Model Predictive Control:
Theory, Computation, and Design
2nd Edition

ISBN 978-0-9759377-8-5

Model Predictive Control:
Theory, Computation, and Design
2nd Edition

James B. Rawlings
Department of Chemical Engineering
University of California
Santa Barbara, California, USA

David Q. Mayne
Department of Electrical and Electronic Engineering
Imperial College London
London, England

Moritz M. Diehl
Department of Microsystems Engineering and
Department of Mathematics
University of Freiburg
Freiburg, Germany

Nob Hill Publishing

Santa Barbara, California

This book was set in Lucida using LaTeX, and printed and bound by Amazon.

Cover design by Cheryl M. and James B. Rawlings.

Copyright © 2026 by Nob Hill Publishing, LLC

All rights reserved.

Nob Hill Publishing, LLC
Cheryl M. Rawlings, publisher
Santa Barbara, CA 93101
orders@nobhillpublishing.com
http://www.nobhillpublishing.com

No part of this book may be reproduced, in any form or by any means, without permission in writing from the publisher.

Library of Congress Control Number: 2020942771

Printed in the United States of America.

First **Edition**	
First Printing	August 2009
Electronic Download (1st)	November 2013
Electronic Download (2nd)	April 2014
Electronic Download (3rd)	July 2014
Electronic Download (4th)	October 2014
Electronic Download (5th)	February 2015
Second **Edition**	
First Printing	October 2017
Electronic Download (1st)	October 2018
Electronic Download (2nd)	February 2019
Paperback **Edition**	
Third Printing	October 2020
Electronic Download (3rd)	October 2020
Electronic Download (4th)	April 2022
Fifth Printing	September 2024
Sixth Printing	May 2026

To Cheryl, Josephine, and Stephanie,

for their love, encouragement, and patience.

Preface to the Second Edition

In the eight years since the publication of the first edition, the field of model predictive control (MPC) has seen tremendous progress. First and foremost, the algorithms and high-level software available for solving challenging nonlinear optimal control problems have advanced significantly. For this reason, we have added a new chapter, Chapter 8, "Numerical Optimal Control," and coauthor, Professor Moritz M. Diehl. This chapter gives an introduction into methods for the numerical solution of the MPC optimization problem. Numerical optimal control builds on two fields: simulation of differential equations, and numerical optimization. Simulation is often covered in undergraduate courses and is therefore only briefly reviewed. Optimization is treated in much more detail, covering topics such as derivative computations, Hessian approximations, and handling inequalities. Most importantly, the chapter presents some of the many ways that the specific structure of optimal control problems arising in MPC can be exploited algorithmically.

We have also added a software release with the second edition of the text. The software enables the solution of all of the examples and exercises in the text requiring numerical calculation. The software is based on the freely available CasADi language, and a high-level set of Octave/MATLAB functions, MPCTools, to serve as an interface to CasADi. These tools have been tested in several MPC short courses to audiences composed of researchers and practitioners. The software can be downloaded from `www.chemengr.ucsb.edu/~jbraw/mpc`.

In Chapter 2, we have added sections covering the following topics:

- economic MPC
- MPC with discrete actuators

We also present a more recent form of suboptimal MPC that is provably robust as well as computationally tractable for online solution of nonconvex MPC problems.

In Chapter 3, we have added a discussion of stochastic MPC, which has received considerable recent research attention.

In Chapter 4, we have added a new treatment of state estimation with persistent, bounded process and measurement disturbances. We have also removed the discussion of particle filtering. There are two

reasons for this removal; first, we wanted to maintain a manageable total length of the text; second, all of the available sampling strategies in particle filtering come up against the "curse of dimensionality," which renders the state estimates inaccurate for dimension higher than about five. The material on particle filtering remains available on the text website.

In Chapter 6, we have added a new section for distributed MPC of nonlinear systems.

In Chapter 7, we have added the software to compute the critical regions in explicit MPC.

Throughout the text, we support the stronger KL-definition of asymptotic stability, in place of the classical definition used in the first edition.

The most significant notational change is to denote a sequence with $(a, b, c, \ldots)$ instead of with $\{a, b, c, \ldots\}$ as in the first edition.

JBR	DQM	MMD
Madison, Wis., USA	London, England	Freiburg, Germany

Added for the second edition, third printing

The second edition, first printing was made available electronically in October 2018. The February 2019 second (electronic only) printing mainly corrected typographical errors. This third printing was printed as a paperback and made available electronically in October 2020.

In this third printing, besides removing typographical and other errors, Chapter 4 was revised significantly. The analysis of Moving Horizon Estimation and Full Information Estimation with bounded disturbances has improved significantly in the last several years due to the research efforts of several groups. We have attempted to bring the material in Chapter 4 up to date with this current literature.

Moreover, the section in Chapter 3 on Stochastic MPC was updated, and a new section on discrete actuators was added to Chapter 8.

JBR	DQM	MMD
Santa Barbara, CA, USA	London, England	Freiburg, Germany

Preface

Our goal in this text is to provide a comprehensive and foundational treatment of the theory and design of model predictive control (MPC). By now several excellent monographs emphasizing various aspects of MPC have appeared (a list appears at the beginning of Chapter 1), and the reader may naturally wonder what is offered here that is new and different. By providing a comprehensive treatment of the MPC foundation, we hope that this text enables researchers to learn and *teach* the fundamentals of MPC without continuously searching the diverse control research literature for omitted arguments and requisite background material. When teaching the subject, it is essential to have a collection of exercises that enables the students to assess their level of comprehension and mastery of the topics. To support the teaching and learning of MPC, we have included more than 200 end-of-chapter exercises. A complete solution manual (more than 300 pages) is available for course instructors.

Chapter 1 is introductory. It is intended for graduate students in engineering who have not yet had a systems course. But it serves a second purpose for those who have already taken the first graduate systems course. It derives all the results of the linear quadratic regulator and optimal Kalman filter using only those arguments that extend to the nonlinear and constrained cases to be covered in the later chapters. Instructors may find that this tailored treatment of the introductory systems material serves both as a review and a preview of arguments to come in the later chapters.

Chapters 2–4 are foundational and should probably be covered in any graduate level MPC course. Chapter 2 covers regulation to the origin for nonlinear and constrained systems. This material presents in a unified fashion many of the major research advances in MPC that took place during the last 20 years. It also includes more recent topics such as regulation to an unreachable setpoint that are only now appearing in the research literature. Chapter 3 addresses MPC design for robustness, with a focus on MPC using tubes or bundles of trajectories in place of the single nominal trajectory. This chapter again unifies a large body of research literature concerned with robust MPC. Chapter 4 covers state estimation with an emphasis on moving horizon estimation, but also

covers extended and unscented Kalman filtering, and particle filtering.

Chapters 5–7 present more specialized topics. Chapter 5 addresses the special requirements of MPC based on output measurement instead of state measurement. Chapter 6 discusses how to design distributed MPC controllers for large-scale systems that are decomposed into many smaller, interacting subsystems. Chapter 7 covers the explicit optimal control of constrained linear systems. The choice of coverage of these three chapters may vary depending on the instructor's or student's own research interests.

Three appendices are included, again, so that the reader is not sent off to search a large research literature for the fundamental arguments used in the text. Appendix A covers the required mathematical background. Appendix B summarizes the results used for stability analysis including the various types of stability and Lyapunov function theory. Since MPC is an optimization-based controller, Appendix C covers the relevant results from optimization theory. In order to reduce the size and expense of the text, the three appendices are available on the web: `www.chemengr.ucsb.edu/~jbraw/mpc`. Note, however, that all material in the appendices is included in the book's printed table of contents, and subject and author indices. The website also includes sample exams, and homework assignments for a one-semester graduate course in MPC. All of the examples and exercises in the text were solved with Octave. Octave is freely available from `www.octave.org`.

JBR
Madison, Wisconsin, USA

DQM
London, England

Acknowledgments

Both authors would like to thank the Department of Chemical and Biological Engineering of the University of Wisconsin for hosting DQM's visits to Madison during the preparation of this monograph. Funding from the Paul A. Elfers Professorship provided generous financial support.

JBR would like to acknowledge the graduate students with whom he has had the privilege to work on model predictive control topics: Rishi Amrit, Dennis Bonné, John Campbell, John Eaton, Peter Findeisen, Rolf Findeisen, Eric Haseltine, John Jørgensen, Nabil Laachi, Scott Meadows, Scott Middlebrooks, Steve Miller, Ken Muske, Brian Odelson, Murali Rajamani, Chris Rao, Brett Stewart, Kaushik Subramanian, Aswin Venkat, and Jenny Wang. He would also like to thank many colleagues with whom he has collaborated on this subject: Frank Allgöwer, Tom Badgwell, Bhavik Bakshi, Don Bartusiak, Larry Biegler, Moritz Diehl, Jim Downs, Tom Edgar, Brian Froisy, Ravi Gudi, Sten Bay Jørgensen, Jay Lee, Fernando Lima, Wolfgang Marquardt, Gabriele Pannocchia, Joe Qin, Harmon Ray, Pierre Scokaert, Sigurd Skogestad, Tyler Soderstrom, Steve Wright, and Robert Young.

DQM would like to thank his colleagues at Imperial College, especially Richard Vinter and Martin Clark, for providing a stimulating and congenial research environment. He is very grateful to Lucien Polak and Graham Goodwin with whom he has collaborated extensively and fruitfully over many years; he would also like to thank many other colleagues, especially Karl Åström, Roger Brockett, Larry Ho, Petar Kokotovic, and Art Krener, from whom he has learned much. He is grateful to past students who have worked with him on model predictive control: Ioannis Chrysochoos, Wilbur Langson, Hannah Michalska, Sasa Raković, and Warren Schroeder; Hannah Michalska and Sasa Raković, in particular, contributed very substantially. He owes much to these past students, now colleagues, as well as to Frank Allgöwer, Rolf Findeisen, Eric Kerrigan, Konstantinos Kouramus, Chris Rao, Pierre Scokaert, and Maria Seron for their collaborative research in MPC.

Both authors would especially like to thank Tom Badgwell, Bob Bird, Eric Kerrigan, Ken Muske, Gabriele Pannocchia, and Maria Seron for their careful and helpful reading of parts of the manuscript. John Eaton

again deserves special mention for his invaluable technical support during the entire preparation of the manuscript.

Added for the second edition. JBR would like to acknowledge the most recent generation of graduate students with whom he has had the privilege to work on model predictive control research topics: Doug Allan, Travis Arnold, Cuyler Bates, Luo Ji, Nishith Patel, Michael Risbeck, and Megan Zagrobelny.

In preparing the second edition, and, in particular, the software release, the current group of graduate students far exceeded expectations to help finish the project. Quite simply, the project could not have been completed in a timely fashion without their generosity, enthusiasm, professionalism, and selfless contribution. Michael Risbeck deserves special mention for creating the MPCTools interface to CasADi, and updating and revising the tools used to create the website to distribute the text- and software-supporting materials. He also wrote code to calculate explicit MPC control laws in Chapter 7. Nishith Patel made a major contribution to the subject index, and Doug Allan contributed generously to the presentation of moving horizon estimation in Chapter 4.

A research leave for JBR in Fall 2016, again funded by the Paul A. Elfers Professorship, was instrumental in freeing up time to complete the revision of the text and further develop computational exercises.

MMD wants to especially thank Jesus Lago Garcia, Jochem De Schutter, Andrea Zanelli, Dimitris Kouzoupis, Joris Gillis, Joel Andersson, and Robin Verschueren for help with the preparation of exercises and examples in Chapter 8; and also wants to acknowledge the following current and former team members that contributed to research and teaching on optimal and model predictive control at the Universities of Leuven and Freiburg: Adrian Bürger, Hans Joachim Ferreau, Jörg Fischer, Janick Frasch, Gianluca Frison, Niels Haverbeke, Greg Horn, Boris Houska, Jonas Koenemann, Attila Kozma, Vyacheslav Kungurtsev, Giovanni Licitra, Rien Quirynen, Carlo Savorgnan, Quoc Tran-Dinh, Milan Vukov, and Mario Zanon. MMD also wants to thank Frank Allgöwer, Alberto Bemporad, Rolf Findeisen, Larry Biegler, Hans Georg Bock, Stephen Boyd, Sébastien Gros, Lars Grüne, Colin Jones, John Bagterp Jørgensen, Christian Kirches, Daniel Leineweber, Katja Mombaur, Yurii Nesterov, Toshiyuki Ohtsuka, Goele Pipeleers, Andreas Potschka, Sebastian Sager, Johannes P. Schlöder, Volker Schulz, Marc Steinbach, Jan Swevers, Philippe Toint, Andrea Walther, Stephen Wright, Joos Vandewalle, and Stefan Vandewalle for inspiring discussions on numerical optimal control

methods and their presentation during the last 20 years.

All three authors would especially like to thank Joel Andersson and Joris Gillis for having developed CasADi and for continuing its support, and for having helped to improve some of the exercises in the text.

Added for the third printing of the second edition. The authors would like to acknowledge and thank Doug Allan again for his suggestions and help with the revision of Chapter 4. Much of the new material on Full Information Estimation and Moving Horizon Estimation is a direct result of Doug's research papers and 2020 PhD thesis on state estimation. Koty McAllister provided expert assistance in the update of stochastic MPC in Chapter 3. Finally, Adrian Buerger and Pratyush Kumar provided valuable assistance on the addition of the discrete actuator numerics to Chapter 8.

Added for the sixth printing of the second edition. All figures in the text and solution manual requiring numerical computation were produced with Python as a replacement language for Octave starting with this printing. Davide Mannini, Titus Quah, and Prithvi Dake were instrumental in producing this transformation of approximately 142 codes. A sabbatical provided to JBR by UCSB in Spring 2025 was invaluable in freeing up time to complete this project.

Contents

1 Getting Started with Model Predictive Control 1
1.1 Introduction . . . 1
1.2 Models and Modeling . . . 1
1.2.1 Linear Dynamic Models . . . 2
1.2.2 Input-Output Models . . . 3
1.2.3 Distributed Models . . . 4
1.2.4 Discrete Time Models . . . 5
1.2.5 Constraints . . . 6
1.2.6 Deterministic and Stochastic . . . 9
1.3 Introductory MPC Regulator . . . 11
1.3.1 Linear Quadratic Problem . . . 11
1.3.2 Optimizing Multistage Functions . . . 12
1.3.3 Dynamic Programming Solution . . . 18
1.3.4 The Infinite Horizon LQ Problem . . . 21
1.3.5 Controllability . . . 23
1.3.6 Convergence of the Linear Quadratic Regulator . . . 24
1.4 Introductory State Estimation . . . 26
1.4.1 Linear Systems and Normal Distributions . . . 27
1.4.2 Linear Optimal State Estimation . . . 29
1.4.3 Least Squares Estimation . . . 33
1.4.4 Moving Horizon Estimation . . . 39
1.4.5 Observability . . . 41
1.4.6 Convergence of the State Estimator . . . 43
1.5 Tracking, Disturbances, and Zero Offset . . . 46
1.5.1 Tracking . . . 46
1.5.2 Disturbances and Zero Offset . . . 49
1.6 Exercises . . . 60

2 Model Predictive Control—Regulation 89
2.1 Introduction . . . 89
2.2 Model Predictive Control . . . 91
2.3 Dynamic Programming Solution . . . 107
2.4 Stability . . . 112
2.4.1 Introduction . . . 112

2.4.2 Stabilizing Conditions 114
2.4.3 Exponential Stability 120
2.4.4 Controllability and Observability 120
2.4.5 Time-Varying Systems 123
2.5 Examples of MPC 131
2.5.1 The Unconstrained Linear Quadratic Regulator . . 132
2.5.2 Unconstrained Linear Periodic Systems 133
2.5.3 Stable Linear Systems with Control Constraints . 135
2.5.4 Linear Systems with Control and State Constraints 136
2.5.5 Constrained Nonlinear Systems 139
2.5.6 Constrained Nonlinear Time-Varying Systems . . . 141
2.6 Is a Terminal Constraint Set $\mathbb{X}_f$ Necessary? 144
2.7 Suboptimal MPC 147
2.7.1 Extended State 150
2.7.2 Asymptotic Stability of Difference Inclusions . . . 150
2.8 Economic Model Predictive Control 153
2.8.1 Asymptotic Average Performance 155
2.8.2 Dissipativity and Asymptotic Stability 156
2.9 Discrete Actuators 160
2.10 Concluding Comments 163
2.11 Notes 166
2.12 Exercises 172

3 Robust and Stochastic Model Predictive Control 193
3.1 Introduction 193
3.1.1 Types of Uncertainty 193
3.1.2 Feedback Versus Open-Loop Control 195
3.1.3 Robust and Stochastic MPC 200
3.1.4 Tubes 202
3.1.5 Difference Inclusion Description of Uncertain Systems 203
3.2 Nominal (*Inherent*) Robustness 204
3.2.1 Introduction 204
3.2.2 Difference Inclusion Description of Discontinuous Systems 206
3.2.3 When Is Nominal MPC Robust? 207
3.2.4 Robustness of Nominal MPC 209
3.3 Min-Max Optimal Control: Dynamic Programming Solution 214
3.3.1 Introduction 214
3.3.2 Properties of the Dynamic Programming Solution 216

3.4 Robust Min-Max MPC 220
3.5 Tube-Based Robust MPC 223
3.5.1 Introduction 223
3.5.2 Outer-Bounding Tube for a Linear System with Additive Disturbance 224
3.5.3 Tube-Based MPC of Linear Systems with Additive Disturbances 228
3.5.4 Improved Tube-Based MPC of Linear Systems with Additive Disturbances 234
3.6 Tube-Based MPC of Nonlinear Systems 236
3.6.1 The Nominal Trajectory 238
3.6.2 Model Predictive Controller 238
3.6.3 Choosing the Nominal Constraint Sets $\bar{\mathbb{U}}$ and $\bar{\mathbb{X}}$ . . 242
3.7 Stochastic MPC 246
3.7.1 Introduction 246
3.7.2 Stability of Stochastic MPC 248
3.7.3 Tube-based stochastic MPC 250
3.8 Notes 257
3.9 Exercises 262

4 State Estimation **269**
4.1 Introduction 269
4.2 Full Information Estimation 269
4.2.1 Nominal Estimator Stability 279
4.2.2 Robust Estimator Stability 284
4.2.3 Interlude—Linear System Review 287
4.3 Moving Horizon Estimation 292
4.3.1 Zero Prior Weighting 293
4.3.2 Nonzero Prior Weighting 296
4.3.3 RGES of MHE under exponential assumptions . . . 297
4.4 Other Nonlinear State Estimators 302
4.4.1 Particle Filtering 302
4.4.2 Extended Kalman Filtering 302
4.4.3 Unscented Kalman Filtering 304
4.4.4 EKF, UKF, and MHE Comparison 306
4.5 On combining MHE and MPC 312
4.6 Notes 318
4.7 Exercises 321

5 Output Model Predictive Control **333**
5.1 Introduction 333
5.2 A Method for Output MPC 335
5.3 Linear Constrained Systems: Time-Invariant Case 338
5.3.1 Introduction 338
5.3.2 State Estimator 338
5.3.3 Controlling $\hat{x}$ 340
5.3.4 Output MPC 342
5.3.5 Computing the Tightened Constraints 346
5.4 Linear Constrained Systems: Time-Varying Case 347
5.5 Offset-Free MPC 347
5.5.1 Estimation 349
5.5.2 Control 350
5.5.3 Convergence Analysis 354
5.6 Nonlinear Constrained Systems 357
5.7 Notes 357
5.8 Exercises 360

6 Distributed Model Predictive Control **363**
6.1 Introduction and Preliminary Results 363
6.1.1 Least Squares Solution 364
6.1.2 Stability of Suboptimal MPC 369
6.2 Unconstrained Two-Player Game 374
6.2.1 Centralized Control 376
6.2.2 Decentralized Control 377
6.2.3 Noncooperative Game 378
6.2.4 Cooperative Game 386
6.2.5 Tracking Nonzero Setpoints 392
6.2.6 State Estimation 399
6.3 Constrained Two-Player Game 400
6.3.1 Uncoupled Input Constraints 402
6.3.2 Coupled Input Constraints 405
6.3.3 Exponential Convergence with Estimate Error . . . 406
6.3.4 Disturbance Models and Zero Offset 408
6.4 Constrained M-Player Game 412
6.5 Nonlinear Distributed MPC 414
6.5.1 Nonconvexity 415
6.5.2 Distributed Algorithm for Nonconvex Functions . 417
6.5.3 Distributed Nonlinear Cooperative Control 419
6.5.4 Stability 422

6.6 Notes 425
6.7 Exercises 429

7 Explicit Control Laws for Constrained Linear Systems 445
7.1 Introduction 445
7.2 Parametric Programming 446
7.3 Parametric Quadratic Programming 451
7.3.1 Preliminaries 451
7.3.2 Preview 452
7.3.3 Optimality Condition for a Convex Program 453
7.3.4 Solution of the Parametric Quadratic Program . . 456
7.3.5 Continuity of $V^0(\cdot)$ and $u^0(\cdot)$ 460
7.4 Constrained Linear Quadratic Control 461
7.5 Parametric Piecewise Quadratic Programming 463
7.6 DP Solution of the Constrained LQ Control Problem . . . 469
7.7 Parametric Linear Programming 470
7.7.1 Preliminaries 470
7.7.2 Minimizer $u^0(x)$ is Unique for all $x \in \mathcal{X}$ 472
7.8 Constrained Linear Control 475
7.9 Computation 476
7.10 Notes 477
7.11 Exercises 478

8 Numerical Optimal Control 485
8.1 Introduction 485
8.1.1 Discrete Time Optimal Control Problem 486
8.1.2 Convex Versus Nonconvex Optimization 487
8.1.3 Simultaneous Versus Sequential Optimal Control 490
8.1.4 Continuous Time Optimal Control Problem 492
8.2 Numerical Simulation 495
8.2.1 Explicit Runge-Kutta Methods 496
8.2.2 Stiff Equations and Implicit Integrators 500
8.2.3 Implicit Runge-Kutta and Collocation Methods . . 501
8.2.4 Differential Algebraic Equations 505
8.2.5 Integrator Adaptivity 507
8.3 Solving Nonlinear Equation Systems 507
8.3.1 Linear Systems 507
8.3.2 Nonlinear Root-Finding Problems 508
8.3.3 Local Convergence of Newton-Type Methods . . . 511
8.3.4 Affine Invariance 513
8.3.5 Globalization for Newton-Type Methods 513

8.4 Computing Derivatives 514
8.4.1 Numerical Differentiation 515
8.4.2 Algorithmic Differentiation 516
8.4.3 Implicit Function Interpretation 517
8.4.4 Algorithmic Differentiation in Forward Mode . . . 520
8.4.5 Algorithmic Differentiation in Reverse Mode . . . 522
8.4.6 Differentiation of Simulation Routines 525
8.4.7 Algorithmic and Symbolic Differentiation Software 527
8.4.8 CasADi for Optimization 527
8.5 Direct Optimal Control Parameterizations 530
8.5.1 Direct Single Shooting 532
8.5.2 Direct Multiple Shooting 534
8.5.3 Direct Transcription and Collocation Methods . . 538
8.6 Nonlinear Optimization 542
8.6.1 Optimality Conditions and Perturbation Analysis 543
8.6.2 Nonlinear Optimization with Equalities 546
8.6.3 Hessian Approximations 547
8.7 Newton-Type Optimization with Inequalities 550
8.7.1 Sequential Quadratic Programming 551
8.7.2 Nonlinear Interior Point Methods 552
8.7.3 Comparison of SQP and Nonlinear IP Methods . . 554
8.8 Structure in Discrete Time Optimal Control 555
8.8.1 Simultaneous Approach 556
8.8.2 Linear Quadratic Problems (LQP) 558
8.8.3 LQP Solution by Riccati Recursion 558
8.8.4 LQP Solution by Condensing 560
8.8.5 Sequential Approaches and Sparsity Exploitation 562
8.8.6 Differential Dynamic Programming 564
8.8.7 Additional Constraints in Optimal Control 566
8.9 Online Optimization Algorithms 567
8.9.1 General Algorithmic Considerations 568
8.9.2 Continuation Methods and Real-Time Iterations . 571
8.10 Discrete Actuators 574
8.11 Notes 579
8.12 Exercises 581

Author Index **600**

Citation Index **608**

Subject Index **614**

A Mathematical Background **624**
A.1 Introduction . . . 624
A.2 Vector Spaces . . . 624
A.3 Range and Nullspace of Matrices . . . 624
A.4 Linear Equations — Existence and Uniqueness . . . 625
A.5 Pseudo-Inverse . . . 625
A.6 Partitioned Matrix Inversion Theorem . . . 628
A.7 Quadratic Forms . . . 629
A.8 Norms in $\mathbb{R}^n$ . . . 631
A.9 Sets in $\mathbb{R}^n$ . . . 631
A.10 Sequences . . . 632
A.11 Continuity . . . 633
A.12 Derivatives . . . 636
A.13 Convex Sets and Functions . . . 641
A.13.1 Convex Sets . . . 641
A.13.2 Convex Functions . . . 646
A.14 Differential Equations . . . 648
A.15 Random Variables and the Probability Density . . . 654
A.16 Multivariate Density Functions . . . 659
A.16.1 Statistical Independence and Correlation . . . 668
A.17 Conditional Probability and Bayes's Theorem . . . 672
A.18 Exercises . . . 678

B Stability Theory **693**
B.1 Introduction . . . 693
B.2 Stability and Asymptotic Stability . . . 696
B.3 Lyapunov Stability Theory . . . 700
B.3.1 Time-Invariant Systems . . . 701
B.3.2 Time-Varying, Constrained Systems . . . 707
B.3.3 Upper bounding $\mathcal{K}$ functions . . . 709
B.4 Robust Stability . . . 709
B.4.1 Nominal Robustness . . . 709
B.4.2 Robustness . . . 711
B.5 Control Lyapunov Functions . . . 713
B.6 Input-to-State Stability . . . 717
B.7 Output-to-State Stability and Detectability . . . 719
B.8 Input/Output-to-State Stability . . . 720
B.9 Incremental-Input/Output-to-State Stability . . . 722
B.10 Observability . . . 722
B.11 Exercises . . . 724

C Optimization **729**
C.1 Dynamic Programming 729
C.1.1 Optimal Control Problem 731
C.1.2 Dynamic Programming 733
C.2 Optimality Conditions 737
C.2.1 Tangent and Normal Cones 737
C.2.2 Convex Optimization Problems 741
C.2.3 Convex Problems: Polyhedral Constraint Set . . . 743
C.2.4 Nonconvex Problems 745
C.2.5 Tangent and Normal Cones 746
C.2.6 Constraint Set Defined by Inequalities 750
C.2.7 Constraint Set; Equalities and Inequalities 753
C.3 Set-Valued Functions and Continuity of Value Function . 755
C.3.1 Outer and Inner Semicontinuity 757
C.3.2 Continuity of the Value Function 759
C.4 Exercises 767

List of Figures

1.1 System with input $\overline{u}$, output $\overline{y}$, and transfer function matrix G connecting them; the model is $\overline{y} = G\overline{u}$. 3
1.2 Typical input constraint sets $\mathbb{U}$ for (a) continuous actuators and (b) mixed continuous/discrete actuators. 9
1.3 Output of a stochastic system versus time. 10
1.4 Two quadratic functions and their sum. 15
1.5 Schematic of the moving horizon estimation problem. . . . 39
1.6 MPC controller consisting of: receding horizon regulator, state estimator, and target selector. 52
1.7 Schematic of the well-stirred reactor. 54
1.8 Three measured outputs versus time after a step change in inlet flowrate at 10 minutes; $n_d = 2$. 57
1.9 Two manipulated inputs versus time after a step change in inlet flowrate at 10 minutes; $n_d = 2$. 57
1.10 Three measured outputs versus time after a step change in inlet flowrate at 10 minutes; $n_d = 3$. 58
1.11 Two manipulated inputs versus time after a step change in inlet flowrate at 10 minutes; $n_d = 3$. 59
1.12 Plug-flow reactor. 60
1.13 Pendulum with applied torque. 62
1.14 Feedback control system with output disturbance d, and setpoint y_{sp}. 84

2.1 Example of MPC. 101
2.2 Feasible region $\mathcal{U}_2$, elliptical cost contours and ellipse center $a(x)$, and constrained minimizers for different values of x. 102
2.3 First element of control constraint set $\mathcal{U}_3(x)$ (shaded) and control law $\kappa_3(x)$ (line) versus $x = (\cos(\theta), \sin(\theta))$, $\theta \in [-\pi, \pi]$ on the unit circle for a nonlinear system with terminal constraint. 106
2.4 Optimal cost $V_3^0(x)$ versus x on the unit circle. 107

2.5 Closed-loop economic MPC versus tracking MPC starting at $x = (-8, 8)$ with optimal steady state $(8, 4)$. Both controllers asymptotically stabilize the steady state. Dashed contours show cost functions for each controller. 159
2.6 Closed-loop evolution under economic MPC. The rotated cost function $\widetilde{V}^0$ is a Lyapunov function for the system. . . 160
2.7 Diagram of tank/cooler system. Each cooling unit can be either on or off, and if on, it must be between its (possibly nonzero) minimum and maximum capacities. 163
2.8 Feasible sets $\mathcal{X}_N$ for two values of $\dot{Q}_{\min}$. Note that for $\dot{Q}_{\min} = 9$ (right-hand side), $\mathcal{X}_N$ for $N \leq 4$ are disconnected sets. 164
2.9 Phase portrait for closed-loop evolution of cooler system with $\dot{Q}_{\min} = 9$. Line colors show value of discrete actuator u_2. 165
2.10 Region of attraction (shaded region) for constrained MPC controller of Exercise 2.6. 174
2.11 The region $\mathbb{X}_f$, in which the unconstrained LQR control law is feasible for Exercise 2.7. 175
2.12 The region of attraction for terminal constraint $x(N) \in \mathbb{X}_f$ and terminal penalty $V_f(x) = (1/2)x'\Pi x$ and the estimate of $\bar{\mathcal{X}}_N$ for Exercise 2.8. 177
2.13 Inconsistent setpoint $(x_{\text{sp}}, u_{\text{sp}})$, unreachable stage cost $\ell(x, u)$, and optimal steady states (x_s, u_s), and stage costs $\ell_s(x, u)$ for constrained and unconstrained systems. . . . 181
2.14 Stage cost versus time for the case of unreachable setpoint. 182
2.15 Rotated cost-function contour $\widetilde{\ell}(x, u) = 0$ (circles) for $\lambda = 0, -8, -12$. Shaded region shows feasible region where $\widetilde{\ell}(x, u) < 0$. 185

3.1 Open-loop and feedback trajectories. 198
3.2 The sets $\mathcal{X}_N$, R_b, and R_c. 214
3.3 Outer-bounding tube $\mathbf{X}(z, \bar{\mathbf{u}})$. 228
3.4 Minimum feasible α for varying N. Note that we require $\alpha \in [0, 1)$. 232
3.5 Bounds on tightened constraint set $\bar{\mathbb{Z}}$ for varying N. Bounds are $|x_1| \leq \chi_1$, $|x_2| \leq \chi_2$, and $|u| \leq \mu$. 233
3.6 Comparison of 100 realizations of standard and tube-based MPC for the chemical reactor example. 244
3.7 Comparison of standard and tube-based MPC with an aggressive model predictive controller. 245

3.8 Concentration versus time for the ancillary model predictive controller with sample time $\Delta = 12$ (left) and $\Delta = 8$ (right). 246
3.9 Observed probability $\varepsilon_{\text{test}}$ of constraint violation for $i = 10$. Distribution is based on 500 trials for each value of ε. Dashed line shows the outcome predicted by formula (3.25), i.e., $\varepsilon_{\text{test}} = \varepsilon$. 255
3.10 Closed-loop robust MPC state evolution with uniformly distributed $|w| \leq 0.1$ from four different x_0. 263

4.1 Smoothing update. 299
4.2 Comparison of filtering and smoothing updates for the batch reactor system. Second column shows absolute estimate error. 300
4.3 Evolution of the state (solid line) and EKF state estimate (dashed line). 308
4.4 Evolution of the state (solid line) and UKF state estimate (dashed line). 309
4.5 Evolution of the state (solid line) and MHE state estimate (dashed line). 310
4.6 Perturbed trajectories terminating in $\mathbb{X}_f$ 315
4.7 Closed-loop performance of combined nonlinear MHE/MPC with no disturbances. First column shows system states, and second column shows estimation error. Dashed line shows concentration setpoint. Vertical lines indicate times of setpoint changes. 317
4.8 Closed-loop performance of combined nonlinear MHE/MPC for varying disturbance size. The system is controlled between two steady states. 318

5.1 State estimator tube. The solid line $\hat{x}(t)$ is the center of the tube, and the dashed line is a sample trajectory of $x(t)$. 336
5.2 The system with disturbance. The state estimate lies in the inner tube, and the state lies in the outer tube. 337

6.1 Convex step from (u_1^p, u_2^p) to (u_1^{p+1}, u_2^{p+1}). 380
6.2 Ten iterations of noncooperative steady-state calculation. . 397
6.3 Ten iterations of cooperative steady-state calculation. . . . 397
6.4 Ten iterations of noncooperative steady-state calculation; reversed pairing. 398

6.5 Ten iterations of cooperative steady-state calculation; reversed pairing. 398
6.6 Cooperative control stuck on the boundary of $\mathbb{U}$ under coupled constraints 406
6.7 Cost contours for a two-player, nonconvex game. 416
6.8 Nonconvex function optimized with the distributed gradient algorithm. 419
6.9 Closed-loop state and control evolution with $(x_1(0), x_2(0)) = (3, -3)$. 424
6.10 Contours of $V(x(0), \mathbf{u}_1, \mathbf{u}_2)$ for $N = 1$. 425
6.11 Optimizing a quadratic function in one set of variables at a time. 434
6.12 Constrained optimality conditions and the normal cone. . 438

7.1 The sets $\mathbb{Z}$, $\mathcal{X}$, and $\mathcal{U}(x)$. 448
7.2 Parametric linear program. 448
7.3 Unconstrained parametric quadratic program. 449
7.4 Parametric quadratic program. 449
7.5 Polar cone. 454
7.6 Regions R_x, $x \in X$ for a second-order example. 462
7.7 Solution to a parametric LP. 473
7.8 Solution times for explicit and implicit MPC for $N = 20$. . . 480

8.1 Feasible set and reduced objective $\psi(u(0))$ of the nonlinear MPC Example 8.1. 490
8.2 Performance of different integration methods. 499
8.3 Polynomial approximation $\widetilde{x}_1(t)$ and true trajectory $x_1(t)$ of the first state and its derivative. 504
8.4 Performance of implicit integration methods on a stiff ODE. 506
8.5 Newton-type iterations for solution of $R(z) = 0$ from Example 8.5. Left: exact Newton method. Right: constant Jacobian approximation. 510
8.6 Convergence of different sequences as a function of k. . . 512
8.7 Relaxed and binary feasible solution for Example 8.17. . . 578
8.8 A hanging chain at rest. See Exercise 8.6(b). 585
8.9 Direct single shooting solution for (8.65) without path constraints. 587
8.10 Open-loop simulation for (8.65) using collocation. 590
8.11 Gauss-Newton iterations for the direct multiple-shooting method 592

A.1 The four fundamental subspaces of matrix A 626
A.2 Matrix A maps into $\mathcal{R}(A)$. 627
A.3 Pseudo-inverse of A maps into $\mathcal{R}(A')$. 627
A.4 Subgradient. 640
A.5 Separating hyperplane. 642
A.6 Polar cone. 645
A.7 A convex function. 646
A.8 Normal distribution. 658
A.9 Multivariate normal in two dimensions. 660
A.10 The geometry of quadratic form $x'Ax = b$. 661
A.11 A nearly singular normal density in two dimensions. 665
A.12 The region $\mathbb{X}(c)$ for $y = \max(x_1, x_2) \leq c$. 667
A.13 A joint density function for the two uncorrelated random variables. 670
A.14 The probability distribution and inverse distribution for random variable ξ. 687

B.1 Stability of the origin. 697
B.2 An attractive but unstable origin. 698

C.1 Routing problem. 730
C.2 Approximation of the set U. 738
C.3 Tangent cones. 738
C.4 Normal at u. 739
C.5 Condition of optimality. 745
C.6 Tangent and normal cones. 747
C.7 Condition of optimality. 749
C.8 $\mathcal{F}_U(u) \neq \mathcal{T}_U(u)$. 751
C.9 Graph of set-valued function $U(\cdot)$. 756
C.10 Graphs of discontinuous set-valued functions. 757
C.11 Outer and inner semicontinuity of $U(\cdot)$. 758
C.12 Subgradient of $f(\cdot)$. 762

List of Examples and Statements

1.1 Example: Sum of quadratic functions 15
1.2 Lemma: Hautus lemma for controllability 24
1.3 Lemma: LQR convergence 24
1.4 Lemma: Hautus lemma for observability 42
1.5 Lemma: Convergence of estimator cost 43
1.6 Lemma: Estimator convergence 44
1.7 Assumption: Target feasibility and uniqueness 48
1.8 Lemma: Detectability of the augmented system 50
1.9 Corollary: Dimension of the disturbance 50
1.10 Lemma: Offset-free control 52
1.11 Example: More measured outputs than inputs and zero offset 53
1.12 Lemma: Hautus lemma for stabilizability 68
1.13 Lemma: Hautus lemma for detectability 72
1.14 Lemma: Stabilizable systems and feasible targets 82

2.1 Proposition: Continuity of system solution 94
2.2 Assumption: Continuity of system and cost 97
2.3 Assumption: Properties of constraint sets 98
2.4 Proposition: Existence of solution to optimal control problem 98
2.5 Example: Linear quadratic MPC 99
2.6 Example: Closer inspection of linear quadratic MPC 101
2.7 Theorem: Continuity of value function and control law . . 104
2.8 Example: Discontinuous MPC control law 105
2.9 Definition: Positive and control invariant sets 109
2.10 Proposition: Existence of solutions to DP recursion 110
2.11 Definition: Asymptotically stable and GAS 112
2.12 Definition: Lyapunov function 113
2.13 Theorem: Lyapunov stability theorem 113
2.14 Assumption: Basic stability assumption 114
2.15 Proposition: The value function $V_N^0(\cdot)$ is locally bounded . 115
2.16 Proposition: Extension of upper bound to $\mathcal{X}_N$ 115
2.17 Assumption: Weak controllability 116
2.18 Proposition: Monotonicity of the value function 118
2.19 Theorem: Asymptotic stability of the origin 119

2.20 Definition: Exponential stability 120
2.21 Theorem: Lyapunov function and exponential stability . . 120
2.22 Definition: Input/output-to-state stable (IOSS) 121
2.23 Assumption: Modified basic stability assumption 121
2.24 Theorem: Asymptotic stability with stage cost $\ell(y,u)$ . . . 122
2.25 Assumption: Continuity of system and cost; time-varying case 124
2.26 Assumption: Properties of constraint sets; time-varying case 124
2.27 Definition: Sequential positive invariance and sequential control invariance 125
2.28 Proposition: Continuous system solution; time-varying case 125
2.29 Proposition: Existence of solution to optimal control problem; time-varying case 125
2.30 Definition: Asymptotically stable and GAS for time-varying systems 125
2.31 Definition: Lyapunov function: time-varying, constrained case 126
2.32 Theorem: Lyapunov theorem for asymptotic stability (time-varying, constrained) 126
2.33 Assumption: Basic stability assumption; time-varying case 127
2.34 Proposition: Optimal cost decrease; time-varying case . . . 127
2.35 Proposition: MPC cost is less than terminal cost 127
2.36 Proposition: Optimal value function properties; time-varying case 127
2.37 Assumption: Uniform weak controllability 128
2.38 Proposition: Conditions for uniform weak controllability . 128
2.39 Theorem: Asymptotic stability of the origin: time-varying MPC 130
2.40 Lemma: Entering the terminal region 146
2.41 Theorem: MPC stability; no terminal constraint 146
2.42 Proposition: Admissible warm start in $\mathbb{X}_f$ 149
2.43 Algorithm: Suboptimal MPC 149
2.44 Proposition: Linking warm start and state 150
2.45 Definition: Asymptotic stability (difference inclusion) . . . 150
2.46 Definition: Lyapunov function (difference inclusion) 151
2.47 Proposition: Asymptotic stability (difference inclusion) . . 151
2.48 Theorem: Asymptotic stability of suboptimal MPC 151
2.49 Assumption: Continuity of system and cost 154
2.50 Assumption: Properties of constraint sets 154
2.51 Assumption: Cost lower bound 154

2.52 Proposition: Asymptotic average performance 155
2.53 Definition: Dissipativity 156
2.54 Assumption: Continuity at the steady state 157
2.55 Assumption: Strict dissipativity 157
2.56 Theorem: Asymptotic stability of economic MPC 157
2.57 Example: Economic MPC versus tracking MPC 158
2.58 Example: MPC with mixed continuous/discrete actuators . 162
2.59 Theorem: Lyapunov theorem for asymptotic stability . . . 177
2.60 Proposition: Convergence of state under IOSS 178
2.61 Lemma: An equality for quadratic functions 178
2.62 Lemma: Evolution in a compact set 179

3.1 Definition: Robust global asymptotic stability 207
3.2 Theorem: Lyapunov function and RGAS 208
3.3 Theorem: Robust global asymptotic stability and regularization 209
3.4 Proposition: Bound for continuous functions 211
3.5 Proposition: Robustness of nominal MPC 214
3.6 Definition: Robust control invariance 217
3.7 Definition: Robust positive invariance 217
3.8 Assumption: Basic stability assumption; robust case 218
3.9 Theorem: Recursive feasibility of control policies 218
3.10 Definition: Set algebra and Hausdorff distance 224
3.11 Definition: Robust asymptotic stability of a set 230
3.12 Proposition: Robust asymptotic stability of tube-based MPC for linear systems 230
3.13 Example: Calculation of tightened constraints 231
3.14 Proposition: Recursive feasibility of tube-based MPC 235
3.15 Proposition: Robust exponential stability of improved tube-based MPC 235
3.16 Proposition: Implicit satisfaction of terminal constraint . . 239
3.17 Proposition: Properties of the value function 240
3.18 Proposition: Neighborhoods of the uncertain system 241
3.19 Proposition: Robust positive invariance of tube-based MPC for nonlinear systems 241
3.20 Example: Robust control of an exothermic reaction 243
3.21 Assumption: Stabilizing conditions, stochastic MPC: Version 1 248
3.22 Assumption: Stabilizing conditions, stochastic MPC: Version 2 249

3.23 Proposition: Expected cost bound 250
3.24 Assumption: Robust terminal set condition 253
3.25 Example: Constraint tightening via sampling 254

4.1 Definition: State Estimator 271
4.2 Definition: Robustly globally asymptotically stable estimation 272
4.3 Proposition: RGAS plus convergent disturbances imply convergent estimates 273
4.4 Example: The Kalman filter of a linear system is RGAS . . . 273
4.5 Definition: i-IOSS 275
4.6 Proposition: RGAS estimator implies i-IOSS 276
4.7 Definition: i-IOSS Lyapunov function 277
4.8 Theorem: i-IOSS and Lyapunov function equivalence 277
4.9 Definition: Incremental Stabilizability with respect to stage cost $L(\cdot)$ 277
4.10 Assumption: Continuity 277
4.11 Assumption: Positive-definite stage cost 278
4.12 Assumption: Stabilizability 278
4.13 Assumption: Detectability 278
4.14 Definition: Q-function for estimation 283
4.15 Theorem: Q-function theorem for global asymptotic stability 283
4.16 Theorem: Stability of full information estimation 283
4.17 Assumption: Stage cost under disturbances 284
4.18 Assumption: Stabilizability under disturbances 284
4.19 Definition: Exponentially i-IOSS 285
4.20 Definition: Robustly globally exponentially stable estimation 285
4.21 Proposition: Equivalent definition of RGES 286
4.22 Assumption: Power-law bounds for stage costs 286
4.23 Assumption: Exponential stabilizability 286
4.24 Assumption: Exponential detectability 286
4.25 Theorem: Robust stability of full information estimation . 287
4.26 Lemma: Duality of controllability and observability 291
4.27 Theorem: Riccati iteration and regulator stability 291
4.28 Definition: Observability 293
4.29 Definition: Final-state observability 294
4.30 Definition: Globally $\mathcal{K}$-continuous 294
4.31 Proposition: Observable and global $\mathcal{K}$-continuous imply FSO 294
4.32 Definition: RGAS estimation (observable case) 295
4.33 Theorem: MHE is RGAS (observable case) 295

4.34 Definition: Full information arrival cost 297
4.35 Lemma: MHE and FIE equivalence 297
4.36 Assumption: MHE prior weighting bounds 297
4.37 Theorem: MHE is RGES 298
4.38 Example: Filtering and smoothing updates 300
4.39 Example: EKF, UKF, and MHE performance comparison . . 306
4.40 Definition: i-UIOSS 312
4.41 Assumption: Bounded estimate error 313
4.42 Definition: Robust positive invariance 313
4.43 Definition: Robust asymptotic stability 314
4.44 Definition: ISS Lyapunov function 314
4.45 Proposition: ISS Lyapunov stability theorem 314
4.46 Theorem: Combined MHE/MPC is RAS 316
4.47 Example: Combined MHE/MPC 317

5.1 Definition: Positive invariance; robust positive invariance . 339
5.2 Proposition: Proximity of state and state estimate 339
5.3 Proposition: Proximity of state estimate and nominal state 341
5.4 Assumption: Constraint bounds 342
5.5 Algorithm: Robust control algorithm (linear constrained systems) 344
5.6 Proposition: Exponential stability of output MPC 345
5.7 Algorithm: Robust control algorithm (offset-free MPC) . . . 353

6.1 Algorithm: Suboptimal MPC (simplified) 369
6.2 Definition: Lyapunov stability 370
6.3 Definition: Uniform Lyapunov stability 371
6.4 Definition: Exponential stability 371
6.5 Lemma: Exponential stability of suboptimal MPC 372
6.6 Lemma: Global asymptotic stability and exponential convergence with mixed powers of norm 373
6.7 Lemma: Converse theorem for exponential stability 374
6.8 Assumption: Unconstrained two-player game 380
6.9 Example: Nash equilibrium is unstable 383
6.10 Example: Nash equilibrium is stable but closed loop is unstable 384
6.11 Example: Nash equilibrium is stable and the closed loop is stable 385
6.12 Example: Stability and offset in the distributed target calculation 395

6.13 Assumption: Constrained two-player game 400
6.14 Lemma: Global asymptotic stability and exponential convergence of perturbed system 408
6.15 Assumption: Disturbance models 409
6.16 Lemma: Detectability of distributed disturbance model . . 409
6.17 Assumption: Constrained M-player game 413
6.18 Lemma: Distributed gradient algorithm properties 418
6.19 Assumption: Basic stability assumption (distributed) 420
6.20 Proposition: Terminal constraint satisfaction 421
6.21 Theorem: Asymptotic stability 423
6.22 Example: Nonlinear distributed control 423
6.23 Lemma: Local detectability 437

7.1 Definition: Polytopic (polyhedral) partition 450
7.2 Definition: Piecewise affine function 450
7.3 Assumption: Strict convexity 451
7.4 Definition: Polar cone 453
7.5 Proposition: Farkas's lemma 453
7.6 Proposition: Optimality conditions for convex set 453
7.7 Proposition: Optimality conditions in terms of polar cone 455
7.8 Proposition: Optimality conditions for linear inequalities . 455
7.9 Proposition: Solution of $\mathbb{P}(w)$, $w \in R_x^0$ 457
7.10 Proposition: Piecewise quadratic (affine) cost (solution) . . 458
7.11 Example: Parametric QP 458
7.12 Example: Explicit optimal control 459
7.13 Proposition: Continuity of cost and solution 461
7.14 Assumption: Continuous, piecewise quadratic function . . 464
7.15 Definition: Active polytope (polyhedron) 465
7.16 Proposition: Solving $\mathbb{P}$ using $\mathbb{P}_i$ 465
7.17 Proposition: Optimality of $u_x^0(w)$ in R_x 468
7.18 Proposition: Piecewise quadratic (affine) solution 468
7.19 Proposition: Optimality conditions for parametric LP . . . 472
7.20 Proposition: Solution of $\mathbb{P}$ 475
7.21 Proposition: Piecewise affine cost and solution 475

8.1 Example: Nonlinear MPC 489
8.2 Example: Sequential approach 492
8.3 Example: Integration methods of different order 498
8.4 Example: Implicit integrators for a stiff ODE system 505
8.5 Example: Finding a fifth root with Newton-type iterations . 510

8.6 Example: Convergence rates 511
8.7 Theorem: Local contraction for Newton-type methods . . . 512
8.8 Corollary: Convergence of exact Newton's method 513
8.9 Example: Function evaluation via elementary operations . 516
8.10 Example: Implicit function representation 518
8.11 Example: Forward algorithmic differentiation 520
8.12 Example: Algorithmic differentiation in reverse mode . . . 522
8.13 Example: Sequential optimal control using CasADi from Octave 528
8.14 Theorem: KKT conditions 543
8.15 Theorem: Strong second-order sufficient conditions for optimality 545
8.16 Theorem: Tangential predictor by quadratic program . . . 545
8.17 Example: MPC with discrete actuator 577

A.1 Theorem: Schur decomposition 629
A.2 Theorem: Real Schur decomposition 630
A.3 Theorem: Bolzano-Weierstrass 632
A.4 Proposition: Convergence of monotone sequences 633
A.5 Proposition: Uniform continuity 634
A.6 Proposition: Compactness of continuous functions of compact sets 635
A.7 Proposition: Weierstrass 636
A.8 Proposition: Derivative and partial derivative 637
A.9 Proposition: Continuous partial derivatives 638
A.10 Proposition: Chain rule 638
A.11 Proposition: Mean value theorem for vector functions . . . 638
A.12 Definition: Convex set 641
A.13 Theorem: Caratheodory 641
A.14 Theorem: Separation of convex sets 642
A.15 Theorem: Separation of convex set from zero 643
A.16 Corollary: Existence of separating hyperplane 643
A.17 Definition: Support hyperplane 644
A.18 Theorem: Convex set and halfspaces 644
A.19 Definition: Convex cone 644
A.20 Definition: Polar cone 644
A.21 Definition: Cone generator 645
A.22 Proposition: Cone and polar cone generator 645
A.23 Theorem: Convexity implies continuity 647
A.24 Theorem: Differentiability and convexity 647

A.25 Theorem: Second derivative and convexity 647
A.26 Definition: Level set 648
A.27 Definition: Sublevel set 648
A.28 Definition: Support function 648
A.29 Proposition: Set membership and support function 648
A.30 Proposition: Lipschitz continuity of support function . . . 648
A.31 Theorem: Existence of solution to differential equations . 651
A.32 Theorem: Maximal interval of existence 651
A.33 Theorem: Continuity of solution to differential equation . 651
A.34 Theorem: Bellman-Gronwall 651
A.35 Theorem: Existence of solutions to forced systems 653
A.36 Example: Fourier transform of the normal density. 659
A.37 Definition: Density of a singular normal 662
A.38 Example: Marginal normal density 663
A.39 Example: Nonlinear transformation 666
A.40 Example: Maximum of two random variables 667
A.41 Example: Independent implies uncorrelated 668
A.42 Example: Does uncorrelated imply independent? 669
A.43 Example: Independent and uncorrelated are equivalent for normals 671
A.44 Example: Conditional normal density 674
A.45 Example: More normal conditional densities 675

B.1 Definition: Equilibrium point 694
B.2 Definition: Positive invariant set 694
B.3 Definition: $\mathcal{K}$, $\mathcal{K}_\infty$, $\mathcal{KL}$, and $\mathcal{PD}$ functions 695
B.4 Definition: Local stability 696
B.5 Definition: Global attraction 697
B.6 Definition: Global asymptotic stability 697
B.7 Definition: Various forms of stability 698
B.8 Definition: Global asymptotic stability (KL version) 699
B.9 Proposition: Connection of classical and KL global asymptotic stability 699
B.10 Definition: Various forms of stability (constrained) 699
B.11 Definition: Asymptotic stability (constrained, KL version) . 700
B.12 Definition: Lyapunov function (unconstrained and constrained) 701
B.13 Theorem: Lyapunov function and GAS (classical definition) 702
B.14 Lemma: From $\mathcal{PD}$ to $\mathcal{K}_\infty$ function (Jiang and Wang (2002)) 703

B.15 Theorem: Lyapunov function and global asymptotic stability (KL definition) 703
B.16 Proposition: Improving convergence (Sontag (1998b)) . . . 705
B.17 Theorem: Converse theorem for global asymptotic stability 705
B.18 Theorem: Lyapunov function for asymptotic stability (constrained) 706
B.19 Theorem: Lyapunov function for exponential stability . . . 706
B.20 Lemma: Lyapunov function for linear systems 707
B.21 Definition: Sequential positive invariance 707
B.22 Definition: Asymptotic stability (time-varying, constrained) 707
B.23 Definition: Lyapunov function: time-varying, constrained case 708
B.24 Theorem: Lyapunov theorem for asymptotic stability (time-varying, constrained) 708
B.25 Proposition: Global K function overbound 709
B.26 Definition: Nominal robust global asymptotic stability . . . 710
B.27 Theorem: Nominal robust global asymptotic stability and Lyapunov function 710
B.28 Definition: Positive invariance with disturbances 711
B.29 Definition: Local stability (disturbances) 712
B.30 Definition: Global attraction (disturbances) 712
B.31 Definition: GAS (disturbances) 712
B.32 Definition: Lyapunov function (disturbances) 712
B.33 Theorem: Lyapunov function for global asymptotic stability (disturbances) 713
B.34 Definition: Global control Lyapunov function (CLF) 714
B.35 Definition: Global stabilizability 714
B.36 Definition: Positive invariance (disturbance and control) . . 715
B.37 Definition: CLF (disturbance and control) 715
B.38 Definition: Control invariance (constrained) 715
B.39 Definition: CLF (constrained) 716
B.40 Definition: Control invariance (disturbances, constrained) 716
B.41 Definition: CLF (disturbances, constrained) 716
B.42 Definition: Input-to-state stable (ISS) 717
B.43 Definition: ISS-Lyapunov function 718
B.44 Lemma: ISS-Lyapunov function implies ISS 718
B.45 Definition: ISS (constrained) 718
B.46 Definition: ISS-Lyapunov function (constrained) 718
B.47 Lemma: ISS-Lyapunov function implies ISS (constrained) . 719
B.48 Definition: Output-to-state stable (OSS) 720

B.49 Definition: OSS-Lyapunov function 720
B.50 Theorem: OSS and OSS-Lyapunov function 720
B.51 Definition: Input/output-to-state stable (IOSS) 721
B.52 Definition: IOSS-Lyapunov function 721
B.53 Theorem: Modified IOSS-Lyapunov function 721
B.54 Conjecture: IOSS and IOSS-Lyapunov function 722
B.55 Definition: Incremental input/output-to-state stable 722
B.56 Definition: Observability 722
B.57 Assumption: Lipschitz continuity of model 723
B.58 Lemma: Lipschitz continuity and state difference bound . 723
B.59 Theorem: Observability and convergence of state 723

C.1 Lemma: Principle of optimality 734
C.2 Theorem: Optimal value function and control law from DP 734
C.3 Example: DP applied to linear quadratic regulator 736
C.4 Definition: Tangent vector 739
C.5 Proposition: Tangent vectors are closed cone 739
C.6 Definition: Regular normal 739
C.7 Proposition: Relation of normal and tangent cones 740
C.8 Proposition: Global optimality for convex problems 741
C.9 Proposition: Optimality conditions—normal cone 742
C.10 Proposition: Optimality conditions—tangent cone 743
C.11 Proposition: Representation of tangent and normal cones . 743
C.12 Proposition: Optimality conditions—linear inequalities . . 744
C.13 Corollary: Optimality conditions—linear inequalities 744
C.14 Proposition: Necessary condition for nonconvex problem . 746
C.15 Definition: General normal 748
C.16 Definition: General tangent 748
C.17 Proposition: Set of regular tangents is closed convex cone 748
C.18 Definition: Regular set 749
C.19 Proposition: Conditions for regular set 749
C.20 Proposition: Quasiregular set 751
C.21 Proposition: Optimality conditions nonconvex problem . . 752
C.22 Proposition: Fritz-John necessary conditions 753
C.23 Definition: Outer semicontinuous function 757
C.24 Definition: Inner semicontinuous function 758
C.25 Definition: Continuous function 758
C.26 Theorem: Equivalent conditions for outer and inner semi-continuity 759
C.27 Proposition: Outer semicontinuity and closed graph 759

C.28 Theorem: Minimum theorem 760
C.29 Theorem: Lipschitz continuity of the value function, constant U 761
C.30 Definition: Subgradient of convex function 762
C.31 Theorem: Clarke et al. (1998) 762
C.32 Corollary: A bound on $d(u, U(x'))$ for $u \in U(x)$ 763
C.33 Theorem: Continuity of $U(\cdot)$ 765
C.34 Theorem: Continuity of the value function 765
C.35 Theorem: Lipschitz continuity of the value function—$U(x)$ 766

Notation

Mathematical notation

$\exists$	there exists
$\in$	is an element of
$\forall$	for all
$\Longrightarrow \quad \Longleftarrow$	implies; is implied by
$\not\Longrightarrow \quad \not\Longleftarrow$	does not imply; is not implied by
$a := b$	a is defined to be equal to b.
$a =: b$	b is defined to be equal to a.
$\approx$	approximately equal
$V(\cdot)$	function V
$V : \mathbb{A} \to \mathbb{B}$	V is a function mapping set $\mathbb{A}$ into set $\mathbb{B}$
$x \mapsto V(x)$	function V maps variable x to value $V(x)$
x^+	value of x at next sample time (discrete time system)
$\dot{x}$	time derivative of x (continuous time system)
f_x	partial derivative of $f(x)$ with respect to x
∇	nabla or del operator
δ	unit impulse or delta function
$\lvert x \rvert$	absolute value of scalar; norm of vector (two-norm unless stated otherwise); induced norm of matrix
$\mathbf{x}$	sequence of vector-valued variable x, $(x(0), x(1), \ldots)$
$\lVert \mathbf{x} \rVert$	sup norm over a sequence, $\sup_{i \geq 0} \lvert x(i) \rvert$
$\lVert \mathbf{x} \rVert_{a:b}$	$\max_{a \leq i \leq b} \lvert x(i) \rvert$
$\mathrm{tr}(A)$	trace of matrix A
$\det(A)$	determinant of matrix A
$\mathrm{eig}(A)$	set of eigenvalues of matrix A
$\rho(A)$	spectral radius of matrix A, $\max_i \lvert \lambda_i \rvert$ for $\lambda_i \in \mathrm{eig}(A)$
A^{-1}	inverse of matrix A
$A^\dagger$	pseudo-inverse of matrix A
A'	transpose of matrix A
inf	infimum or greatest lower bound
min	minimum
sup	supremum or least upper bound
max	maximum

arg	argument or solution of an optimization
s.t.	subject to
$\mathbb{I}$	integers
$\mathbb{I}_{\geq 0}$	nonnegative integers
$\mathbb{I}_{n:m}$	integers in the interval $[n, m]$
$\mathbb{R}$	real numbers
$\mathbb{R}_{\geq 0}$	nonnegative real numbers
$\mathbb{R}^n$	real-valued n-vectors
$\mathbb{R}^{m\times n}$	real-valued $m \times n$ matrices
$\mathbb{C}$	complex numbers
$\mathcal{B}$	ball in $\mathbb{R}^n$ of unit radius
$x \sim p_x$	random variable x has probability density p_x
$\mathcal{E}(x)$	expectation of random variable x
$\mathrm{var}(x)$	variance of random variable x
$\mathrm{cov}(x, y)$	covariance of random variables x and y
$N(m, P)$	normal distribution (mean m, covariance P), $x \sim N(m, P)$
$n(x, m, P)$	normal probability density, $p_x(x) = n(x, m, P)$
$\emptyset$	the empty set
$\mathrm{aff}(\mathbb{A})$	affine hull of set $\mathbb{A}$
$\mathrm{int}(\mathbb{A})$	interior of set $\mathbb{A}$
$\mathrm{co}(\mathbb{A})$	convex hull of the set $\mathbb{A}$
$\overline{\mathbb{A}}$	closure of set $\mathbb{A}$
$\mathrm{lev}_a V$	sublevel set of function V, $\{x \mid V(x) \leq a\}$
$f \circ g$	composition of functions f and g, $f \circ g\ (s) := f(g(s))$
$a \oplus b$	maximum of scalars a and b, Chapter 4
$\bigoplus_{i=1}^n a_i$	$a_1 \oplus a_2 \oplus \cdots \oplus a_n$, Chapter 4
$\mathbb{A} \oplus \mathbb{B}$	set addition of sets $\mathbb{A}$ and $\mathbb{B}$, Chapters 3 and 5
$\mathbb{A} \ominus \mathbb{B}$	set subtraction of set $\mathbb{B}$ from set $\mathbb{A}$
$\mathbb{A} \setminus \mathbb{B}$	elements of set $\mathbb{A}$ not in set $\mathbb{B}$
$\mathbb{A} \cup \mathbb{B}$	union of sets $\mathbb{A}$ and $\mathbb{B}$
$\mathbb{A} \cap \mathbb{B}$	intersection of sets $\mathbb{A}$ and $\mathbb{B}$
$\mathbb{A} \subseteq \mathbb{B}$	set $\mathbb{A}$ is a subset of set $\mathbb{B}$
$\mathbb{A} \supseteq \mathbb{B}$	set $\mathbb{A}$ is a superset of set $\mathbb{B}$
$\mathbb{A} \subset \mathbb{B}$	set $\mathbb{A}$ is a proper (or strict) subset of set $\mathbb{B}$
$\mathbb{A} \supset \mathbb{B}$	set $\mathbb{A}$ is a proper (or strict) superset of set $\mathbb{B}$
$d(a, \mathbb{B})$	Distance between element a and set $\mathbb{B}$
$d_H(\mathbb{A}, \mathbb{B})$	Hausdorff distance between sets $\mathbb{A}$ and $\mathbb{B}$
$x \searrow y$ ($x \nearrow y$)	x converges to y from above (below)
$\mathrm{sat}(x)$	saturation, $\mathrm{sat}(x) = x$ if $\lvert x\rvert \leq 1, -1$ if $x < -1, 1$ if $x > 1$

Symbols

A, B, C	system matrices, discrete time, $x^+ = Ax + Bu$, $y = Cx$
A_c, B_c	system matrices, continuous time, $\dot{x} = A_c x + B_c u$
A_{ij}	state transition matrix for player i to player j
A_i	state transition matrix for player i
A_{Li}	estimate error transition matrix $A_i - L_i C_i$
B_d	input disturbance matrix
B_{ij}	input matrix of player i for player j's inputs
B_i	input matrix of player i
C_{ij}	output matrix of player i for player j's interaction states
C_i	output matrix of player i
C_d	output disturbance matrix
$\mathcal{C}$	controllability matrix
C^*	polar cone of cone C
d	integrating disturbance
E, F	constraint matrices, $Fx + Eu \leq e$
f, h	system functions, discrete time, $x^+ = f(x,u)$, $y = h(x)$
$f_c(x,u)$	system function, continuous time, $\dot{x} = f_c(x,u)$
$F(x,u)$	difference inclusion, $x^+ \in F(x,u)$, F is set valued
G	input noise-shaping matrix
G_{ij}	steady-state gain of player i to player j
H	controlled variable matrix
$I(x,u)$	index set of constraints active at (x,u)
$I^0(x)$	index set of constraints active at $(x, u^0(x))$
k	sample time
K	optimal controller gain
$\ell(x,u)$	stage cost
$\ell_N(x,u)$	final stage cost
L	optimal estimator gain
m	input dimension
M	cross-term penalty matrix $x'Mu$
M	number of players, Chapter 6
$\mathcal{M}$	class of admissible input policies, $\boldsymbol{\mu} \in \mathcal{M}$
n	state dimension
N	horizon length
$\mathcal{O}$	observability matrix, Chapters 1 and 4
$\mathcal{O}$	compact robust control invariant set containing the origin, Chapter 3
p	output dimension

p	optimization iterate, Chapter 6
p_ξ	probability density of random variable ξ
$p_s(x)$	sampled probability density, $p_s(x) = \sum_i w_i \delta(x - x_i)$
P	covariance matrix in the estimator
P_f	terminal penalty matrix
$\mathcal{P}$	polytopic partition, Chapter 3
$\mathcal{P}$	polytopic partition, Chapter 7
$\mathbb{P}_N(x)$	MPC optimization problem; horizon N and initial state x
q	importance function in importance sampling
Q	state penalty matrix
r	controlled variable, $r = Hy$
R	input penalty matrix
s	number of samples in a sampled probability density
S	input rate of change penalty matrix
$S(x,u)$	index set of active polytopes at (x,u)
$S^0(x)$	index set of active polytopes at $(x,u^0(x))$
t	time
T	current time in estimation problem
u	input (manipulated variable) vector
$\tilde{\mathbf{u}}^+$	warm start for input sequence
$\mathbf{u}^+$	improved input sequence
$\mathcal{U}_N(x)$	control constraint set
$\mathbb{U}$	input constraint set
v	output disturbance, Chapters 1 and 4
v	nominal control input, Chapters 3 and 5
$V_N(x,\mathbf{u})$	MPC objective function
$V_N^0(x)$	MPC optimal value function
$V_T(\chi,\boldsymbol{\omega})$	Full information state estimation objective function at time T with initial state χ and disturbance sequence $\boldsymbol{\omega}$
$\hat{V}_T(\chi,\boldsymbol{\omega})$	MHE objective function at time T with initial state χ and disturbance sequence $\boldsymbol{\omega}$
$V_f(x)$	terminal penalty
$\mathcal{V}_N(z)$	nominal control input constraint set
$\mathbb{V}$	output disturbance constraint set
w	disturbance to the state evolution
w_i	weights in a sampled probability density, Chapter 4
w_i	convex weight for player i, Chapter 6
$\overline{w}_i$	normalized weights in a sampled probability density
$\mathcal{W}$	class of admissible disturbance sequences, $\mathbf{w} \in \mathcal{W}$

$\mathbb{W}$	state disturbance constraint set
x	state vector
x_i	sample values in a sampled probability density
x_{ij}	state interaction vector from player i to player j
$\overline{x}(0)$	mean of initial state density
$X(k; x, \boldsymbol{\mu})$	state tube at time k with initial state x and control policy $\boldsymbol{\mu}$
$\mathcal{X}_j$	set of feasible states for optimal control problem at stage j
$\mathbb{X}$	state constraint set
$\mathbb{X}_f$	terminal region
y	output (measurement) vector
$\mathbb{Y}$	output constraint set
z	nominal state, Chapters 3 and 5
$Z_T(x)$	full information arrival cost
$\hat{Z}_T(x)$	MHE arrival cost
$\widetilde{Z}_T(x)$	MHE smoothing arrival cost
$\mathbb{Z}$	system constraint region, $(x, u) \in \mathbb{Z}$
$\mathbb{Z}_f$	terminal constraint region, $(x, u) \in \mathbb{Z}_f$
$\mathbb{Z}_N(x, \mathbf{u})$	constraint set for state and input sequence

Greek letters

$\Gamma_T(\chi)$	MHE prior weighting on state at time T
Δ	sample time
κ	control law
κ_j	control law at stage j
κ_f	control law applied in terminal region $\mathbb{X}_f$
$\mu_i(x)$	control law at stage i
$\boldsymbol{\mu}(x)$	control policy or sequence of control laws
ν	output disturbance decision variable in estimation problem
Π	cost-to-go matrix in regulator, Chapter 1
Π	covariance matrix in the estimator, Chapter 5
ρ_i	objective function weight for player i
Σ_i	Solution to Lyapunov equation for player i
$\phi(k; x, \mathbf{u})$	state at time k given initial state x and input sequence $\mathbf{u}$
$\phi(k; x, i, \mathbf{u})$	state at time k given state at time i is x and input sequence $\mathbf{u}$
$\phi(k; x, \mathbf{u}, \mathbf{w})$	state at time k given initial state is x, input sequence is $\mathbf{u}$, and disturbance sequence is $\mathbf{w}$
χ	state decision variable in estimation problem
ω	state disturbance decision variable in estimation problem

Subscripts, superscripts, and accents

$\hat{x}$	estimate
$\hat{x}^-$	estimate before measurement
$\tilde{x}$	estimate error
x_s	steady state
x_i	subsystem i in a decomposed large-scale system
x_{sp}	setpoint
V^0	optimal
V^{uc}	unconstrained
V^{sp}	unreachable setpoint

Acronyms

AD	algorithmic (or automatic) differentiation
AS	asymptotically stable
BFGS	Broyden-Fletcher-Goldfarb-Shanno
CIA	combinatorial integral approximation
CLF	control Lyapunov function
DAE	differential algebraic equation
DARE	discrete algebraic Riccati equation
DDP	differential dynamic programming
DP	dynamic programming
END	external numerical differentiation
FIE	full information estimation
FLOP	floating point operation
FSO	final-state observable
GAS	globally asymptotically stable
GES	globally exponentially stable
GL	Gauss-Legendre
GPC	generalized predictive control
EKF	extended Kalman filter
i-IOSS	incrementally input/output-to-state stable
IND	internal numerical differentiation
i-OSS	incrementally output-to-state stable
IOSS	input/output-to-state stable
IP	interior point
ISS	input-to-state stable
i-UIOSS	incrementally uniformly input/output-to-state stable
KF	Kalman filter
KKT	Karush-Kuhn-Tucker
LAR	linear absolute regulator
LICQ	linear independence constraint qualification
LP	linear program
LQ	linear quadratic
LQG	linear quadratic Gaussian
LQP	linear quadratic problem
LQR	linear quadratic regulator

MHE	moving horizon estimation
MILP	mixed-integer linear program
MINLP	mixed-integer nonlinear program
MIQP	mixed-integer quadratic program
NLP	nonlinear program
MPC	model predictive control
OCP	optimal control problem
ODE	ordinary differential equation
OSS	output-to-state stable
PID	proportional-integral-derivative
QP	quadratic program
RGA	relative gain array
RAS	robustly asymptotically stable
RGAS	robustly globally asymptotically stable
RGES	robustly globally exponentially stable
RHC	receding horizon control
RK	Runge-Kutta
SQP	sequential quadratic programming
SVD	singular-value decomposition
UKF	unscented Kalman filter

1

Getting Started with Model Predictive Control

1.1 Introduction

The main purpose of this chapter is to provide a compact and accessible overview of the essential elements of model predictive control (MPC). We introduce deterministic and stochastic models, regulation, state estimation, dynamic programming (DP), tracking, disturbances, and some important performance properties such as closed-loop stability and zero offset to disturbances. The reader with background in MPC and linear systems theory may wish to skim this chapter briefly and proceed to Chapter 2. Other introductory texts covering the basics of MPC include Maciejowski (2002); Camacho and Bordons (2004); Rossiter (2004); Goodwin, Serón, and De Doná (2005); Kwon (2005); Wang (2009).

1.2 Models and Modeling

Model predictive control has its roots in optimal control. The basic concept of MPC is to use a dynamic model to forecast system behavior, and optimize the forecast to produce the best decision—the control move at the current time. Models are therefore central to every form of MPC. Because the optimal control move depends on the initial state of the dynamic system, a second basic concept in MPC is to use the past record of measurements to determine the most likely initial state of the system. The state estimation problem is to examine the record of past data, and reconcile these measurements with the model to determine the most likely value of the state at the current time. Both the regulation problem, in which a model forecast is used to produce the optimal control action, and the estimation problem, in which the past record

of measurements is used to produce an optimal state estimate, involve dynamic models and optimization.

We first discuss the dynamic models used in this text. We start with the familiar differential equation models

$$\begin{aligned} \frac{dx}{dt} &= f(x,u,t) \\ y &= h(x,u,t) \\ x(t_0) &= x_0 \end{aligned}$$

in which $x \in \mathbb{R}^n$ is the state, $u \in \mathbb{R}^m$ is the input, $y \in \mathbb{R}^p$ is the output, and $t \in \mathbb{R}$ is time. We use $\mathbb{R}^n$ to denote the set of real-valued n-vectors. The initial condition specifies the value of the state x at time $t = t_0$, and we seek a solution to the differential equation for time greater than t_0, $t \in \mathbb{R}_{\geq t_0}$. Often we define the initial time to be zero, with a corresponding initial condition, in which case $t \in \mathbb{R}_{\geq 0}$.

1.2.1 Linear Dynamic Models

Time-varying model. The most general *linear* state space model is the time-varying model

$$\begin{aligned} \frac{dx}{dt} &= A(t)x + B(t)u \\ y &= C(t)x + D(t)u \\ x(0) &= x_0 \end{aligned}$$

in which $A(t) \in \mathbb{R}^{n\times n}$ is the state transition matrix, $B(t) \in \mathbb{R}^{n\times m}$ is the input matrix, $C(t) \in \mathbb{R}^{p\times n}$ is the output matrix, and $D(t) \in \mathbb{R}^{p\times m}$ allows a direct coupling between u and y. In many applications $D = 0$.

Time-invariant model. If A, B, C, and D are time invariant, the linear model reduces to

$$\begin{aligned} \frac{dx}{dt} &= Ax + Bu \\ y &= Cx + Du \\ x(0) &= x_0 \end{aligned} \tag{1.1}$$

One of the main motivations for using linear models to approximate physical systems is the ease of solution and analysis of linear models. Equation (1.1) can be solved to yield

$$x(t) = e^{At}x_0 + \int_0^t e^{A(t-\tau)}Bu(\tau)d\tau \tag{1.2}$$

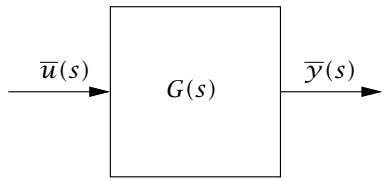


Figure 1.1: System with input $\overline{u}$, output $\overline{y}$, and transfer function matrix G connecting them; the model is $\overline{y} = G\overline{u}$.

in which $e^{At} \in \mathbb{R}^{n\times n}$ is the matrix exponential.[1] Notice the solution is a convolution integral of the entire $u(t)$ behavior weighted by the matrix exponential of At. We will see later that the eigenvalues of A determine whether the past $u(t)$ has more effect or less effect on the current $x(t)$ as time increases.

1.2.2 Input-Output Models

If we know little about the internal structure of a system, it may be convenient to take another approach in which we suppress the state variable, and focus attention only on the manipulatable inputs and measurable outputs. As shown in Figure 1.1, we consider the system to be the connection between u and y. In this viewpoint, we usually perform system identification experiments in which we manipulate u and measure y, and develop simple linear models for G. To take advantage of the usual block diagram manipulation of simple series and feedback connections, it is convenient to consider the Laplace transform of the signals rather than the time functions

$$\overline{y}(s) := \int_0^\infty e^{-st} y(t)\,dt$$

in which $s \in \mathbb{C}$ is the complex-valued Laplace transform variable, in contrast to t, which is the real-valued time variable. The symbol := means "equal by definition" or "is defined by." The transfer function matrix is then identified from the data, and the block diagram represents the

[1] We can define the exponential of matrix X in terms of its Taylor series

$$e^X := \frac{1}{0!}I + \frac{1}{1!}X + \frac{1}{2!}X^2 + \frac{1}{3!}X^3 + \cdots$$

This series converges for all X.

following mathematical relationship between input and output

$$\overline{y}(s) = G(s)\overline{u}(s)$$

$G(s) \in \mathbb{C}^{p\times m}$ is the transfer function matrix. Notice the state does not appear in this input-output description. If we are obtaining $G(s)$ instead from a state space model, then $G(s) = C(sI - A)^{-1}B + D$, and we assume $x(0) = 0$ as the system initial condition.

1.2.3 Distributed Models

Distributed models arise whenever we consider systems that are not spatially uniform. Consider, for example, a multicomponent, chemical mixture undergoing convection and chemical reaction. The microscopic mass balance for species A is

$$\frac{\partial c_A}{\partial t} + \nabla \cdot (c_A v_A) - R_A = 0$$

in which c_A is the molar concentration of species A, v_A is the velocity of species A, and R_A is the production rate of species A due to chemical reaction, in which

$$\nabla \coloneqq \delta_x \frac{\partial}{\partial x} + \delta_y \frac{\partial}{\partial y} + \delta_z \frac{\partial}{\partial z}$$

and the $\delta_{x,y,z}$ are the respective unit vectors in the (x, y, z) spatial coordinates.

We also should note that the distribution does not have to be "spatial." Consider a particle size distribution $f(r,t)$ in which $f(r,t)dr$ represents the number of particles of size r to $r + dr$ in a particle reactor at time t. The reactor volume is considered well mixed and spatially homogeneous. If the particles nucleate at zero size with nucleation rate $B(t)$ and grow with growth rate, $G(t)$, the evolution of the particle size distribution is given by

$$\begin{aligned}
\frac{\partial f}{\partial t} &= -G\frac{\partial f}{\partial r} \\
f(r,t) &= B/G \qquad r = 0 \qquad t \geq 0 \\
f(r,t) &= f_0(r) \qquad r \geq 0 \qquad t = 0
\end{aligned}$$

Again we have partial differential equation descriptions even though the particle reactor is well mixed and spatially uniform.

1.2.4 Discrete Time Models

Discrete time models are often convenient if the system of interest is sampled at discrete times. If the sampling rate is chosen appropriately, the behavior between the samples can be safely ignored and the model describes exclusively the behavior at the sample times. The finite dimensional, linear, time-invariant, discrete time model is

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k) + Du(k) \\ x(0) &= x_0 \end{aligned} \tag{1.3}$$

in which $k \in \mathbb{I}_{\geq 0}$ is a nonnegative integer denoting the sample number, which is connected to time by $t = k\Delta$ in which Δ is the sample time. We use $\mathbb{I}$ to denote the set of integers and $\mathbb{I}_{\geq 0}$ to denote the set of nonnegative integers. The linear discrete time model is a linear difference equation.

It is sometimes convenient to write the time index with a subscript

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k \\ y_k &= Cx_k + Du_k \\ x_0 &\quad \text{given} \end{aligned}$$

but we avoid this notation in this text. To reduce the notational complexity we usually express (1.3) as

$$\begin{aligned} x^+ &= Ax + Bu \\ y &= Cx + Du \\ x(0) &= x_0 \end{aligned}$$

in which the superscript $^+$ means the state at the next sample time. The linear discrete time model is convenient for presenting the ideas and concepts of MPC in the simplest possible mathematical setting. Because the model is linear, analytical solutions are readily derived. The solution to (1.3) is

$$x(k) = A^k x_0 + \sum_{j=0}^{k-1} A^{k-j-1} Bu(j) \tag{1.4}$$

Notice that a convolution sum corresponds to the convolution integral of (1.2) and powers of A correspond to the matrix exponential. Because (1.4) involves only multiplication and addition, it is convenient to program for computation.

The discrete time analog of the continuous time input-output model is obtained by defining the Z-transform of the signals

$$\overline{y}(z) := \sum_{k=0}^{\infty} z^k y(k)$$

The discrete transfer function matrix $G(z)$ then represents the discrete input-output model

$$\overline{y}(z) = G(z)\overline{u}(z)$$

and $G(z) \in \mathbb{C}^{p\times m}$ is the transfer function matrix. Notice the state does not appear in this input-output description. We make only passing reference to transfer function models in this text.

1.2.5 Constraints

The manipulated inputs (valve positions, voltages, torques, etc.) to most physical systems are bounded. We include these constraints by linear inequalities

$$Eu(k) \le e \qquad k \in \mathbb{I}_{\ge 0}$$

in which

$$E = \begin{bmatrix} I \\ -I \end{bmatrix} \qquad e = \begin{bmatrix} \overline{u} \\ -\underline{u} \end{bmatrix}$$

are chosen to describe simple bounds such as

$$\underline{u} \le u(k) \le \overline{u} \qquad k \in \mathbb{I}_{\ge 0}$$

We sometimes wish to impose constraints on states or outputs for reasons of safety, operability, product quality, etc. These can be stated as

$$Fx(k) \le f \qquad k \in \mathbb{I}_{\ge 0}$$

Practitioners find it convenient in some applications to limit the rate of change of the input, $u(k) - u(k-1)$. To maintain the state space form of the model, we may augment the state as

$$\widetilde{x}(k) = \begin{bmatrix} x(k) \\ u(k-1) \end{bmatrix}$$

and the augmented system model becomes

$$\widetilde{x}^+ = \widetilde{A}\widetilde{x} + \widetilde{B}u$$
$$y = \widetilde{C}\widetilde{x}$$

in which

$$\tilde{A} = \begin{bmatrix} A & 0 \\ 0 & 0 \end{bmatrix} \qquad \tilde{B} = \begin{bmatrix} B \\ I \end{bmatrix} \qquad \tilde{C} = \begin{bmatrix} C & 0 \end{bmatrix}$$

A rate of change constraint such as

$$\underline{\Delta} \leq u(k) - u(k-1) \leq \overline{\Delta} \qquad k \in \mathbb{I}_{\geq 0}$$

is then stated as

$$F\tilde{x}(k) + Eu(k) \leq e \qquad F = \begin{bmatrix} 0 & -I \\ 0 & I \end{bmatrix} \qquad E = \begin{bmatrix} I \\ -I \end{bmatrix} \qquad e = \begin{bmatrix} \overline{\Delta} \\ -\underline{\Delta} \end{bmatrix}$$

To simplify analysis, it pays to maintain linear constraints when using linear dynamic models. So if we want to consider fairly general constraints for a linear system, we choose the form

$$Fx(k) + Eu(k) \leq e \qquad k \in \mathbb{I}_{\geq 0}$$

which subsumes all the forms listed previously.

When we consider nonlinear systems, analysis of the controller is not significantly simplified by maintaining linear inequalities, and we generalize the constraints to set membership

$$x(k) \in \mathbb{X} \qquad u(k) \in \mathbb{U} \qquad k \in \mathbb{I}_{\geq 0}$$

or, more generally

$$(x(k), u(k)) \in \mathbb{Z} \qquad k \in \mathbb{I}_{\geq 0}$$

We should bear in mind one general distinction between input constraints, and output or state constraints. The input constraints often represent *physical limits*. In these cases, if the controller does not respect the input constraints, the physical system enforces them. In contrast, the output or state constraints are usually *desirables*. They may not be achievable depending on the disturbances affecting the system. It is often the function of an MPC controller to determine in real time that the output or state constraints are not achievable, and relax them in some satisfactory manner. As we discuss in Chapter 2, these considerations lead implementers of MPC often to set up the optimization problem using hard constraints for the input constraints and some form of soft constraints for the output or state constraints.

Soft state or output constraints. A simple formulation for soft state or output constraints is presented next. Consider a set of hard input and state constraints such as those described previously

$$Eu(k) \le e \qquad Fx(k) \le f \qquad k \in \mathbb{I}_{\ge 0}$$

To soften state constraints one introduces slack variables, $\varepsilon(k)$, which are considered decision variables, like the manipulated inputs. One then relaxes the state constraints via

$$Fx(k) \le f + \varepsilon(k) \qquad k \in \mathbb{I}_{\ge 0}$$

and adds the new "input" constraint

$$\varepsilon(k) \ge 0 \qquad k \in \mathbb{I}_{\ge 0}$$

Consider the augmented input to be $\widetilde{u}(k) = (u(k), \varepsilon(k))$, the soft state constraint formulation is then a set of mixed input-state constraints

$$\widetilde{F}x(k) + \widetilde{E}\widetilde{u}(k) \le \widetilde{e} \qquad k \ge 0$$

with

$$\widetilde{F} = \begin{bmatrix} 0 \\ 0 \\ F \end{bmatrix} \qquad \widetilde{E} = \begin{bmatrix} E & 0 \\ 0 & -I \\ 0 & -I \end{bmatrix} \qquad \widetilde{u} = \begin{bmatrix} u \\ \varepsilon \end{bmatrix} \qquad \widetilde{e} = \begin{bmatrix} e \\ 0 \\ f \end{bmatrix}$$

As we discuss subsequently, one then formulates a stage-cost penalty that weights how much one cares about the state x, the input u and the violation of the hard state constraint, which is given by ε. The hard state constraint has been replaced by a mixed state-input constraint. The benefit of this reformulation is that the state constraint cannot cause an infeasiblity in the control problem because it can be relaxed by choosing ε; large values of ε may be undesirable as measured by the stage-cost function, but they are not infeasible.

Discrete actuators and integrality constraints. In many industrial applications, a subset of the actuators or decision variables may be integer valued or discrete. A common case arises when the process has banks of similar units such as furnaces, heaters, chillers, compressors, etc., operating in parallel. In this kind of process, part of the control problem is to decide how many and which of these discrete units should be on or off during process operation to meet the setpoint or reject a disturbance. Discrete decisions also arise in many scheduling problems. In chemical production scheduling, for example, the discrete decisions can be whether or not to produce a certain chemical in a certain

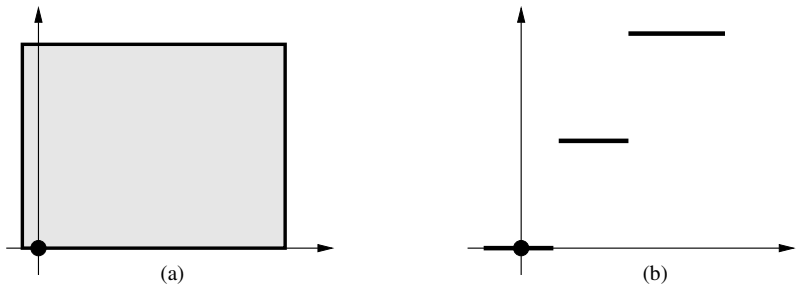


Figure 1.2: Typical input constraint sets $\mathbb{U}$ for (a) continuous actuators and (b) mixed continuous/discrete actuators. The origin (circle) represents the steady-state operating point.

reactor during the production schedule. Since these decisions are often made repeatedly as new measurement information becomes available, these (re)scheduling problems are also feedback control problems.

To define discrete-valued actuators, one may add constraints like

$$u_i(k) \in \{0,1\} \qquad i \in I_D, \quad k \in \mathbb{I}_{\geq 0}$$

in which the set $I_D \subset \{1,2,\ldots,m\}$ represents the indices of the actuators that are discrete, which are binary (on/off) decisions in the case illustrated above. Alternatively, one may use the general set membership constraint $u(k) \in \mathbb{U}$, and employ the set $\mathbb{U}$ to define the discrete actuators as shown in Figure 1.2. In the remainder of this introductory chapter we focus exclusively on continuous actuators, but return to discrete actuators in later chapters.

1.2.6 Deterministic and Stochastic

If one examines measurements coming from any complex, physical process, fluctuations in the data as depicted in Figure 1.3 are invariably present. For applications at small length scales, the fluctuations may be caused by the random behavior of small numbers of molecules. This type of application is becoming increasingly prevalent as scientists and engineers study applications in nanotechnology. This type of system also arises in life science applications when modeling the interactions of a few virus particles or protein molecules with living cells. In these applications there is no deterministic simulation model; the only system model available is stochastic.

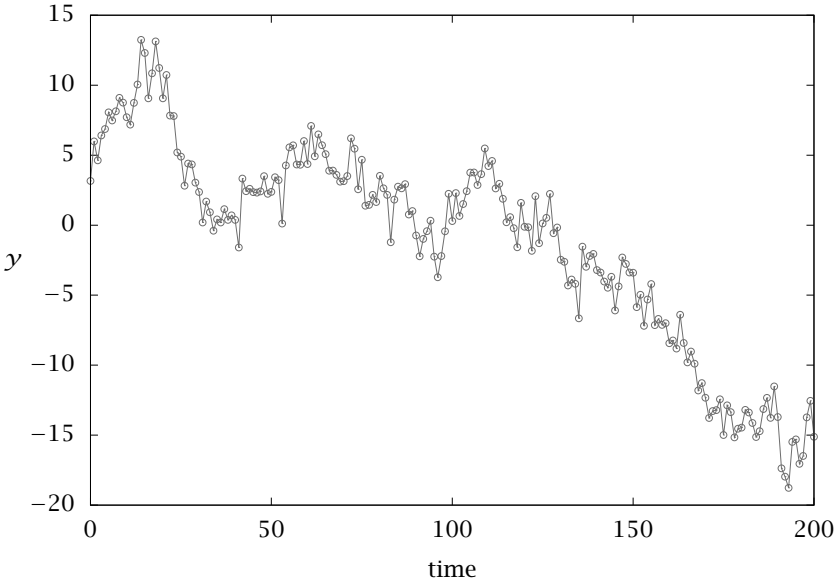


Figure 1.3: Output of a stochastic system versus time.

Linear time-invariant models. In mainstream, classical process control problems, we are usually concerned with modeling, monitoring and controlling macroscopic systems, i.e., we are not considering systems composed of small numbers of molecules. So one may naturally ask (many do) what is the motivation for stochastic models in this arena? The motivation for stochastic models is to account for the unmodeled effects of the environment (disturbances) on the system under study. If we examine the measurement from any process control system of interest, no matter how "macroscopic," we are confronted with the physical reality that the measurement still looks a lot like Figure 1.3. If it is important to model the observed measurement fluctuations, we turn to stochastic models.

Some of the observed fluctuation in the data is assignable to the measurement device. This source of fluctuation is known as measurement "noise." Some of the observed fluctuation in the data is assignable to unmodeled disturbances from the environment affecting the state of the system. The simplest stochastic model for representing these two possible sources of disturbances is a linear model with added random

variables

$$x^+ = Ax + Bu + Gw$$
$$y = Cx + Du + v$$

with initial condition $x(0) = x_0$. The variable $w \in \mathbb{R}^g$ is the random variable acting on the state transition, $v \in \mathbb{R}^p$ is a random variable acting on the measured output, and x_0 is a random variable specifying the initial state. The random variable v is used to model the measurement noise and w models the process disturbance. The matrix $G \in \mathbb{R}^{n\times g}$ allows further refinement of the modeling between the source of the disturbance and its effect on the state. Often G is chosen to be the identity matrix with $g = n$.

1.3 Introductory MPC Regulator

1.3.1 Linear Quadratic Problem

We start by designing a controller to take the state of a deterministic, linear system to the origin. If the setpoint is not the origin, or we wish to track a time-varying setpoint trajectory, we will subsequently make modifications of the zero setpoint problem to account for that. The system model is

$$x^+ = Ax + Bu$$
$$y = Cx \tag{1.5}$$

In this first problem, we assume that the state is measured, or $C = I$. We will handle the output measurement problem with state estimation in the next section. Using the model we can predict how the state evolves given any set of inputs we are considering. Consider N time steps into the future and collect the input sequence into $\mathbf{u}$

$$\mathbf{u} = (u(0), u(1), \ldots, u(N-1))$$

Constraints on the $\mathbf{u}$ sequence (i.e., valve saturations, etc.) are covered extensively in Chapter 2. The constraints are the main feature that distinguishes MPC from the standard linear quadratic (LQ) control.

We first define an objective function $V(\cdot)$ to measure the deviation of the trajectory of $x(k), u(k)$ from zero by summing the weighted squares

$$V(x(0), \mathbf{u}) = \frac{1}{2}\sum_{k=0}^{N-1}\left[x(k)'Qx(k) + u(k)'Ru(k)\right] + \frac{1}{2}x(N)'P_f x(N)$$

subject to

$$x^+ = Ax + Bu$$

The objective function depends on the input sequence and state sequence. The initial state is available from the measurement. The remainder of the state trajectory, $x(k), k = 1, \ldots, N$, is determined by the model and the input sequence $\mathbf{u}$. So we show the objective function's explicit dependence on the input sequence and initial state. The tuning parameters in the controller are the matrices Q and R. We allow the final state penalty to have a different weighting matrix, P_f, for generality. Large values of Q in comparison to R reflect the designer's intent to drive the state to the origin quickly at the expense of large control action. Penalizing the control action through large values of R relative to Q is the way to reduce the control action and slow down the rate at which the state approaches the origin. Choosing appropriate values of Q and R (i.e., tuning) is not always obvious, and this difficulty is one of the challenges faced by industrial practitioners of LQ control. Notice that MPC inherits this tuning challenge.

We then formulate the following optimal LQ control problem

$$\min_{\mathbf{u}} V(x(0), \mathbf{u}) \tag{1.6}$$

The Q, P_f, and R matrices often are chosen to be diagonal, but we do not assume that here. We assume, however, that Q, P_f, and R are *real and symmetric*; Q and P_f are *positive semidefinite*; and R is *positive definite*. These assumptions guarantee that the solution to the optimal control problem exists and is unique.

1.3.2 Optimizing Multistage Functions

We next provide a brief introduction to methods for solving multistage optimization problems like (1.6). Consider the set of variables w, x, y, and z, and the following function to be optimized

$$f(w, \underbrace{x) + g(x,} \overbrace{y) + h(y,} z)$$

Notice that the objective function has a special structure in which each stage's cost function in the sum depends only on adjacent variable pairs. For the first version of this problem, we consider w to be a fixed parameter, and we would like to solve the problem

$$\min_{x,y,z} f(w, x) + g(x, y) + h(y, z) \qquad w \text{ fixed}$$

One option is to optimize simultaneously over all three decision variables. Because of the objective function's special structure, however, we can obtain the solution by optimizing a sequence of three single-variable problems defined as follows

$$\min_x \left[f(w,x) + \min_y \left[g(x,y) + \min_z h(y,z) \right] \right]$$

We solve the inner problem over z first, and denote the optimal value and solution as follows

$$\underline{h}^0(y) = \min_z h(y,z) \qquad \underline{z}^0(y) = \arg\min_z h(y,z)$$

Notice that the optimal z and value function for this problem are both expressed as a function of the y variable. We then move to the next optimization problem and solve for the y variable

$$\min_y g(x,y) + \underline{h}^0(y)$$

and denote the solution and value function as

$$\underline{g}^0(x) = \min_y g(x,y) + \underline{h}^0(y) \qquad \underline{y}^0(x) = \arg\min_y g(x,y) + \underline{h}^0(y)$$

The optimal solution for y is a function of x, the remaining variable to be optimized. The third and final optimization is

$$\min_x f(w,x) + \underline{g}^0(x)$$

with solution and value function

$$\underline{f}^0(w) = \min_x f(w,x) + \underline{g}^0(x) \qquad \underline{x}^0(w) = \arg\min_x f(w,x) + \underline{g}^0(x)$$

We summarize the recursion with the following annotated equation

$$\underbrace{\min_x \left[f(w,x) + \overbrace{\min_y \left[g(x,y) + \underbrace{\min_z h(y,z)}_{\underline{h}^0(y),\, \underline{z}^0(y)} \right]}^{\underline{g}^0(x),\, \underline{y}^0(x)} \right]}_{\underline{f}^0(w),\, \underline{x}^0(w)}$$

If we are mainly interested in the first variable x, then the function $\underline{x}^0(w)$ is of primary interest and we have obtained this function quite efficiently. This nested solution approach is an example of a class of

techniques known as dynamic programming (DP). DP was developed by Bellman (Bellman, 1957; Bellman and Dreyfus, 1962) as an efficient means for solving these kinds of multistage optimization problems. Bertsekas (1987) provides an overview of DP.

The version of the method we just used is called *backward* DP because we find the variables in reverse order: first z, then y, and finally x. Notice we find the optimal solutions as *functions* of the variables to be optimized at the next stage. If we wish to find the other variables y and z as a function of the known parameter w, then we nest the optimal solutions found by the backward DP recursion

$$\underset{\sim}{y}^{0}(w) = \underline{y}^{0}(\underline{x}^{0}(w)) \qquad \underset{\sim}{z}^{0}(w) = \underline{z}^{0}(\underset{\sim}{y}^{0}(w)) = \underline{z}^{0}(\underline{y}^{0}(\underline{x}^{0}(w)))$$

As we see shortly, backward DP is the method of choice for the regulator problem.

In the state estimation problem to be considered later in this chapter, w becomes a variable to be optimized, and z plays the role of a parameter. We wish to solve the problem

$$\min_{w,x,y} f(w,x) + g(x,y) + h(y,z) \qquad z \text{ fixed}$$

We can still break the problem into three smaller nested problems, but the order is reversed

$$\underbrace{\min_{y}\left[h(y,z) + \overbrace{\min_{x}\left[g(x,y) + \underbrace{\min_{w} f(w,x)}_{\overline{f}^{0}(x),\,\overline{w}^{0}(x)}\right]}^{\overline{g}^{0}(y),\,\overline{x}^{0}(y)}\right]}_{\overline{h}^{0}(z),\,\overline{y}^{0}(z)} \tag{1.7}$$

This form is called *forward* DP because we find the variables in the order given: first w, then x, and finally y. The optimal value functions and optimal solutions at each of the three stages are shown in (1.7). This version is preferable if we are primarily interested in finding the final variable y as a function of the parameter z. As before, if we need the other optimized variables x and w as a function of the parameter z, we must insert the optimal functions found by the forward DP recursion

$$\tilde{x}^{0}(z) = \overline{x}^{0}(\overline{y}^{0}(z)) \qquad \tilde{w}^{0}(z) = \overline{w}^{0}(\tilde{x}^{0}(z)) = \overline{w}^{0}(\overline{x}^{0}(\overline{y}^{0}(z)))$$

For the reader interested in trying some exercises to reinforce the concepts of DP, Exercise 1.15 considers finding the function $\tilde{w}^{0}(z)$ with

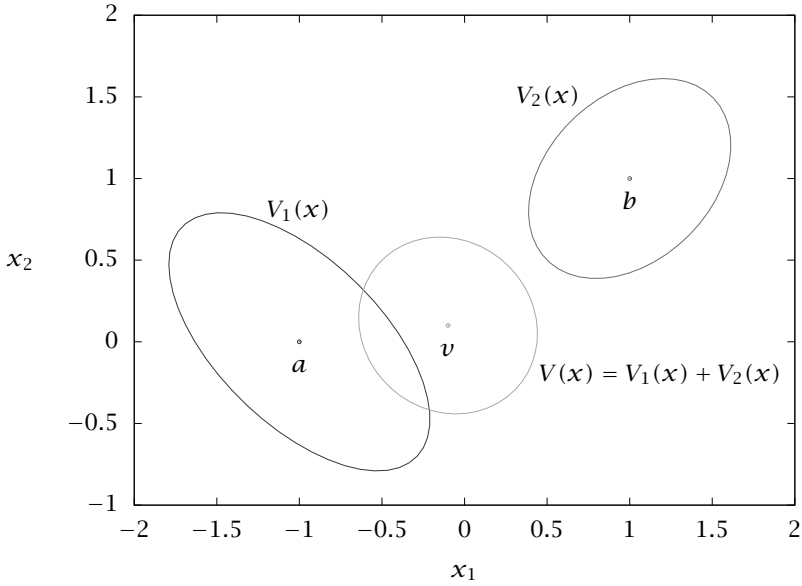


Figure 1.4: Level sets of two quadratic functions $V_1(x) = (1/4)$, $V_2(x) = (1/4)$, and their sum; $V(x) = V_1(x) + V_2(x) = 2$.

backward DP instead of forward DP as we just did here. Exercise C.1 discusses showing that the nested optimizations indeed give the same answer as simultaneous optimization over all decision variables.

Finally, if we optimize over all four variables, including the one considered as a fixed parameter in the two versions of DP we used, then we have two equivalent ways to express the value of the complete optimization

$$\min_{w,x,y,z} f(w,x) + g(x,y) + h(y,z) = \min_{w} \underline{f}^0(w) = \min_{z} \overline{h}^0(z)$$

The result in the next example proves useful in combining quadratic functions to solve the LQ problem.

Example 1.1: Sum of quadratic functions

Consider the two quadratic functions given by

$$V_1(x) = (1/2)(x-a)'A(x-a) \qquad V_2(x) = (1/2)(x-b)'B(x-b)$$

in which $A, B > 0$ are positive definite matrices and a and b are n-vectors locating the minimum of each function. Figure 1.4 displays the ellipses defined by the level sets $V_1(x) = 1/4$ and $V_2(x) = 1/4$ for the following data

$$A = \begin{bmatrix} 1.25 & 0.75 \\ 0.75 & 1.25 \end{bmatrix} \qquad a = \begin{bmatrix} -1 \\ 0 \end{bmatrix} \qquad B = \begin{bmatrix} 1.5 & -0.5 \\ -0.5 & 1.5 \end{bmatrix} \qquad b = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

(a) Show that the sum $V(x) = V_1(x) + V_2(x)$ is also quadratic

$$V(x) = (1/2)((x - v)'H(x - v) + d)$$

in which

$$H = A + B \qquad v = H^{-1}(Aa + Bb)$$
$$d = -(Aa + Bb)'H^{-1}(Aa + Bb) + a'Aa + b'Bb$$

and verify the three ellipses given in Figure 1.4.

(b) Consider a generalization useful in the discussion of the upcoming regulation and state estimation problems. Let

$$V_1(x) = (1/2)(x-a)'A(x-a) \qquad V_2(x) = (1/2)(Cx-b)'B(Cx-b)$$

Derive the formulas for H, v, d for this case.

(c) Use the matrix inversion lemma (see Exercise 1.12) and show that $V(x)$ of part (b) can be expressed also in an inverse form, which is useful in state estimation problems

$$V(x) = (1/2)((x - v)'\tilde{H}^{-1}(x - v) + d)$$
$$\tilde{H} = A^{-1} - A^{-1}C'(CA^{-1}C' + B^{-1})^{-1}CA^{-1}$$
$$v = a + A^{-1}C'(CA^{-1}C' + B^{-1})^{-1}(b - Ca)$$
$$d = (b - Ca)'(CA^{-1}C' + B^{-1})^{-1}(b - Ca)$$

Solution

(a) The sum of two quadratics is also quadratic, so we parameterize the sum as

$$V(x) = (1/2)\left((x - v)'H(x - v) + d\right)$$

and solve for v, H, and d. Comparing the expansion of the quadratics of the right- and left-hand sides gives

$$x'Hx-2x'Hv+v'Hv+d = x'(A+B)x-2x'(Aa+Bb)+a'Aa+b'Bb$$

Equating terms at each order gives

$$\begin{aligned} H &= A + B \\ v &= H^{-1}(Aa + Bb) \\ d &= -v'Hv + a'Aa + b'Bb \\ &= -(Aa + Bb)'H^{-1}(Aa + Bb) + a'Aa + b'Bb \end{aligned}$$

Notice that H is positive definite since A and B are positive definite. Substituting the values of a, A, b, and B gives

$$H = \begin{bmatrix} 2.75 & 0.25 \\ 0.25 & 2.75 \end{bmatrix} \qquad v = \begin{bmatrix} -0.1 \\ 0.1 \end{bmatrix} \qquad d = 3.2$$

The level set $V(x) = 2$ is also plotted in Figure 1.4.

(b) Expanding and comparing terms as before, we obtain

$$\begin{aligned} H &= A + C'BC \\ v &= H^{-1}(Aa + C'Bb) \\ d &= -(Aa + C'Bb)'H^{-1}(Aa + C'Bb) + a'Aa + b'Bb \end{aligned} \tag{1.8}$$

Notice that H is positive definite since A is positive definite and $C'BC$ is positive semidefinite for any C.

(c) Define $\overline{x} = x - a$ and $\overline{b} = b - Ca$, and express the problem as

$$\begin{aligned} V(x) &= (1/2)\overline{x}'A\overline{x} + (1/2)(C(\overline{x} + a) - b)'B(C(\overline{x} + a) - b) \\ &= (1/2)\overline{x}'A\overline{x} + (1/2)(C\overline{x} - \overline{b})'B(C\overline{x} - \overline{b}) \end{aligned}$$

Apply the solution of part (b) to obtain

$$\begin{gathered} V(x) = (1/2)((\overline{x} - \overline{v})'H(\overline{x} - \overline{v}) + d) \\ H = A + C'BC \qquad \overline{v} = H^{-1}C'B\overline{b} \\ d = (b - Ca)'(B - BCH^{-1}C'B)^{-1}(b - Ca) \end{gathered}$$

From the matrix inversion lemma, use (1.54) on H and (1.55) on $\overline{v}$ to obtain

$$\begin{aligned} H^{-1} = \widetilde{H} &= A^{-1} - A^{-1}C'(CA^{-1}C' + B^{-1})^{-1}CA^{-1} \\ \overline{v} &= A^{-1}C'(CA^{-1}C' + B^{-1})^{-1}\overline{b} \\ d &= (b - Ca)'(CA^{-1}C' + B^{-1})^{-1}(b - Ca) \end{aligned}$$

The function $V(x)$ is then given by

$$V(x) = (1/2)((x - v)'\widetilde{H}^{-1}(x - v) + d)$$

with $v = a + A^{-1}C'(CA^{-1}C' + B^{-1})^{-1}\ (b - Ca)$. □

1.3.3 Dynamic Programming Solution

After this brief introduction to DP, we apply it to solve the LQ control problem. We first rewrite (1.6) in the following form to see the structure clearly

$$V(x(0), \mathbf{u}) = \sum_{k=0}^{N-1} \ell(x(k), u(k)) + \ell_N(x(N)) \qquad \text{s.t. } x^+ = Ax + Bu$$

in which the *stage cost* $\ell(x, u) = (1/2)(x'Qx + u'Ru), k = 0, \ldots, N-1$ and the terminal stage cost $\ell_N(x) = (1/2)x'P_f x$. Since $x(0)$ is known, we choose *backward* DP as the convenient method to solve this problem. We first rearrange the overall objective function so we can optimize over input $u(N-1)$ and state $x(N)$

$$\begin{aligned} \min_{u(0),x(1),\ldots u(N-2),x(N-1)} & \ell(x(0), u(0)) + \ell(x(1), u(1)) + \cdots + \\ & \min_{u(N-1),x(N)} \ell(x(N-1), u(N-1)) + \ell_N(x(N)) \end{aligned}$$

subject to

$$x(k+1) = Ax(k) + Bu(k) \qquad k = 0, \ldots N-1$$

The problem to be solved at the last stage is

$$\min_{u(N-1),x(N)} \ell(x(N-1), u(N-1)) + \ell_N(x(N)) \tag{1.9}$$

subject to

$$x(N) = Ax(N-1) + Bu(N-1)$$

in which $x(N-1)$ appears in this stage as a parameter. We denote the optimal cost by $V^0_{N-1}(x(N-1))$ and the optimal decision variables by $u^0_{N-1}(x(N-1))$ and $x^0_N(x(N-1))$. The optimal cost and decisions at the last stage are parameterized by the state at the previous stage as we expect in backward DP. We next solve this optimization. First we substitute the state equation for $x(N)$ and combine the two quadratic terms using (1.8)

$$\begin{aligned}
&\ell(x(N-1),u(N-1)) + \ell_N(x(N)) \\
&\quad = (1/2)\left(|x(N-1)|^2_Q + |u(N-1)|^2_R + |Ax(N-1)+Bu(N-1)|^2_{P_f}\right) \\
&\quad = (1/2)\left(|x(N-1)|^2_Q + |(u(N-1)-v)|^2_H + d\right)
\end{aligned}$$

in which

$$\begin{aligned}
H &= R + B'P_fB \\
v &= -(B'P_fB+R)^{-1}B'P_fA\,x(N-1) \\
d &= x(N-1)'\left(A'P_fA - A'P_fB(B'P_fB+R)^{-1}B'P_fA\right)x(N-1)
\end{aligned}$$

Given this form of the cost function, we see by inspection that the optimal input for $u(N-1)$ is v, so the optimal control law at stage $N-1$ is a *linear* function of the state $x(N-1)$. Then using the model equation, the optimal final state is also a linear function of state $x(N-1)$. The optimal cost is $(1/2)(|x(N-1)|^2_Q + d)$, which makes the optimal cost a quadratic function of $x(N-1)$. Summarizing, for all x

$$\begin{aligned}
u^0_{N-1}(x) &= K(N-1)\,x \\
x^0_N(x) &= (A+BK(N-1))\,x \\
V^0_{N-1}(x) &= (1/2)x'\,\Pi(N-1)\,x
\end{aligned}$$

with the definitions

$$\begin{aligned}
K(N-1) &:= -(B'P_fB+R)^{-1}B'P_fA \\
\Pi(N-1) &:= Q + A'P_fA - A'P_fB(B'P_fB+R)^{-1}B'P_fA
\end{aligned}$$

The function $V^0_{N-1}(x)$ defines the optimal *cost to go* from state x for the last stage under the optimal control law $u^0_{N-1}(x)$. Having this function allows us to move to the next stage of the DP recursion. For the next stage we solve the optimization

$$\min_{u(N-2),x(N-1)} \ell(x(N-2),u(N-2)) + V^0_{N-1}(x(N-1))$$

subject to

$$x(N-1) = Ax(N-2) + Bu(N-2)$$

Notice that this problem is identical in structure to the stage we just solved, (1.9), and we can write out the solution by simply renaming variables

$$\begin{aligned}
u^0_{N-2}(x) &= K(N-2)\,x \\
x^0_{N-1}(x) &= (A + BK(N-2))\,x \\
V^0_{N-2}(x) &= (1/2)x'\,\Pi(N-2)\,x \\
K(N-2) &:= -(B'\Pi(N-1)B + R)^{-1}B'\Pi(N-1)A \\
\Pi(N-2) &:= Q + A'\Pi(N-1)A - \\
&\qquad A'\Pi(N-1)B(B'\Pi(N-1)B + R)^{-1}B'\Pi(N-1)A
\end{aligned}$$

The recursion from $\Pi(N-1)$ to $\Pi(N-2)$ is known as a backward Riccati iteration. To summarize, the backward Riccati iteration is defined as follows

$$\Pi(k-1) = Q + A'\Pi(k)A - A'\Pi(k)B\left(B'\Pi(k)B + R\right)^{-1}B'\Pi(k)A \qquad k = N, N-1, \ldots, 1 \quad (1.10)$$

with terminal condition

$$\Pi(N) = P_f \quad (1.11)$$

The terminal condition replaces the typical initial condition because the iteration is running backward. The optimal control policy at each stage is

$$u^0_k(x) = K(k)x \qquad k = N-1, N-2, \ldots, 0 \quad (1.12)$$

The optimal gain at time k is computed from the Riccati matrix at time $k+1$

$$K(k) = -\left(B'\Pi(k+1)B + R\right)^{-1}B'\Pi(k+1)A \qquad k = N-1, N-2, \ldots, 0 \quad (1.13)$$

and the optimal cost to go from time k to time N is

$$V^0_k(x) = (1/2)x'\Pi(k)x \qquad k = N, N-1, \ldots, 0 \quad (1.14)$$

1.3.4 The Infinite Horizon LQ Problem

Let us motivate the infinite horizon problem by showing a weakness of the finite horizon problem. Kalman (1960b, p.113) pointed out in his classic 1960 paper that optimality does not ensure stability.

> In the engineering literature it is often assumed (tacitly and incorrectly) that a system with optimal control law (6.8) is necessarily stable.

Assume that we use as our control law the first feedback gain of the finite horizon problem, $K(0)$

$$u(k) = K(0)x(k)$$

Then the stability of the closed-loop system is determined by the eigenvalues of $A+BK(0)$. We now construct an example that shows choosing $Q > 0$, $R > 0$, and $N \geq 1$ does not ensure stability. In fact, we can find reasonable values of these parameters such that the controller destabilizes a stable system.[2] Let

$$A = \begin{bmatrix} 4/3 & -2/3 \\ 1 & 0 \end{bmatrix} \qquad B = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \qquad C = [-2/3 \;\; 1]$$

This system is chosen so that $G(z)$ has a *zero* at $z = 3/2$, i.e., an unstable zero. We now construct an LQ controller that inverts this zero and hence produces an unstable system. We would like to choose $Q = C'C$ so that y itself is penalized, but that Q is only semidefinite. We add a small positive definite piece to $C'C$ so that Q is positive definite, and choose a *small* positive R penalty (to encourage the controller to misbehave), and $N = 5$

$$Q = C'C + 0.001I = \begin{bmatrix} 4/9 + .001 & -2/3 \\ -2/3 & 1.001 \end{bmatrix} \qquad R = 0.001$$

We now iterate the Riccati equation four times starting from $\Pi = P_f = Q$ and compute $K(0)$ for $N = 5$; then we compute the eigenvalues of $A + BK(0)$ and achieve[3]

$$\text{eig}(A + BK_5(0)) = \{1.307, 0.001\}$$

[2] In Chapter 2, we present several controller design methods that prevent this kind of instability.

[3] Please check this answer with Octave or MATLAB.

Using this controller the closed-loop system evolution is $x(k) = (A + BK_5(0))^k x_0$. Since an eigenvalue of $A + BK_5(0)$ is greater than unity, $x(k) \to \infty$ as $k \to \infty$. In other words the closed-loop system is unstable.

If we continue to iterate the Riccati equation, which corresponds to increasing the horizon in the controller, we obtain for $N = 7$

$$\text{eig}(A + BK_7(0)) = \{0.989, 0.001\}$$

and the controller is stabilizing. If we continue iterating the Riccati equation, we converge to the following steady-state closed-loop eigenvalues

$$\text{eig}(A + BK_\infty(0)) = \{0.664, 0.001\}$$

This controller corresponds to an infinite horizon control law. Notice that it is stabilizing and has a reasonable stability margin. Nominal stability is a guaranteed property of infinite horizon controllers as we prove in the next section.

With this motivation, we are led to consider directly the infinite horizon case

$$V(x(0), \mathbf{u}) = \frac{1}{2} \sum_{k=0}^{\infty} x(k)'Qx(k) + u(k)'Ru(k) \tag{1.15}$$

in which $x(k)$ is the solution at time k of $x^+ = Ax+Bu$ if the initial state is $x(0)$ and the input sequence is $\mathbf{u}$. If we are interested in a continuous process (i.e., no final time), then the natural cost function is an infinite horizon cost. If we were truly interested in a batch process (i.e., the process does stop at $k = N$), then stability is not a relevant property, and we naturally would use the finite horizon LQ controller and the *time-varying* controller, $u(k) = K(k)x(k), k = 0, 1, \ldots, N$.

In considering the infinite horizon problem, we first restrict attention to systems for which there exist input sequences that give bounded cost. Consider the case $A = I$ and $B = 0$, for example. Regardless of the choice of input sequence, (1.15) is unbounded for $x(0) \neq 0$. It seems clear that we are not going to stabilize an unstable system ($A = I$) without any input ($B = 0$). This is an example of an *uncontrollable* system. In order to state the sharpest results on stabilization, we require the concepts of controllability, stabilizability, observability, and detectability. We shall define these concepts subsequently.

1.3.5 Controllability

A system is *controllable* if, for any pair of states x, z in the state space, z can be reached in finite time from x (or x controlled to z) (Sontag, 1998, p.83). A *linear discrete time* system $x^+ = Ax + Bu$ is therefore controllable if there exists a finite time N and a sequence of inputs

$$(u(0), u(1), \ldots u(N-1))$$

that can transfer the system from any x to any z in which

$$z = A^N x + \begin{bmatrix} B & AB & \cdots & A^{N-1}B \end{bmatrix} \begin{bmatrix} u(N-1) \\ u(N-2) \\ \vdots \\ u(0) \end{bmatrix}$$

We can simplify this condition by noting that the matrix powers A^k for $k \geq n$ are expressible as linear combinations of the powers 0 to $n-1$. This result is a consequence of the Cayley-Hamilton theorem (Horn and Johnson, 1985, pp. 86–87). Therefore the range of the matrix $\begin{bmatrix} B & AB & \cdots & A^{N-1}B \end{bmatrix}$ for $N \geq n$ is the same as $\begin{bmatrix} B & AB & \cdots & A^{n-1}B \end{bmatrix}$. In other words, for an unconstrained linear system, if we cannot reach z in n moves, we cannot reach z in any number of moves. The question of *controllability* of a linear time-invariant system is therefore a question of *existence* of solutions to linear equations for an arbitrary right-hand side

$$\begin{bmatrix} B & AB & \cdots & A^{n-1}B \end{bmatrix} \begin{bmatrix} u(n-1) \\ u(n-2) \\ \vdots \\ u(0) \end{bmatrix} = z - A^n x$$

The matrix appearing in this equation is known as the *controllability matrix* C

$$C = \begin{bmatrix} B & AB & \cdots & A^{n-1}B \end{bmatrix} \tag{1.16}$$

From the fundamental theorem of linear algebra, we know a solution exists for all right-hand sides if and only if the *rows* of the $n \times nm$ controllability matrix are linearly independent.[4] Therefore, the system (A, B) is controllable if and only if

$$\operatorname{rank}(C) = n$$

[4] See Section A.4 of Appendix A or (Strang, 1980, pp.87–88) for a review of this result.

The following result for checking controllability also proves useful (Hautus, 1972).

Lemma 1.2 (Hautus lemma for controllability)**.** *A system is controllable if and only if*

$$\text{rank}\begin{bmatrix} \lambda I - A & B \end{bmatrix} = n \qquad \text{for all } \lambda \in \mathbb{C} \tag{1.17}$$

in which $\mathbb{C}$ is the set of complex numbers.

Notice that the first n columns of the matrix in (1.17) are linearly independent if λ is not an eigenvalue of A, so (1.17) is equivalent to checking the rank at just the eigenvalues of A

$$\text{rank}\begin{bmatrix} \lambda I - A & B \end{bmatrix} = n \qquad \text{for all } \lambda \in \text{eig}(A)$$

1.3.6 Convergence of the Linear Quadratic Regulator

We now show that the infinite horizon regulator asymptotically stabilizes the origin for the closed-loop system. Define the infinite horizon objective function

$$V(x, \mathbf{u}) = \frac{1}{2} \sum_{k=0}^{\infty} x(k)'Qx(k) + u(k)'Ru(k)$$

subject to

$$\begin{aligned} x^+ &= Ax + Bu \\ x(0) &= x \end{aligned}$$

with $Q, R > 0$. If (A, B) is controllable, the solution to the optimization problem

$$\min_{\mathbf{u}} V(x, \mathbf{u})$$

exists and is unique for all x. We denote the optimal solution by $\mathbf{u}^0(x)$, and the first input in the optimal sequence by $u^0(x)$. The feedback control law $\kappa_\infty(\cdot)$ for this infinite horizon case is then defined as $u = \kappa_\infty(x)$ in which $\kappa_\infty(x) = u^0(x) = \mathbf{u}^0(0; x)$. As stated in the following lemma, this infinite horizon linear quadratic regulator (LQR) is stabilizing.

Lemma 1.3 (LQR convergence)**.** *For (A, B) controllable, the infinite horizon LQR with $Q, R > 0$ gives a convergent closed-loop system*

$$x^+ = Ax + B\kappa_\infty(x)$$

Proof. The cost of the infinite horizon objective is bounded above for all $x(0)$ because (A, B) is controllable. Controllability implies that there exists a sequence of n inputs $(u(0), u(1), \ldots, u(n-1))$ that transfers the state from any $x(0)$ to $x(n) = 0$. A zero control sequence after $k = n$ for $(u(n+1), u(n+2), \ldots)$ generates zero cost for all terms in V after $k = n$, and the objective function for this infinite control sequence is therefore finite. The cost function is strictly convex in $\mathbf{u}$ because $R > 0$ so the solution to the optimization is unique.

If we consider the sequence of costs to go along the closed-loop trajectory, we have

$$V_{k+1} = V_k - (1/2)\left(x(k)'Qx(k) + u(k)'Ru(k)\right)$$

in which $V_k = V^0(x(k))$ is the cost at time k for state value $x(k)$ and $u(k) = u^0(x(k))$ is the optimal control for state $x(k)$. The cost along the closed-loop trajectory is nonincreasing and bounded below (by zero). Therefore, the sequence (V_k) converges and

$$x(k)'Qx(k) \to 0 \qquad u(k)'Ru(k) \to 0 \qquad \text{as } k \to \infty$$

Since $Q, R > 0$, we have

$$x(k) \to 0 \qquad u(k) \to 0 \qquad \text{as } k \to \infty$$

and closed-loop convergence is established. ■

In fact we know more. From the previous sections, we know the optimal solution is found by iterating the Riccati equation, and the optimal infinite horizon control law and optimal cost are given by

$$u^0(x) = Kx \qquad V^0(x) = (1/2)x'\Pi x$$

in which

$$\begin{aligned} K &= -(B'\Pi B + R)^{-1}B'\Pi A \\ \Pi &= Q + A'\Pi A - A'\Pi B(B'\Pi B + R)^{-1}B'\Pi A \end{aligned} \tag{1.18}$$

Proving Lemma 1.3 has shown also that for (A, B) controllable and Q, $R > 0$, a positive definite solution to the discrete algebraic Riccati equation (DARE), (1.18), exists and the eigenvalues of $(A+BK)$ are asymptotically stable for the K corresponding to this solution (Bertsekas, 1987, pp.58–64).

This basic approach to establishing regulator stability will be generalized in Chapter 2 to handle constrained and nonlinear systems, so it

is helpful for the new student to first become familiar with these ideas in the unconstrained, linear setting. For linear systems, asymptotic convergence is equivalent to asymptotic stability, and we delay the discussion of stability until Chapter 2. In Chapter 2 the optimal cost is shown to be a Lyapunov function for the closed-loop system. We also can strengthen the stability for linear systems from asymptotic stability to exponential stability based on the form of the Lyapunov function.

The LQR convergence result in Lemma 1.3 is the simplest to establish, but we can enlarge the class of systems and penalties for which closed-loop stability is guaranteed. The system restriction can be weakened from controllability to *stabilizability*, which is discussed in Exercises 1.19 and 1.20. The restriction on the allowable state penalty Q can be weakened from $Q > 0$ to $Q \geq 0$ and (A, Q) *detectable*, which is also discussed in Exercise 1.20. The restriction $R > 0$ is retained to ensure uniqueness of the control law. In applications, if one cares little about the cost of the control, then R is chosen to be small, but positive definite.

1.4 Introductory State Estimation

The next topic is state estimation. In most applications, the variables that are conveniently or economically measurable (y) are a small subset of the variables required to model the system (x). Moreover, the measurement is corrupted with sensor noise and the state evolution is corrupted with process noise. Determining a good state estimate for use in the regulator in the face of a noisy and incomplete output measurement is a challenging task. That is the challenge of state estimation.

To fully appreciate the fundamentals of state estimation, we must address the fluctuations in the data. Probability theory has proven itself as the most successful and versatile approach to modeling these fluctuations. In this section we introduce the probability fundamentals necessary to develop an optimal state estimator in the simplest possible setting: a linear discrete time model subject to normally distributed process and measurement noise. This optimal state estimator is known as the Kalman filter (Kalman, 1960a). In Chapter 4 we revisit the state estimation problem in a much wider setting, and consider nonlinear models and constraints on the system that preclude an analytical solution such as the Kalman filter. The probability theory presented here is also preparation for understanding that chapter.

1.4.1 Linear Systems and Normal Distributions

This section summarizes the probability and random variable results required for deriving a linear optimal estimator such as the Kalman filter. We assume that the reader is familiar with the concepts of a random variable, probability density and distribution, the multivariate normal distribution, mean and variance, statistical independence, and conditional probability. Readers unfamiliar with these terms should study the material in Appendix A before reading this and the next sections.

In the following discussion let x, y, and z be vectors of random variables. We use the notation

$$x \sim N(m, P)$$
$$p_x(x) = n(x, m, P)$$

to denote random variable x is normally distributed with mean m and covariance (or simply variance) P, in which

$$n(x, m, P) = \frac{1}{(2\pi)^{n/2}(\det P)^{1/2}} \exp\left[-\frac{1}{2}(x-m)'P^{-1}(x-m)\right] \tag{1.19}$$

and $\det P$ denotes the determinant of matrix P. Note that if $x \in \mathbb{R}^n$, then $m \in \mathbb{R}^n$ and $P \in \mathbb{R}^{n\times n}$ is a positive definite matrix. We require three main results. The simplest version can be stated as follows.

Joint independent normals. If x and y are normally distributed and (statistically) independent[5]

$$x \sim N(m_x, P_x) \qquad y \sim N(m_y, P_y)$$

then their joint density is given by

$$p_{x,y}(x, y) = n(x, m_x, P_x)\, n(y, m_y, P_y)$$
$$\begin{bmatrix} x \\ y \end{bmatrix} \sim N\left(\begin{bmatrix} m_x \\ m_y \end{bmatrix}, \begin{bmatrix} P_x & 0 \\ 0 & P_y \end{bmatrix}\right) \tag{1.20}$$

Note that, depending on convenience, we use both (x, y) and the vector $\left[\begin{smallmatrix} x \\ y \end{smallmatrix}\right]$ to denote the pair of random variables.

Linear transformation of a normal. If x is normally distributed with mean m and variance P, and y is a linear transformation of x, $y = Ax$, then y is distributed with mean Am and variance APA'

$$x \sim N(m, P) \qquad y = Ax \qquad y \sim N(Am, APA') \tag{1.21}$$

[5] We may emphasize that two vectors of random variables are independent using *statistically independent* to distinguish this concept from linear independence of vectors.

Conditional of a joint normal. If x and y are jointly normally distributed as

$$\begin{bmatrix} x \\ y \end{bmatrix} \sim N\left(\begin{bmatrix} m_x \\ m_y \end{bmatrix} \begin{bmatrix} P_x & P_{xy} \\ P_{yx} & P_y \end{bmatrix}\right)$$

then the conditional density of x given y is also normal

$$p_{x|y}(x|y) = n(x, m, P) \tag{1.22}$$

in which the mean is

$$m = m_x + P_{xy}P_y^{-1}(y - m_y)$$

and the covariance is

$$P = P_x - P_{xy}P_y^{-1}P_{yx}$$

Note that the conditional mean m is itself a random variable because it depends on the random variable y.

To derive the optimal estimator, we actually require these three main results conditioned on additional random variables. The analogous results are the following.

Joint independent normals. If $p_{x|z}(x|z)$ is normal, and y is statistically independent of x and z and normally distributed

$$p_{x|z}(x|z) = n(x, m_x, P_x)$$
$$y \sim N(m_y, P_y) \qquad y \text{ independent of } x \text{ and } z$$

then the conditional joint density of (x, y) given z is

$$p_{x,y|z}(x, y|z) = n(x, m_x, P_x)\, n(y, m_y, P_y)$$
$$p_{x,y|z}\left(\begin{bmatrix} x \\ y \end{bmatrix} \middle| z\right) = n\left(\begin{bmatrix} x \\ y \end{bmatrix}, \begin{bmatrix} m_x \\ m_y \end{bmatrix}, \begin{bmatrix} P_x & 0 \\ 0 & P_y \end{bmatrix}\right) \tag{1.23}$$

Linear transformation of a normal.

$$p_{x|z}(x|z) = n(x, m, P) \qquad y = Ax$$
$$p_{y|z}(y|z) = n(y, Am, APA') \tag{1.24}$$

Conditional of a joint normal. If x and y are jointly normally distributed as

$$p_{x,y|z}\left(\begin{bmatrix} x \\ y \end{bmatrix} \middle| z\right) = n\left(\begin{bmatrix} x \\ y \end{bmatrix}, \begin{bmatrix} m_x \\ m_y \end{bmatrix}, \begin{bmatrix} P_x & P_{xy} \\ P_{yx} & P_y \end{bmatrix}\right)$$

then the conditional density of x given y, z is also normal

$$p_{x|y,z}(x|y,z) = n(x, m, P) \tag{1.25}$$

in which

$$\begin{aligned} m &= m_x + P_{xy}P_y^{-1}(y - m_y) \\ P &= P_x - P_{xy}P_y^{-1}P_{yx} \end{aligned}$$

1.4.2 Linear Optimal State Estimation

We start by assuming the initial state $x(0)$ is normally distributed with some mean and covariance

$$x(0) \sim N(\overline{x}(0), Q(0))$$

In applications, we often do not know $\overline{x}(0)$ or $Q(0)$. In such cases we often set $\overline{x}(0) = 0$ and choose a large value for $Q(0)$ to indicate our lack of prior knowledge. The choice of a large variance prior forces the upcoming $y(k)$ measurements to determine the state estimate $\hat{x}(k)$.

Combining the measurement. We obtain noisy measurement $y(0)$ satisfying

$$y(0) = Cx(0) + v(0)$$

in which $v(0) \sim N(0, R)$ is the measurement noise. If the measurement process is quite noisy, then R is large. If the measurements are highly accurate, then R is small. We choose a zero mean for v because all of the deterministic effects with nonzero mean are considered part of the model, and the measurement noise reflects what is left after all these other effects have been considered. Given the measurement $y(0)$, we want to obtain the conditional density $p_{x(0)|y(0)}(x(0)|y(0))$. This conditional density describes the change in our knowledge about $x(0)$ after we obtain measurement $y(0)$. This step is the essence of state estimation. To derive this conditional density, first consider the pair of variables $(x(0), y(0))$ given as

$$\begin{bmatrix} x(0) \\ y(0) \end{bmatrix} = \begin{bmatrix} I & 0 \\ C & I \end{bmatrix} \begin{bmatrix} x(0) \\ v(0) \end{bmatrix}$$

We assume that the noise $v(0)$ is statistically independent of $x(0)$, and use the independent joint normal result (1.20) to express the joint density of $(x(0), v(0))$

$$\begin{bmatrix} x(0) \\ v(0) \end{bmatrix} \sim N\left(\begin{bmatrix} \overline{x}(0) \\ 0 \end{bmatrix}, \begin{bmatrix} Q(0) & 0 \\ 0 & R \end{bmatrix} \right)$$

From the previous equation, the pair $(x(0), y(0))$ is a linear transformation of the pair $(x(0), v(0))$. Therefore, using the linear transformation of normal result (1.21), and the density of $(x(0), v(0))$ gives the density of $(x(0), y(0))$

$$\begin{bmatrix} x(0) \\ y(0) \end{bmatrix} \sim N\left(\begin{bmatrix} \overline{x}(0) \\ C\overline{x}(0) \end{bmatrix}, \begin{bmatrix} Q(0) & Q(0)C' \\ CQ(0) & CQ(0)C' + R \end{bmatrix} \right)$$

Given this joint density, we then use the conditional of a joint normal result (1.22) to obtain

$$p_{x(0)|y(0)}\left(x(0)|y(0)\right) = n\left(x(0), m, P\right)$$

in which

$$\begin{aligned} m &= \overline{x}(0) + L(0)\left(y(0) - C\overline{x}(0)\right) \\ L(0) &= Q(0)C'(CQ(0)C' + R)^{-1} \\ P &= Q(0) - Q(0)C'(CQ(0)C' + R)^{-1}CQ(0) \end{aligned}$$

We see that the conditional density $p_{x(0)|y(0)}$ is normal. The *optimal* state estimate is the value of $x(0)$ that maximizes this conditional density. For a normal, that is the mean, and we choose $\hat{x}(0) = m$. We also denote the variance in this conditional after measurement $y(0)$ by $P(0) = P$ with P given in the previous equation. The change in variance after measurement ($Q(0)$ to $P(0)$) quantifies the information increase by obtaining measurement $y(0)$. The variance after measurement, $P(0)$, is always less than or equal to $Q(0)$, which implies that we can only gain information by measurement; but the information gain may be small if the measurement device is poor and the measurement noise variance R is large.

Forecasting the state evolution. Next we consider the state evolution from $k = 0$ to $k = 1$, which satisfies

$$x(1) = \begin{bmatrix} A & I \end{bmatrix} \begin{bmatrix} x(0) \\ w(0) \end{bmatrix}$$

in which $w(0) \sim N(0, Q)$ is the process noise. If the state is subjected to large disturbances, then Q is large, and if the disturbances are small, Q is small. Again we choose zero mean for w because the nonzero-mean disturbances should have been accounted for in the system model. We next calculate the conditional density $p_{x(1)|y(0)}$. Now we require the conditional version of the joint density $(x(0), w(0))$. We assume that the process noise $w(0)$ is statistically independent of both $x(0)$ and $v(0)$, hence it is also independent of $y(0)$, which is a linear combination of $x(0)$ and $v(0)$. Therefore we use (1.23) to obtain

$$\begin{bmatrix} x(0) \\ w(0) \end{bmatrix} \sim N\left(\begin{bmatrix} \hat{x}(0) \\ 0 \end{bmatrix}, \begin{bmatrix} P(0) & 0 \\ 0 & Q \end{bmatrix} \right)$$

We then use the conditional version of the linear transformation of a normal (1.24) to obtain

$$p_{x(1)|y(0)}(x(1)|y(0)) = n(x(1), \hat{x}^-(1), P^-(1))$$

in which the mean and variance are

$$\hat{x}^-(1) = A\hat{x}(0) \qquad P^-(1) = AP(0)A' + Q$$

We see that forecasting forward one time step may increase or decrease the conditional variance of the state. If the eigenvalues of A are less than unity, for example, the term $AP(0)A'$ *may* be smaller than $P(0)$, but the process noise Q adds a positive contribution. If the system is unstable, $AP(0)A'$ *may* be larger than $P(0)$, and then the conditional variance definitely increases upon forecasting. See also Exercise 1.27 for further discussion of this point.

Given that $p_{x(1)|y(0)}$ is also a normal, we are situated to add measurement $y(1)$ and continue the process of adding measurements followed by forecasting forward one time step until we have processed all the available data. Because this process is recursive, the storage requirements are small. We need to store only the current state estimate and variance, and can discard the measurements as they are processed. The required online calculation is minor. These features make the optimal linear estimator an ideal candidate for rapid online application. We next summarize the state estimation recursion.

Summary. Denote the measurement trajectory by

$$\mathbf{y}(k) := (y(0), y(1), \ldots y(k))$$

At time k the conditional density with data $\mathbf{y}(k-1)$ is normal

$$p_{x(k)|\mathbf{y}(k-1)}(x(k)|\mathbf{y}(k-1)) = n(x(k), \hat{x}^-(k), P^-(k))$$

and we denote the mean and variance with a superscript minus to indicate these are the statistics *before* measurement $y(k)$. At $k = 0$, the recursion starts with $\hat{x}^-(0) = \overline{x}(0)$ and $P^-(0) = Q(0)$ as discussed previously. We obtain measurement $y(k)$ which satisfies

$$\begin{bmatrix} x(k) \\ y(k) \end{bmatrix} = \begin{bmatrix} I & 0 \\ C & I \end{bmatrix} \begin{bmatrix} x(k) \\ v(k) \end{bmatrix}$$

The density of $(x(k), v(k))$ follows from (1.23) since measurement noise $v(k)$ is independent of $x(k)$ and $\mathbf{y}(k-1)$

$$\begin{bmatrix} x(k) \\ v(k) \end{bmatrix} \sim N\left(\begin{bmatrix} \hat{x}^-(k) \\ 0 \end{bmatrix}, \begin{bmatrix} P^-(k) & 0 \\ 0 & R \end{bmatrix} \right)$$

Equation (1.24) then gives the joint density

$$\begin{bmatrix} x(k) \\ y(k) \end{bmatrix} \sim N\left(\begin{bmatrix} \hat{x}^-(k) \\ C\hat{x}^-(k) \end{bmatrix}, \begin{bmatrix} P^-(k) & P^-(k)C' \\ CP^-(k) & CP^-(k)C' + R \end{bmatrix} \right)$$

We note $(\mathbf{y}(k-1), y(k)) = \mathbf{y}(k)$, and using the conditional density result (1.25) gives

$$p_{x(k)|\mathbf{y}(k)}(x(k)|\mathbf{y}(k)) = n(x(k), \hat{x}(k), P(k))$$

in which

$$\begin{aligned} \hat{x}(k) &= \hat{x}^-(k) + L(k)\left(y(k) - C\hat{x}^-(k)\right) \\ L(k) &= P^-(k)C'(CP^-(k)C' + R)^{-1} \\ P(k) &= P^-(k) - P^-(k)C'(CP^-(k)C' + R)^{-1}CP^-(k) \end{aligned}$$

We forecast from k to $k+1$ using the model

$$x(k+1) = \begin{bmatrix} A & I \end{bmatrix} \begin{bmatrix} x(k) \\ w(k) \end{bmatrix}$$

Because $w(k)$ is independent of $x(k)$ and $\mathbf{y}(k)$, the joint density of $(x(k), w(k))$ follows from a second use of (1.23)

$$\begin{bmatrix} x(k) \\ w(k) \end{bmatrix} \sim N\left(\begin{bmatrix} \hat{x}(k) \\ 0 \end{bmatrix}, \begin{bmatrix} P(k) & 0 \\ 0 & Q \end{bmatrix} \right)$$

and a second use of the linear transformation result (1.24) gives

$$p_{x(k+1)|\mathbf{y}(k)}(x(k+1)|\mathbf{y}(k)) = n(x(k+1), \hat{x}^-(k+1), P^-(k+1))$$

in which

$$\begin{aligned}\hat{x}^-(k+1) &= A\hat{x}(k)\\ P^-(k+1) &= AP(k)A' + Q\end{aligned}$$

and the recursion is complete.

1.4.3 Least Squares Estimation

We next consider the state estimation problem as a deterministic optimization problem rather than an exercise in maximizing conditional density. This viewpoint proves valuable in Chapter 4 when we wish to add constraints to the state estimator. Consider a time horizon with measurements $y(k), k = 0, 1, \ldots, T$. We consider the prior information to be our best initial guess of the initial state $x(0)$, denoted $\overline{x}(0)$, and weighting matrices $P^-(0)$, Q, and R for the initial state, process disturbance, and measurement disturbance. A reasonably flexible choice for objective function is

$$\begin{aligned}V_T(\mathbf{x}(T)) = \frac{1}{2}\Bigg(&|x(0) - \overline{x}(0)|^2_{(P^-(0))^{-1}} + \\ &\sum_{k=0}^{T-1} |x(k+1) - Ax(k)|^2_{Q^{-1}} + \sum_{k=0}^{T} |y(k) - Cx(k)|^2_{R^{-1}} \Bigg)\end{aligned} \tag{1.26}$$

in which $\mathbf{x}(T) := (x(0), x(1), \ldots, x(T))$. We claim and then show that the following (deterministic) least squares optimization problem produces the same result as the conditional density function maximization of the Kalman filter

$$\min_{\mathbf{x}(T)} V_T(\mathbf{x}(T)) \tag{1.27}$$

Game plan. Using forward DP, we can decompose and solve recursively the least squares state estimation problem. To see clearly how the procedure works, first we write out the terms in the state estimation least squares problem (1.27)

$$\min_{x(0),\ldots,x(T)} \frac{1}{2}\Big(|x(0)-\overline{x}(0)|^2_{(P^-(0))^{-1}} + |y(0)-Cx(0)|^2_{R^{-1}} + |x(1)-Ax(0)|^2_{Q^{-1}} + |y(1)-Cx(1)|^2_{R^{-1}} + |x(2)-Ax(1)|^2_{Q^{-1}} + \cdots + |x(T)-Ax(T-1)|^2_{Q^{-1}} + |y(T)-Cx(T)|^2_{R^{-1}} \Big) \quad (1.28)$$

We decompose this T-stage optimization problem with forward DP. First we combine the prior and the measurement $y(0)$ into the quadratic function $V_0(x(0))$ as shown in the following equation

$$\min_{x(T),\ldots,x(1)} \overbrace{\min_{x(0)} \underbrace{\frac{1}{2}\Big(|x(0)-\overline{x}(0)|^2_{(P^-(0))^{-1}} + |y(0)-Cx(0)|^2_{R^{-1}} }_{\text{combine } V_0(x(0))} + |x(1)-Ax(0)|^2_{Q^{-1}} + |y(1)-Cx(1)|^2_{R^{-1}} + |x(2)-Ax(1)|^2_{Q^{-1}} + \cdots + |x(T)-Ax(T-1)|^2_{Q^{-1}} + |y(T)-Cx(T)|^2_{R^{-1}} \Big)}^{\text{arrival cost } V_1^-(x(1))}$$

Then we optimize over the first state, $x(0)$. This produces the arrival cost for the first stage, $V_1^-(x(1))$, which we will show is also quadratic

$$V_1^-(x(1)) = (1/2)\left(|x(1)-\hat{x}^-(1)|^2_{(P^-(1))^{-1}} + d(0)\right)$$

Next we combine the arrival cost of the first stage with the next measurement $y(1)$ to obtain $V_1(x(1))$

$$\min_{x(T),\ldots,x(2)} \overbrace{\min_{x(1)} \underbrace{\frac{1}{2}\Big(|x(1)-\hat{x}^-(1)|^2_{(P^-(1))^{-1}} + d(0) + |y(1)-Cx(1)|^2_{R^{-1}} }_{\text{combine } V_1(x(1))} + |x(2)-Ax(1)|^2_{Q^{-1}} + |y(2)-Cx(2)|^2_{R^{-1}} + |x(3)-Ax(2)|^2_{Q^{-1}} + \cdots + |x(T)-Ax(T-1)|^2_{Q^{-1}} + |y(T)-Cx(T)|^2_{R^{-1}} \Big)}^{\text{arrival cost } V_2^-(x(2))} \quad (1.29)$$

We optimize over the second state, $x(1)$, which defines arrival cost for the first two stages, $V_2^-(x(2))$. We continue in this fashion until we have optimized finally over $x(T)$ and have solved (1.28). Now that we have in mind an overall game plan for solving the problem, we look at each step in detail and develop the recursion formulas of forward DP.

Combine prior and measurement. Combining the prior and measurement defines V_0

$$V_0(x(0)) = \frac{1}{2}\Big(\underbrace{|x(0)-\overline{x}(0)|^2_{(P^-(0))^{-1}}}_{\text{prior}} + \underbrace{|\,y(0)-Cx(0)\,|^2_{R^{-1}}}_{\text{measurement}} \Big) \quad (1.30)$$

which can be expressed also as

$$V_0(x(0)) = \frac{1}{2}\Big(|x(0)-\overline{x}(0)|^2_{(P^-(0))^{-1}} + |\,(y(0)-C\overline{x}(0)) - C(x(0)-\overline{x}(0))\,|^2_{R^{-1}} \Big)$$

Using the third form in Example 1.1 we can combine these two terms into a single quadratic function

$$V_0(x(0)) = (1/2)\,\big((x(0)-\overline{x}(0)-v)'\widetilde{H}^{-1}(x(0)-\overline{x}(0)-v)+d(0)\big)$$

in which

$$\begin{aligned} v &= P^-(0)C'(CP^-(0)C'+R)^{-1}\big(y(0)-C\overline{x}(0)\big)\\ \widetilde{H} &= P^-(0)-P^-(0)C'(CP^-(0)C'+R)^{-1}CP^-(0)\\ d(0) &= |\,y(0)-C\overline{x}(0)\,|^2_{(CP^-(0)C'+R)^{-1}} \end{aligned}$$

If we define

$$\begin{aligned} P(0) &= P^-(0)-P^-(0)C'(CP^-(0)C'+R)^{-1}CP^-(0)\\ L(0) &= P^-(0)C'(CP^-(0)C'+R)^{-1} \end{aligned}$$

and define the state estimate $\hat{x}(0)$ as follows

$$\begin{aligned} \hat{x}(0) &= \overline{x}(0)+v\\ \hat{x}(0) &= \overline{x}(0)+L(0)\big(y(0)-C\overline{x}(0)\big) \end{aligned}$$

then we have the following compact expression for the function V_0.

$$V_0(x(0)) = (1/2)(|x(0)-\hat{x}(0)|^2_{P(0)^{-1}}+d(0))$$

State evolution and arrival cost. Now we add the next term in (1.28) to the function $V_0(\cdot)$ and denote the sum as $V(\cdot)$

$$\begin{aligned} V(x(0),x(1)) &= V_0(x(0)) + (1/2)\,|x(1)-Ax(0)|^2_{Q^{-1}}\\ V(x(0),x(1)) &= \frac{1}{2}\big(\,|x(0)-\hat{x}(0)|^2_{P(0)^{-1}}+|x(1)-Ax(0)|^2_{Q^{-1}}+d(0)\big) \end{aligned}$$

Again using the third form in Example 1.1, we can add the two quadratics to obtain

$$V(x(0), x(1)) = (1/2)(|x(0) - v|^2_{\tilde{H}^{-1}} + d)$$

in which

$$\begin{aligned}
v &= \hat{x}(0) + P(0)A' \left(AP(0)A' + Q\right)^{-1} (x(1) - A\hat{x}(0)) \\
d &= (x(1) - A\hat{x}(0))' \left(AP(0)A' + Q\right)^{-1} (x(1) - A\hat{x}(0)) + d(0) \\
\tilde{H} &= P(0) - P(0)A' \left(AP(0)A' + Q\right)^{-1} AP(0)
\end{aligned}$$

This form is convenient for optimization over the first decision variable $x(0)$; by inspection the solution is $x(0) = v$ and the cost is $(1/2)d$. We define the arrival cost to be the result of this optimization

$$V_1^-(x(1)) = \min_{x(0)} V(x(0), x(1))$$

and we have that

$$V_1^-(x(1)) = (1/2)(|x(1) - \hat{x}^-(1)|^2_{(P^-(1))^{-1}} + d(0))$$

with

$$\begin{aligned}
\hat{x}^-(1) &= A\hat{x}(0) \\
P^-(1) &= AP(0)A' + Q
\end{aligned}$$

Combine arrival cost and measurement. We now combine the arrival cost and measurement for the next stage of the optimization to obtain

$$V_1(x(1)) = \underbrace{V_1^-(x(1))}_{\text{prior}} + \underbrace{(1/2)\,|(y(1) - Cx(1))|^2_{R^{-1}}}_{\text{measurement}}$$

$$V_1(x(1)) = \frac{1}{2}\Big(|x(1) - \hat{x}^-(1)|^2_{(P^-(1))^{-1}} + |y(1) - Cx(1)|^2_{R^{-1}} + d(0) \Big)$$

We can see that this equation is exactly the form as (1.30) of the previous step, and, by simply changing the variable names, we have that

$$\begin{aligned}
P(1) &= P^-(1) - P^-(1)C'(CP^-(1)C' + R)^{-1}CP^-(1) \\
L(1) &= P^-(1)C'(CP^-(1)C' + R)^{-1} \\
\hat{x}(1) &= \hat{x}^-(1) + L(1)(y(1) - C\hat{x}^-(1)) \\
d(1) &= d(0) + |y(1) - C\hat{x}^-(1)|^2_{(CP^-(1)C'+R)^{-1}}
\end{aligned}$$

and the cost function V_1 is defined as

$$V_1(x(1)) = (1/2)(|x(1) - \hat{x}(1)|^2_{P(1)^{-1}} + d(1))$$

Recursion and termination. The recursion can be summarized by two steps. Adding the measurement at time k produces

$$\begin{aligned}
P(k) &= P^-(k) - P^-(k)C'(CP^-(k)C' + R)^{-1}CP^-(k) \\
L(k) &= P^-(k)C'(CP^-(k)C' + R)^{-1} \\
\hat{x}(k) &= \hat{x}^-(k) + L(k)(y(k) - C\hat{x}^-(k)) \\
d(k) &= d(k-1) + \left| y(k) - C\hat{x}^-(k) \right|^2_{(CP^-(k)C'+R)^{-1}}
\end{aligned}$$

Propagating the model to time $k+1$ produces

$$\begin{aligned}
\hat{x}^-(k+1) &= A\hat{x}(k) \\
P^-(k+1) &= AP(k)A' + Q
\end{aligned}$$

and the recursion starts with the prior information $\hat{x}^-(0) = \overline{x}(0)$ and $P^-(0)$. The arrival cost, V_k^-, and arrival cost plus measurement, V_k, for each stage are given by

$$\begin{aligned}
V_k^-(x(k)) &= (1/2)\left(\left| x(k) - \hat{x}^-(k) \right|^2_{(P^-(k))^{-1}} + d(k-1) \right) \\
V_k(x(k)) &= (1/2)\left(|x(k) - \hat{x}(k)|^2_{(P(k))^{-1}} + d(k) \right)
\end{aligned}$$

The process terminates with the final measurement $y(T)$, at which point we have recursively solved the original problem (1.28).

We see by inspection that the recursion formulas given by forward DP of (1.28) are the same as those found by calculating the conditional density function in Section 1.4.2. Moreover, the conditional densities before and after measurement are closely related to the least squares value functions as shown below

$$\begin{aligned}
p(x(k)|\mathbf{y}(k-1)) = \frac{1}{(2\pi)^{n/2}(\det P^-(k))^{1/2}} \\
\exp\left(- (V_k^-(x(k)) - (1/2)d(k-1)) \right)
\end{aligned}$$

$$\begin{aligned}
p(x(k)|\mathbf{y}(k)) = \frac{1}{(2\pi)^{n/2}(\det P(k))^{1/2}} \\
\exp\left(- (V_k(x(k)) - (1/2)d(k)) \right) \quad (1.31)
\end{aligned}$$

The discovery (and rediscovery) of the close connection between recursive least squares and optimal statistical estimation has not always been greeted happily by researchers:

> The recursive least squares approach was actually inspired by probabilistic results that automatically produce an equation of evolution for the estimate (the conditional mean). In fact, much of the recent least squares work did nothing more than rederive the probabilistic results (perhaps in an attempt to understand them). As a result, much of the least squares work contributes very little to estimation theory.
> —Jazwinski (1970, pp.152–153)

In contrast with this view, we find both approaches valuable in the subsequent development. The probabilistic approach, which views the state estimator as maximizing conditional density of the state given measurement, offers the most insight. It provides a rigorous basis for comparing different estimators based on the variance of their estimate error. It also specifies what information is required to define an optimal estimator, with variances Q and R of primary importance. In the probabilistic framework, these parameters should be found from modeling and data. The main deficiency in the least squares viewpoint is that the objective function, although reasonable, is ad hoc and not justified. The choice of weighting matrices Q and R is arbitrary. Practitioners generally choose these parameters based on a tradeoff between the competing goals of speed of estimator response and insensitivity to measurement noise. But a careful statement of this tradeoff often just leads back to the probabilistic viewpoint in which the process disturbance and measurement disturbance are modeled as normal distributions. If we restrict attention to unconstrained linear systems, the probabilistic viewpoint is clearly superior.

Approaching state estimation with the perspective of least squares pays off, however, when the models are significantly more complex. It is generally intractable to find and maximize the conditional density of the state given measurements for complex, nonlinear and constrained models. Although the state estimation problem can be stated in the language of probability, it cannot be solved with current methods. But reasonable objective functions can be chosen for even complex, nonlinear and constrained models. Moreover, knowing which least squares problems correspond to which statistically optimal estimation problems for the simple linear case, provides the engineer with valuable insight in choosing useful objective functions for nonlinear estimation. We explore these more complex and realistic estimation problems in Chapter 4. The perspective of least squares also leads to succinct arguments for establishing estimator stability, which we take up shortly.

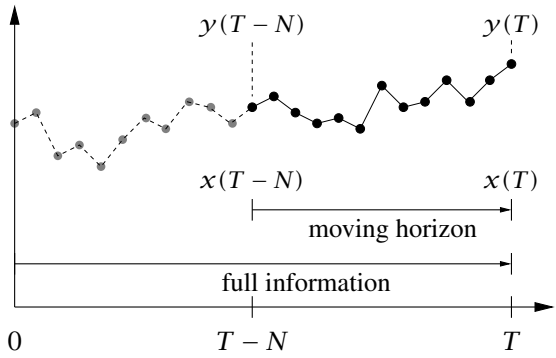


Figure 1.5: Schematic of the moving horizon estimation problem.

First we consider situations in which it is advantageous to use moving horizon estimation.

1.4.4 Moving Horizon Estimation

When using nonlinear models or considering constraints on the estimates, we cannot calculate the conditional density recursively in closed form as we did in Kalman filtering. Similarly, we cannot solve recursively the least squares problem. If we use least squares we must optimize all the states in the trajectory $\mathbf{x}(T)$ simultaneously to obtain the state estimates. This optimization problem becomes computationally intractable as T increases. Moving horizon estimation (MHE) removes this difficulty by considering only the most recent N measurements and finds only the most recent N values of the state trajectory as sketched in Figure 1.5. The states to be estimated are $\mathbf{x}_N(T) = \big(x(T-N),\ldots, x(T)\big)$ given measurements $\mathbf{y}_N(T) = \big(y(T-N),\ldots,y(T)\big)$. The data have been broken into two sections with $(\mathbf{y}(T-N-1),\mathbf{y}_N(T)) = \mathbf{y}(T)$. We assume here that $T \geq N-1$ to ignore the initial period in which the estimation window fills with measurements and assume that the window is always full.

The simplest form of MHE is the following least squares problem

$$\min_{\mathbf{x}_N(T)} \hat{V}_T(\mathbf{x}_N(T)) \tag{1.32}$$

in which the objective function is

$$\hat{V}_T(\mathbf{x}_N(T)) = \frac{1}{2}\Big(\sum_{k=T-N}^{T-1} |x(k+1) - Ax(k)|^2_{Q^{-1}} + \sum_{k=T-N}^{T} |y(k) - Cx(k)|^2_{R^{-1}} \Big) \quad (1.33)$$

We use the circumflex (hat) to indicate this is the MHE cost function considering data sequence from $T-N$ to T rather than the full information or least squares cost considering the data from 0 to T.

MHE in terms of least squares. Notice that from our previous DP recursion in (1.29), we can write the full least squares problem as

$$V_T(\mathbf{x}_N(T)) = V^-_{T-N}(x(T-N)) + \frac{1}{2}\Big(\sum_{k=T-N}^{T-1} |x(k+1) - Ax(k)|^2_{Q^{-1}} + \sum_{k=T-N}^{T} |y(k) - Cx(k)|^2_{R^{-1}} \Big)$$

in which $V^-_{T-N}(\cdot)$ is the arrival cost at time $T-N$. Comparing these two objective functions, it is clear that the simplest form of MHE is equivalent to setting up a full least squares problem, but then setting the arrival cost function $V^-_{T-N}(\cdot)$ to zero.

MHE in terms of conditional density. Because we have established the close connection between least squares and conditional density in (1.31), we can write the full least squares problem also as an equivalent conditional density maximization

$$\max_{x(T)} p_{x(T)|\mathbf{y}_N(T)}(x(T)|\mathbf{y}_N(T))$$

with prior density

$$p_{x(T-N)|\mathbf{y}(T-N-1)}(x|\mathbf{y}(T-N-1)) = c\exp(-V^-_{T-N}(x)) \quad (1.34)$$

in which the constant c can be found from (1.19) if desired, but its value does not change the solution to the optimization. We can see from (1.34) that setting $V^-_{T-N}(\cdot)$ to zero in the simplest form of MHE is equivalent to giving infinite variance to the conditional density of $x(T-N)|\mathbf{y}(T-N-1)$. This means we are using no information about the state $x(T-N)$ and completely discounting the previous measurements $\mathbf{y}(T-N-1)$.

To provide a more flexible MHE problem, we therefore introduce a penalty on the first state to account for the neglected data $\mathbf{y}(T-N-1)$

$$\hat{V}_T(\mathbf{x}_N(T)) = \Gamma_{T-N}(x(T-N)) + \frac{1}{2}\Big(\sum_{k=T-N}^{T-1} |x(k+1) - Ax(k)|^2_{Q^{-1}} + \sum_{k=T-N}^{T} |y(k) - Cx(k)|^2_{R^{-1}} \Big)$$

For the linear Gaussian case, we can account for the neglected data exactly with no approximation by setting Γ equal to the arrival cost, or, equivalently, the negative logarithm of the conditional density of the state given the prior measurements. Indeed, there is no need to use MHE for the linear Gaussian problem at all because we can solve the full problem recursively. When addressing nonlinear and constrained problems in Chapter 4, however, we must approximate the conditional density of the state given the prior measurements in MHE to obtain a computationally tractable and high-quality estimator.

1.4.5 Observability

We next explore the convergence properties of the state estimators. For this we require the concept of system observability. The basic idea of observability is that any two distinct states can be *distinguished* by applying some input and observing the two system outputs over some finite time interval (Sontag, 1998, p.262–263). We discuss this general definition in more detail when treating nonlinear systems in Chapter 4, but observability for linear systems is much simpler. First of all, the applied input is irrelevant and we can set it to zero. Therefore consider the linear time-invariant system (A, C) with zero input

$$\begin{aligned} x(k+1) &= Ax(k) \\ y(k) &= Cx(k) \end{aligned}$$

The system is observable if there exists a finite N, such that for every $x(0)$, N measurements $(y(0), y(1), \ldots, y(N-1))$ distinguish uniquely the initial state $x(0)$. Similarly to the case of controllability, if we cannot determine the initial state using n measurements, we cannot determine it using $N > n$ measurements. Therefore we can develop a convenient test for observability as follows. For n measurements, the

system model gives

$$\begin{bmatrix} y(0) \\ y(1) \\ \vdots \\ y(n-1) \end{bmatrix} = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix} x(0) \tag{1.35}$$

The question of *observability* is therefore a question of *uniqueness* of solutions to these linear equations. The matrix appearing in this equation is known as the *observability matrix* $\mathcal{O}$

$$\mathcal{O} = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix} \tag{1.36}$$

From the fundamental theorem of linear algebra, we know the solution to (1.35) is unique if and only if the *columns* of the $np \times n$ observability matrix are linearly independent.[6] Therefore, we have that the system (A, C) is observable if and only if

$$\text{rank}(\mathcal{O}) = n$$

The following result for checking observability also proves useful (Hautus, 1972).

Lemma 1.4 (Hautus lemma for observability)**.** *A system is observable if and only if*

$$\text{rank} \begin{bmatrix} \lambda I - A \\ C \end{bmatrix} = n \qquad \textit{for all } \lambda \in \mathbb{C} \tag{1.37}$$

in which $\mathbb{C}$ is the set of complex numbers.

Notice that the first n rows of the matrix in (1.37) are linearly independent if $\lambda \notin \text{eig}(A)$, so (1.37) is equivalent to checking the rank at just the eigenvalues of A

$$\text{rank} \begin{bmatrix} \lambda I - A \\ C \end{bmatrix} = n \qquad \text{for all } \lambda \in \text{eig}(A)$$

[6] See Section A.4 of Appendix A or (Strang, 1980, pp.87–88) for a review of this result.

1.4.6 Convergence of the State Estimator

Next we consider the question of convergence of the estimates of several of the estimators we have considered. The simplest convergence question to ask is the following. Given an initial estimate error, and zero state and measurement noises, does the state estimate converge to the state as time increases and more measurements become available? If the answer to this question is yes, we say the estimates converge; sometimes we say the estimator converges. As with the regulator, optimality of an estimator does not ensure its stability. Consider the case $A = I, C = 0$. The optimal estimate is $\hat{x}(k) = \overline{x}(0)$, which does not converge to the true state unless we have luckily chosen $\overline{x}(0) = x(0)$.[7] Obviously the lack of stability is caused by our choosing an unobservable (undetectable) system.

We treat first the Kalman filtering or full least squares problem. Recall that this estimator optimizes over the entire state trajectory $\mathbf{x}(T) := (x(0), \ldots, x(T))$ based on all measurements $\mathbf{y}(T) := (y(0), \ldots, y(T))$. In order to establish convergence, the following result on the optimal estimator cost function proves useful.

Lemma 1.5 (Convergence of estimator cost)**.** *Given noise-free measurements* $\mathbf{y}(T) = (Cx(0), CAx(0), \ldots, CA^T x(0))$*, the optimal estimator cost* $V_T^0(\mathbf{y}(T))$ *converges as* $T \to \infty$*.*

Proof. Denote the optimal state sequence at time T given measurement $\mathbf{y}(T)$ by

$$(\hat{x}(0|T),\ \hat{x}(1|T),\ \ldots,\ \hat{x}(T|T))$$

We wish to compare the optimal costs at time T and $T-1$. Therefore, consider using the first $T-1$ elements of the solution at time T as decision variables in the state estimation problem at time $T-1$. The cost for those decision variables at time $T-1$ is given by

$$V_T^0 \ - \ \frac{1}{2}\Big(|\hat{x}(T|T) - A\hat{x}(T-1|T)|^2_{Q^{-1}} \ + \ |y(T) - C\hat{x}(T|T)|^2_{R^{-1}} \Big)$$

In other words, we have the full cost at time T and we deduct the cost of the last stage, which is not present at $T-1$. Now this choice of decision variables is not necessarily optimal at time $T-1$, so we have the inequality

$$V_{T-1}^0 \leq V_T^0 - \frac{1}{2}\Big(|\hat{x}(T|T) - A\hat{x}(T-1|T)|^2_{Q^{-1}} + |y(T) - C\hat{x}(T|T)|^2_{R^{-1}} \Big)$$

[7] If we could count on that kind of luck, we would have no need for state estimation.

Because the quadratic terms are nonnegative, the sequence of optimal estimator costs is nondecreasing with increasing T. We can establish that the optimal cost is bounded above as follows: at any time T we can choose the decision variables to be $(x(0), Ax(0), \ldots, A^T x(0))$, which achieves cost $|x(0) - \overline{x}(0)|^2_{(P^-(0))^{-1}}$ independent of T. The optimal cost sequence is nondecreasing and bounded above and, therefore, converges. ■

The optimal estimator cost converges regardless of system observability. But if we want the optimal estimate to converge to the state, we have to restrict the system further. The following lemma provides an example of what is required.

Lemma 1.6 (Estimator convergence)**.** *For (A, C) observable, $Q, R > 0$, and noise-free measurements $\mathbf{y}(T) = (Cx(0), CAx(0), \ldots, CA^T x(0))$, the optimal linear state estimate converges to the state*

$$\hat{x}(T) \to x(T) \quad \text{as } T \to \infty$$

Proof. To compress the notation somewhat, let $\hat{w}_T(j) = \hat{x}(T + j + 1|T + n - 1) - A\hat{x}(T + j|T + n - 1)$. Using the optimal solution at time $T + n - 1$ as decision variables at time $T - 1$ allows us to write the following inequality

$$V^0_{T-1} \leq V^0_{T+n-1} - \frac{1}{2}\left(\sum_{j=-1}^{n-2} |\hat{w}_T(j)|^2_{Q^{-1}} + \sum_{j=0}^{n-1} |y(T + j) - C\hat{x}(T + j|T + n - 1)|^2_{R^{-1}} \right)$$

Because the sequence of optimal costs converges with increasing T, and $Q^{-1}, R^{-1} > 0$, we have established that for increasing T

$$\begin{aligned} \hat{w}_T(j) &\to 0 \quad j = -1, \ldots, n - 2 \\ y(T + j) - C\hat{x}(T + j|T + n - 1) &\to 0 \quad j = 0, \ldots, n - 1 \end{aligned} \tag{1.38}$$

From the system model we have the following relationship between the last n stages in the optimization problem at time $T + n - 1$ with data

$$\mathbf{y}(T+n-1)$$

$$\begin{bmatrix} \hat{x}(T|T+n-1) \\ \hat{x}(T+1|T+n-1) \\ \vdots \\ \hat{x}(T+n-1|T+n-1) \end{bmatrix} = \begin{bmatrix} I \\ A \\ \vdots \\ A^{n-1} \end{bmatrix} \hat{x}(T|T+n-1) + \begin{bmatrix} 0 & & & \\ I & 0 & & \\ \vdots & \vdots & \ddots & \\ A^{n-2} & A^{n-3} & \cdots & I \end{bmatrix} \begin{bmatrix} \hat{w}_T(0) \\ \hat{w}_T(1) \\ \vdots \\ \hat{w}_T(n-2) \end{bmatrix} \tag{1.39}$$

We note the measurements satisfy

$$\begin{bmatrix} y(T) \\ y(T+1) \\ \vdots \\ y(T+n-1) \end{bmatrix} = \mathcal{O}x(T)$$

Multiplying (1.39) by C and subtracting gives

$$\begin{bmatrix} y(T) - C\hat{x}(T|T+n-1) \\ y(T+1) - C\hat{x}(T+1|T+n-1) \\ \vdots \\ y(T+n-1) - C\hat{x}(T+n-1|T+n-1) \end{bmatrix} = \mathcal{O}(x(T) - \hat{x}(T|T+n-1)) - \begin{bmatrix} 0 & & & \\ C & 0 & & \\ \vdots & \vdots & \ddots & \\ CA^{n-2} & CA^{n-3} & \cdots & C \end{bmatrix} \begin{bmatrix} \hat{w}_T(0) \\ \hat{w}_T(1) \\ \vdots \\ \hat{w}_T(n-2) \end{bmatrix}$$

Applying (1.38) to this equation, we conclude $\mathcal{O}(x(T) - \hat{x}(T|T+n-1)) \to 0$ with increasing T. Because the observability matrix has independent columns, we conclude $x(T) - \hat{x}(T|T+n-1) \to 0$ as $T \to \infty$. Thus we conclude that the *smoothed* estimate $\hat{x}(T|T+n-1)$ converges to the state $x(T)$. Because the $\hat{w}_T(j)$ terms go to zero with increasing T, the last line of (1.39) gives $\hat{x}(T+n-1|T+n-1) \to A^{n-1}\hat{x}(T|T+n-1)$ as $T \to \infty$. From the system model $A^{n-1}x(T) = x(T+n-1)$ and, therefore, after replacing $T+n-1$ by T, we have

$$\hat{x}(T|T) \to x(T) \quad \text{as } T \to \infty$$

and asymptotic convergence of the estimator is established. ■

This convergence result also covers MHE with prior weighting set to the exact arrival cost because that is equivalent to Kalman filtering and full least squares. The simplest form of MHE, which discounts prior data completely, is also a convergent estimator, however, as discussed in Exercise 1.28.

The estimator convergence result in Lemma 1.6 is the simplest to establish, but, as in the case of the LQ regulator, we can enlarge the class of systems and weighting matrices (variances) for which estimator convergence is guaranteed. The system restriction can be weakened from observability to *detectability*, which is discussed in Exercises 1.31 and 1.32. The restriction on the process disturbance weight (variance) Q can be weakened from $Q > 0$ to $Q \geq 0$ and (A, Q) *stabilizable*, which is discussed in Exercise 1.33. The restriction $R > 0$ remains to ensure existence of the estimator for all measurements y.

1.5 Tracking, Disturbances, and Zero Offset

In the last section of this chapter we show briefly how to use the MPC regulator and MHE estimator to handle different kinds of control problems, including setpoint tracking and rejecting nonzero disturbances.

1.5.1 Tracking

It is a standard objective in applications to use a feedback controller to move the measured outputs of a system to a specified and constant setpoint. This problem is known as setpoint tracking. In Chapter 5 we consider the case in which the system is nonlinear and constrained, but for simplicity here we consider the linear unconstrained system in which y_{sp} is an arbitrary constant. In the regulation problem of Section 1.3 we assumed that the goal was to take the state of the system to the origin. Such a regulator can be used to treat the setpoint tracking problem with a coordinate transformation. Denote the desired output setpoint as y_{sp}. Denote a steady state of the system model as (x_s, u_s). From (1.5), the steady state satisfies

$$\begin{bmatrix} I - A & -B \end{bmatrix} \begin{bmatrix} x_s \\ u_s \end{bmatrix} = 0$$

For *unconstrained* systems, we also impose the requirement that the steady state satisfies $Cx_s = y_{\text{sp}}$ for the tracking problem, giving the

set of equations

$$\begin{bmatrix} I - A & -B \\ C & 0 \end{bmatrix} \begin{bmatrix} x_s \\ u_s \end{bmatrix} = \begin{bmatrix} 0 \\ y_{\text{sp}} \end{bmatrix} \tag{1.40}$$

If this set of equations has a solution, we can then define deviation variables

$$\begin{aligned} \widetilde{x}(k) &= x(k) - x_s \\ \widetilde{u}(k) &= u(k) - u_s \end{aligned}$$

that satisfy the dynamic model

$$\begin{aligned} \widetilde{x}(k+1) &= x(k+1) - x_s \\ &= Ax(k) + Bu(k) - (Ax_s + Bu_s) \\ \widetilde{x}(k+1) &= A\widetilde{x}(k) + B\widetilde{u}(k) \end{aligned}$$

so that the deviation variables satisfy the same model equation as the original variables. The zero regulation problem applied to the system in deviation variables finds $\widetilde{u}(k)$ that takes $\widetilde{x}(k)$ to zero, or, equivalently, which takes $x(k)$ to x_s, so that at steady state, $Cx(k) = Cx_s = y_{\text{sp}}$, which is the goal of the setpoint tracking problem. After solving the regulation problem in deviation variables, the input applied to the system is $u(k) = \widetilde{u}(k) + u_s$.

We next discuss when we can solve (1.40). We also note that for *constrained* systems, we must impose the constraints on the steady state (x_s, u_s). The matrix in (1.40) is a $(n+p) \times (n+m)$ matrix. For (1.40) to have a solution for all y_{sp}, it is sufficient that the rows of the matrix are linearly independent. That requires $p \leq m$: we require at least as many inputs as outputs with setpoints. But it is not uncommon in applications to have many more measured outputs than manipulated inputs. To handle these more general situations, we choose a matrix H and denote a new variable $r = Hy$ as a selection of linear combinations of the measured outputs. The variable $r \in \mathbb{R}^{n_c}$ is known as the *controlled variable*. For cases in which $p > m$, we choose some set of outputs $n_c \leq m$, as controlled variables, and assign setpoints to r, denoted r_{sp}.

We also wish to treat systems with more inputs than outputs, $m > p$. For these cases, the solution to (1.40) may exist for some choice of H and r_{sp}, but cannot be unique. If we wish to obtain a unique steady state, then we also must provide desired values for the steady inputs, u_{sp}. To handle constrained systems, we simply impose the constraints on (x_s, u_s).

Steady-state target problem. Our candidate optimization problem is therefore

$$\min_{x_s,u_s} \frac{1}{2}\left(|u_s - u_{\text{sp}}|^2_{R_s} + |Cx_s - y_{\text{sp}}|^2_{Q_s} \right) \tag{1.41a}$$

subject to

$$\begin{bmatrix} I - A & -B \\ HC & 0 \end{bmatrix} \begin{bmatrix} x_s \\ u_s \end{bmatrix} = \begin{bmatrix} 0 \\ r_{\text{sp}} \end{bmatrix} \tag{1.41b}$$

$$Eu_s \le e \tag{1.41c}$$

$$FCx_s \le f \tag{1.41d}$$

We make the following assumptions.

Assumption 1.7 (Target feasibility and uniqueness)**.**

(a) The target problem is feasible for the controlled variable setpoints of interest r_{sp}.

(b) The steady-state input penalty R_s is positive definite.

Assumption 1.7 (a) ensures that the solution (x_s, u_s) exists, and Assumption 1.7 (b) ensures that the solution is unique. If one chooses $n_c = 0$, then no controlled variables are required to be at setpoint, and the problem is feasible for any $(u_{\text{sp}}, y_{\text{sp}})$ because $(x_s, u_s) = (0, 0)$ is a feasible point. Exercises 1.56 and 1.57 explore the connection between feasibility of the equality constraints and the number of controlled variables relative to the number of inputs and outputs. One restriction is that the number of controlled variables chosen to be offset free must be less than or equal to the number of manipulated variables and the number of measurements, $n_c \le m$ and $n_c \le p$.

Dynamic regulation problem. Given the steady-state solution, we define the following multistage objective function

$$V(\tilde{x}(0), \tilde{\mathbf{u}}) = \frac{1}{2} \sum_{k=0}^{N-1} |\tilde{x}(k)|^2_Q + |\tilde{u}(k)|^2_R \qquad \text{s.t. } \tilde{x}^+ = A\tilde{x} + B\tilde{u}$$

in which $\tilde{x}(0) = \hat{x}(k) - x_s$, i.e., the initial condition for the regulation problem comes from the state estimate shifted by the steady-state x_s. The regulator solves the following dynamic, zero-state regulation problem

$$\min_{\tilde{\mathbf{u}}} V(\tilde{x}(0), \tilde{\mathbf{u}})$$

subject to

$$E\tilde{u} \le e - Eu_s$$
$$FC\tilde{x} \le f - FCx_s$$

in which the constraints also are shifted by the steady state (x_s, u_s). The optimal cost and solution are $V^0(\tilde{x}(0))$ and $\tilde{\mathbf{u}}^0(\tilde{x}(0))$. The moving horizon control law uses the first move of this optimal sequence, $\tilde{u}^0(\tilde{x}(0)) = \tilde{\mathbf{u}}^0(0;\tilde{x}(0))$, so the controller output is $u(k) = \tilde{u}^0(\tilde{x}(0)) + u_s$.

1.5.2 Disturbances and Zero Offset

Another common objective in applications is to use a feedback controller to compensate for an unmeasured disturbance to the system with the input so the disturbance's effect on the controlled variable is mitigated. This problem is known as disturbance rejection. We may wish to design a feedback controller that compensates for nonzero disturbances such that the selected controlled variables asymptotically approach their setpoints without offset. This property is known as zero offset. In this section we show a simple method for constructing an MPC controller to achieve zero offset.

In Chapter 5, we address the full problem. Here we must be content to limit our objective. We will ensure that *if the system is stabilized in the presence of the disturbance*, then there is zero offset. But we will not attempt to construct the controller that ensures stabilization over an interesting class of disturbances. That topic is treated in Chapter 5.

This more limited objective is similar to what one achieves when using the integral mode in proportional-integral-derivative (PID) control of an unconstrained system: either there is zero steady offset, or the system trajectory is unbounded. In a constrained system, the statement is amended to: either there is zero steady offset, or the system trajectory is unbounded, or the system constraints are active at steady state. In both constrained and unconstrained systems, the zero-offset property *precludes* one undesirable possibility: the system settles at an unconstrained steady state, and the steady state displays offset in the controlled variables.

A simple method to compensate for an unmeasured disturbance is to (i) model the disturbance, (ii) use the measurements and model to estimate the disturbance, and (iii) find the inputs that minimize the effect of the disturbance on the controlled variables. The choice of

disturbance model is motivated by the zero-offset goal. To achieve offset-free performance we augment the system state with an *integrating* disturbance d driven by a white noise w_d

$$d^+ = d + w_d \tag{1.42}$$

This choice is motivated by the works of Davison and Smith (1971, 1974); Qiu and Davison (1993) and the Internal Model Principle of Francis and Wonham (1976). To remove offset, one designs a control system that can remove asymptotically constant, nonzero disturbances (Davison and Smith, 1971), (Kwakernaak and Sivan, 1972, p.278). To accomplish this end, the original system is augmented with a replicate of the constant, nonzero disturbance model, (1.42). Thus the states of the original system are moved onto the manifold that cancels the effect of the disturbance on the controlled variables. The augmented system model used for the state estimator is given by

$$\begin{bmatrix} x \\ d \end{bmatrix}^+ = \begin{bmatrix} A & B_d \\ 0 & I \end{bmatrix} \begin{bmatrix} x \\ d \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u + w \tag{1.43a}$$

$$y = \begin{bmatrix} C & C_d \end{bmatrix} \begin{bmatrix} x \\ d \end{bmatrix} + v \tag{1.43b}$$

and we are free to choose how the integrating disturbance affects the states and measured outputs through the choice of B_d and C_d. The only restriction is that the augmented system is detectable. That restriction can be easily checked using the following result.

Lemma 1.8 (Detectability of the augmented system)**.** *The augmented system* (1.43) *is detectable if and only if the unaugmented system* (A, C) *is detectable, and the following condition holds*

$$\operatorname{rank} \begin{bmatrix} I - A & -B_d \\ C & C_d \end{bmatrix} = n + n_d \tag{1.44}$$

Corollary 1.9 (Dimension of the disturbance)**.** *The maximal dimension of the disturbance* d *in* (1.43) *such that the augmented system is detectable is equal to the number of measurements, that is*

$$n_d \leq p$$

A pair of matrices (B_d, C_d) such that (1.44) is satisfied always exists. In fact, since (A, C) is detectable, the submatrix $\left[\begin{smallmatrix} I-A \\ C \end{smallmatrix}\right] \in \mathbb{R}^{(p+n)\times n}$ has

rank n. Thus, we can choose any $n_d \le p$ columns in $\mathbb{R}^{p+n}$ independent of $\begin{bmatrix} I-A \\ C \end{bmatrix}$ for $\begin{bmatrix} -B_d \\ C_d \end{bmatrix}$.

The state and the additional integrating disturbance are estimated from the plant measurement using a Kalman filter designed for the augmented system. The variances of the stochastic disturbances w and v may be treated as adjustable parameters or found from input-output measurements (Odelson, Rajamani, and Rawlings, 2006). The estimator provides $\hat{x}(k)$ and $\hat{d}(k)$ at each time k. The best forecast of the steady-state disturbance using (1.42) is simply

$$\hat{d}_s = \hat{d}(k)$$

The steady-state target problem is therefore modified to account for the nonzero disturbance $\hat{d}_s$

$$\min_{x_s,u_s} \frac{1}{2}\left(|u_s - u_{\text{sp}}|^2_{R_s} + \left| Cx_s + C_d\hat{d}_s - y_{\text{sp}} \right|^2_{Q_s} \right) \tag{1.45a}$$

subject to

$$\begin{bmatrix} I-A & -B \\ HC & 0 \end{bmatrix} \begin{bmatrix} x_s \\ u_s \end{bmatrix} = \begin{bmatrix} B_d\hat{d}_s \\ r_{\text{sp}} - HC_d\hat{d}_s \end{bmatrix} \tag{1.45b}$$

$$Eu_s \le e \tag{1.45c}$$

$$FCx_s \le f - FC_d\hat{d}_s \tag{1.45d}$$

Comparing (1.41) to (1.45), we see the disturbance model affects the steady-state target determination in four ways.

1. The output target is modified in (1.45a) to account for the effect of the disturbance on the measured output ($y_{\text{sp}} \to y_{\text{sp}} - C_d\hat{d}_s$).

2. The output constraint in (1.45d) is similarly modified ($f \to f - FC_d\hat{d}_s$).

3. The system steady-state relation in (1.45b) is modified to account for the effect of the disturbance on the state evolution ($0 \to B_d\hat{d}_s$).

4. The controlled variable target in (1.45b) is modified to account for the effect of the disturbance on the controlled variable ($r_{\text{sp}} \to r_{\text{sp}} - HC_d\hat{d}_s$).

Given the steady-state target, the same dynamic regulation problem as presented in the tracking section, Section 1.5, is used for the regulator.

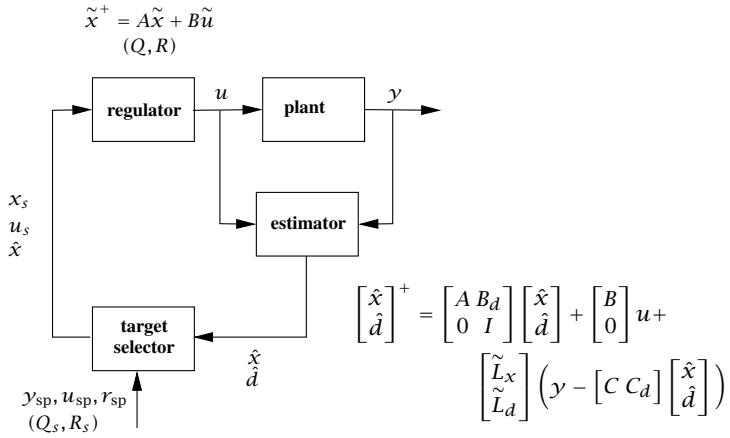


Figure 1.6: MPC controller consisting of: receding horizon regulator, state estimator, and target selector; for simplicity we show the steady-state Kalman predictor form of the state estimator where $\hat{x} := \hat{x}(k \mid k-1)$ and $\tilde{L}_x := AL_x + B_dL_d$ and $\tilde{L}_d := L_d$.

In other words, the regulator is based on the deterministic system (A, B) in which the current state is $\hat{x}(k) - x_s$ and the goal is to take the system to the origin.

The following lemma summarizes the offset-free control property of the combined control system.

Lemma 1.10 (Offset-free control)**.** *Consider a system controlled by the MPC algorithm as shown in Figure 1.6. The target problem* (1.45) *is assumed feasible. Augment the system model with a number of integrating disturbances equal to the number of measurements* ($n_d = p$)*; choose any* $B_d \in \mathbb{R}^{n\times p}$, $C_d \in \mathbb{R}^{p\times p}$ *such that*

$$\operatorname{rank}\begin{bmatrix} I - A & -B_d \\ C & C_d \end{bmatrix} = n + p$$

If the plant output $y(k)$ *goes to steady state* y_s*, the closed-loop system is stable, and constraints are not active at steady state, then there is zero offset in the controlled variables, that is*

$$Hy_s = r_{sp}$$

The proof of this lemma is given in Pannocchia and Rawlings (2003). It may seem surprising that the number of integrating disturbances

must be equal to the number of *measurements* used for feedback rather than the number of *controlled variables* to guarantee offset-free control. To gain insight into the reason, consider the disturbance part (bottom half) of the Kalman filter equations shown in Figure 1.6

$$\hat{d}^+ = \hat{d} + L_d\left(y - \begin{bmatrix} C & C_d \end{bmatrix}\begin{bmatrix} \hat{x} \\ \hat{d} \end{bmatrix}\right)$$

Because of the integrator, the disturbance estimate cannot converge until

$$L_d\left(y - \begin{bmatrix} C & C_d \end{bmatrix}\begin{bmatrix} \hat{x} \\ \hat{d} \end{bmatrix}\right) = 0$$

But notice this condition merely restricts the output prediction error to lie in the nullspace of the matrix L_d, which is an $n_d \times p$ matrix. If we choose $n_d = n_c < p$, then the number of columns of L_d is greater than the number of rows and L_d has a nonzero nullspace.[8] In general, we require the output prediction error to be *zero* to achieve zero offset independently of the regulator tuning. For L_d to have only the zero vector in its nullspace, we require $n_d \geq p$. Since we also know $n_d \leq p$ from Corollary 1.9, we conclude $n_d = p$.

Notice also that Lemma 1.10 does not require that the plant output be generated by the model. The theorem applies regardless of what generates the plant output. *If the plant is identical to the system plus disturbance model assumed in the estimator*, then the conclusion can be strengthened. In the nominal case without measurement or process noise ($w = 0$, $v = 0$), *for a set of plant initial states*, the closed-loop system *converges to a steady state* and the feasible steady-state target is achieved leading to zero offset in the controlled variables. Characterizing the set of initial states in the region of convergence, and stabilizing the system when the plant and the model differ, are treated in Chapters 3 and 5. We conclude the chapter with a nonlinear example that demonstrates the use of Lemma 1.10.

Example 1.11: More measured outputs than inputs and zero offset

We consider a well-stirred chemical reactor depicted in Figure 1.7, as in Pannocchia and Rawlings (2003). An irreversible, first-order reaction A⟶ B occurs in the liquid phase and the reactor temperature is

[8] This is another consequence of the fundamental theorem of linear algebra. The result is depicted in Figure A.1.

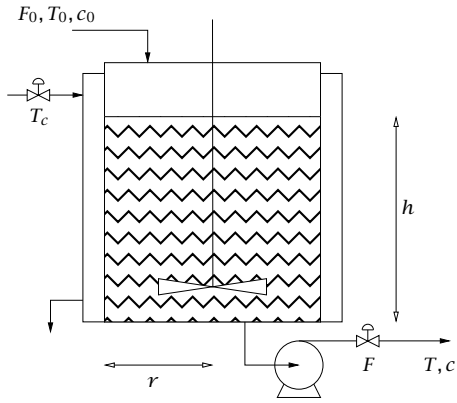


Figure 1.7: Schematic of the well-stirred reactor.

regulated with external cooling. Mass and energy balances lead to the following nonlinear state space model

$$\begin{aligned}
\frac{dc}{dt} &= \frac{F_0(c_0 - c)}{\pi r^2 h} - k_0 \exp\left(-\frac{E}{RT}\right) c \\
\frac{dT}{dt} &= \frac{F_0(T_0 - T)}{\pi r^2 h} + \frac{-\Delta H}{\rho C_p} k_0 \exp\left(-\frac{E}{RT}\right) c + \frac{2U}{r \rho C_p}(T_c - T) \\
\frac{dh}{dt} &= \frac{F_0 - F}{\pi r^2}
\end{aligned}$$

The controlled variables are h, the level of the tank, and c, the molar concentration of species A. The additional state variable is T, the reactor temperature; while the manipulated variables are T_c, the coolant liquid temperature, and F, the outlet flowrate. Moreover, it is assumed that the inlet flowrate acts as an unmeasured disturbance. The model parameters in nominal conditions are reported in Table 1.1. The open-loop stable steady-state operating conditions are the following

$$c^s = 0.878\,\text{kmol/m}^3 \qquad T^s = 324.5\,\text{K} \qquad h^s = 0.659\,\text{m}$$
$$T_c^s = 300\,\text{K} \qquad F^s = 0.1\,\text{m}^3/\text{min}$$

Using a sampling time of 1 min, a linearized discrete state space model is obtained and, assuming that all the states are measured, the state space variables are

$$x = \begin{bmatrix} c - c^s \\ T - T^s \\ h - h^s \end{bmatrix} \qquad u = \begin{bmatrix} T_c - T_c^s \\ F - F^s \end{bmatrix} \qquad y = \begin{bmatrix} c - c^s \\ T - T^s \\ h - h^s \end{bmatrix} \qquad p = F_0 - F_0^s$$

Parameter	Nominal value	Units
F_0	0.1	m^3/min
T_0	350	K
c_0	1	$kmol/m^3$
r	0.219	m
k_0	7.2×10^{10}	min^{-1}
E/R	8750	K
U	54.94	$kJ/min \cdot m^2 \cdot K$
ρ	1000	kg/m^3
C_p	0.239	$kJ/kg \cdot K$
ΔH	-5×10^4	kJ/kmol

Table 1.1: Parameters of the well-stirred reactor.

The corresponding linear model is

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) + B_p p \\ y(k) &= Cx(k) \end{aligned}$$

in which

$$A = \begin{bmatrix} 0.2681 & -0.00338 & -0.00728 \\ 9.703 & 0.3279 & -25.44 \\ 0 & 0 & 1 \end{bmatrix} \qquad C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$B = \begin{bmatrix} -0.00537 & 0.1655 \\ 1.297 & 97.91 \\ 0 & -6.637 \end{bmatrix} \qquad B_p = \begin{bmatrix} -0.1175 \\ 69.74 \\ 6.637 \end{bmatrix}$$

(a) Since we have two inputs, T_c and F, we try to remove offset in two controlled variables, c and h. Model the disturbance with *two* integrating output disturbances on the two controlled variables. Assume that the covariances of the state noises are zero except for the two integrating states. Assume that the covariances of the three measurements' noises are also zero.

Notice that although there are only two controlled variables, this choice of *two* integrating disturbances does not follow the prescription of Lemma 1.10 for zero offset.

Simulate the response of the controlled system after a 10% increase in the inlet flowrate F_0 at time $t = 10$ min. Use the nonlin-

ear differential equations for the plant model. Do you have steady offset in any of the outputs? Which ones?

(b) Follow the prescription of Lemma 1.10 and choose a disturbance model with *three* integrating modes. Can you choose three integrating output disturbances for this plant? If so, prove it. If not, state why not.

(c) Again choose a disturbance model with three integrating modes; choose two integrating output disturbances on the two controlled variables. Choose one integrating input disturbance on the outlet flowrate F. Is the augmented system detectable?

Simulate again the response of the controlled system after a 10% increase in the inlet flowrate F_0 at time $t = 10$ min. Again use the nonlinear differential equations for the plant model. Do you have steady offset in any of the outputs? Which ones?

Compare and contrast the closed-loop performance for the design with two integrating disturbances and the design with three integrating disturbances. Which control system do you recommend and why?

Solution

(a) Integrating disturbances are added to the two controlled variables (first and third outputs) by choosing

$$C_d = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \qquad B_d = 0$$

The results with two integrating disturbances are shown in Figures 1.8 and 1.9. Notice that despite adding integrating disturbances to the two controlled variables, c and h, both of these controlled variables as well as the third output, T, all display nonzero offset at steady state.

(b) A third integrating disturbance is added to the second output giving

$$C_d = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \qquad B_d = 0$$

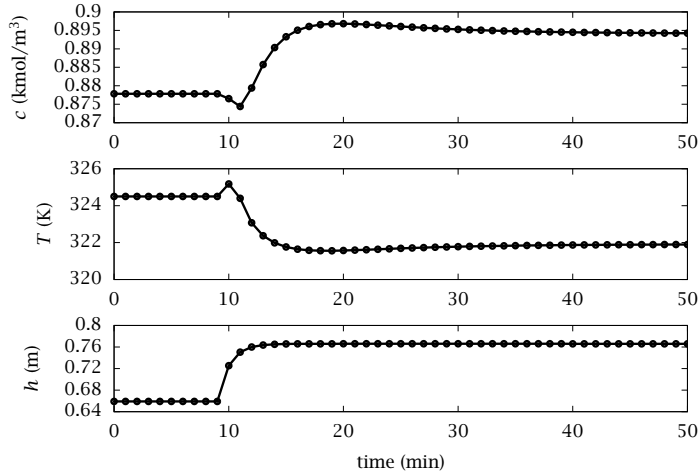


Figure 1.8: Three measured outputs versus time after a step change in inlet flowrate at 10 minutes; $n_d = 2$.

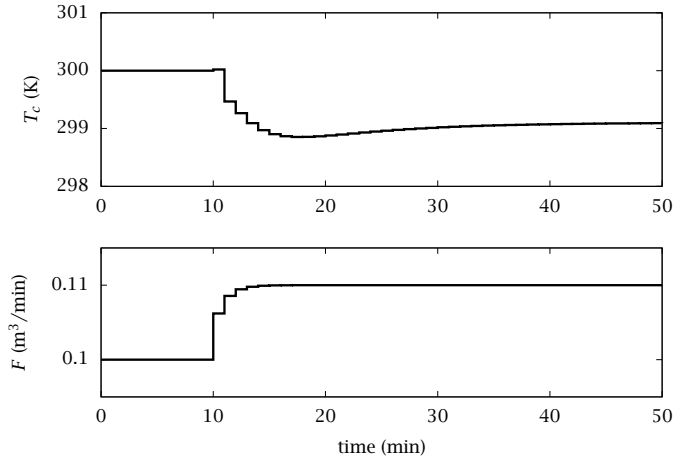


Figure 1.9: Two manipulated inputs versus time after a step change in inlet flowrate at 10 minutes; $n_d = 2$.

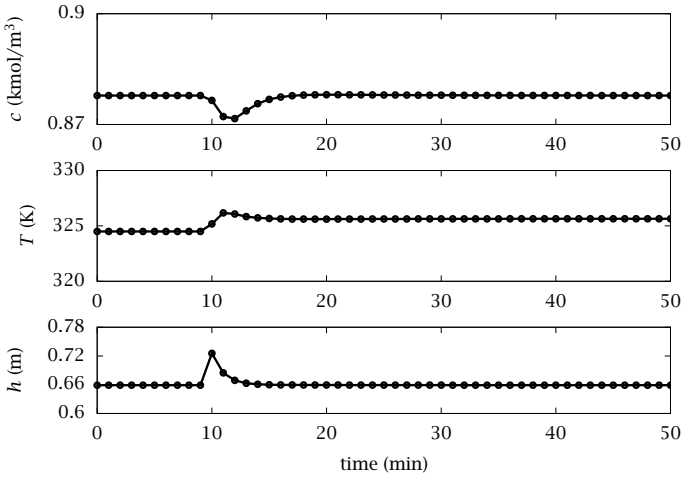


Figure 1.10: Three measured outputs versus time after a step change in inlet flowrate at 10 minutes; $n_d = 3$.

The augmented system is not detectable with this disturbance model. The rank of $\begin{bmatrix} I-A & -B_d \\ C & C_d \end{bmatrix}$ is only 5 instead of 6. The problem here is that the system level is itself an integrator, and we cannot distinguish h from the integrating disturbance added to h.

(c) Next we try three integrating disturbances: two added to the two controlled variables, and one added to the second manipulated variable

$$C_d = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \qquad B_d = \begin{bmatrix} 0 & 0 & 0.1655 \\ 0 & 0 & 97.91 \\ 0 & 0 & -6.637 \end{bmatrix}$$

The augmented system is detectable for this disturbance model.

The results for this choice of three integrating disturbances are shown in Figures 1.10 and 1.11. Notice that we have zero offset in the two controlled variables, c and h, and have successfully forced the steady-state effect of the inlet flowrate disturbance entirely into the second output, T.

Notice also that the dynamic behavior of all three outputs is superior to that achieved with the model using two integrating disturbances. The true disturbance, which is a step at the inlet flowrate,

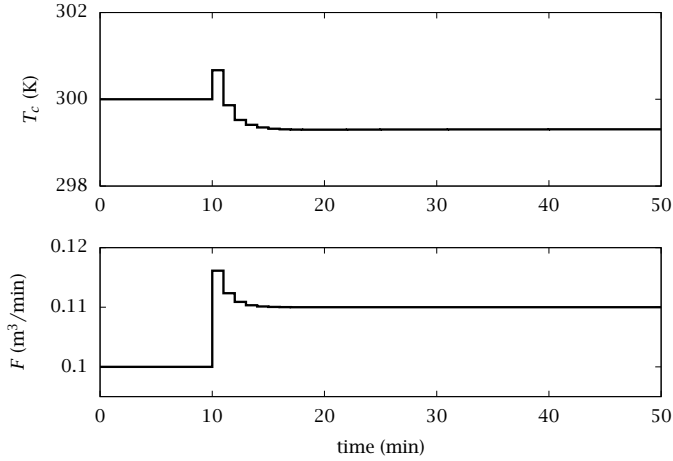


Figure 1.11: Two manipulated inputs versus time after a step change in inlet flowrate at 10 minutes; $n_d = 3$.

is better represented by including the integrator in the outlet flowrate. With a more accurate disturbance model, better overall control is achieved. The controller uses smaller manipulated variable action and also achieves better output variable behavior. An added bonus is that steady offset is removed in the maximum possible number of outputs. □

Further notation

G	transfer function matrix
m	mean of normally distributed random variable
T	reactor temperature
$\tilde{u}$	input deviation variable
x, y, z	spatial coordinates for a distributed system
$\tilde{x}$	state deviation variable

1.6 Exercises

Exercise 1.1: State space form for chemical reaction model

Consider the following chemical reaction kinetics for a two-step series reaction

$$\mathrm{A} \xrightarrow{k_1} \mathrm{B} \qquad \mathrm{B} \xrightarrow{k_2} \mathrm{C}$$

We wish to follow the reaction in a constant volume, well-mixed, batch reactor. As taught in the undergraduate chemical engineering curriculum, we proceed by writing material balances for the three species giving

$$\frac{dc_A}{dt} = -r_1 \qquad \frac{dc_B}{dt} = r_1 - r_2 \qquad \frac{dc_C}{dt} = r_2$$

in which c_j is the concentration of species j, and r_1 and r_2 are the rates (mol/(time·vol)) at which the two reactions occur. We then assume some rate law for the reaction kinetics, such as

$$r_1 = k_1 c_A \qquad r_2 = k_2 c_B$$

We substitute the rate laws into the material balances and specify the starting concentrations to produce three differential equations for the three species concentrations.

(a) Write the linear state space model for the deterministic series chemical reaction model. Assume we can measure the component A concentration. What are x, y, A, B, C, and D for this model?

(b) Simulate this model with initial conditions and parameters given by

$$c_{A0} = 1 \quad c_{B0} = c_{C0} = 0 \qquad k_1 = 2 \quad k_2 = 1$$

Exercise 1.2: Distributed systems and time delay

We assume familiarity with the transfer function of a time delay from an undergraduate systems course

$$\overline{y}(s) = e^{-\theta s}\overline{u}(s)$$

Let's see the connection between the delay and the distributed systems, which give rise to it. A simple physical example of a time delay is the delay caused by transport in a flowing system. Consider plug flow in a tube depicted in Figure 1.12.

(a) Write down the equation of change for moles of component j for an arbitrary volume element and show that

$$\frac{\partial c_j}{\partial t} = -\nabla \cdot (c_j v_j) + R_j$$

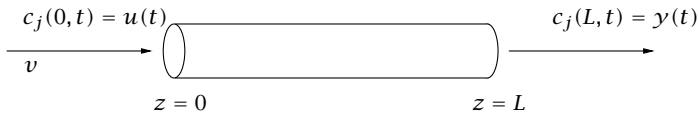


Figure 1.12: Plug-flow reactor.

in which c_j is the molar concentration of component j, v_j is the velocity of component j, and R_j is the production rate of component j due to chemical reaction.[9]

Plug flow means the fluid velocity of all components is purely in the z direction, and is independent of r and θ and, we assume here, z

$$v_j = v\delta_z$$

(b) Assuming plug flow and neglecting chemical reaction in the tube, show that the equation of change reduces to

$$\frac{\partial c_j}{\partial t} = -v\frac{\partial c_j}{\partial z} \tag{1.46}$$

This equation is known as a hyperbolic, first-order partial differential equation. Assume the boundary and initial conditions are

$$c_j(z,t) = u(t) \qquad 0 = z \qquad t \geq 0 \tag{1.47}$$

$$c_j(z,t) = c_{j0}(z) \qquad 0 \leq z \leq L \quad t = 0 \tag{1.48}$$

In other words, we are using the feed concentration as the manipulated variable, $u(t)$, and the tube starts out with some initial concentration profile of component j, $c_{j0}(z)$.

(c) Show that the solution to (1.46) with these boundary conditions is

$$c_j(z,t) = \begin{cases} u(t - z/v) & vt > z \\ c_{j0}(z - vt) & vt < z \end{cases} \tag{1.49}$$

(d) If the reactor starts out empty of component j, show that the transfer function between the outlet concentration, $y = c_j(L,t)$, and the inlet concentration, $c_j(0, t) = u(t)$, is a time delay. What is the value of θ?

Exercise 1.3: Pendulum in state space

Consider the pendulum suspended at the end of a rigid link depicted in Figure 1.13. Let r and θ denote the polar coordinates of the center of the pendulum, and let $p = r\delta_r$ be the position vector of the pendulum, in which δ_r and δ_θ are the unit vectors in polar coordinates. We wish to determine a state space description of the system. We are able to apply a torque T to the pendulum as our manipulated variable. The pendulum has mass m, the only other external force acting on the pendulum is gravity, and we neglect friction. The link provides force $-t\delta_r$ necessary to maintain the pendulum at distance $r = R$ from the axis of rotation, and we measure this force t.

(a) Provide expressions for the four partial derivatives for changes in the unit vectors with r and θ

$$\frac{\partial \delta_r}{\partial r} \qquad \frac{\partial \delta_r}{\partial \theta} \qquad \frac{\partial \delta_\theta}{\partial r} \qquad \frac{\partial \delta_\theta}{\partial \theta}$$

(b) Use the chain rule to find the velocity of the pendulum in terms of the time derivatives of r and θ. Do not simplify yet by assuming r is constant. We want the general result.

[9] You will need the Gauss divergence theorem and 3D Leibniz formula to go from a mass balance on a volume element to the equation of continuity.

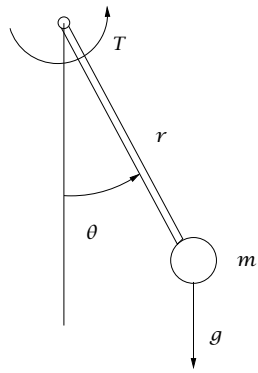


Figure 1.13: Pendulum with applied torque.

(c) Differentiate again to show that the acceleration of the pendulum is

$$\ddot{p} = (\ddot{r} - r\dot{\theta}^2)\delta_r + (r\ddot{\theta} + 2\dot{r}\dot{\theta})\delta_\theta$$

(d) Use a momentum balance on the pendulum mass (you may assume it is a point mass) to determine both the force exerted by the link

$$t = mR\dot{\theta}^2 + mg\cos\theta$$

and an equation for the acceleration of the pendulum due to gravity and the applied torque

$$mR\ddot{\theta} - T/R + mg\sin\theta = 0$$

(e) Define a state vector and give a state space description of your system. What is the physical significance of your state. Assume you measure the force exerted by the link.

One answer is

$$\begin{aligned}\frac{dx_1}{dt} &= x_2\\ \frac{dx_2}{dt} &= -(g/R)\sin x_1 + u\\ y &= mRx_2^2 + mg\cos x_1\end{aligned}$$

in which $u = T/(mR^2)$

Exercise 1.4: Time to Laplace domain

Take the Laplace transform of the following set of differential equations and find the transfer function, $G(s)$, connecting $\overline{u}(s)$ and $\overline{y}(s)$, $\overline{y} = G\overline{u}$

$$\begin{aligned}\frac{dx}{dt} &= Ax + Bu\\ y &= Cx + Du\end{aligned} \tag{1.50}$$

For $x \in \mathbb{R}^n$, $y \in \mathbb{R}^p$, and $u \in \mathbb{R}^m$, what is the dimension of the G matrix? What happens to the initial condition, $x(0) = x_0$?

Exercise 1.5: Converting between continuous and discrete time models

Given a prescribed $u(t)$, derive and check the solution to (1.50). Given a prescribed $u(k)$ sequence, what is the solution to the discrete time model

$$x(k+1) = \widetilde{A}x(k) + \widetilde{B}u(k)$$
$$y(k) = \widetilde{C}x(k) + \widetilde{D}u(k)$$

(a) Compute $\widetilde{A}, \widetilde{B}, \widetilde{C}$, and $\widetilde{D}$ so that the two solutions agree at the sample times for a zero-order hold input, i.e., $y(k) = y(t_k)$ for $u(t) = u(k),\ t \in (t_k, t_{k+1})$ in which $t_k = k\Delta$ for sample time Δ.

(b) Is your result valid for A singular? If not, how can you find $\widetilde{A}, \widetilde{B}, \widetilde{C}$, and $\widetilde{D}$ for this case?

Exercise 1.6: Continuous to discrete time conversion for nonlinear models

Consider the autonomous nonlinear differential equation model

$$\frac{dx}{dt} = f(x,u)$$
$$x(0) = x_0 \tag{1.51}$$

Given a zero-order hold on the input, let $s(t,u,x_0), 0 \le t \le \Delta$, be the solution to (1.51) given initial condition x_0 at time $t = 0$, and constant input u is applied for t in the interval $0 \le t \le \Delta$. Consider also the nonlinear discrete time model

$$x(k+1) = F(x(k), u(k))$$

(a) What is the relationship between F and s so that the solution of the discrete time model agrees at the sample times with the continuous time model with a zero-order hold?

(b) Assume f is linear and apply this result to check the result of Exercise 1.5.

Exercise 1.7: Commuting functions of a matrix

Although matrix multiplication does not commute in general

$$AB \neq BA$$

multiplication of functions of the same matrix do commute. You may have used the following fact in Exercise 1.5

$$A^{-1}\exp(At) = \exp(At)A^{-1} \tag{1.52}$$

(a) Prove that (1.52) is true assuming A has distinct eigenvalues and can therefore be represented as

$$A = Q\Lambda Q^{-1} \qquad \Lambda = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{bmatrix}$$

in which Λ is a diagonal matrix containing the eigenvalues of A, and Q is the matrix of eigenvectors such that

$$Aq_i = \lambda_i q_i, \qquad i = 1, \ldots, n$$

in which q_i is the ith column of matrix Q.

(b) Prove the more general relationship

$$f(A)g(A) = g(A)f(A) \tag{1.53}$$

in which f and g are any functions definable by Taylor series.

(c) Prove that (1.53) is true without assuming the eigenvalues are distinct.

Hint: use the Taylor series defining the functions and apply the Cayley-Hamilton theorem (Horn and Johnson, 1985, pp. 86–87).

Exercise 1.8: Finite difference formula and approximating the exponential

Instead of computing the exact conversion of a continuous time to a discrete time system as in Exercise 1.5, assume instead one simply approximates the time derivative with a first-order finite difference formula

$$\frac{dx}{dt} \approx \frac{x(t_{k+1}) - x(t_k)}{\Delta}$$

with step size equal to the sample time, Δ. For this approximation of the continuous time system, compute $\tilde{A}$ and $\tilde{B}$ so that the discrete time system agrees with the approximate continuous time system at the sample times. Comparing these answers to the exact solution, what approximation of $e^{A\Delta}$ results from the finite difference approximation? When is this a good approximation of $e^{A\Delta}$?

Exercise 1.9: Mapping eigenvalues of continuous time systems to discrete time systems

Consider the continuous time differential equation and discrete time difference equation

$$\frac{dx}{dt} = Ax$$
$$x^+ = \tilde{A}x$$

and the transformation

$$\tilde{A} = e^{A\Delta}$$

Consider the scalar A case.

(a) What A represents an integrator in continuous time? What is the corresponding $\tilde{A}$ value for the integrator in discrete time?

(b) What A give purely oscillatory solutions? What are the corresponding $\tilde{A}$?

(c) For what A is the solution of the ODE stable? Unstable? What are the corresponding $\tilde{A}$?

(d) Sketch and label these A and $\tilde{A}$ regions in two complex-plane diagrams.

Exercise 1.10: State space realization

Define a state vector and realize the following models as state space models **by hand**. One should do a few by hand to understand what the Octave or MATLAB calls are doing. Answer the following questions. What is the connection between the poles of G and the state space description? For what kinds of $G(s)$ does one obtain a nonzero D matrix? What is the order and gain of these systems? Is there a connection between order and the numbers of inputs and outputs?

(a) $G(s) = \dfrac{1}{2s+1}$

(b) $G(s) = \dfrac{1}{(2s+1)(3s+1)}$

(c) $G(s) = \dfrac{2s+1}{3s+1}$

(d) $y(k+1) = y(k) + 2u(k)$

(e) $y(k+1) = a_1 y(k) + a_2 y(k-1) + b_1 u(k) + b_2 u(k-1)$

Exercise 1.11: Minimal realization

Find minimal realizations of the state space models you found by hand in Exercise 1.10. Use Octave or MATLAB for computing minimal realizations. Were any of your hand realizations nonminimal?

Exercise 1.12: Partitioned matrix inversion lemma

Let matrix Z be partitioned into

$$Z = \begin{bmatrix} B & C \\ D & E \end{bmatrix}$$

and assume Z^{-1}, B^{-1} and E^{-1} exist.

(a) Perform row elimination and show that

$$Z^{-1} = \begin{bmatrix} B^{-1} + B^{-1}C(E - DB^{-1}C)^{-1}DB^{-1} & -B^{-1}C(E - DB^{-1}C)^{-1} \\ -(E - DB^{-1}C)^{-1}DB^{-1} & (E - DB^{-1}C)^{-1} \end{bmatrix}$$

Note that this result is still valid if E is singular.

(b) Perform column elimination and show that

$$Z^{-1} = \begin{bmatrix} (B - CE^{-1}D)^{-1} & -(B - CE^{-1}D)^{-1}CE^{-1} \\ -E^{-1}D(B - CE^{-1}D)^{-1} & E^{-1} + E^{-1}D(B - CE^{-1}D)^{-1}CE^{-1} \end{bmatrix}$$

Note that this result is still valid if B is singular.

(c) A host of other useful control-related inversion formulas follow from these results. Equate the (1,1) or (2,2) entries of Z^{-1} and derive the identity

$$(A + BCD)^{-1} = A^{-1} - A^{-1}B(DA^{-1}B + C^{-1})^{-1}DA^{-1} \tag{1.54}$$

A useful special case of this result is

$$(I + X^{-1})^{-1} = I - (I + X)^{-1}$$

(d) Equate the (1,2) or (2,1) entries of Z^{-1} and derive the identity

$$(A + BCD)^{-1}BC = A^{-1}B(DA^{-1}B + C^{-1})^{-1} \tag{1.55}$$

Equations (1.54) and (1.55) prove especially useful in rearranging formulas in least squares estimation.

Exercise 1.13: Perturbation to an asymptotically stable linear system

Given the system

$$x^+ = Ax + Bu$$

If A is an asymptotically stable matrix, prove that if $u(k) \to 0$, then $x(k) \to 0$.

Exercise 1.14: Exponential stability of a perturbed linear system

Given the system

$$x^+ = Ax + Bu$$

If A is an asymptotically stable matrix, prove that if $u(k)$ decreases exponentially to zero, then $x(k)$ decreases exponentially to zero.

Exercise 1.15: Are we going forward or backward today?

In the chapter we derived the solution to

$$\min_{w,x,y} f(w,x) + g(x,y) + h(y,z)$$

in which z is a fixed parameter using forward dynamic programming (DP)

$$\begin{aligned}
&\overline{y}^0(z) \\
&\tilde{\overline{x}}^0(z) = \overline{x}^0(\overline{y}^0(z)) \\
&\tilde{\overline{w}}^0(z) = \overline{w}^0(\overline{x}^0(\overline{y}^0(z)))
\end{aligned}$$

(a) Solve for optimal w as a function of z using backward DP.

(b) Is forward or backward DP more efficient if you want optimal w as a function of z?

Exercise 1.16: Method of Lagrange multipliers

Consider the objective function $V(x) = (1/2)x'Hx + h'x$ and optimization problem

$$\min_x V(x) \tag{1.56}$$

subject to

$$Dx = d$$

in which $H > 0$, $x \in \mathbb{R}^n$, $d \in \mathbb{R}^m$, $m < n$, i.e., fewer constraints than decisions. Rather than partially solving for x using the constraint and eliminating it, we make use of the method of Lagrange multipliers for treating the equality constraints (Fletcher, 1987; Nocedal and Wright, 2006).

In the method of Lagrange multipliers, we augment the objective function with the constraints to form the Lagrangian function, L

$$L(x,\lambda) = (1/2)x'Hx + h'x - \lambda'(Dx - d)$$

in which $\lambda \in \mathbb{R}^m$ is the vector of Lagrange multipliers. The necessary and sufficient conditions for a global minimizer are that the partial derivatives of L with respect to x and λ vanish (Nocedal and Wright, 2006, p. 451), (Fletcher, 1987, p.198,236).

(a) Show that the necessary and sufficient conditions are equivalent to the matrix equation

$$\begin{bmatrix} H & -D' \\ -D & 0 \end{bmatrix} \begin{bmatrix} x \\ \lambda \end{bmatrix} = - \begin{bmatrix} h \\ d \end{bmatrix} \tag{1.57}$$

The solution to (1.57) then provides the solution to the original problem (1.56).

(b) We note one other important feature of the Lagrange multipliers, their relationship to the optimal cost of the purely quadratic case. For $h = 0$, the cost is given by

$$V^0 = (1/2)(x^0)'Hx^0$$

Show that this can also be expressed in terms of λ^0 by the following

$$V^0 = (1/2)d'\lambda^0$$

Exercise 1.17: Minimizing a constrained, quadratic function

Consider optimizing the positive definite quadratic function subject to a linear constraint

$$\min_x (1/2)x'Hx \qquad \text{s.t. } Ax = b$$

Using the method of Lagrange multipliers presented in Exercise 1.16, show that the optimal solution, multiplier, and cost are given by

$$\begin{aligned} x^0 &= H^{-1}A'(AH^{-1}A')^{-1}b \\ \lambda^0 &= (AH^{-1}A')^{-1}b \\ V^0 &= (1/2)b'(AH^{-1}A')^{-1}b \end{aligned}$$

Exercise 1.18: Minimizing a partitioned quadratic function

Consider the partitioned constrained minimization

$$\min_{x_1,x_2} (1/2) \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}' \begin{bmatrix} H_1 & \\ & H_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

subject to

$$\begin{bmatrix} D & I \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = d$$

The solution to this optimization is required in two different forms, depending on whether one is solving an estimation or regulation problem. Show that the solution can be expressed in the following two forms if both H_1 and H_2 are full rank.

- Regulator form

$$\begin{aligned} V^0(d) &= (1/2)d'(H_2 - H_2D(D'H_2D + H_1)^{-1}D'H_2)d \\ x_1^0(d) &= \widetilde{K}d \qquad \widetilde{K} = (D'H_2D + H_1)^{-1}D'H_2 \\ x_2^0(d) &= (I - D\widetilde{K})d \end{aligned}$$

- Estimator form

$$\begin{aligned} V^0(d) &= (1/2)d'(DH_1^{-1}D' + H_2^{-1})^{-1}d \\ x_1^0(d) &= \widetilde{L}d \qquad \widetilde{L} = H_1^{-1}D'(DH_1^{-1}D' + H_2^{-1})^{-1} \\ x_2^0(d) &= (I - D\widetilde{L})d \end{aligned}$$

Exercise 1.19: Stabilizability and controllability canonical forms

Consider the partitioned system

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}^+ = \begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} B_1 \\ 0 \end{bmatrix} u$$

with (A_{11}, B_1) controllable. This form is known as controllability canonical form.

(a) Show that the system is *not* controllable by checking the rank of the controllability matrix.

(b) Show that the modes x_1 can be controlled from any $x_1(0)$ to any $x_1(n)$ with a sequence of inputs $u(0), \ldots, u(n-1)$, but the modes x_2 *cannot* be controlled from any $x_2(0)$ to any $x_2(n)$. The states x_2 are termed the uncontrollable modes.

(c) If A_{22} is stable the system is termed *stabilizable*. Although not all modes can be controlled, the uncontrollable modes are stable and decay to steady state.

The following lemma gives an equivalent condition for stabilizability.

Lemma 1.12 (Hautus lemma for stabilizability)**.** *A system is stabilizable if and only if*

$$\operatorname{rank} \begin{bmatrix} \lambda I - A & B \end{bmatrix} = n \qquad \text{for all } |\lambda| \geq 1$$

Prove this lemma using Lemma 1.2 as the condition for controllability.

Exercise 1.20: Regulator stability, stabilizable systems, and semidefinite state penalty

(a) Show that the infinite horizon LQR is stabilizing for (A, B) *stabilizable* with R, $Q > 0$.

(b) Show that the infinite horizon LQR is stabilizing for (A, B) stabilizable and $R > 0$, $Q \geq 0$, and (A, Q) detectable. Discuss what happens to the controller's stabilizing property if Q is not positive semidefinite or (A, Q) is not detectable.

Exercise 1.21: Time-varying linear quadratic problem

Consider the time-varying version of the LQ problem solved in the chapter. The system model is

$$x(k+1) = A(k)x(k) + B(k)u(k)$$

The objective function also contains time-varying penalties

$$\min_{\mathbf{u}} V(x(0), \mathbf{u}) = \frac{1}{2} \left(\sum_{k=0}^{N-1} \left(x(k)' Q(k) x(k) + u(k)' R(k) u(k) \right) + x(N)' Q(N) x(N) \right)$$

subject to the model. Notice the penalty on the final state is now simply $Q(N)$ instead of P_f.

Apply the DP argument to this problem and determine the optimal input sequence and cost. Can this problem also be solved in closed form like the time-invariant case?

Exercise 1.22: Steady-state Riccati equation

Generate a random A and B for a system model for whatever $n(\geq 3)$ and $m(\geq 3)$ you wish. Choose a positive semidefinite Q and positive definite R of the appropriate sizes.

(a) Iterate the DARE by hand with Octave or MATLAB until Π stops changing. Save this result. Now call the MATLAB or Octave function to solve the steady-state DARE. Do the solutions agree? Where in the complex plane are the eigenvalues of $A+BK$? Increase the size of Q relative to R. Where do the eigenvalues move?

(b) Repeat for a singular A matrix. What happens to the two solution techniques?

(c) Repeat for an unstable A matrix.

Exercise 1.23: Positive definite Riccati iteration

If $\Pi(k), Q, R > 0$ in (1.10), show that $\Pi(k-1) > 0$.
Hint: apply (1.54) to the term $(B'\Pi(k)B + R)^{-1}$.

Exercise 1.24: Existence and uniqueness of the solution to constrained least squares

Consider the least squares problem subject to linear constraint

$$\min_x (1/2)x'Qx \qquad \text{subject to} \quad Ax = b$$

in which $x \in \mathbb{R}^n$, $b \in \mathbb{R}^p$, $Q \in \mathbb{R}^{n\times n}$, $Q \geq 0$, $A \in \mathbb{R}^{p\times n}$. Show that this problem has a solution for every b and the solution is unique if and only if

$$\text{rank}(A) = p \qquad \text{rank}\begin{bmatrix} Q \\ A \end{bmatrix} = n$$

Exercise 1.25: Rate-of-change penalty

Consider the generalized LQR problem with the cross term between $x(k)$ and $u(k)$

$$V(x(0), \mathbf{u}) = \frac{1}{2}\sum_{k=0}^{N-1} \left(x(k)'Qx(k) + u(k)'Ru(k) + 2x(k)'Mu(k)\right) + (1/2)x(N)'P_f x(N)$$

(a) Solve this problem with backward DP and write out the Riccati iteration and feedback gain.

(b) Control engineers often wish to tune a regulator by penalizing the rate of change of the input rather than the absolute size of the input. Consider the additional positive definite penalty matrix S and the modified objective function

$$V(x(0), \mathbf{u}) = \frac{1}{2}\sum_{k=0}^{N-1} \left(x(k)'Qx(k) + u(k)'Ru(k) + \Delta u(k)'S\Delta u(k)\right) + (1/2)x(N)'P_f x(N)$$

in which $\Delta u(k) = u(k) - u(k-1)$. Show that you can augment the state to include $u(k-1)$ via

$$\widetilde{x}(k) = \begin{bmatrix} x(k) \\ u(k-1) \end{bmatrix}$$

and reduce this new problem to the standard LQR with the cross term. What are $\widetilde{A}$, $\widetilde{B}$, $\widetilde{Q}$, $\widetilde{R}$, and $\widetilde{M}$ for the augmented problem (Rao and Rawlings, 1999)?

Exercise 1.26: Existence, uniqueness and stability with the cross term

Consider the linear quadratic problem with system

$$x^+ = Ax + Bu \tag{1.58}$$

and infinite horizon cost function

$$V(x(0), \mathbf{u}) = (1/2) \sum_{k=0}^{\infty} x(k)'Qx(k) + u(k)'Ru(k)$$

The existence, uniqueness and stability conditions for this problem are: (A, B) stabilizable, $Q \geq 0$, (A, Q) detectable, and $R > 0$. Consider the modified objective function with the cross term

$$V = (1/2) \sum_{k=0}^{\infty} x(k)'Qx(k) + u(k)'Ru(k) + 2x(k)'Mu(k) \tag{1.59}$$

(a) Consider reparameterizing the input as

$$v(k) = u(k) + Tx(k) \tag{1.60}$$

Choose T such that the cost function in x and v does not have a cross term, and express the existence, uniqueness and stability conditions for the transformed system. Goodwin and Sin (1984, p.251) discuss this procedure in the state estimation problem with nonzero covariance between state and output measurement noises.

(b) Translate and simplify these to obtain the existence, uniqueness and stability conditions for the original system with cross term.

Exercise 1.27: Forecasting and variance increase or decrease

Given positive definite initial state variance $P(0)$ and process disturbance variance Q, the variance after forecasting one sample time was shown to be

$$P^-(1) = AP(0)A' + Q$$

(a) If A is stable, is it true that $AP(0)A' < P(0)$? If so, prove it. If not, provide a counterexample.

(b) If A is unstable, is it true that $AP(0)A' > P(0)$? If so, prove it. If not, provide a counterexample.

(c) If the magnitudes of *all* the eigenvalues of A are unstable, is it true that $AP(0)A' > P(0)$? If so, prove it. If not, provide a counterexample.

Exercise 1.28: Convergence of MHE with zero prior weighting

Show that the simplest form of MHE defined in (1.32) and (1.33) is also a convergent estimator for an observable system. What restrictions on the horizon length N do you require for this result to hold?

Hint: you can solve the MHE optimization problem by inspection when there is no prior weighting of the data.

Exercise 1.29: Symmetry in regulation and estimation

In this exercise we display the symmetry of the backward DP recursion for regulation, and the forward DP recursion for estimation. In the regulation problem we solve at stage k

$$\min_{x,u} \ell(z,u) + V_k^0(x) \qquad \text{s.t. } x = Az + Bu$$

In backward DP, x is the state at the current stage and z is the state at the previous stage. The stage cost and cost to go are given by

$$\ell(z,u) = (1/2)(z'Qz + u'Ru) \qquad V_k^0(x) = (1/2)x'\Pi(k)x$$

and the optimal cost is $V_{k-1}^0(z)$ since z is the state at the previous stage.

In estimation we solve at stage k

$$\min_{x,w} \ell(z,w) + V_k^0(x) \qquad \text{s.t. } z = Ax + w$$

In forward DP, x is the state at the current stage, z is the state at the next stage. The stage cost and arrival cost are given by

$$\ell(z,w) = (1/2)(\,|y(k+1) - Cz|_{R^{-1}}^2 + w'Q^{-1}w) \qquad V_k^0(x) = (1/2)\,|x - \hat{x}(k)|_{P(k)^{-1}}^2$$

and we wish to find $V_{k+1}^0(z)$ in the estimation problem.

(a) In the estimation problem, take the z term outside the optimization and solve

$$\min_{x,w} \frac{1}{2}\Big(w'Q^{-1}w + (x - \hat{x}(k))'P(k)^{-1}(x - \hat{x}(k))\Big) \qquad \text{s.t. } z = Ax + w$$

using the inverse form in Exercise 1.18, and show that the optimal cost is given by

$$V^0(z) = (1/2)(z - A\hat{x}(k))'(P^-(k+1))^{-1}(z - A\hat{x}(k))$$
$$P^-(k+1) = AP(k)A' + Q$$

Add the z term to this cost using the third part of Example 1.1 and show that

$$V_{k+1}^0(z) = (1/2)(z - \hat{x}(k+1))'P^{-1}(k+1)(z - \hat{x}(k+1))$$
$$P(k+1) = P^-(k+1) - P^-(k+1)C'(CP^-(k+1)C' + R)^{-1}CP^-(k+1)$$
$$\hat{x}(k+1) = A\hat{x}(k) + L(k+1)(y(k+1) - CA\hat{x}(k))$$
$$L(k+1) = P^-(k+1)C'(CP^-(k+1)C' + R)^{-1}$$

(b) In the regulator problem, take the z term outside the optimization and solve the remaining two-term problem using the regulator form of Exercise 1.18. Then add the z term and show that

$$V_{k-1}^0(z) = (1/2)z'\Pi(k-1)z$$
$$\Pi(k-1) = Q + A'\Pi(k)A - A'\Pi(k)B(B'\Pi(k)B + R)^{-1}B'\Pi(k)A$$
$$u^0(z) = K(k-1)z$$
$$x^0(z) = (A + BK(k-1))z$$
$$K(k-1) = -(B'\Pi(k)B + R)^{-1}B'\Pi(k)A$$

This symmetry can be developed further if we pose an output tracking problem rather than zero state regulation problem in the regulator.

Exercise 1.30: Symmetry in the Riccati iteration

Show that the covariance before measurement $P^-(k+1)$ in estimation satisfies an identical iteration to the cost to go $\Pi(k-1)$ in regulation under the change of variables $P^- \longrightarrow \Pi$, $A \longrightarrow A'$, $C \longrightarrow B'$.

Exercise 1.31: Detectability and observability canonical forms

Consider the partitioned system

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}^+ = \begin{bmatrix} A_{11} & 0 \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$
$$y = \begin{bmatrix} C_1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

with (A_{11}, C_1) observable. This form is known as observability canonical form.

(a) Show that the system is *not* observable by checking the rank of the observability matrix.

(b) Show that the modes x_1 can be uniquely determined from a sequence of measurements, but the modes x_2 *cannot* be uniquely determined from the measurements. The states x_2 are termed the unobservable modes.

(c) If A_{22} is stable the system is termed *detectable*. Although not all modes can be observed, the unobservable modes are stable and decay to steady state.

The following lemma gives an equivalent condition for detectability.

Lemma 1.13 (Hautus lemma for detectability)**.** *A system is detectable if and only if*

$$\operatorname{rank} \begin{bmatrix} \lambda I - A \\ C \end{bmatrix} = n \qquad \textit{for all } |\lambda| \geq 1$$

Prove this lemma using Lemma 1.4 as the condition for observability.

Exercise 1.32: Estimator stability and detectable systems

Show that the least squares estimator given in (1.27) is stable for (A, C) *detectable* with $Q > 0$.

Exercise 1.33: Estimator stability and semidefinite state noise penalty

We wish to show that the least squares estimator is stable for (A, C) detectable and $Q \geq 0$, (A, Q) stabilizable.

(a) Because Q^{-1} is not defined in this problem, the objective function defined in (1.26) requires modification. Show that the objective function with semidefinite $Q \geq 0$ can be converted into the following form

$$V(x(0), \mathbf{w}(T)) = \frac{1}{2}\Bigg(|x(0) - \overline{x}(0)|^2_{(P^-(0))^{-1}} + \sum_{k=0}^{T-1} |w(k)|^2_{\tilde{Q}^{-1}} + \sum_{k=0}^{T} |y(k) - Cx(k)|^2_{R^{-1}} \Bigg)$$

in which

$$x^+ = Ax + Gw \qquad \widetilde{Q} > 0$$

Find expressions for $\widetilde{Q}$ and G in terms of the original semidefinite Q. How are the dimension of $\widetilde{Q}$ and G related to the rank of Q?

(b) What is the probabilistic interpretation of the state estimation problem with semidefinite Q?

(c) Show that (A, Q) stabilizable implies (A, G) stabilizable in the converted form.

(d) Show that this estimator is stable for (A, C) detectable and (A, G) stabilizable with $\widetilde{Q}, R > 0$.

(e) Discuss what happens to the estimator's stability if Q is not positive semidefinite or (A, Q) is not stabilizable.

Exercise 1.34: Calculating mean and variance from data

We are sampling a real-valued scalar random variable $x(k) \in \mathbb{R}$ at time k. Assume the random variable comes from a distribution with mean $\overline{x}$ and variance P, and the samples at different times are statistically independent.

A colleague has suggested the following formulas for estimating the mean and variance from N samples

$$\hat{x}_N = \frac{1}{N}\sum_{j=1}^{N} x(j) \qquad \hat{P}_N = \frac{1}{N}\sum_{j=1}^{N} (x(j) - \hat{x}_N)^2$$

(a) Prove that the estimate of the mean is unbiased for all N, i.e., show that for all N

$$\mathcal{E}(\hat{x}_N) = \overline{x}$$

(b) Prove that the estimate of the variance is not unbiased for any N, i.e., show that for all N

$$\mathcal{E}(\hat{P}_N) \neq P$$

(c) Using the result above, provide an alternative formula for the variance estimate that is unbiased for all N. How large does N have to be before these two estimates of P are within 1%?

Exercise 1.35: Expected sum of squares

Given that a random variable x has mean m and covariance P, show that the expected sum of squares is given by the formula (Selby, 1973, p.138)

$$\mathcal{E}(x'Qx) = m'Qm + \mathrm{tr}(QP)$$

The trace of a square matrix A, written $\mathrm{tr}(A)$, is defined to be the sum of the diagonal elements

$$\mathrm{tr}(A) := \sum_i A_{ii}$$

Exercise 1.36: Normal distribution

Given a normal distribution with scalar parameters m and σ

$$p_\xi(x) = \sqrt{\frac{1}{2\pi\sigma^2}} \exp\left[-\frac{1}{2}\left(\frac{x-m}{\sigma}\right)^2\right] \tag{1.61}$$

By direct calculation, show that

(a)

$$\mathcal{E}(\xi) = m$$
$$\text{var}(\xi) = \sigma^2$$

(b) Show that the mean and the maximum likelihood are equal for the normal distribution. Draw a sketch of this result. The maximum likelihood estimate, $\hat{x}$, is defined as

$$\hat{x} := \arg\max_x p_\xi(x)$$

in which arg returns the solution to the optimization problem.

Exercise 1.37: Conditional densities are positive definite

We show in Example A.44 that if ξ and η are jointly normally distributed as

$$\begin{bmatrix} \xi \\ \eta \end{bmatrix} \sim N(m, P)$$
$$\sim N\left(\begin{bmatrix} m_x \\ m_y \end{bmatrix}, \begin{bmatrix} P_x & P_{xy} \\ P_{yx} & P_y \end{bmatrix}\right)$$

then the conditional density of ξ given η is also normal

$$(\xi|\eta) \sim N(m_{x|y}, P_{x|y})$$

in which the conditional mean is

$$m_{x|y} = m_x + P_{xy}P_y^{-1}(y - m_y)$$

and the conditional covariance is

$$P_{x|y} = P_x - P_{xy}P_y^{-1}P_{yx}$$

Given that the joint density is well defined, prove the marginal densities and the conditional densities also are well defined, i.e., given $P > 0$, prove $P_x > 0$, $P_y > 0$, $P_{x|y} > 0$, $P_{y|x} > 0$.

Exercise 1.38: Expectation and covariance under linear transformations

Consider the random variable $x \in \mathbb{R}^n$ with density p_x and mean and covariance

$$\mathcal{E}(x) = m_x \qquad \text{cov}(x) = P_x$$

Consider the random variable $y \in \mathbb{R}^p$ defined by the linear transformation

$$y = Cx$$

(a) Show that the mean and covariance for y are given by

$$\mathcal{E}(y) = Cm_x \qquad \text{cov}(y) = CP_xC'$$

Does this result hold for all C? If yes, prove it; if no, provide a counterexample.

(b) Apply this result to solve Exercise A.35.

Exercise 1.39: Normal distributions under linear transformations

Given the normally distributed random variable, $\xi \in \mathbb{R}^n$, consider the random variable, $\eta \in \mathbb{R}^n$, obtained by the linear transformation

$$\eta = A\xi$$

in which A is a nonsingular matrix. Using the result on transforming probability densities, show that if $\xi \sim N(m, P)$, then $\eta \sim N(Am, APA')$. This result basically says that linear transformations of normal random variables are normal.

Exercise 1.40: More on normals and linear transformations

Consider a normally distributed random variable $x \in \mathbb{R}^n$, $x \sim N(m_x, P_x)$. You showed in Exercise 1.39 for $C \in \mathbb{R}^{n \times n}$ invertible, that the random variable y defined by the linear transformation $y = Cx$ is also normal and is distributed as

$$y \sim N(Cm_x, CP_xC')$$

Does this result hold for all C? If yes, prove it; if no, provide a counterexample.

Exercise 1.41: Signal processing in the good old days—recursive least squares

Imagine we are sent back in time to 1960 and the only computers available have extremely small memories. Say we have a large amount of data coming from a process and we want to compute the least squares estimate of model parameters from these data. Our immediate challenge is that we cannot load all of these data into memory to make the standard least squares calculation.

Alternatively, go 150 years further back in time and consider the situation from Gauss's perspective,

> It occasionally happens that after we have completed all parts of an extended calculation on a sequence of observations, we learn of a new observation that we would like to include. In many cases we will not want to have to redo the entire elimination but instead to find the modifications due to the new observation in the most reliable values of the unknowns and in their weights.
> C.F. Gauss, 1823
> G.W. Stewart Translation, 1995, p. 191.

Given the linear model

$$y_i = X_i'\theta$$

in which scalar y_i is the measurement at sample i, X_i' is the independent model variable (row vector, $1 \times p$) at sample i, and θ is the parameter vector ($p \times 1$) to be estimated from these data. Given the weighted least squares objective and n measurements, we wish to compute the usual estimate

$$\hat{\theta} = (X'X)^{-1}X'y \tag{1.62}$$

in which

$$y = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \qquad X = \begin{bmatrix} X_1' \\ \vdots \\ X_n' \end{bmatrix}$$

We do not wish to store the large matrices $X(n \times p)$ and $y(n \times 1)$ required for this calculation. Because we are planning to process the data one at a time, we first modify our usual least squares problem to deal with small n. For example, we wish to estimate

the parameters when $n < p$ and the inverse in (1.62) does not exist. In such cases, we may choose to regularize the problem by modifying the objective function as follows

$$\Phi(\theta) = (\theta - \overline{\theta})' P_0^{-1} (\theta - \overline{\theta}) + \sum_{i=1}^{n} (y_i - X_i'\theta)^2$$

in which $\overline{\theta}$ and P_0 are chosen by the user. In Bayesian estimation, we call $\overline{\theta}$ and P_0 the prior information, and often assume that the prior density of θ (without measurements) is normal

$$\theta \sim N(\overline{\theta}, P_0)$$

The solution to this modified least squares estimation problem is

$$\hat{\theta} = \overline{\theta} + (X'X + P_0^{-1})^{-1} X'(y - X\overline{\theta}) \tag{1.63}$$

Devise a means to *recursively* estimate θ so that:

1. We never store more than one measurement at a time in memory.
2. After processing all the measurements, we obtain the same least squares estimate given in (1.63).

Exercise 1.42: Least squares parameter estimation and Bayesian estimation

Consider a model linear in the parameters

$$y = X\theta + e \tag{1.64}$$

in which $y \in \mathbb{R}^p$ is a vector of measurements, $\theta \in \mathbb{R}^m$ is a vector of parameters, $X \in \mathbb{R}^{p \times m}$ is a matrix of known constants, and $e \in \mathbb{R}^p$ is a random variable modeling the measurement error. The standard parameter estimation problem is to find the best estimate of θ given the measurements y corrupted with measurement error e, which we assume is distributed as

$$e \sim N(0, R)$$

(a) Consider the case in which the errors in the measurements are independently and identically distributed with variance σ^2, $R = \sigma^2 I$. For this case, the classic least squares problem and solution are

$$\min_{\theta} |y - X\theta|^2 \qquad \hat{\theta} = (X'X)^{-1} X'y$$

Consider the measurements to be sampled from (1.64) with true parameter value θ_0. Show that using the least squares formula, the parameter estimate is distributed as

$$\hat{\theta} \sim N(\theta_0, P_{\hat{\theta}}) \qquad P_{\hat{\theta}} = \sigma^2 (X'X)^{-1}$$

(b) Now consider again the model of (1.64) and a Bayesian estimation problem. Assume a prior distribution for the random variable θ

$$\theta \sim N(\overline{\theta}, \overline{P})$$

Compute the conditional density of θ given measurement y, show that this density is normal, and find its mean and covariance

$$p_{\theta|y}(\theta|y) = n(\theta, m, P)$$

Show that Bayesian estimation and least squares estimation give the same result in the limit of an infinite variance prior. In other words, if the covariance of the prior is large compared to the covariance of the measurement error, show that

$$m \approx (X'X)^{-1} X'y \qquad P \approx P_{\hat{\theta}}$$

(c) What (weighted) least squares minimization problem is solved for the general measurement error covariance

$$e \sim N(0, R)$$

Derive the least squares estimate formula for this case.

(d) Again consider the measurements to be sampled from (1.64) with true parameter value θ_0. Show that the weighted least squares formula gives parameter estimates that are distributed as

$$\hat{\theta} \sim N(\theta_0, P_{\hat{\theta}})$$

and find $P_{\hat{\theta}}$ for this case.

(e) Show again that Bayesian estimation and least squares estimation give the same result in the limit of an infinite variance prior.

Exercise 1.43: Least squares and minimum variance estimation

Consider again the model linear in the parameters and the least squares estimator from Exercise 1.42

$$y = X\theta + e \qquad e \sim N(0, R)$$
$$\hat{\theta} = \left(X'R^{-1}X\right)^{-1} X'R^{-1}y$$

Show that the covariance of the least squares estimator is the smallest covariance of all linear unbiased estimators.

Exercise 1.44: Two stages are not better than one

We often can decompose an estimation problem into stages. Consider the following case in which we wish to estimate x from measurements of z, but we have the model between x and an intermediate variable, y, and the model between y and z

$$y = Ax + e_1 \qquad \text{cov}(e_1) = Q_1$$
$$z = By + e_2 \qquad \text{cov}(e_2) = Q_2$$

(a) Write down the optimal least squares problem to solve for $\hat{y}$ given the z measurements and the second model. Given $\hat{y}$, write down the optimal least squares problem for $\hat{x}$ in terms of $\hat{y}$. Combine these two results together and write the resulting estimate of $\hat{x}$ given measurements of z. Call this the two-stage estimate of x.

(b) Combine the two models together into a single model and show that the relationship between z and x is

$$z = BAx + e_3 \qquad \text{cov}(e_3) = Q_3$$

Express Q_3 in terms of Q_1, Q_2 and the models A, B. What is the optimal least squares estimate of $\hat{x}$ given measurements of z and the one-stage model? Call this the one-stage estimate of x.

(c) Are the one-stage and two-stage estimates of x the same? If yes, prove it. If no, provide a counterexample. Do you have to make any assumptions about the models A, B?

Exercise 1.45: Time-varying Kalman filter

Derive formulas for the conditional densities of $x(k)|\mathbf{y}(k-1)$ and $x(k)|\mathbf{y}(k)$ for the time-varying linear system

$$x(k+1) = A(k)x(k) + G(k)w(k)$$
$$y(k) = C(k)x(k) + v(k)$$

in which the initial state, state noise and measurement noise are independently distributed as

$$x(0) \sim N(\overline{x}_0, Q_0) \qquad w(k) \sim N(0, Q) \qquad v(k) \sim N(0, R)$$

Exercise 1.46: More on conditional densities

In deriving the discrete time Kalman filter, we have $p_{x|\mathbf{y}}(x(k)|\mathbf{y}(k))$ and we wish to calculate recursively $p_{x|\mathbf{y}}(x(k+1)|\mathbf{y}(k+1))$ after we collect the output measurement at time $k+1$. It is straightforward to calculate $p_{x,y|\mathbf{y}}(x(k+1), y(k+1)|\mathbf{y}(k))$ from our established results on normal densities and knowledge of $p_{x|\mathbf{y}}(x(k)|\mathbf{y}(k))$, but we still need to establish a formula for pushing the $y(k+1)$ to the other side of the conditional density bar. Consider the following statement as a possible lemma to aid in this operation.

$$p_{a|b,c}(a|b,c) = \frac{p_{a,b|c}(a,b|c)}{p_{b|c}(b|c)}$$

If this statement is true, prove it. If it is false, give a counterexample.

Exercise 1.47: Other useful conditional densities

Using the definitions of marginal and conditional density, establish the following useful conditional density relations

1. $p_{A|B}(a|b) = \int p_{A|B,C}(a|b,c)p_{C|B}(c|b)dc$
2. $p_{A|B,C}(a|b,c) = p_{C|A,B}(c|a,b)\dfrac{p_{A|B}(a|b)}{p_{C|B}(c|b)}$

Exercise 1.48: Optimal filtering and deterministic least squares

Given the data sequence $(y(0), \ldots, y(k))$ and the system model

$$x^+ = Ax + w$$
$$y = Cx + v$$

(a) Write down a least squares problem whose solution would provide a good state estimate for $x(k)$ in this situation. What probabilistic interpretation can you assign to the estimate calculated from this least squares problem?

(b) Now consider the nonlinear model

$$x^+ = f(x) + w$$
$$y = g(x) + v$$

What is the corresponding nonlinear least squares problem for estimating $x(k)$ in this situation? What probabilistic interpretation, if any, can you assign to this estimate in the nonlinear model context?

(c) What is the motivation for changing from these least squares estimators to the moving horizon estimators we discussed in the chapter?

Exercise 1.49: A nonlinear transformation and conditional density

Consider the following relationship between the random variable y, and x and v

$$y = f(x) + v$$

The author of a famous textbook wants us to believe that

$$p_{y|x}(y|x) = p_v(y - f(x))$$

Derive this result and state what additional assumptions on the random variables x and v are required for this result to be correct.

Exercise 1.50: Some smoothing

One of the problems with asking you to derive the Kalman filter is that the derivation is in so many textbooks that it is difficult to tell if you are thinking independently. So here's a variation on the theme that should help you evaluate your level of understanding of these ideas. Let's calculate a smoothed rather than filtered estimate and covariance. Here's the problem.

We have the usual setup with a prior on $x(0)$

$$x(0) \sim N(\overline{x}(0), Q_0)$$

and we receive data from the following system

$$\begin{aligned} x(k+1) &= Ax(k) + w(k) \\ y(k) &= Cx(k) + v(k) \end{aligned}$$

in which the random variables $w(k)$ and $v(k)$ are independent, identically distributed normals, $w(k) \sim N(0, Q), v(k) \sim N(0, R)$.

(a) Calculate the standard density for the filtering problem, $p_{x(0)|y(0)}(x(0)|y(0))$.

(b) Now calculate the density for the smoothing problem

$$p_{x(0)|y(0),y(1)}(x(0)|y(0), y(1))$$

that is, *not* the usual $p_{x(1)|y(0),y(1)}(x(1)|y(0), y(1))$.

Exercise 1.51: Alive on arrival

The following two optimization problems are helpful in understanding the arrival cost decomposition in state estimation.

(a) Let $V(x, y, z)$ be a positive, strictly convex function consisting of the sum of two functions, one of which depends on both x and y, and the other of which depends on y and z

$$V(x, y, z) = g(x, y) + h(y, z) \qquad V : \mathbb{R}^m \times \mathbb{R}^n \times \mathbb{R}^p \to \mathbb{R}_{\geq 0}$$

Consider the optimization problem

$$P1 : \min_{x,y,z} V(x, y, z)$$

The arrival cost decomposes this three-variable optimization problem into two, smaller dimensional optimization problems. Define the "arrival cost" $\tilde{g}$ for this problem as the solution to the following single-variable optimization problem

$$\tilde{g}(y) = \min_x g(x, y)$$

and define optimization problem $P2$ as follows

$$P2 : \min_{y,z} \widetilde{g}(y) + h(y,z)$$

Let (x', y', z') denote the solution to $P1$ and (x^0, y^0, z^0) denote the solution to $P2$, in which

$$x^0 = \arg\min_x g(x, y^0)$$

Prove that the two solutions are equal

$$(x', y', z') = (x^0, y^0, z^0)$$

(b) Repeat the previous part for the following optimization problems

$$V(x,y,z) = g(x) + h(y,z)$$

Here the y variables do not appear in g but restrict the x variables through a linear constraint. The two optimization problems are

$$P1 : \min_{x,y,z} V(x,y,z) \qquad \text{subject to } Ex = y$$

$$P2 : \min_{y,z} \widetilde{g}(y) + h(y,z)$$

in which

$$\widetilde{g}(y) = \min_x g(x) \qquad \text{subject to } Ex = y$$

Exercise 1.52: On-time arrival

Consider the deterministic, full information state estimation optimization problem

$$\min_{x(0),\mathbf{w},\mathbf{v}} \frac{1}{2}\left(|x(0) - \overline{x}(0)|^2_{(P^-(0))^{-1}} + \sum_{i=0}^{T-1} |w(i)|^2_{Q^{-1}} + |v(i)|^2_{R^{-1}} \right) \tag{1.65}$$

subject to

$$\begin{aligned} x^+ &= Ax + w \\ y &= Cx + v \end{aligned} \tag{1.66}$$

in which the sequence of measurements $\mathbf{y}(T)$ are known values. Notice we assume the noise-shaping matrix, G, is an identity matrix here. See Exercise 1.53 for the general case. Using the result of the first part of Exercise 1.51, show that this problem is equivalent to the following problem

$$\min_{x(T-N),\mathbf{w},\mathbf{v}} V^-_{T-N}(x(T-N)) + \frac{1}{2} \sum_{i=T-N}^{T-1} |w(i)|^2_{Q^{-1}} + |v(i)|^2_{R^{-1}}$$

subject to (1.66). The arrival cost is defined as

$$V^-_N(a) := \min_{x(0),\mathbf{w},\mathbf{v}} \frac{1}{2}\left(|x(0) - \overline{x}(0)|^2_{(P^-(0))^{-1}} + \sum_{i=0}^{N-1} |w(i)|^2_{Q^{-1}} + |v(i)|^2_{R^{-1}} \right)$$

subject to (1.66) and $x(N) = a$. Notice that any value of N, $0 \le N \le T$, can be used to split the cost function using the arrival cost.

Exercise 1.53: Arrival cost with noise-shaping matrix G

Consider the deterministic, full information state estimation optimization problem

$$\min_{x(0),\mathbf{w},\mathbf{v}} \frac{1}{2}\Big(|x(0)-\overline{x}(0)|^2_{(P^-(0))^{-1}} + \sum_{i=0}^{T-1} |w(i)|^2_{Q^{-1}} + |v(i)|^2_{R^{-1}} \Big)$$

subject to

$$\begin{aligned} x^+ &= Ax + Gw \\ y &= Cx + v \end{aligned} \tag{1.67}$$

in which the sequence of measurements $\mathbf{y}$ are known values. Using the result of the second part of Exercise 1.51, show that this problem also is equivalent to the following problem

$$\min_{x(T-N),\mathbf{w},\mathbf{v}} V^-_{T-N}(x(T-N)) + \frac{1}{2}\Big(\sum_{i=T-N}^{T-1} |w(i)|^2_{Q^{-1}} + |v(i)|^2_{R^{-1}} \Big)$$

subject to (1.67). The arrival cost is defined for all $k \geq 0$ and $a \in \mathbb{R}^n$ by

$$V^-_k(a) := \min_{x(0),\mathbf{w},\mathbf{v}} \frac{1}{2}\Big(|x(0)-\overline{x}(0)|^2_{(P^-(0))^{-1}} + \sum_{i=0}^{k-1} |w(i)|^2_{Q^{-1}} + |v(i)|^2_{R^{-1}} \Big)$$

subject to $x(k) = a$ and the model (1.67). Notice that any value of N, $0 \leq N \leq T$, can be used to split the cost function using the arrival cost.

Exercise 1.54: Where is the steady state?

Consider the two-input, two-output system

$$A = \begin{bmatrix} 0.5 & 0 & 0 & 0 \\ 0 & 0.6 & 0 & 0 \\ 0 & 0 & 0.5 & 0 \\ 0 & 0 & 0 & 0.6 \end{bmatrix} \quad B = \begin{bmatrix} 0.5 & 0 \\ 0 & 0.4 \\ 0.25 & 0 \\ 0 & 0.6 \end{bmatrix} \quad C = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

(a) The output setpoint is $y_{\text{sp}} = \begin{bmatrix} 1 & -1 \end{bmatrix}'$ and the input setpoint is $u_{\text{sp}} = \begin{bmatrix} 0 & 0 \end{bmatrix}'$. Calculate the target triple (x_s, u_s, y_s). Is the output setpoint feasible, i.e., does $y_s = y_{\text{sp}}$?

(b) Assume only input one u_1 is available for control. Is the output setpoint feasible? What is the target in this case using $Q_s = I$?

(c) Assume both inputs are available for control but only the first output has a setpoint, $y_{1t} = 1$. What is the solution to the target problem for $R_s = I$?

Exercise 1.55: Detectability of integrating disturbance models

(a) Prove Lemma 1.8; the augmented system is detectable if and only if the system (A, C) is detectable and

$$\operatorname{rank} \begin{bmatrix} I - A & -B_d \\ C & C_d \end{bmatrix} = n + n_d$$

(b) Prove Corollary 1.9; the augmented system is detectable only if $n_d \leq p$.

Exercise 1.56: Unconstrained tracking problem

(a) For an *unconstrained* system, show that the following condition is *sufficient* for feasibility of the target problem for any r_{sp}.

$$\operatorname{rank}\begin{bmatrix} I-A & -B \\ HC & 0 \end{bmatrix} = n + n_c \tag{1.68}$$

(b) Show that (1.68) implies that the number of controlled variables without offset is less than or equal to the number of manipulated variables and the number of measurements, $n_c \le m$ and $n_c \le p$.

(c) Show that (1.68) implies the rows of H are independent.

(d) Does (1.68) imply that the rows of C are independent? If so, prove it; if not, provide a counterexample.

(e) By choosing H, how can one satisfy (1.68) if one has installed redundant sensors so several rows of C are identical?

Exercise 1.57: Unconstrained tracking problem for stabilizable systems

If we restrict attention to stabilizable systems, the sufficient condition of Exercise 1.56 becomes a necessary and sufficient condition. Prove the following lemma.

Lemma 1.14 (Stabilizable systems and feasible targets)**.** *Consider an unconstrained, stabilizable system* (A, B)*. The target is feasible for any* r_{sp} *if and only if*

$$\operatorname{rank}\begin{bmatrix} I-A & -B \\ HC & 0 \end{bmatrix} = n + n_c$$

Exercise 1.58: Existence and uniqueness of the unconstrained target

Assume a system having p controlled variables $z = Hx$, with setpoints r_{sp}, and m manipulated variables u, with setpoints u_{sp}. Consider the steady-state target problem

$$\min_{x,u} (1/2)(u - u_{sp})' R (u - u_{sp}) \qquad R > 0$$

subject to

$$\begin{bmatrix} I-A & -B \\ H & 0 \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix} = \begin{bmatrix} 0 \\ r_{sp} \end{bmatrix}$$

Show that the steady-state solution (x, u) exists for any (r_{sp}, u_{sp}) and is unique if

$$\operatorname{rank}\begin{bmatrix} I-A & -B \\ H & 0 \end{bmatrix} = n + p \qquad \operatorname{rank}\begin{bmatrix} I-A \\ H \end{bmatrix} = n$$

Exercise 1.59: Choose a sample time

Consider the unstable continuous time system

$$\frac{dx}{dt} = Ax + Bu \qquad y = Cx$$

in which

$$A = \begin{bmatrix} -0.281 & 0.935 & 0.035 & 0.008 \\ 0.047 & -0.116 & 0.053 & 0.383 \\ 0.679 & 0.519 & 0.030 & 0.067 \\ 0.679 & 0.831 & 0.671 & -0.083 \end{bmatrix} \qquad B = \begin{bmatrix} 0.687 \\ 0.589 \\ 0.930 \\ 0.846 \end{bmatrix} \qquad C = I$$

Consider regulator tuning parameters and constraints

$$Q = \text{diag}(1,2,1,2) \qquad R = 1 \qquad N = 10 \qquad |x| \le \begin{bmatrix} 1 \\ 2 \\ 1 \\ 3 \end{bmatrix}$$

(a) Compute the eigenvalues of A. Choose a sample time of $\Delta = 0.04$ and simulate the MPC regulator response given $x(0) = \begin{bmatrix} -0.9 & -1.8 & 0.7 & 2 \end{bmatrix}'$ until $t = 20$. Use an ODE solver to simulate the continuous time plant response. Plot all states and the input versus time.

Now add an input disturbance to the regulator so the control applied to the plant is u_d instead of u in which

$$u_d(k) = (1 + 0.1w_1)u(k) + 0.1w_2$$

and w_1 and w_2 are zero-mean, normally distributed random variables with unit variance. Simulate the regulator's performance given this disturbance. Plot all states and $u_d(k)$ versus time.

(b) Repeat the simulations with and without disturbance for $\Delta = 0.4$ and $\Delta = 2$.

(c) Compare the simulations for the different sample times. What happens if the sample time is too large? Choose an appropriate sample time for this system and justify your choice.

Exercise 1.60: Disturbance models and offset

Consider the following two-input, three-output plant discussed in Example 1.11

$$x^+ = Ax + Bu + B_p p$$
$$y = Cx$$

in which

$$A = \begin{bmatrix} 0.2681 & -0.00338 & -0.00728 \\ 9.703 & 0.3279 & -25.44 \\ 0 & 0 & 1 \end{bmatrix} \qquad C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$B = \begin{bmatrix} -0.00537 & 0.1655 \\ 1.297 & 97.91 \\ 0 & -6.637 \end{bmatrix} \qquad B_p = \begin{bmatrix} -0.1175 \\ 69.74 \\ 6.637 \end{bmatrix}$$

The input disturbance p results from a reactor inlet flowrate disturbance.

(a) Since there are two inputs, choose two outputs in which to remove steady-state offset. Build an output disturbance model with two integrators. Is your augmented model detectable?

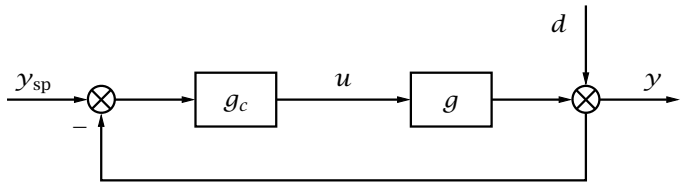


Figure 1.14: Feedback control system with output disturbance d, and setpoint y_{sp}.

(b) Implement your controller using $p = 0.01$ as a step disturbance at $k = 0$. Do you remove offset in your chosen outputs? Do you remove offset in any outputs?

(c) Can you find any two-integrator disturbance model that removes offset in two outputs? If so, which disturbance model do you use? If not, why not?

Exercise 1.61: MPC, PID, and time delay

Consider the following first-order system with time delay shown in Figure 1.14

$$g(s) = \frac{k}{\tau s + 1} e^{-\theta s}, \qquad k = 1, \tau = 1, \theta = 5$$

Consider a unit step change in setpoint y_{sp}, at $t = 0$.

(a) Choose a reasonable sample time, Δ, and disturbance model, and simulate an offset-free discrete time MPC controller for this setpoint change. List all of your chosen parameters.

(b) Choose PID tuning parameters to achieve "good performance" for this system. List your PID tuning parameters. Compare the performances of the two controllers.

Exercise 1.62: CSTR heat-transfer coefficient

Your mission is to design the controller for the nonlinear CSTR model given in Example 1.11. We wish to use a linear controller and estimator with three integrating disturbances to remove offset in two controlled variables: *temperature* and *level*; use the nonlinear CSTR model as the plant.

(a) You are particularly concerned about disturbances to the heat-transfer rate (parameter U) for this reactor. If changes to U are the primary disturbance, what disturbance model do you recommend and what covariances do you recommend for the three disturbances so that the disturbance state accounting for heat transfer is used primarily to explain the output error in the state estimator? First do a simulation with no measurement noise to test your estimator design. In the simulation let the reactor's heat-transfer coefficient decrease (and increase) by 20% at 10 minutes to test your control system design. Comment on the performance of the control system.

(b) Now let's add some measurement noise to all three sensors. So we all work on the same problem, choose the variance of the measurement error R_v to be

$$R_v = 10^{-3}\,\mathrm{diag}\,(c_s^2, T_s^2, h_s^2)$$

in which (c_s, T_s, h_s) are the nominal steady states of the three measurements. Is the performance from the previous part assuming no measurement noise acceptable? How do you adjust your estimator from the previous part to obtain good performance? Rerun the simulation with measurement noise and your adjusted state estimator. Comment on the change in the performance of your new design that accounts for the measurement noise.

(c) Recall that the offset lemma 1.10 is an either-or proposition, i.e., *either* the controller removes steady offset in the controlled variables *or* the system is closed-loop unstable. From closed-loop simulation, approximate the range of plant U values for which the controller is stabilizing (with zero measurement noise). From a stabilization perspective, which disturbance is worse, an increase or decrease in the plant's heat-transfer coefficient?

Exercise 1.63: System identification of the nonlinear CSTR

In many practical applications, it may not be convenient to express system dynamics from first principles. Hence, identifying a suitable model from data is a critical step in the design of an MPC controller. Your final mission is to obtain a 2-input, 3-output process model for the nonlinear CSTR given in Example 1.11 using the System Identification Toolbox in MATLAB. Relevant functions are provided.

(a) Begin first by creating a dataset for identification. Generate a pseudo-random, binary signal (PRBS) for the inputs using `idinput`. Ensure you have generated uncorrelated signals for each input. Think about the amplitude of the PRBS to use when collecting data from a nonlinear process keeping in mind that large perturbations may lead to undesirable phenomena such as reactor ignition. Inject these generated input sequences into the nonlinear plant of Example 1.11 and simulate the system by solving the nonlinear ODEs. Add measurement noise to the simulation so that you have a realistic dataset for the ID and plot the input-output data.

(b) Use the data to identify a third-order linear state space model by calling `iddata` and `ssest`. Compare the step tests of your identified model with those from the linear model used in Example 1.11. Which is more accurate compared to the true plant simulation?

(c) Using the code for Example 1.11 as a starting point, replace the linear model in the MPC controller with your identified model and recalculate Figures 1.10 and 1.11 from the example. Is your control system robust enough to obtain good closed-loop control of the nonlinear plant using your linear model identified from data in the MPC controller? Do you maintain zero offset in the controlled variables?

Bibliography

R. E. Bellman. *Dynamic Programming*. Princeton University Press, Princeton, New Jersey, 1957.

R. E. Bellman and S. E. Dreyfus. *Applied Dynamic Programming*. Princeton University Press, Princeton, New Jersey, 1962.

D. P. Bertsekas. *Dynamic Programming*. Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1987.

E. F. Camacho and C. Bordons. *Model Predictive Control*. Springer-Verlag, London, second edition, 2004.

E. J. Davison and H. W. Smith. Pole assignment in linear time-invariant multivariable systems with constant disturbances. *Automatica*, 7:489–498, 1971.

E. J. Davison and H. W. Smith. A note on the design of industrial regulators: Integral feedback and feedforward controllers. *Automatica*, 10:329–332, 1974.

R. Fletcher. *Practical Methods of Optimization*. John Wiley & Sons, New York, 1987.

B. A. Francis and W. M. Wonham. The internal model principle of control theory. *Automatica*, 12:457–465, 1976.

G. C. Goodwin and K. S. Sin. *Adaptive Filtering Prediction and Control*. Prentice-Hall, Englewood Cliffs, New Jersey, 1984.

G. C. Goodwin, M. M. Serón, and J. A. De Doná. *Constrained control and estimation: an optimization approach*. Springer, New York, 2005.

M. L. J. Hautus. Controllability and stabilizability of sampled systems. *IEEE Trans. Auto. Cont.*, 17(4):528–531, August 1972.

R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press, 1985.

A. H. Jazwinski. *Stochastic Processes and Filtering Theory*. Academic Press, New York, 1970.

R. E. Kalman. A new approach to linear filtering and prediction problems. *Trans. ASME, J. Basic Engineering*, pages 35–45, March 1960a.

R. E. Kalman. Contributions to the theory of optimal control. *Bull. Soc. Math. Mex.*, 5:102–119, 1960b.

H. Kwakernaak and R. Sivan. *Linear Optimal Control Systems*. John Wiley and Sons, New York, 1972.

W. H. Kwon. *Receding horizon control: model predictive control for state models*. Springer-Verlag, London, 2005.

J. M. Maciejowski. *Predictive Control with Contraints*. Prentice-Hall, Harlow, UK, 2002.

J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, New York, second edition, 2006.

B. J. Odelson, M. R. Rajamani, and J. B. Rawlings. A new autocovariance least-squares method for estimating noise covariances. *Automatica*, 42(2):303–308, February 2006.

G. Pannocchia and J. B. Rawlings. Disturbance models for offset-free MPC control. *AIChE J.*, 49(2):426–437, 2003.

L. Qiu and E. J. Davison. Performance limitations of non-minimum phase systems in the servomechanism problem. *Automatica*, 29(2):337–349, 1993.

C. V. Rao and J. B. Rawlings. Steady states and constraints in model predictive control. *AIChE J.*, 45(6):1266–1278, 1999.

J. A. Rossiter. *Model-based predictive control: a practical approach*. CRC Press LLC, Boca Raton, FL, 2004.

S. M. Selby. *CRC Standard Mathematical Tables*. CRC Press, twenty-first edition, 1973.

E. D. Sontag. *Mathematical Control Theory*. Springer-Verlag, New York, second edition, 1998.

G. Strang. *Linear Algebra and its Applications*. Academic Press, New York, second edition, 1980.

L. Wang. *Model Predictive Control System Design and Implementation Using Matlab*. Springer, New York, 2009.

2
Model Predictive Control—Regulation

2.1 Introduction

In Chapter 1 we investigated a special, but useful, form of model predictive control (MPC); an important feature of this form of MPC is that, *if* the terminal cost is chosen to be the value function of infinite horizon *unconstrained* optimal control problem, there exists a set of initial states for which MPC is actually optimal for the *infinite horizon constrained* optimal control problem and therefore inherits its associated advantages. Just as there are many methods other than infinite horizon linear quadratic control for stabilizing linear systems, there are alternative forms of MPC that can stabilize linear and even nonlinear systems. We explore these alternatives in the remainder of this chapter. But first we place MPC in a more general setting to facilitate comparison with other control methods.

MPC is, as we have seen earlier, a form of control in which the control action is obtained by solving *online*, at each sampling instant, a *finite horizon* optimal control problem in which the initial state is the current state of the plant. Optimization yields a finite control sequence, and the first control action in this sequence is applied to the plant. MPC differs, therefore, from conventional control in which the control law is precomputed offline. But this is not an essential difference; MPC implicitly implements a control law that can, in principle, be computed offline as we shall soon see. Specifically, if the current state of the system being controlled is x, MPC obtains, by solving an open-loop optimal control problem for this initial state, a specific control action u to apply to the plant.

Dynamic programming (DP) may be used to solve a feedback version of the same optimal control problem, however, yielding a receding horizon control *law* $\kappa(\cdot)$. The important fact is that if x is the current state,

the optimal control u obtained by MPC (by solving an open-loop optimal control problem) satisfies $u = \kappa(x)$; MPC computes the *value* $\kappa(x)$ of the optimal receding horizon control law for the current state x, while DP yields the control *law* $\kappa(\cdot)$ that can be used for *any* state. DP would appear to be preferable since it provides a control law that can be implemented simply (as a look-up table). However, obtaining a DP solution is difficult, if not impossible, for most optimal control problems if the state dimension is reasonably high — unless the system is linear, the cost quadratic and there are no control or state constraints. The great advantage of MPC is that open-loop optimal control problems often can be solved rapidly enough, using standard mathematical programming algorithms, to permit the use of MPC even though the system being controlled is nonlinear, and hard constraints on states and controls must be satisfied. Thus MPC permits the application of a DP solution, even though explicit determination of the optimal control law is intractable. MPC is an effective *implementation* of the DP solution.

In this chapter we study MPC for the case when the state is known. This case is particularly important, even though it rarely arises in practice, because important properties, such as stability and performance, may be relatively easily established. The relative simplicity of this case arises from the fact that if the state is known and if there are no disturbances or model error, the problem is *deterministic*, i.e., there is no uncertainty making feedback unnecessary in principle. As we pointed out previously, for deterministic systems the MPC action for a given state is identical to the receding horizon control law, determined using DP, and evaluated at the given state. When the state is *not* known, it has to be estimated and state estimation error, together with model error and disturbances, makes the system uncertain in that future trajectories cannot be precisely predicted. The simple connection between MPC and the DP solution is lost because there does not exist an open-loop optimal control problem whose solution yields a control action that is the same as that obtained by the DP solution. A practical consequence is that special techniques are required to ensure robustness against these various forms of uncertainty. So the results of this chapter hold when there is no uncertainty. We prove, in particular, that the optimal control problem that defines the model predictive control can always be solved if the initial optimal control problem can be solved (recursive feasibility), and that the optimal cost can always be reduced allowing us to prove asymptotic or exponential stability of the target state. We

refer to stability in the absence of uncertainty as *nominal or inherent stability*.

When uncertainty is present, however, neither of these two assertions is necessarily true; uncertainty may cause the state to wander outside the region where the optimal control problem can be solved and may lead to instability. Procedures for overcoming the problems arising from uncertainty are presented in Chapters 3 and 5. In most of the control algorithms presented in this chapter, the decrease in the optimal cost, on which the proof of stability is founded, is based on the assumption that the next state is exactly as predicted and that the global solution to the optimal control problem can be computed. In the suboptimal control algorithm presented in Chapter 6, where global optimality is not required, the decrease in the optimal cost is still based on the assumption that the current state is exactly the state as predicted at the previous time.

2.2 Model Predictive Control

As discussed briefly in Chapter 1, most nonlinear system descriptions derived from physical arguments are continuous time models in the form of nonlinear differential equations

$$\frac{dx}{dt} = f(x, u)$$

For this class of systems, the control law with the best closed-loop properties is the solution to the following infinite horizon, constrained optimal control problem. The cost is defined to be

$$V_\infty(x, u(\cdot)) = \int_0^\infty \ell(x(t), u(t))dt$$

in which $x(t)$ and $u(t)$ satisfy $\dot{x} = f(x, u)$. The optimal control problem $\mathbb{P}_\infty(x)$ is defined by

$$\min_{u(\cdot)} V_\infty(x, u(\cdot))$$

subject to

$$\dot{x} = f(x, u) \quad x(0) = x_0$$
$$(x(t), u(t)) \in \mathbb{Z} \text{ for all } t \in \mathbb{R}_{\geq 0}$$

If $\ell(\cdot)$ is positive definite, the goal of the regulator is to steer the state of the system to the origin.

We denote the solution to this problem (when it exists) by $u_\infty^0(\cdot;x)$ and the resultant optimal value function by $V_\infty^0(x)$. The closed-loop system under this optimal control law evolves as

$$\frac{dx(t)}{dt} = f(x(t), u_\infty^0(t;x))$$

If $f(\cdot)$, $\ell(\cdot)$ and $V_f(\cdot)$ satisfy certain differentiability and growth assumptions, and if the class of admissible controls is sufficiently rich, then a solution to $\mathbb{P}_\infty(x)$ exists for all x and satisfies

$$\dot{V}_\infty^0(x) = -\ell(x, u_\infty^0(0;x))$$

Using this and upper and lower bounds on $V_\infty^0(\cdot)$ enables global asymptotic stability of the origin to be established.

Although the control law $u_\infty^0(0;\cdot)$ provides excellent closed-loop properties, there are several impediments to its use. A feedback, rather than an open-loop, solution of the optimal control problem is desirable because of uncertainty; solution of the optimal control problem $\mathbb{P}_\infty(x)$ yields the optimal control sequence $u_\infty^0(0;x)$ for the state x but does not provide a control law. Dynamic programming may, in principle, be employed, but is generally impractical if the state dimension and the horizon are not small.

If we turn instead to an MPC approach in which we generate online only the value of optimal control sequence $u_\infty^0(\cdot;x)$ for the currently measured value of x, rather than for all x, the problem remains formidable for the following reasons. First, we are optimizing a time *function*, $u(\cdot)$, and functions are infinite dimensional. Secondly, the time interval of interest, $[0, \infty)$, is a semi-infinite interval, which poses other numerical challenges. Finally, the cost function $V(x, u(\cdot))$ is usually not a convex function of $u(\cdot)$, which presents significant optimization difficulties, especially in an online setting. Even proving existence of the optimal control in this general setting is a challenge. However, see Pannocchia, Rawlings, Mayne, and Mancuso (2015) in which it is shown how an *infinite horizon* optimal control may be solved online if the system is linear, the cost quadratic and the control but not the state is constrained.

Our task in this chapter may therefore be viewed as restricting the system and control parameterization to make problem $\mathbb{P}_\infty(x)$ more easily computable. We show how to pose various problems for which we can establish existence of the optimal solution and asymptotic closed-loop stability of the resulting controller. For these problems, we almost

always replace the continuous time differential equation with a discrete time difference equation. We often replace the semi-infinite time interval with a finite time interval and append a terminal region so that we can approximate the cost to go for the semi-infinite interval once the system enters the terminal region. Although the solution of problem $\mathbb{P}_\infty(x)$ in its full generality is out of reach with today's computational methods, its value lies in distinguishing what is *desirable* in the control problem formulation and what is *achievable* with available computing technology.

We develop here MPC for the control of constrained nonlinear time-invariant systems. The nonlinear system is described by the nonlinear difference equation

$$x^+ = f(x,u) \qquad f : \mathbb{X} \times \mathbb{U} \to \mathbb{X} \tag{2.1}$$

in which $x \in \mathbb{X} \subseteq \mathbb{R}^n$ is the current state, $u \in \mathbb{U} \subseteq \mathbb{R}^m$, is the current control, (sets $\mathbb{X}$ and $\mathbb{U}$ are assumed closed), and x^+ the successor state; $x^+ = f(x,u)$ is the discrete time analog of the continuous time differential equation $\dot{x} = f(x,u)$. The function $f(\cdot)$ is assumed to be continuous and to satisfy $f(0,0) = 0$; $(0,0)$ is the desired equilibrium pair. The subsequent analysis is easily extended to the case when the desired equilibrium pair is (x_s, u_s) satisfying $x_s = f(x_s, u_s)$.

We introduce here some notation that we employ in the sequel. The set $\mathbb{I}$ denotes the set of integers, $\mathbb{I}_{\geq 0} := \{0,1,2,\ldots\}$ and, for any two integers m and n satisfying $m \leq n$, $\mathbb{I}_{m:n} := \{m, m+1, \ldots, n\}$. We refer to the pair (x,i) as an event; an event (x,i) denotes that the state at time i is x. We use $\mathbf{u}$ to denote the possibly infinite control sequence $(u(k))_{k\in\mathbb{I}_{\geq 0}} = (u(0), u(1), u(2), \ldots)$. In the context of MPC, $\mathbf{u}$ frequently denotes the finite sequence $\mathbf{u}_{\mathbb{I}_{0:N-1}} = (u(0), u(1), \ldots, u(N-1))$ in which N is the control *horizon*. For any integer $j \in \mathbb{I}_{\geq 0}$, we sometimes employ $\mathbf{u}_j$ to denote the finite sequence $(u(0), u(1), \ldots, u(j-1))$. Similarly $\mathbf{x}$ denotes the possibly infinite state sequence $(x(0), x(1), x(2), \ldots)$ and $\mathbf{x}_j$ the finite sequence $(x(0), x(1), \ldots, x(j))$. When no confusion can arise we often employ, for simplicity in notation, $\mathbf{u}$ in place of $\mathbf{u}_N$ and $\mathbf{x}$ in place of $\mathbf{x}_N$. Also for simplicity in notation, $\mathbf{u}$, when used in algebraic expressions, denotes the column vector $(u(0)', u(1)', \ldots, u(N-1)')'$; similarly $\mathbf{x}$ in algebraic expressions denotes the column vector $(x(0)', x(1)', \ldots, x(N)')'$.

The solution of (2.1) at time k, if the initial state at time zero is x and the control sequence is $\mathbf{u}$, is denoted by $\phi(k; x, \mathbf{u})$; the solution at time k depends only on $u(0), u(1), \ldots, u(k-1)$. Similarly, the solution

of the system (2.1) at time k, if the initial state at time i is x and the control sequence is $\mathbf{u}$, is denoted by $\phi(k;(x,i),\mathbf{u})$. Because the system is time invariant, the solution does not depend on the initial time; if the initial state is x at time i, the solution at time $j \geq i$ is $\phi(j-i;x,\mathbf{u})$. Thus the solution at time k if the initial event is (x,i) is identical to the solution at time $k-i$ if the initial event is $(x,0)$. For each k, the function $(x,\mathbf{u}) \mapsto \phi(k;x,\mathbf{u})$ is continuous as we show next.

Proposition 2.1 (Continuity of system solution)**.** *Suppose the function $f(\cdot)$ is continuous. Then, for each integer $k \in \mathbb{I}$, the function $(x,\mathbf{u}) \mapsto \phi(k;x,\mathbf{u})$ is continuous.*

Proof.
Since $\phi(1;x,u(0)) = f(x,u(0))$, the function $(x,u(0)) \mapsto \phi(1;x,u(0))$ is continuous. Suppose the function $(x,\mathbf{u}_{j-1}) \mapsto \phi(j;x,\mathbf{u}_{j-1})$ is continuous and consider the function $(x,\mathbf{u}_j) \mapsto \phi(j+1;x,\mathbf{u}_j)$. Since

$$\phi(j+1;x,\mathbf{u}_j) = f(\phi(j;x,\mathbf{u}_{j-1}),u(j))$$

in which $f(\cdot)$ and $\phi(j;\cdot)$ are continuous and since $\phi(j+1;\cdot)$ is the composition of two continuous functions $f(\cdot)$ and $\phi(j;\cdot)$, it follows that $\phi(j+1;\cdot)$ is continuous. By induction $\phi(k;\cdot)$ is continuous for any positive integer k. ■

The system (2.1) is subject to hard constraints which may take the form

$$(x(k),u(k)) \in \mathbb{Z} \qquad \text{for all } k \in \mathbb{I}_{\geq 0} \tag{2.2}$$

in which $\mathbb{Z} \subseteq \mathbb{X} \times \mathbb{U}$ is often polyhedral, i.e., $\mathbb{Z} = \{(x,u) \mid Fx + Eu \leq e\}$ for some F,E,e. For example, many problems have a rate constraint $|u(k) - u(k-1)| \leq c$ on the control. This constraint may equivalently be expressed as $|u(k) - z(k)| \leq c$ in which z is an extra state satisfying $z^+ = u$ so that $z(k) = u(k-1)$. The constraint $(x,u) \in \mathbb{Z}$ implies the control constraint is possibly state-dependent, i.e., $(x,u) \in \mathbb{Z}$ implies that

$$u \in \mathbb{U}(x) := \{u \in \mathbb{U} \mid (x,u) \in \mathbb{Z}\}$$

It also implies that the state must satisfy the constraint

$$x \in \{x \in \mathbb{X} \mid \mathbb{U}(x) \neq \emptyset\}$$

If there are no mixed constraints, then $\mathbb{Z} = \mathbb{X} \times \mathbb{U}$ so the system constraints become $x(k) \in \mathbb{X}$ and $u(k) \in \mathbb{U}$.

We assume in this chapter that the state x is known; if the state x is estimated, uncertainty (state estimation error) is introduced and *robust* MPC, discussed in Chapter 3, is required.

The next ingredient of the optimal control problem is the cost function. Practical considerations normally require that the cost be defined over a finite horizon N to ensure the resultant optimal control problem can be solved sufficiently rapidly to permit effective control. We consider initially the regulation problem in which the target state is the origin. If x is the current state and i the current time, then the optimal control problem may be posed as minimizing a cost defined over the interval from time i to time $N+i$. The optimal control problem $\mathbb{P}_N(x,i)$ at event (x,i) is the problem of minimizing the cost

$$\sum_{k=i}^{i+N-1} \ell(x(k),u(k)) + V_f(x(N+i))$$

with respect to the sequences $\mathbf{x} := (x(i),x(i+1),\ldots,x(i+N))$ and $\mathbf{u} := (u(i),u(i+1),\ldots,u(i+N-1))$ subject to the constraints that $\mathbf{x}$ and $\mathbf{u}$ satisfy the difference equation (2.1), the initial condition $x(i) = x$, and the state and control constraints (2.2). We assume that $\ell(\cdot)$ is continuous and that $\ell(0,0) = 0$. The optimal control and state sequences, obtained by solving $\mathbb{P}_N(x,i)$, are functions of the initial event (x,i)

$$\mathbf{u}^0(x,i) = \left(u^0(i;(x,i)),u^0(i+1;(x,i)),\ldots,u^0(i+N-1;(x,i))\right)$$
$$\mathbf{x}^0(x,i) = \left(x^0(i;(x,i)),x^0(i+1;(x,i)),\ldots,x^0(i+N;(x,i))\right)$$

with $x^0(i;(x,i)) = x$. In MPC, the first control action $u^0(i;(x,i))$ in the optimal control sequence $\mathbf{u}^0(x,i)$ is applied to the plant, i.e., $u(i) = u^0(i;(x,i))$. Because the system $x^+ = f(x,u)$, the stage cost $\ell(\cdot)$, and the terminal cost $V_f(\cdot)$ are all time invariant, however, the solution of $\mathbb{P}_N(x,i)$, for any time $i \in \mathbb{I}_{\geq 0}$, is identical to the solution of $\mathbb{P}_N(x,0)$ so that

$$\mathbf{u}^0(x,i) = \mathbf{u}^0(x,0)$$
$$\mathbf{x}^0(x,i) = \mathbf{x}^0(x,0)$$

In particular, $u^0(i;(x,i)) = u^0(0;(x,0))$, i.e., the control $u^0(i;(x,i))$ applied to the plant is equal to $u^0(0;(x,0))$, the first element in the sequence $\mathbf{u}^0(x,0)$. Hence we may as well merely consider problem

$\mathbb{P}_N(x,0)$ which, since the initial time is irrelevant, we call $\mathbb{P}_N(x)$. Similarly, for simplicity in notation, we replace $\mathbf{u}^0(x,0)$ and $\mathbf{x}^0(x,0)$ by, respectively, $\mathbf{u}^0(x)$ and $\mathbf{x}^0(x)$.

The optimal control problem $\mathbb{P}_N(x)$ may then be expressed as minimization of

$$\sum_{k=0}^{N-1} \ell(x(k),u(k)) + V_f(x(N))$$

with respect to the *decision variables* $(\mathbf{x},\mathbf{u})$ subject to the constraints that the state and control sequences $\mathbf{x}$ and $\mathbf{u}$ satisfy the difference equation (2.1), the initial condition $x(0) = x$, and the state, control constraints (2.2). Here $\mathbf{u}$ denotes the control sequence $\big(u(0),u(1),\ldots,u(N-1)\big)$ and $\mathbf{x}$ the state sequence $(x(0),x(1),\ldots,x(N))$. Retaining the state sequence in the set of decision variables is discussed in Chapters 6 and 8. For the purpose of analysis, however, it is preferable to constrain the state sequence $\mathbf{x}$ *a priori* to be a solution of $x^+ = f(x,u)$ enabling us to express the problem in the equivalent form of minimizing, with respect to the decision variable $\mathbf{u}$, a cost that is purely a function of the initial state x and the control sequence $\mathbf{u}$. This formulation is possible since the state sequence $\mathbf{x}$ may be expressed, via the difference equation $x^+ = f(x,u)$, as a function of $(x,\mathbf{u})$. The cost becomes $V_N(x,\mathbf{u})$ defined by

$$V_N(x,\mathbf{u}) := \sum_{k=0}^{N-1} \ell(x(k),u(k)) + V_f(x(N)) \tag{2.3}$$

in which $x(k) := \phi(k;x,\mathbf{u})$ for all $k \in \mathbb{I}_{0:N}$. Similarly the constraints (2.2), together with an additional terminal constraint

$$x(N) \in \mathbb{X}_f \subseteq \mathbb{X}$$

impose an implicit constraint on the control sequence of the form

$$\mathbf{u} \in \mathcal{U}_N(x) \tag{2.4}$$

The control constraint set $\mathcal{U}_N(x)$ is the set of control sequences $\mathbf{u} := (u(0),u(1),\ldots,u(N-1))$ satisfying the state and control constraints. It is therefore defined by

$$\mathcal{U}_N(x) := \{\mathbf{u} \mid (x,\mathbf{u}) \in \mathbb{Z}_N\} \tag{2.5}$$

in which the set $\mathbb{Z}_N \subset \mathbb{X} \times \mathbb{U}^N$ is defined by

$$\mathbb{Z}_N := \big\{(x,\mathbf{u}) \mid (\phi(k;x,\mathbf{u}),u(k)) \in \mathbb{Z}, \forall k \in \mathbb{I}_{0:N-1}, \; \phi(N;x,\mathbf{u}) \in \mathbb{X}_f\big\} \tag{2.6}$$

The optimal control problem $\mathbb{P}_N(x)$, is, therefore

$$\mathbb{P}_N(x): \qquad V_N^0(x) := \min_{\mathbf{u}} \{V_N(x,\mathbf{u}) \mid \mathbf{u} \in \mathcal{U}_N(x)\} \tag{2.7}$$

Problem $\mathbb{P}_N(x)$ is a *parametric* optimization problem in which the decision variable is $\mathbf{u}$, and both the cost and the constraint set depend on the *parameter* x. The set $\mathbb{Z}_N$ is the set of admissible $(x,\mathbf{u})$, i.e., the set of $(x,\mathbf{u})$ for which the constraints of $\mathbb{P}_N(x)$ are satisfied. Let $\mathcal{X}_N$ be the set of states in $\mathbb{X}$ for which $\mathbb{P}_N(x)$ has a solution

$$\mathcal{X}_N := \{x \in \mathbb{X} \mid \mathcal{U}_N(x) \neq \emptyset\} \tag{2.8}$$

It follows from (2.5) and (2.8) that

$$\mathcal{X}_N = \{x \in \mathbb{X} \mid \exists \mathbf{u} \in \mathbb{U}^N \text{ such that } (x,\mathbf{u}) \in \mathbb{Z}_N\}$$

which is the orthogonal projection of $\mathbb{Z}_N \subset \mathbb{X} \times \mathbb{U}^N$ onto $\mathbb{X}$. The domain of $V_N^0(\cdot)$, i.e., the set of states in $\mathbb{X}$ for which $\mathbb{P}_N(x)$ has a solution, is $\mathcal{X}_N$.

Not every optimization problem has a solution. For example, the problem $\min\{x \mid x \in (0,1)\}$ does not have a solution; $\inf\{x \mid x \in (0,1)\} = 0$ but $x = 0$ does not lie in the constraint set $(0,1)$. By Weierstrass's theorem, however, an optimization problem does have a solution if the cost is continuous (in the decision variable) and the constraint set compact (see Proposition A.7). This is the case for our problem as shown subsequently in Proposition 2.4. We assume, without further comment, that the following two standing conditions are satisfied in the sequel.

Assumption 2.2 (Continuity of system and cost)**.** The functions $f : \mathbb{Z} \to \mathbb{X}$, $\ell : \mathbb{Z} \to \mathbb{R}_{\geq 0}$ and $V_f : \mathbb{X}_f \to \mathbb{R}_{\geq 0}$ are continuous, $f(0,0) = 0$, $\ell(0,0) = 0$ and $V_f(0) = 0$.

In by far the majority of applications the set of controls $\mathbb{U}$ is bounded. Nevertheless, it is of theoretical interest to consider the case when $\mathbb{U}$ is not bounded; e.g., when the optimal control problem has no constraints on the control. To analyze this case we employ an implicit control constraint set $\bar{\mathcal{U}}_N^c(x)$ defined as follows. Choose $c \geq 0$ and define

$$\bar{\mathcal{U}}_N^c(x) := \{\mathbf{u} \in \mathcal{U}_N(x) \mid V_N(x,\mathbf{u}) \leq c\}$$

We also define the feasible set $\bar{\mathcal{X}}_N^c$ for the optimal control problem with no constraints on the control by

$$\bar{\mathcal{X}}_N^c := \{x \in \mathbb{X} \mid \bar{\mathcal{U}}_N^c(x) \neq \emptyset\}$$

Assumption 2.3 (Properties of constraint sets)**.** The set $\mathbb{Z}$ is closed and the set $\mathbb{X}_f \subseteq \mathbb{X}$ is compact. Each set contains the origin. If $\mathbb{U}$ is bounded (hence compact), the set $\mathbb{U}(x)$ is compact for all $x \in \mathbb{X}$. If $\mathbb{U}$ is unbounded, the function $\mathbf{u} \mapsto V_N(x, \mathbf{u})$ is coercive, i.e., $V_N(x, \mathbf{u}) \to \infty$ as $|\mathbf{u}| \to \infty$ for all $x \in \mathbb{X}$).

It is implicitly assumed that the desired equilibrium pair is $(\bar{x}, \bar{u}) = (0, 0)$ because the first problem we tackle is regulation to the origin.

Proposition 2.4 (Existence of solution to optimal control problem)**.** *Suppose Assumptions 2.2 and 2.3 hold. Then*

(a) The function $V_N(\cdot)$ is continuous in $\mathbb{Z}_N$.

(b) For each $x \in \mathcal{X}_N$ (for each $x \in \bar{\mathcal{X}}_N^c$, each $c \in \mathbb{R}_{>0}$), the control constraint set $\mathcal{U}_N(x)$ ($\bar{\mathcal{U}}_N^c(x)$) is compact.

(c) For each $x \in \mathcal{X}_N$ (for each $\bar{\mathcal{X}}_N^c$, each $c \in \mathbb{R}_{>0}$) a solution to $\mathbb{P}_N(x)$ exists.

Proof.

(a) That $(x, \mathbf{u}) \mapsto V_N(x, \mathbf{u})$ is continuous follows from continuity of $\ell(\cdot)$ and $V_f(\cdot)$ in Assumption 2.2, and the continuity of $(x, \mathbf{u}) \mapsto \phi(j; x, \mathbf{u})$ for each $j \in \mathbb{I}_{0:N-1}$, established in Proposition 2.1.

(b) We first show that $\mathcal{U}_N(x)$ is closed for all $x \in \mathcal{X}_N$. By Assumption 2.2 and Proposition 2.1, the function $\phi(k; \cdot)$ is continuous for any $k \in \mathbb{I}_{\geq 0}$. Since $\mathbb{Z}$ and $\mathbb{X}_f$ are closed by Assumption 2.3, any sequence $((x, \mathbf{u})_i) \in \mathbb{Z}_N$ that converges to, say, $(\bar{x}, \bar{\mathbf{u}})$ satisfies $(\phi(k; \bar{x}, \bar{\mathbf{u}}), \bar{u}(k)) \in \mathbb{Z}$ for all $k \in \mathbb{I}_{0:N-1}$, and $\phi(N; \bar{x}, \bar{\mathbf{u}}) \in \mathbb{X}_f$. Hence $(\bar{x}, \bar{\mathbf{u}}) \in \mathbb{Z}_N$ so that $\mathbb{Z}_N$ is closed. It follows from (2.5) that $\mathcal{U}_N(x)$ is closed for all $x \in \mathcal{X}_N$.
If $\mathbb{U}$ is bounded, so is $\mathcal{U}_N(x)$, and $\mathcal{U}_N(x)$ is therefore compact for all $x \in \mathcal{X}_N$. If instead $\mathbb{U}$ is unbounded, the set $\widetilde{\mathcal{U}}_N^c := \{\mathbf{u} \mid V_N(x, \mathbf{u}) \leq c\}$ for $c \in \mathbb{R}_{>0}$ is closed for all c and x because $V_N(\cdot)$ is continuous; $\bar{\mathcal{U}}_N^c(x)$ is the intersection of this set with $\mathcal{U}_N(x)$, just shown to be closed. So $\bar{\mathcal{U}}_N^c(x)$ is the intersection of closed sets and is closed. To prove $\bar{\mathcal{U}}_N^c(x)$ is bounded for all c, suppose the contrary: there exists a c such that $\bar{\mathcal{U}}_N^c(x)$ is unbounded. Then there exists a sequence $(\mathbf{u}_i)_{i \in \mathbb{I}_{\geq 0}}$ in $\bar{\mathcal{U}}_N^c(x)$ such that $\mathbf{u}_i \to \infty$ as $i \to \infty$. Because $V_N(\cdot)$ is coercive, $V_N(x, \mathbf{u}_i) \to \infty$ as $i \to \infty$, a contradiction. Hence $\bar{\mathcal{U}}_N^c(x)$ is closed and bounded and, hence, compact.

(c) Since $V_N(x, \cdot)$ is continuous and $\mathcal{U}_N(x)$ ($\bar{\mathcal{U}}_N^c(x)$) is compact, it follows from Weierstrass's theorem (Proposition A.7) a solution to $\mathbb{P}_N(x)$

exists for each $x \in \mathcal{X}_N$ ($\tilde{\mathcal{X}}_N^c$). ■

Although the function $(x, \mathbf{u}) \mapsto V_N(x, \mathbf{u})$ is continuous, the function $x \mapsto V_N^0(x)$ is not necessarily continuous; we discuss this possibility and its implications later. For each $x \in \mathcal{X}_N$, the solution of $\mathbb{P}_N(x)$ is

$$\mathbf{u}^0(x) = \arg\min_{\mathbf{u}} \{V_N(x, \mathbf{u}) \mid \mathbf{u} \in \mathcal{U}_N(x)\}$$

If $\mathbf{u}^0(x) = (u^0(0;x), u^0(1;x), \ldots, u^0(N-1;x))$ is unique for each $x \in \mathcal{X}_N$, then $\mathbf{u}^0 : \mathbb{X} \to \mathbb{U}^N$ is a function; otherwise it is a set-valued function.[1] In MPC, the control applied to the plant is the first element $u^0(0;x)$ of the optimal control sequence. At the next sampling instant, the procedure is repeated for the successor state. Although MPC computes $\mathbf{u}^0(x)$ only for specific values of the state x, it could, in principle, be used to compute $\mathbf{u}^0(x)$ and, hence, $u^0(0;x)$ for every x for which $\mathbb{P}_N(x)$ is feasible, yielding the MPC control law $\kappa_N(\cdot)$ defined by

$$\kappa_N(x) := u^0(0;x), \qquad x \in \mathcal{X}_N$$

MPC does *not* require determination of the control law $\kappa_N(\cdot)$, a task that is usually intractable when constraints or nonlinearities are present and the state dimension is large; it is this fact that makes MPC so useful.

If, at a given state x, the solution of $\mathbb{P}_N(x)$ is not unique, then $\kappa_N(\cdot) = u^0(0;\cdot)$ is set valued and the model predictive controller selects one element from the set $\kappa_N(x)$.

Example 2.5: Linear quadratic MPC

Suppose the system is described by

$$x^+ = f(x,u) := x + u$$

with initial state x. The stage cost and terminal cost are

$$\ell(x,u) := (1/2)(x^2 + u^2) \qquad V_f(x) := (1/2)x^2$$

The control constraint is

$$u \in [-1, 1]$$

and there are no state or terminal constraints. Suppose the horizon is $N = 2$. Under the first approach, the decision variables are $\mathbf{u}$ and $\mathbf{x}$,

[1] A set-valued function $\phi(\cdot)$ is a function whose value $\phi(x)$ for each x in its domain is a set.

and the optimal control problem is minimization of

$$V_N(x(0), x(1), x(2), u(0), u(1)) = \\ (1/2)\left(x(0)^2 + x(1)^2 + x(2)^2 + u(0)^2 + u(1)^2\right)$$

with respect to $(x(0), x(1), x(2))$, and $(u(0), u(1))$ subject to the following constraints

$$x(0) = x \qquad x(1) = x(0) + u(0) \qquad x(2) = x(1) + u(1)$$
$$u(0) \in [-1, 1] \qquad u(1) \in [-1, 1]$$

The constraint $u \in [-1, 1]$ is equivalent to two inequality constraints, $u \leq 1$ and $-u \leq 1$. The first three constraints are equality constraints enforcing satisfaction of the difference equation.

In the second approach, the decision variable is merely $\mathbf{u}$ because the first three constraints are automatically enforced by requiring $\mathbf{x}$ to be a solution of the difference equation. Hence, the optimal control problem becomes minimization with respect to $\mathbf{u} = (u(0), u(1))$ of

$$\begin{aligned} V_N(x, \mathbf{u}) &= (1/2)\big(x^2 + (x + u(0))^2 + (x + u(0) + u(1))^2 + \\ &\qquad u(0)^2 + u(1)^2\big) \\ &= (3/2)x^2 + \begin{bmatrix} 2x & x \end{bmatrix} \mathbf{u} + (1/2)\mathbf{u}' H \mathbf{u} \end{aligned}$$

in which

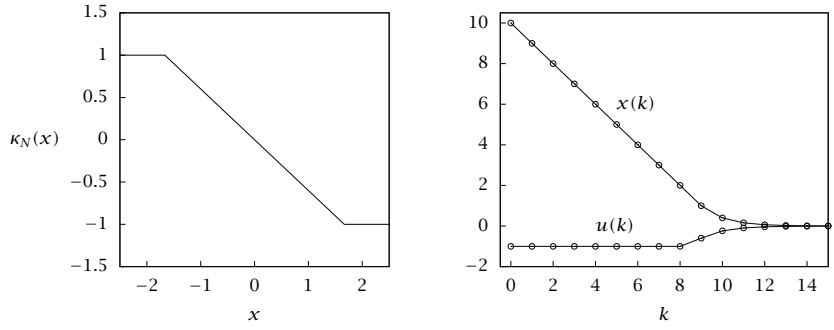
$$H = \begin{bmatrix} 3 & 1 \\ 1 & 2 \end{bmatrix}$$

subject to the constraint $\mathbf{u} \in \mathcal{U}_N(x)$ where

$$\mathcal{U}_N(x) = \{\mathbf{u} \mid |u(k)| \leq 1 \;\; k = 0, 1\}$$

Because there are no state or terminal constraints, the set $\mathcal{U}_N(x) = \mathcal{U}_N$ for this example does not depend on the parameter x; often it does. Both optimal control problems are quadratic programs.[2] The solution for $x = 10$ is $u^0(1; 10) = u^0(2; 10) = -1$ so the optimal state trajectory is $x^0(0; 10) = 10$, $x^0(1; 10) = 9$ and $x^0(2; 10) = 8$. The value $V_N^0(10) = 124$. By solving $\mathbb{P}_N(x)$ for every $x \in [-10, 10]$, the optimal control law $\kappa_N(\cdot)$ on this set can be determined, and is shown in Figure 2.1(a). The implicit MPC control law is *time invariant* since the system

[2] A quadratic program is an optimization problem in which the cost is quadratic and the constraint set is polyhedral, i.e., defined by linear inequalities.



(a) Implicit MPC control law. (b) Trajectories of controlled system.

Figure 2.1: Example of MPC.

being controlled, the cost, and the constraints are all time invariant. For our example, the controlled system (the system with MPC) satisfies the difference equation

$$x^+ = x + \kappa_N(x) \qquad \kappa_N(x) = -\text{sat}(3x/5)$$

and the state and control trajectories for an initial state of $x = 10$ are shown in Figure 2.1(b). It turns out that the origin is exponentially stable for this simple case; often, however, the terminal cost and terminal constraint set have to be carefully chosen to ensure stability. □

Example 2.6: Closer inspection of linear quadratic MPC

We revisit the MPC problem discussed in Example 2.5. The objective function is

$$V_N(x, \mathbf{u}) = (1/2)\mathbf{u}'H\mathbf{u} + c(x)'\mathbf{u} + d(x)$$

where $c(x)' = [2\ 1]x$ and $d(x) = (3/2)x^2$. The objective function may be written in the form

$$V_N(x, \mathbf{u}) = (1/2)(\mathbf{u} - a(x))'H(\mathbf{u} - a(x)) + e(x)$$

Expanding the second form shows the two forms are equal if

$$a(x) = -H^{-1}c(x) = K_1 x \qquad K_1 = -(1/5)\begin{bmatrix}3\\1\end{bmatrix}$$

and

$$e(x) + (1/2)a(x)'Ha(x) = d(x)$$

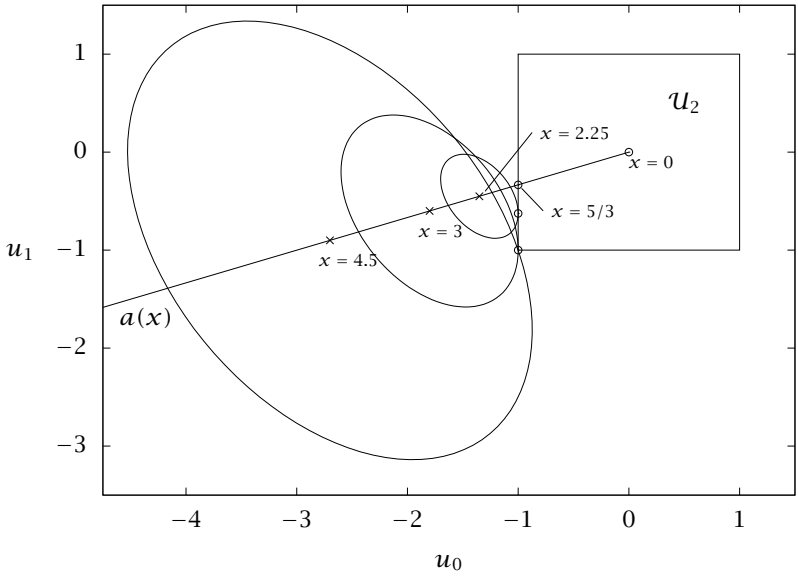


Figure 2.2: Feasible region $\mathcal{U}_2$, elliptical cost contours and ellipse center $a(x)$, and constrained minimizers for different values of x.

Since H is positive definite, $a(x)$ is the unconstrained minimizer of the objective function; indeed $\nabla_{\mathbf{u}} V_N(x, a(x)) = 0$ since

$$\nabla_{\mathbf{u}} V_N(x, \mathbf{u}) = H\mathbf{u} + c(x)$$

The locus of $a(x)$ for $x \geq 0$ is shown in Figure 2.2. Clearly the unconstrained minimizer $a(x) = K_1 x$ is equal to the constrained minimizer $\mathbf{u}^0(x)$ for all x such that $a(x) \in \mathcal{U}_2$ where $\mathcal{U}_2$ is the unit square illustrated in Figure 2.2; since $a(x) = K_1 x$, $a(x) \in \mathcal{U}_2$ for all $x \in \mathbb{X}_1 = [0, x_{c1}]$ where $x_{c1} = 5/3$. For $x > x_{c1}$, the unconstrained minimizer lies outside $\mathcal{U}_2$ as shown in Figure 2.2 for $x = 2.25$, $x = 3$ and $x = 5$. For such x, the constrained minimizer $\mathbf{u}^0(x)$ is a point that lies on the intersection of a level set of the objective function (which is an ellipse) and the boundary of $\mathcal{U}_2$. For $x \in [x_{c1}, x_{c2})$, $\mathbf{u}^0(x)$ lies on the left face of the box $\mathcal{U}_2$ and for $x \geq x_{c2} = 3$, $\mathbf{u}^0(x)$ remains at $(-1, -1)$, the bottom left vertex of $\mathcal{U}_2$.

When $u^0(x)$ lies on the left face of $\mathcal{U}_2$, the gradient $\nabla_{\mathbf{u}} V_N(x, \mathbf{u}^0(x))$ of the objective function is normal to the left face of $\mathcal{U}_2$, i.e., the level

set of $V_N^0(\cdot)$ passing through $\mathbf{u}^0(x)$ is tangential to the left face of $\mathcal{U}_2$. The outward normal to $\mathcal{U}_2$ at a point on the left face is $-e_1 = (-1, 0)$ so that at $\mathbf{u} = \mathbf{u}^0(x)$

$$\nabla_{\mathbf{u}} V(x, \mathbf{u}^0(x)) + \lambda(-e_1) = 0$$

for some $\lambda > 0$; this is a standard condition of optimality. Since $\mathbf{u} = [-1 \ v]'$ for some $v \in [-1, 1]$ and since $\nabla_{\mathbf{u}} V(x, \mathbf{u}) = H(\mathbf{u} - a(x)) = H\mathbf{u} + c(x)$, the condition of optimality is

$$\begin{bmatrix} 3 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} -1 \\ v \end{bmatrix} + \begin{bmatrix} 2 \\ 1 \end{bmatrix} x - \begin{bmatrix} \lambda \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

or

$$\begin{aligned} -3 + v + 2x - \lambda &= 0 \\ -1 + 2v + x &= 0 \end{aligned}$$

which, when solved, yields $v = (1/2) - (1/2)x$ and $\lambda = -(5/2) + (3/2)x$. Hence

$$\mathbf{u}^0(x) = b_2 + K_2 x \qquad b_2 = \begin{bmatrix} -1 \\ (1/2) \end{bmatrix} \qquad K_2 = \begin{bmatrix} 0 \\ -(1/2) \end{bmatrix}$$

for all $x \in \mathbb{X}_2 = [x_{c1}, x_{c2}]$ where $x_{c2} = 3$ since $\mathbf{u}^0(x) \in \mathcal{U}_2$ for all x in this range. For all $x \in \mathbb{X}_3 = [x_{c_2}, \infty)$, $\mathbf{u}^0(x) = (-1, -1)'$. Summarizing

$$\begin{aligned} x \in [0, (5/3)] &\implies \mathbf{u}^0(x) = K_1 x \\ x \in [(5/3), 3] &\implies \mathbf{u}^0(x) = K_2 x + b_2 \\ x \in [3, \infty) &\implies \mathbf{u}^0(x) = b_3 \end{aligned}$$

in which

$$K_1 = \begin{bmatrix} -(3/5) \\ -(1/5) \end{bmatrix} \qquad K_2 = \begin{bmatrix} 0 \\ -(1/2) \end{bmatrix} \qquad b_2 = \begin{bmatrix} -1 \\ (1/2) \end{bmatrix} \qquad b_3 = \begin{bmatrix} -1 \\ -1 \end{bmatrix}$$

The optimal control for $x \leq 0$ may be obtained by symmetry; $\mathbf{u}^0(-x) = -\mathbf{u}^0(x)$ for all $x \geq 0$ so that

$$\begin{aligned} x \in [0, -(5/3)] &\implies \mathbf{u}^0(x) = -K_1 x \\ x \in [-(5/3), -3] &\implies \mathbf{u}^0(x) = -K_2 x - b_2 \\ x \in [-3, -\infty) &\implies \mathbf{u}^0(x) = -b_3 \end{aligned}$$

It is easily checked that $\mathbf{u}^0(\cdot)$ is continuous and satisfies the constraint for all $x \in \mathbb{R}$. The MPC control law $\kappa_N(\cdot)$ is the first component of $\mathbf{u}^0(\cdot)$ and, therefore, is defined by

$$\begin{aligned} \kappa_N(x) &= 1 && x \in [-(5/3), -\infty) \\ \kappa_N(x) &= -(3/5)x && x \in [-(5/3), (5/3)] \\ \kappa_N(x) &= -1 && x \in [(5/3), \infty) \end{aligned}$$

i.e., $\kappa_N(x) = -\text{sat}(3x/5)$ which is the saturating control law depicted in Figure 2.1(a). The control law is piecewise affine and the value function piecewise quadratic. The structure of the solution to constrained linear quadratic optimal control problems is explored more fully in Chapter 7. □

As we show in Chapter 3, continuity of the value function is desirable. Unfortunately, this is not true in general; the major difficulty is in establishing that the set-valued function $\mathcal{U}_N(\cdot)$ has certain continuity properties. Continuity of the value function $V_N^0(\cdot)$ and of the implicit control law $\kappa_N(\cdot)$ may be established for a few important cases, however, as is shown by the next result, which assumes satisfaction of our standing assumptions: 2.2 and 2.3 so that the cost function $V_N(\cdot)$ is continuous in $(x, \mathbf{u})$.

Theorem 2.7 (Continuity of value function and control law)**.** *Suppose that Assumptions 2.2 and 2.3 ($\mathbb{U}$ bounded) hold.*

(a) Suppose that there are no state constraints so that $\mathbb{Z} = \mathbb{X} \times \mathbb{U}$ in which $\mathbb{X} = \mathbb{X}_f = \mathbb{R}^n$. Then the value function $V_N^0 : \mathcal{X}_N \to \mathbb{R}$ is continuous and $\mathcal{X}_N = \mathbb{R}^n$.

(b) Suppose $f(\cdot)$ is linear ($x^+ = Ax + Bu$) and that the state-control constraint set $\mathbb{Z}$ is polyhedral.[3] Then the value function $V_N^0 : \mathcal{X}_N \to \mathbb{R}$ is continuous.

(c) If, in addition, the solution $\mathbf{u}^0(x)$ of $\mathbb{P}_N(x)$ is unique at each $x \in \mathcal{X}_N$, then the implicit MPC control law $\kappa_N(\cdot)$ is continuous.

The proof of this theorem is given in Section C.3 of Appendix C. The following example, due to Meadows, Henson, Eaton, and Rawlings (1995), shows that there exist nonlinear examples where the value function and implicit control law are not continuous.

[3] A set $\mathbb{Z}$ is polyhedral if it may be defined as set of linear inequalities, i.e., if it may be expressed in the form $\mathbb{Z} = \{z \mid Mz \le m\}$.

Example 2.8: Discontinuous MPC control law

Consider the nonlinear system defined by

$$x_1^+ = x_1 + u$$
$$x_2^+ = x_2 + u^3$$

The control horizon is $N = 3$ and the cost function $V_3(\cdot)$ is defined by

$$V_3(x, \mathbf{u}) := \sum_{k=0}^{2} \ell(x(k), u(k))$$

and the stage cost $\ell(\cdot)$ is defined by

$$\ell(x, u) := |x|^2 + u^2$$

The constraint sets are $\mathbb{X} = \mathbb{R}^2$, $\mathbb{U} = \mathbb{R}$, and $\mathbb{X}_f := \{0\}$, i.e., there are no state and control constraints, and the terminal state must satisfy the constraint $x(3) = 0$. Hence, although there are three control actions, $u(0)$, $u(1)$, and $u(2)$, two must be employed to satisfy the terminal constraint, leaving only one degree of freedom. Choosing $u(0)$ to be the free decision variable automatically constrains $u(1)$ and $u(2)$ to be functions of the initial state x and the first control action $u(0)$. Solving the equation

$$\begin{aligned} x_1(3) &= x_1 + u(0) + u(1) + u(2) &&= 0 \\ x_2(3) &= x_2 + u(0)^3 + u(1)^3 + u(2)^3 &&= 0 \end{aligned}$$

for $u(1)$ and $u(2)$ yields

$$u(1) = -x_1/2 - u(0)/2 \pm \sqrt{b}$$
$$u(2) = -x_1/2 - u(0)/2 \mp \sqrt{b}$$

in which

$$b = \frac{3u(0)^3 - 3u(0)^2 x_1 - 3u(0)x_1^2 - x_1^3 + 4x_2}{12(u(0) + x_1)}$$

Clearly a real solution exists only if b is positive, i.e., if both the numerator and denominator in the expression for b have the same sign. The optimal control problem $\mathbb{P}_3(x)$ is defined by

$$V_3^0(x) = \min_{\mathbf{u}} \{V_3(x, \mathbf{u}) \mid \phi(3; x, \mathbf{u}) = 0\}$$

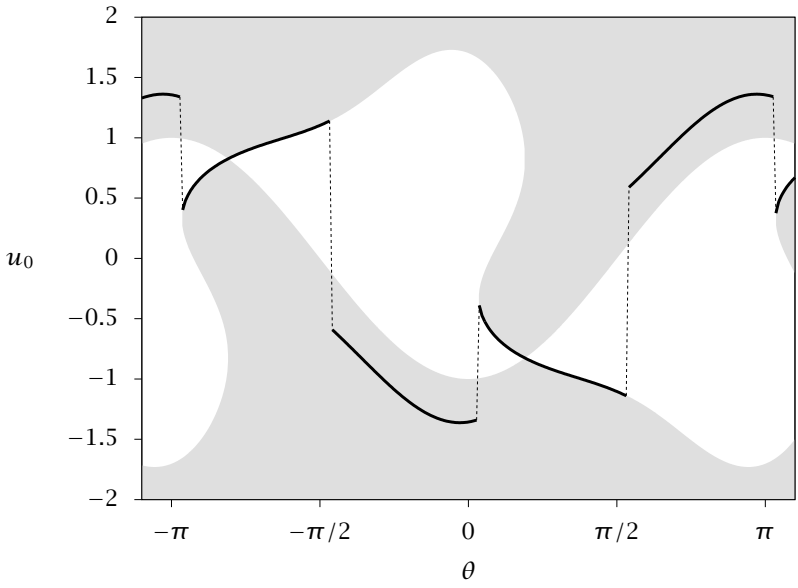


Figure 2.3: First element of control constraint set $\mathcal{U}_3(x)$ (shaded) and control law $\kappa_3(x)$ (line) versus $x = (\cos(\theta), \sin(\theta))$, $\theta \in [-\pi, \pi]$ on the unit circle for a nonlinear system with terminal constraint.

and the implicit MPC control law is $\kappa_3(\cdot)$ where $\kappa_3(x) = u^0(0; x)$, the first element in the minimizing sequence $\mathbf{u}^0(x)$. It can be shown, using analysis presented later in this chapter, that the origin is asymptotically stable for the controlled system $x^+ = f(x, \kappa_N(x))$. That this control law is necessarily discontinuous may be shown as follows. If the control is strictly positive, any trajectory originating in the first quadrant (x_1, $x_2 > 0$) moves away from the origin. If the control is strictly negative, any control originating in the third quadrant ($x_1, x_2 < 0$) also moves away from the origin. But the control cannot be zero at any nonzero point lying in the domain of attraction. If it were, this point would be a fixed point for the controlled system, contradicting the fact that it lies in the domain of attraction.

In fact, both the value function $V_3^0(\cdot)$ and the MPC control law $\kappa_3(\cdot)$ are discontinuous. Figures 2.3 and 2.4 show how $\mathcal{U}_3(x)$, $\kappa_3(x)$, and $V_3^0(x)$ vary as $x = (\cos(\theta), \sin(\theta))$ ranges over the unit circle. A further conclusion that can be drawn from this example is that it is possible

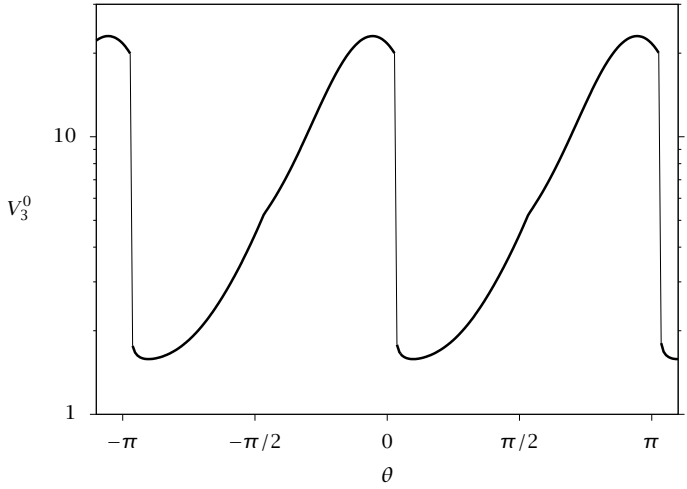


Figure 2.4: Optimal cost $V_3^0(x)$ versus $x = (\cos(\theta), \sin(\theta))$, $\theta \in [-\pi, \pi]$ on the unit circle; the discontinuity in V_3^0 is caused by the discontinuity in $\mathcal{U}_3$ as θ crosses the dashed line in Figure 2.3.

for the MPC control law to be discontinuous at points where the value function is continuous. □

2.3 Dynamic Programming Solution

We examine next the DP solution of the optimal control problem $\mathbb{P}_N(x)$, not because it provides a practical procedure but because of the insight it provides. DP can rarely be used for constrained and/or nonlinear control problems unless the state dimension n is small. MPC is best regarded as a practical means of implementing the DP solution; for a given state x it provides $V_N^0(x)$ and $\kappa_N(x)$, the value, respectively, of the value function and control law at a *point* x. DP, on the other hand, yields the value function $V_N^0(\cdot)$ and the implicit MPC control law $\kappa_N(\cdot)$.

The optimal control problem $\mathbb{P}_N(x)$ is defined, as before, by (2.7) with the cost function $V_N(\cdot)$ defined by (2.3) and the constraints by (2.4). DP yields an optimal policy $\boldsymbol{\mu}^0 = \left(\mu_0^0(\cdot), \mu_1^0(\cdot), \ldots, \mu_{N-1}^0(\cdot)\right)$, i.e., a sequence of control laws $\mu_i : \mathcal{X}_i \to \mathbb{U}$, $i = 0, 1, \ldots, N-1$. The domain $\mathcal{X}_i$ of each control law will be defined later. The optimal controlled

system is time varying and satisfies

$$x^+ = f(x, \mu_i^0(x)),\ i = 0, 1, \ldots, N-1$$

in contrast with the system using MPC, which is time invariant and satisfies

$$x^+ = f(x, \kappa_N(x)),\ i = 0, 1, \ldots, N-1$$

with $\kappa_N(\cdot) = \mu_0^0(\cdot)$. The optimal control law at time i is $\mu_i^0(\cdot)$, but receding horizon control (RHC) uses the time-invariant control law $\kappa_N(\cdot) = \mu_0(\cdot)$ obtained by assuming that at each time t, the terminal time or *horizon* is $t + N$ so that the horizon $t + N$ recedes as t increases. One consequence is that the time-invariant control law $\kappa_N(\cdot)$ is *not* optimal for the problem of controlling $x^+ = f(x, u)$ over the fixed interval $[0, T]$ in such a way as to minimize V_N and satisfy the constraints.

For all $j \in \mathbb{I}_{0:N-1}$, let $V_j(x, \mathbf{u})$, $\mathcal{U}_j(x)$, $\mathbb{Z}_j$, $\mathbb{P}_j(x)$ (and $V_j^0(x)$) be defined, respectively, by (2.3), (2.5), (2.6), and (2.7), with N replaced by j. As shown in Section C.1 of Appendix C, DP solves not only $\mathbb{P}_N(x)$ for all $x \in \mathcal{X}_N$, the domain of $V_N^0(\cdot)$, but also $\mathbb{P}_j(x)$ for all $x \in \mathcal{X}_j$, the domain of $V_j^0(\cdot)$, all $j \in \mathbb{I}_{0:N-1}$. The DP equations are, for all $x \in \mathcal{X}_j$

$$V_j^0(x) = \min_{u \in \mathbb{U}(x)} \{\ell(x, u) + V_{j-1}^0(f(x, u)) \mid f(x, u) \in \mathcal{X}_{j-1}\} \tag{2.9}$$

$$\kappa_j(x) = \arg\min_{u \in \mathbb{U}(x)} \{\ell(x, u) + V_{j-1}^0(f(x, u)) \mid f(x, u) \in \mathcal{X}_{j-1}\} \tag{2.10}$$

with

$$\mathcal{X}_j = \{x \in \mathbb{X} \mid \exists u \in \mathbb{U}(x) \text{ such that } f(x, u) \in \mathcal{X}_{j-1}\} \tag{2.11}$$

for $j = 1, 2, \ldots, N$ (j is *time to go*), with terminal conditions

$$V_0^0(x) = V_f(x)\ \forall x \in \mathcal{X}_0 \qquad \mathcal{X}_0 = \mathbb{X}_f$$

For each j, $V_j^0(x)$ is the optimal cost for problem $\mathbb{P}_j(x)$ if the current state is x, current time is $N - j$, and the terminal time is N; $\mathcal{X}_j$ is the domain of $V_j^0(x)$ and is also the set of states in $\mathbb{X}$ that can be steered to the terminal set $\mathbb{X}_f$ in j steps by an *admissible* control sequence, i.e., a control sequence that satisfies the control, state, and terminal constraints. Hence, for each j

$$\mathcal{X}_j = \{x \in \mathbb{X} \mid \mathcal{U}_j(x) \neq \varnothing\}$$

DP yields much more than an optimal control sequence for a given initial state; it yields an optimal feedback *policy* $\boldsymbol{\mu}^0$ or sequence of control laws where

$$\boldsymbol{\mu}^0 := (\mu_0(\cdot), \mu_1(\cdot), \ldots, \mu_{N-1}(\cdot)) = (\kappa_N(\cdot), \kappa_{N-1}(\cdot), \ldots, \kappa_1(\cdot))$$

At event (x, i), i.e., at state x at time i, the time to go is $N - i$ and the optimal control is

$$\mu_i^0(x) = \kappa_{N-i}(x)$$

i.e., $\mu_i^0(\cdot)$ is the optimal control law at time i. Consider an initial *event* $(x, 0)$, i.e., state x at time zero. If the terminal time (horizon) is N, the optimal control for $(x, 0)$ is $\kappa_N(x)$. The successor state, at time 1, is

$$x^+ = f(x, \kappa_N(x))$$

At event $(x^+, 1)$, the time to go to the terminal set $\mathbb{X}_f$ is $N - 1$ and the optimal control is $\kappa_{N-1}(x^+) = \kappa_{N-1}(f(x, \kappa_N(x)))$. For a given initial event $(x, 0)$, the optimal policy generates the optimal state and control trajectories $\mathbf{x}^0(x)$ and $\mathbf{u}^0(x)$ that satisfy the difference equations

$$x(0) = x \qquad u(0) = \kappa_N(x) = \mu_0(x) \tag{2.12}$$

$$x(i+1) = f(x(i), u(i)) \qquad u(i) = \kappa_{N-i}(x(i)) = \mu_i(x(i)) \tag{2.13}$$

for $i = 0, 1, \ldots, N-1$. These state and control trajectories are identical to those obtained, as in MPC, by solving $\mathbb{P}_N(x)$ directly for the particular initial event $(x, 0)$ using a mathematical programming algorithm. Dynamic programming, however, provides a solution for *any* event (x, i) such that $i \in \mathbb{I}_{0:N-1}$ and $x \in \mathcal{X}_i$.

Optimal control, in the classic sense of determining a control that minimizes a cost over the interval $[0, N]$ (so that the cost for $k > N$ is irrelevant), is generally time varying (at event (x, i), $i \in \mathbb{I}_{0:N}$, the optimal control is $\mu_i(x) = \kappa_{N-i}(x)$). Under fairly general conditions, $\mu_i(\cdot) \to \kappa_\infty(\cdot)$ as $N \to \infty$ where $\kappa_\infty(\cdot)$ is the stationary infinite horizon optimal control law. MPC and RHC, on the other hand, employ the time-invariant control $\kappa_N(x)$ for all $i \in \mathbb{I}_{\geq 0}$. Thus the state and control trajectories $\mathbf{x}_{\text{mpc}}(x)$ and $\mathbf{u}_{\text{mpc}}(x)$ generated by MPC for an initial event $(x, 0)$ satisfy the difference equations

$$x(0) = x \qquad u(0) = \kappa_N(x)$$

$$x(i+1) = f(x(i), u(i)) \qquad u(i) = \kappa_N(x(i))$$

and can be seen to differ in general from $\mathbf{x}^0(x)$ and $\mathbf{u}^0(x)$, which satisfy (2.12) and (2.13).

Before leaving this section, we obtain some properties of the solution to each partial problem $\mathbb{P}_j(x)$. For this, we require a few definitions (Blanchini and Miani, 2008).

Definition 2.9 (Positive and control invariant sets)**.**

(a) A set $X \subseteq \mathbb{R}^n$ is positive invariant for $x^+ = f(x)$ if $x \in X$ implies $f(x) \in X$.

(b) A set $X \subseteq \mathbb{R}^n$ is control invariant for $x^+ = f(x,u)$, $u \in \mathbb{U}$, if, for all $x \in X$, there exists a $u \in \mathbb{U}$ such that $f(x,u) \in X$.

We recall from our standing assumptions 2.2 and 2.3 that $f(\cdot)$, $\ell(\cdot)$ and $V_f(\cdot)$ are continuous, that $\mathbb{X}$ and $\mathbb{X}_f$ are closed, $\mathbb{U}$ is compact and that each of these sets contains the origin.

Proposition 2.10 (Existence of solutions to DP recursion)**.** *Suppose Assumptions 2.2 and 2.3 ($\mathbb{U}$ bounded) hold. Then*

(a) For all $j \in \mathbb{I}_{\geq 0}$, the cost function $V_j(\cdot)$ is continuous in $\mathbb{Z}_j$, and, for each $x \in \mathcal{X}_j$, the control constraint set $\mathcal{U}_j(x)$ is compact and a solution $\mathbf{u}^0(x) \in \mathcal{U}_j(x)$ to $\mathbb{P}_j(x)$ exists.

(b) If $\mathcal{X}_0 := \mathbb{X}_f$ is control invariant for $x^+ = f(x,u)$, $u \in \mathbb{U}(x)$ and $0 \in \mathbb{X}_f$, then, for each $j \in \mathbb{I}_{\geq 0}$, the set $\mathcal{X}_j$ is also control invariant, $\mathcal{X}_j \supseteq \mathcal{X}_{j-1}$, and $0 \in \mathcal{X}_j$. In addition, the sets $\mathcal{X}_j$ and $\mathcal{X}_{j-1}$ are positive invariant for $x^+ = f(x, \kappa_j(x))$ for $j \in \mathbb{I}_{\geq 1}$.

(c) For all $j \in \mathbb{I}_{\geq 0}$, the set $\mathcal{X}_j$ is closed.

(d) If, in addition, $\mathbb{X}_f$ is compact and the function $f^{-1}(\cdot)$[4] is bounded on bounded sets ($f^{-1}(S)$ is bounded for every bounded set $S \in \mathbb{R}^n$), then, for all $j \in \mathbb{I}_{\geq 0}$, $\mathcal{X}_j$ is compact.

Proof.

(a) This proof is almost identical to the proof of Proposition 2.4.

(b) By assumption, $\mathcal{X}_0 = \mathbb{X}_f \subseteq \mathbb{X}$ is control invariant. By (2.11)

$$\mathcal{X}_1 = \{x \in \mathbb{X} \mid \exists u \in \mathbb{U}(x) \text{ such that } f(x,u) \in \mathcal{X}_0\}$$

Since $\mathcal{X}_0$ is control invariant for $x^+ = f(x,u), u \in \mathbb{U}$, for every $x \in \mathcal{X}_0$ there exist a $u \in \mathbb{U}$ such that $f(x,u) \in \mathcal{X}_0$ so that $x \in \mathcal{X}_1$. Hence $\mathcal{X}_1 \supseteq \mathcal{X}_0$. Since for every $x \in \mathcal{X}_1$, there exists a $u \in \mathbb{U}$ such that $f(x, u) \in \mathcal{X}_0 \subseteq \mathcal{X}_1$, it follows that $\mathcal{X}_1$ is control invariant for $x^+ = f(x,u)$, $u \in \mathbb{U}(x)$. If for some integer $j \in \mathbb{I}_{\geq 0}$, $\mathcal{X}_{j-1}$ is control invariant for $x^+ = f(x,u)$, it follows by similar reasoning that $\mathcal{X}_j \supseteq \mathcal{X}_{j-1}$ and that $\mathcal{X}_j$ is control invariant. By induction $\mathcal{X}_j$ is control invariant and $\mathcal{X}_j \supseteq \mathcal{X}_{j-1}$ for all $j > 0$. Hence $0 \in \mathcal{X}_j$ for all $j \in \mathbb{I}_{\geq 0}$. That $\mathcal{X}_j$ is positive invariant for $x^+ = f(x, \kappa_j(x))$ follows from (2.10), which

[4] For any $S \subset \mathbb{R}^n$, $f^{-1}(S) := \{z \in \mathbb{Z} \mid f(z) \in S\}$

shows that $\kappa_j(\cdot)$ steers every $x \in \mathcal{X}_j$ into $\mathcal{X}_{j-1} \subseteq \mathcal{X}_j$. Since $\mathcal{X}_{j-1} \subseteq \mathcal{X}_j$, $\kappa_j(\cdot)$ also steers every $x \in \mathcal{X}_{j-1}$ into $\mathcal{X}_{j-1}$, so $\mathcal{X}_{j-1}$ is positive invariant under control law $\kappa_j(\cdot)$ as well.

(c) By Assumption 2.3, $\mathcal{X}_0 = \mathbb{X}_f$ is closed. Suppose, for some $j \in \mathbb{I}_{\geq 1}$, that $\mathcal{X}_{j-1}$ is closed. Then $\mathcal{Z}_j \coloneqq \{(x,u) \in \mathbb{Z} \mid f(x,u) \in \mathcal{X}_{j-1}\}$ is closed since $f(\cdot)$ is continuous. To prove that $\mathcal{X}_j$ is closed, take any sequence $(x_i)_{i \in \mathbb{I}_{\geq 0}}$ in $\mathcal{X}_j$ that converges to, say, $\bar{x}$. For each i, select a $u_i \in \mathbb{U}(x_i)$ such that $z_i = (x_i, u_i) \in \mathcal{Z}_j$; this is possible since $x_i \in \mathcal{X}_j$ implies $x_i \in \{\mathbb{X} \mid \mathcal{U}_j(x) \neq \emptyset\}$. Since $\mathcal{U}_j(x) \subseteq \mathbb{U}$ and $\mathbb{U}$ is bounded, by the Bolzano-Weierstrass theorem there exists a subsequence, indexed by $\mathbb{I}$, such that $u_i \to \bar{u}$ (and $x_i \to \bar{x}$) as $i \to \infty$, $i \in \mathbb{I}$. The sequence $(x_i, u_i) \in \mathcal{Z}_j, i \in \mathbb{I}$ converges, and, since $\mathcal{Z}_j$ is closed, $(\bar{x}, \bar{u}) \in \mathcal{Z}_j$. Therefore $f(\bar{x}, \bar{u}) \in \mathcal{X}_{j-1}$ and $\bar{x} \in \mathcal{X}_j$ so that $\mathcal{X}_j$ is closed. By induction $\mathcal{X}_j$ is closed for all $j \in \mathbb{I}_{\geq 0}$.

(d) Since $\mathbb{X}_f$ and $\mathbb{U}$ are bounded, so is $\mathcal{Z}_1 \subset f^{-1}(\mathbb{X}_f) \coloneqq \{(x,u) \in \mathbb{Z} \mid f(x,u) \in \mathbb{X}_f\}$ and its projection $\mathcal{X}_1$ onto $\mathbb{R}^n$. Assume then, for some $j \in \mathbb{I}_{\geq 0}$ that $\mathcal{Z}_{j-1}$ is bounded; its projection $\mathcal{X}_{j-1}$ is also bounded. Consequently, $\mathcal{Z}_j \subset f^{-1}(\mathcal{X}_{j-1})$ is also bounded and so is its projection $\mathcal{X}_j$. By induction, $\mathcal{X}_j$ is bounded, and hence, compact, for all $j \in \mathbb{I}_{\geq 0}$.

■

Part (d) of Proposition 2.10 requires that the function $f^{-1}(\cdot)$ is bounded on bounded sets. This is a mild requirement if $f(\cdot)$ is the discrete time version of a continuous system as is almost always the case in process control. If the continuous time system satisfies $\dot{x} = f_c(x,u)$ and if the sample time is Δ, then

$$f(x,u) = x + \int_0^\Delta f_c(x(s;x),u)ds$$

in which $x(s;x)$ is the solution of $\dot{x} = f_c(x,u)$ at time s if $x(0) = x$ and u is the constant input in the interval $[0,\Delta]$. It is easily shown that $f^{-1}(\cdot)$ is bounded on bounded sets if $\mathbb{U}$ is bounded and either $f(x,u) = Ax + Bu$ and A is nonsingular, or $f_c(x,u)$ is Lipschitz in x (see Exercise 2.2).

The fact that $\mathcal{X}_N$ is positive invariant for $x^+ = f(x, \kappa_N(x))$ can also be established by observing that $\mathcal{X}_N$ is the set of states x in $\mathbb{X}$ for which there exists a $\mathbf{u}$ that is feasible for $\mathbb{P}_N(x)$, i.e., for which there exists a control $\mathbf{u}$ satisfying the control, state and terminal constraints. It is shown in the next section that for every $x \in \mathcal{X}_N$, there exists a feasible control sequence $\tilde{\mathbf{u}}$ for $\mathbb{P}_N(x^+)$ ($x^+ = f(x, \kappa_N(x))$ is the successor

state) provided that $\mathbb{X}_f$ is control invariant, i.e., $\mathcal{X}_N$ is positive invariant for $x^+ = f(x, \kappa_N(x))$ if $\mathbb{X}_f$ is control invariant. An important practical consequence is that if $\mathbb{P}_N(x(0))$ can be solved for the initial state $x(0)$, then $\mathbb{P}_N(x(i))$ can be solved for any subsequent state $x(i)$ of the controlled system $x^+ = f(x, \kappa_N(x))$, a property that is sometimes called recursive feasibility. Uncertainty, in the form of additive disturbances, model error or state estimation error, may destroy this important property; techniques to restore this property when uncertainty is present are discussed in Chapter 3.

2.4 Stability

2.4.1 Introduction

The classical definition of stability was employed in the first edition of this text. This states the origin in $\mathbb{R}^n$ is globally asymptotically stable (GAS) for $x^+ = f(x)$ if the origin is *locally stable* and if the origin is *globally attractive*. The origin is *locally stable* if, for all $\varepsilon > 0$, there exists a $\delta > 0$ such that $|x| < \delta$ implies $|\phi(k;x)| < \varepsilon$ for all $k \in \mathbb{I}_{\geq 0}$ (small perturbations of the initial state from the origin cause subsequent perturbations to be small). The origin is *globally attractive* for $x^+ = f(x)$ if $|\phi(k;x)| \to 0$ as $k \to \infty$ for all $x \in \mathbb{R}^n$. This definition of stability has been widely used and is equivalent to the recently defined stronger definition given below if $f(\cdot)$ is continuous but has some disadvantages; there exist examples of systems that are asymptotically stable (AS) in the classical sense in which small perturbations in the initial state from its initial value, not the origin, can cause subsequent perturbations to be arbitrarily large. Hence we employ in this section, as discussed more fully in Appendix B, a stronger definition of asymptotic stability that avoids this undesirable behavior.

To establish stability we make use of Lyapunov theorems that are defined in terms of the function classes $\mathcal{K}$, $\mathcal{K}_\infty$ and $\mathcal{KL}$. A function belongs to class $\mathcal{K}$ if it is continuous, zero at zero, and strictly increasing; a function belongs to class $\mathcal{K}_\infty$ if it is in class $\mathcal{K}$ and unbounded; a function $\beta(\cdot)$ belongs to class $\mathcal{KL}$ if it is continuous and if, for each $k \geq 0$, $\beta(\cdot, k)$ is a class $\mathcal{K}$ function and for each $s \geq 0$, $\beta(s, \cdot)$ is nonincreasing and $\beta(s, i)$ converges to zero as $i \to \infty$. We can now state the stronger definition of stability.

Definition 2.11 (Asymptotically stable and GAS)**.** Suppose $\mathbb{X}$ is positive invariant for $x^+ = f(x)$. The origin is AS for $x^+ = f(x)$ in $\mathbb{X}$ if there

exists a $\mathcal{KL}$ function $\beta(\cdot)$ such that, for each $x \in \mathbb{X}$

$$|\phi(i;x)| \leq \beta(|x|, i) \qquad \forall i \in \mathbb{I}_{\geq 0}$$

If $\mathbb{X} = \mathbb{R}^n$, the origin is GAS for $x^+ = f(x)$.

The set $\mathbb{X}$ is called a region of attraction. Energy in a passive electrical or mechanical system provides a useful analogy to Lyapunov stability theory. In a lumped mechanical system, the total stored energy, the sum of the potential and kinetic energy, is dissipated by friction and decays to zero at which point the dynamic system is in equilibrium. Lyapunov theory follows a similar path; if a real-valued function (a Lyapunov function) can be found that is positive and decreasing if the state is not the origin, then the state converges to the origin.

Definition 2.12 (Lyapunov function)**.** Suppose that $\mathbb{X}$ is positive invariant for $x^+ = f(x)$. A function $V : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ is said to be a Lyapunov function in $\mathbb{X}$ for $x^+ = f(x)$ if there exist functions $\alpha_1, \alpha_2 \in \mathcal{K}_\infty$ and a continuous, positive definite function α_3 such that for any $x \in \mathbb{X}$

$$V(x) \geq \alpha_1(|x|) \tag{2.14}$$

$$V(x) \leq \alpha_2(|x|) \tag{2.15}$$

$$V(f(x)) - V(x) \leq -\alpha_3(|x|) \tag{2.16}$$

We now employ the following stability theorem.

Theorem 2.13 (Lyapunov stability theorem)**.** *Suppose $\mathbb{X} \subset \mathbb{R}^n$ is positive invariant for $x^+ = f(x)$. If there exists a Lyapunov function in $\mathbb{X}$ for the system $x^+ = f(x)$, then the origin is asymptotically stable in $\mathbb{X}$ for $x^+ = f(x)$. If $\mathbb{X} = \mathbb{R}^n$, then the origin is globally asymptotically stable. If $\alpha_i(|x|) = c_i |x|^a$, $a, c_i \in \mathbb{R}_{>0}$, $i = 1, 2, 3$, then the origin is exponentially stable.*

A standard approach to establish stability is to employ the value function of an infinite horizon optimal control problem as a Lyapunov function. This suggests the use of $V_N^0(\cdot)$, the value function for the finite horizon optimal control problem whose solution yields the model predictive controller, as a Lyapunov function. It is simple to show, under mild assumptions on $\ell(\cdot)$, that $V_N^0(\cdot)$ has property (2.14) for all $x \in \mathcal{X}_N$. The value function $V_\infty(\cdot)$ for infinite horizon optimal control problems does satisfy, under mild conditions, $V_\infty^0(f(x, \kappa_\infty(x))) = V_\infty^0(x) - \ell(x, \kappa_\infty(x))$ thereby ensuring satisfaction of property (2.16). Since, as is often pointed out, optimality does not imply stability, this

property does not usually hold when the horizon is finite. One of the main tasks of this chapter is show that if the ingredients $V_f(\cdot)$, $\ell(\cdot)$, and $\mathbb{X}_f$ of the finite horizon optimal control problem are chosen appropriately, then $V_N^0(f(x,\kappa_N(x))) \leq V_N^0(x) - \ell(x,\kappa_N(x))$ for all x in $\mathcal{X}_N$ enabling property (2.16) to be obtained. Property (2.15), an upper bound on the value function, is more difficult to establish but we also show that appropriate ingredients that ensures satisfaction of property (2.16) also ensures satisfaction of property (2.15).

We now address a point that we have glossed over. The solution to an optimization problem is not necessarily unique. Thus $\mathbf{u}^0(x)$ and $\kappa_N(x)$ may be set valued; any point in the set $\mathbf{u}^0(x)$ is a solution of $\mathbb{P}_N(x)$. Similarly $\mathbf{x}^0(x)$ is set valued. Uniqueness may be obtained by choosing that element in the set $\mathbf{u}^0(x)$ that has least norm; and if the minimum-norm solution is not unique, applying an arbitrary selection map in the set of minimum-norm solutions. To avoid expressions such as "let $\mathbf{u}$ be any element of the minimizing set $\mathbf{u}^0(x)$," we shall, in the sequel, use $\mathbf{u}^0(x)$ to denote any sequence in the set of minimizing sequences and use $\kappa_N(x)$ to denote $u^0(0;x)$, the first element of this sequence.

2.4.2 Stabilizing Conditions

To show that the value function $V_N^0(\cdot)$ is a valid Lyapunov function for the closed-loop system $x^+ = f(x,\kappa_N(x))$ we have to show that it satisfies (2.14), (2.15), and (2.16). We show below that $V_N^0(\cdot)$ is a valid Lyapunov function if, in addition to Assumptions 2.2 and 2.3, the following assumption is satisfied.

Assumption 2.14 (Basic stability assumption)**.** $V_f(\cdot)$, $\mathbb{X}_f$ and $\ell(\cdot)$ have the following properties:

(a) For all $x \in \mathbb{X}_f$, there exists a u (such that $(x,u) \in \mathbb{Z}$) satisfying

$$\begin{aligned} &f(x,u) \in \mathbb{X}_f \\ &V_f(f(x,u)) - V_f(x) \leq -\ell(x,u) \end{aligned}$$

(b) There exist $\mathcal{K}_\infty$ functions $\alpha_1(\cdot)$ and $\alpha_f(\cdot)$ satisfying

$$\begin{aligned} \ell(x,u) &\geq \alpha_1(|x|) \qquad \forall x \in \mathcal{X}_N, \forall u \text{ such that } (x,u) \in \mathbb{Z} \\ V_f(x) &\leq \alpha_f(|x|) \qquad \forall x \in \mathbb{X}_f \end{aligned}$$

We now show that $V_N^0(\cdot)$ is a Lyapunov function satisfying (2.14), (2.15), and (2.16) if Assumptions 2.2, 2.3, and 2.14 hold.

Lower bound for $V_N^0(\cdot)$. The lower-bound property (2.14) is easily obtained. Since $V_N^0(x) \geq \ell(x, \kappa_N(x))$ for all $x \in \mathcal{X}_N$, the lower bound (2.14) follows from Assumption 2.14(b) in which it is assumed that there exists a $\mathcal{K}_\infty$ function $\alpha_1(\cdot)$ such that $\ell(x,u) \geq \alpha_1(|x|)$ for all $x \in \mathcal{X}_N$, for all u such that $(x,u) \in \mathbb{Z}$. This assumption is satisfied by the usual choice $\ell(x,u) = (1/2)(x'Qx + u'Ru)$ with Q and R positive definite. Condition (2.14) is satisfied.

Upper bound for $V_N^0(\cdot)$. If $\mathbb{X}_f$ contains the origin in its interior, the upper bound property (2.15) can be established as follows. We show below in Proposition 2.18 that, under Assumption 2.14, $V_j^0(x) \leq V_f(x)$ for all $x \in \mathbb{X}_f$, all $j \in \mathbb{I}_{\geq 0}$. Also, under the same Assumption, there exists a $\mathcal{K}_\infty$ function $\alpha_f(\cdot)$ such that $V_f(x) \leq \alpha_f(|x|)$ for all $x \in \mathbb{X}_f$. It follows that $V_N^0(\cdot)$ has the same upper bound $\alpha_f(|x|)$ in $\mathbb{X}_f$. We now have to show that this bound on $V_N^0(\cdot)$ in $\mathbb{X}_f$ can be extended to a similar bound on $V_N^0(\cdot)$ in $\mathcal{X}_N$. We do this through two propositions. The first proposition proves that the value function $V_N^0(\cdot)$ is locally bounded.

Proposition 2.15 (The value function $V_N^0(\cdot)$ is locally bounded)**.** *Suppose Assumptions 2.2 and 2.3 ($\mathbb{U}$ bounded) hold. Then $V_N^0(\cdot)$ is locally bounded on $\mathcal{X}_N$.*

Proof. Let X be an arbitrary compact subset of $\mathcal{X}_N$. The function $V_N : \mathbb{R}^n \times \mathbb{R}^{Nm} \to \mathbb{R}_{\geq 0}$ is continuous and therefore has an upper bound on the compact set $X \times \mathbb{U}^N$. Since $\mathcal{U}_N(x) \subset \mathbb{U}^N$ for all $x \in \mathcal{X}_N$, $V_N^0 : \mathcal{X}_N \to \mathbb{R}_{\geq 0}$ has the same upper bound on X. Since X is arbitrary, $V_N^0(\cdot)$ is locally bounded on $\mathcal{X}_N$. ∎

The second proposition shows the upper bound of $V_N^0(\cdot)$ in $\mathbb{X}_f$ implies the existence of a similar upper bound in the larger set $\mathcal{X}_N$.

Proposition 2.16 (Extension of upper bound to $\mathcal{X}_N$)**.** *Suppose Assumptions 2.2 and 2.3 ($\mathbb{U}$ bounded) hold and that $\mathbb{X}_f \subseteq \mathbb{X}$ is control invariant for $x^+ = f(x,u)$, $u \in \mathbb{U}(x)$ and contains the origin in its interior. Suppose also that there exists a $\mathcal{K}_\infty$ function $\alpha(\cdot)$ such that $V_f(x) \leq \alpha(|x|)$ for all $x \in \mathbb{X}_f$. Then there exists a $\mathcal{K}_\infty$ function $\alpha_2(\cdot)$ such that*

$$V_N^0(x) \leq \alpha_2(|x|) \;\; \forall x \in \mathcal{X}_N$$

Proof. We have that $0 \leq V_N^0(x) \leq V_f(x) \leq \alpha(|x|)$ for $x \in \mathbb{X}_f$ which contains a neighborhood of zero (see also Proposition 2.18). Therefore $V_N^0(\cdot)$ is continuous at zero. The set $\mathcal{X}_N$ is closed, and $V_N^0(\cdot)$ is locally bounded on $\mathcal{X}_N$. Therefore Proposition B.25 of Appendix B applies, and the result is established. ■

In situations where $\mathbb{X}_f$ does not have an interior, such as when $\mathbb{X}_f = \{0\}$, we cannot establish an upper bound for $V_N^0(\cdot)$ and resort to the following assumption.

Assumption 2.17 (Weak controllability)**.** There exists a $\mathcal{K}_\infty$ function $\alpha(\cdot)$ such that

$$V_N^0(x) \leq \alpha(|x|) \;\forall x \in \mathcal{X}_N$$

Assumption 2.17 is weaker than a controllability assumption. It confines attention to those states that can be steered to $\mathbb{X}_f$ in N steps and merely requires that the cost of doing so is not excessive.

Descent property for $V_N^0(\cdot)$. Let x be any state in $\mathcal{X}_N$ at time zero. Then

$$V_N^0(x) = V_N(x, \mathbf{u}^0(x))$$

in which

$$\mathbf{u}^0(x) = \left(u^0(0;x), u^0(1;x), \ldots, u^0(N-1;x)\right)$$

is any minimizing control sequence. The resultant optimal state sequence is

$$\mathbf{x}^0(x) = \left(x^0(0;x), x^0(1;x), \ldots, x^0(N;x)\right)$$

in which $x^0(0;x) = x$ and $x^0(1;x) = x^+$. The successor state to x at time zero is $x^+ = f(x, \kappa_N(x)) = x^0(1;x)$ at time 1 where $\kappa_N(x) = u^0(0;x)$, and

$$V_N^0(x^+) = V_N(x^+, \mathbf{u}^0(x^+))$$

in which

$$\mathbf{u}^0(x^+) = \left(u^0(0;x^+), u^0(1;x^+), \ldots, u^0(N-1;x^+)\right)$$

It is difficult to compare $V_N^0(x)$ and $V_N^0(x^+)$ directly, but

$$V_N^0(x^+) = V_N(x^+, \mathbf{u}^0(x^+)) \leq V_N(x^+, \tilde{\mathbf{u}})$$

where $\tilde{\mathbf{u}}$ is any feasible control sequence for $\mathbb{P}_N(x^+)$, i.e., any control sequence in $\mathcal{U}_N(x)$. To facilitate comparison of $V_N(x^+, \tilde{\mathbf{u}})$ with $V_N^0(x) = V_N(x, \mathbf{u}^0(x))$, we choose

$$\tilde{\mathbf{u}} = \left(u^0(1;x), \ldots, u^0(N-1;x), u\right)$$

in which $u \in \mathbb{U}$ still has to be chosen. Comparing $\tilde{\mathbf{u}}$ with $\mathbf{u}^0(x)$ shows that $\tilde{\mathbf{x}}$, the state sequence due to control sequence $\tilde{\mathbf{u}}$, is

$$\tilde{\mathbf{x}} = \left(x^0(1;x), x^0(2;x), \ldots, x^0(N;x), f(x^0(N;x), u)\right)$$

in which $x^0(1;x) = x^+ = f(x, \kappa_N(x))$. Because $\mathbf{x}^0$ coincides with $\tilde{\mathbf{x}}$ and $\mathbf{u}(\cdot)$ coincides with $\tilde{\mathbf{u}}$ for $i = 1, 2, \ldots, N-1$ (but not for $i = N$), a simple calculation yields

$$V_N(x^+, \tilde{\mathbf{u}}) = \sum_{j=1}^{N-1} \ell(x^0(j;x), u^0(j;x)) + \ell(x^0(N;x)) + V_f(f(x^0(N;x), u))$$

But

$$\begin{aligned} V_N^0(x) &= V_N(x, \mathbf{u}^0(x)) \\ &= \ell(x, \kappa_N(x)) + \sum_{j=1}^{N-1} \ell(x^0(j;x), u^0(j;x)) + V_f(x^0(N;x)) \end{aligned}$$

so that

$$\sum_{j=1}^{N-1} \ell(x^0(j;x), u^0(j;x)) = V_N^0(x) - \ell(x, \kappa_N(x)) - V_f(x^0(N;x))$$

Hence

$$\begin{aligned} V_N^0(x) \le V_N(x^+, \tilde{\mathbf{u}}) = V_N^0(x) - \ell(x, \kappa_N(x)) - V_f(x^0(N;x)) + \\ \ell(x^0(N;x), u) + V_f(f(x^0(N;x), u)) \end{aligned}$$

It follows that

$$V_N^0(f(x, \kappa_N(x))) \le V_N^0(x) - \ell(x, \kappa_N(x)) \tag{2.17}$$

for all $x \in \mathbb{X}$ *if* the function $V_f(\cdot)$ and the set $\mathbb{X}_f$ have the property that, for all $x \in \mathbb{X}_f$, there exists a $u \in \mathbb{U}$ such that

$$(x, u) \in \mathbb{Z},\ V_f(f(x, u)) \le V_f(x) - \ell(x, u),\ \text{and}\ f(x, u) \in \mathbb{X}_f \tag{2.18}$$

But this condition is satisfied by the stabilizing condition, Assumption 2.14. Since $\ell(x, \kappa_N(x)) \geq \alpha_1(|x|)$ for all $x \in \mathbb{X}$, $V_N^0(\cdot)$ has the desired descent property (2.16).

To complete the proof that the value function satisfies (2.14), (2.15), and (2.16), we have to prove the assertion, made in obtaining the upper bound for $V_N^0(\cdot)$, that $V_j^0(x) \leq V_f(x)$ for all $x \in \mathbb{X}_f$, all $j \in \mathbb{I}_{\geq 0}$. This assertion follows from the monotonicity property of the value function $V_N^0(\cdot)$. This interesting result was first obtained for the unconstrained linear quadratic optimal control problem.

Proposition 2.18 (Monotonicity of the value function)**.** *Suppose that Assumptions 2.2, 2.3 (𝕌 bounded), and 2.14 hold. Then*

$$V_{j+1}^0(x) \leq V_j^0(x) \ \forall x \in \mathcal{X}_j, \ \forall j \in \mathbb{I}_{\geq 0}$$

and

$$V_j^0(x) \leq V_f(x) \ \forall x \in \mathbb{X}_f, \ \forall j \in \mathbb{I}_{\geq 0}$$

Proof. From the DP recursion (2.9)

$$V_1^0(x) = \min_{u \in \mathbb{U}(x)} \{\ell(x,u) + V_0^0(f(x,u)) \mid f(x,u) \in \mathcal{X}_0\}$$

But $V_0^0(\cdot) \coloneqq V_f(\cdot)$ and $\mathcal{X}_0 \coloneqq \mathbb{X}_f$. Also, by Assumption 2.14

$$\min_{u \in \mathbb{U}(x)} \{\ell(x,u) + V_f(f(x,u)) \mid f(x,u) \in \mathbb{X}_f\} \leq V_f(x) \qquad \forall x \in \mathbb{X}_f$$

so that

$$V_1^0(x) \leq V_0^0(x) \qquad \forall x \in \mathcal{X}_0 = \mathbb{X}_f$$

Next, suppose that for some $j \geq 1$

$$V_j^0(x) \leq V_{j-1}^0(x) \qquad \forall x \in \mathcal{X}_{j-1}$$

Then, using the DP equation (2.9)

$$\begin{aligned} V_{j+1}^0(x) - V_j^0(x) &= \ell(x, \kappa_{j+1}(x)) + V_j^0(f(x, \kappa_{j+1}(x))) \\ &\quad - \ell(x, \kappa_j(x)) - V_{j-1}^0(f(x, \kappa_j(x))) \qquad \forall x \in \mathcal{X}_j \subseteq \mathcal{X}_{j+1} \end{aligned}$$

Since $\kappa_j(x)$ may *not* be optimal for $\mathbb{P}_{j+1}(x)$ for all $x \in \mathcal{X}_j \subseteq \mathcal{X}_{j+1}$, we have

$$\begin{aligned} V_{j+1}^0(x) - V_j^0(x) &\leq \ell(x, \kappa_j(x)) + V_j^0(f(x, \kappa_j(x))) \\ &\quad - \ell(x, \kappa_j(x)) - V_{j-1}^0(f(x, \kappa_j(x))) \qquad \forall x \in \mathcal{X}_j \end{aligned}$$

Also, from (2.11), $x \in \mathcal{X}_j$ implies $f(x, \kappa_j(x)) \in \mathcal{X}_{j-1}$ so that, by assumption, $V_j^0(f(x, \kappa_j(x))) \leq V_{j-1}^0(f(x, \kappa_j(x)))$ for all $x \in \mathcal{X}_j$. Hence

$$V_{j+1}^0(x) \leq V_j^0(x) \qquad \forall x \in \mathcal{X}_j$$

By induction

$$V_{j+1}^0(x) \leq V_j^0(x) \qquad \forall x \in \mathcal{X}_j, \ \forall j \in \mathbb{I}_{\geq 0}$$

Since the set sequence $(\mathcal{X}_j)_{\mathbb{I}_{\geq 0}}$ has the nested property $\mathcal{X}_j \subset \mathcal{X}_{j+1}$ for all $j \in \mathbb{I}_{\geq 0}$, it follows that $V_j^0(x) \leq V_f(x)$ for all $x \in \mathbb{X}_f$, all $j \in \mathbb{I}_{\geq 0}$. ■

The monotonicity property Proposition 2.18 also holds even if $\mathbb{U}(x)$ is not compact provided that the minimizer in the DP recursion always exists; this is the case for the linear-quadratic problem.

The monotonicity property can also be used to establish the (previously established) descent property of $V_N^0(\cdot)$ by noting that

$$\begin{aligned} V_N^0(x) &= \ell(x, \kappa_N(x)) + V_{N-1}^0(f(x, \kappa_N(x))) \\ &= \ell(x, \kappa_N(x)) + V_N^0(f(x, \kappa_N(x))) + \\ &\quad [V_{N-1}^0(f(x, \kappa_N(x))) - V_N^0(f(x, \kappa_N(x)))] \end{aligned}$$

so that using the monotonicity property

$$\begin{aligned} V_N^0(f(x, \kappa_N(x))) &= V_N^0(x) - \ell(x, \kappa_N(x)) + \\ &\qquad [V_N^0(f(x, \kappa_N(x))) - V_{N-1}^0(f(x, \kappa_N(x)))] \\ &\leq V_N^0(x) - \ell(x, \kappa_N(x)) \qquad \forall x \in \mathcal{X}_N \end{aligned}$$

which is the desired descent property.

Since inequalities (2.14), (2.15), and (2.16) are all satisfied we have proved (for $\mathbb{U}$ bounded)

Theorem 2.19 (Asymptotic stability of the origin)**.** *Suppose Assumptions 2.2, 2.3, 2.14, and 2.17 are satisfied. Then*

(a) There exists $\mathcal{K}_\infty$ functions $\alpha_1(\cdot)$ and $\alpha_2(\cdot)$ such that for all $x \in \mathcal{X}_N$ ($\tilde{\mathcal{X}}_N^c$, for each $c \in \mathbb{R}_{>0}$)

$$\begin{aligned} \alpha_1(|x|) \leq V_N^0(x) &\leq \alpha_2(|x|) \\ V_N^0(f(x, \kappa_N(x))) - V_N^0(x) &\leq -\alpha_1(|x|) \end{aligned}$$

(b) The origin is asymptotically stable in $\mathcal{X}_N$ ($\tilde{\mathcal{X}}_N^c$, for each $c \in \mathbb{R}_{>0}$) for $x^+ = f(x, \kappa_N(x))$.

For the proof with $\mathbb{U}$ unbounded, note that the lower bound and descent property remain satisfied as before. For the upper bound, if $\mathbb{X}_f$ contains the origin in its interior, we have that, since $V_f(\cdot)$ is continuous, for each $c > 0$ there exists $0 < \tau \leq c$, such that $\text{lev}_\tau V_f$ contains a neighborhood of the origin and is a subset of both $\mathbb{X}_f$ and $\tilde{\mathcal{X}}_N^c$. One can then show that $V_N^0(\cdot) \leq V_f(\cdot)$ for each $N \geq 0$ on this sublevel set, and therefore $V_N^0(\cdot)$ is continuous at the origin so that again Proposition B.25 applies, and Assumption 2.17 is satisfied on $\tilde{\mathcal{X}}_N^c$ for each $c \in \mathbb{R}_{>0}$.

As discussed above, Assumption 2.17 is immediate if the origin lies in the interior of $\mathbb{X}_f$. In other cases, e.g., when the stabilizing ingredient is the terminal equality constraint $x(N) = 0$ ($\mathbb{X}_f = \{0\}$), Assumption 2.17 is taken directly. See Proposition 2.38 for some additional circumstances in which Assumption 2.17 is satisfied.

2.4.3 Exponential Stability

Exponential stability is defined as follows.

Definition 2.20 (Exponential stability)**.** Suppose $X \subseteq \mathbb{R}^n$ is positive invariant for $x^+ = f(x)$. The origin is exponentially stable for $x^+ = f(x)$ in X if there exist $c \in \mathbb{R}_{>0}$ and $\gamma \in (0, 1)$ such that

$$|\phi(i; x)| \leq c\,|x|\,\gamma^i$$

for all $x \in X$, all $i \in \mathbb{I}_{\geq 0}$.

Theorem 2.21 (Lyapunov function and exponential stability)**.** *Suppose $X \subset \mathbb{R}^n$ is positive invariant for $x^+ = f(x)$. If there exists a Lyapunov function in X for the system $x^+ = f(x)$ with $\alpha_i(\cdot) = c_i\,|\cdot|^a$ in which a, $c_i \in \mathbb{R}_{>0}$ $i = 1, 2, 3$, then the origin is exponentially stable for $x^+ = f(x)$ in X.*

The proof of this result is left as an exercise.

2.4.4 Controllability and Observability

We have not yet made any assumptions on controllability (stabilizability) or observability (detectability) of the system (2.1) being controlled, which may be puzzling since such assumptions are commonly required in optimal control to, for example, establish existence of a solution to the optimal control problem. The reasons for this omission are that such assumptions are implicitly required, at least locally, for the basic stability Assumption 2.14, and that we restrict attention to $\mathcal{X}_N$, the set of states that can be steered to $\mathbb{X}_f$ in N steps satisfying all constraints.

Stage cost $\ell(\cdot)$ not positive definite. In the previous stability analysis we assume that the function $(x,u) \mapsto \ell(x,u)$ is positive definite; more precisely, we assume that there exists a $\mathcal{K}_\infty$ function $\alpha_1(\cdot)$ such that $\ell(x,u) \geq \alpha_1(|x|)$ for all (x,u). Often we assume that $\ell(\cdot)$ is quadratic, satisfying $\ell(x,u) = (1/2)(x'Qx + u'Ru)$ where Q and R are positive definite. In this section we consider the case where the stage cost is $\ell(y,u)$ where $y = h(x)$ and the function $h(\cdot)$ is not necessarily invertible. An example is the quadratic stage cost $\ell(y,u) = (1/2)(y'Q_y y + u'Ru)$ where Q_y and R are positive definite, $y = Cx$, and C is not invertible; hence the stage cost is $(1/2)(x'Qx + u'Ru)$ where $Q = C'Q_y C$ is merely positive semidefinite. Since now $\ell(\cdot)$ does not satisfy $\ell(x,u) \geq \alpha_1(|x|)$ for all $(x,u) \in \mathbb{Z}$ and some $\mathcal{K}_\infty$ function $\alpha_1(\cdot)$, we have to make an additional assumption in order to establish asymptotic stability of the origin for the closed-loop system. An appropriate assumption is input/output-to-state-stability (IOSS), which ensures the state goes to zero as the input and output go to zero. We recall Definition B.51, restated here.

Definition 2.22 (Input/output-to-state stable (IOSS))**.** The system $x^+ = f(x,u)$, $y = h(x)$ is IOSS if there exist functions $\beta(\cdot) \in \mathcal{KL}$ and $\gamma_1(\cdot)$, $\gamma_2(\cdot) \in \mathcal{K}$ such that for every initial state $x \in \mathbb{R}^n$, every control sequence $\mathbf{u}$, and all $i \geq 0$

$$|x(i)| \leq \max\{\beta(|x|, i), \gamma_1(\|\mathbf{u}\|_{0:i-1}), \gamma_2(\|\mathbf{y}\|_{0:i})\}$$

in which $x(i) := \phi(i;x,\mathbf{u})$ is the solution of $x^+ = f(x,u)$ at time i if the initial state is x and the input sequence is $\mathbf{u}$; $y(i) := h(x(i))$ is the output, and $\|\mathbf{d}\|_{a:b} := \max_{a \leq j \leq b} |d(j)|$ denotes the max norm of a sequence.

Note that for linear systems, IOSS is equivalent to detectability of (A,C) (see Exercise 4.5).

We assume as usual that Assumptions 2.2 and 2.3 are satisfied, but we replace Assumption 2.14 by the following.

Assumption 2.23 (Modified basic stability assumption)**.** $V_f(\cdot)$, $\mathbb{X}_f$ and $\ell(\cdot)$ have the following properties.

(a) For all $x \in \mathbb{X}_f$, there exists a u (such that $(x,u) \in \mathbb{Z}$) satisfying

$$V_f(f(x,u)) - V_f(x) \leq -\ell(h(x),u), \quad f(x,u) \in \mathbb{X}_f$$

(b) There exist $\mathcal{K}_\infty$ functions $\alpha_1(\cdot)$ and $\alpha_f(\cdot)$ satisfying

$$\begin{aligned} \ell(y,u) &\geq \alpha_1(|(y,u)|) \quad && \forall (y,u) \in \mathbb{R}^p \times \mathbb{R}^m \\ V_f(x) &\leq \alpha_f(|x|) && \forall x \in \mathbb{X}_f \end{aligned}$$

Note that in the modification of Assumption 2.14 we have changed only the lower-bound inequality for stage cost $\ell(y,u)$. With these assumptions we can then establish asymptotic stability of the origin.

Theorem 2.24 (Asymptotic stability with stage cost $\ell(y,u)$)**.** *Suppose Assumptions 2.2, 2.3, 2.17 and 2.23 are satisfied, and the system $x^+ = f(x,u), y = h(x)$ is IOSS. Then there exists a Lyapunov function in $\mathcal{X}_N$ ($\tilde{\mathcal{X}}_N^c$, for each $c \in \mathbb{R}_{>0}$) for the closed-loop system $x^+ = f(x, \kappa_N(x))$, and the origin is asymptotically stable in $\mathcal{X}_N$ ($\tilde{\mathcal{X}}_N^c$, for each $c \in \mathbb{R}_{>0}$).*

Proof. For the case of bounded $\mathbb{U}$, Assumptions 2.2, 2.3, and 2.23(a) guarantee the existence of the optimal solution of the MPC problem and the positive invariance of $\mathcal{X}_N$ for $x^+ = f(x, \kappa_N(x))$, but the nonpositive definite stage cost gives the following modified inequalities

$$\begin{aligned} \ell(h(x),u) \leq V_N^0(x) &\leq \alpha_2(|x|) \\ V_N^0(f(x,\kappa_N(x))) - V_N^0(x) &\leq -\ell(h(x),u) \end{aligned}$$

so $V_N^0(\cdot)$ is no longer a Lyapunov function for the closed-loop system. Because the system is IOSS and $\ell(y,u) \geq \alpha_1(|(y,u)|)$, however, Theorem B.53 in Appendix B provides that for *any* $\gamma(\cdot) \in \mathcal{K}_\infty$ there exists an IOSS-Lyapunov function $\Lambda(\cdot)$ for which the following holds for all $(x,u) \in \mathbb{Z}$ for which $f(x,u) \in \mathbb{X}$

$$\begin{aligned} \gamma_1(|x|) \leq \Lambda(x) &\leq \gamma_2(|x|) \\ \Lambda(f(x,u)) - \Lambda(x) &\leq -\rho(|x|) + \gamma(\ell(h(x),u)) \end{aligned}$$

with $\gamma_1, \gamma_2 \in \mathcal{K}_\infty$ and *continuous* function $\rho \in \mathcal{PD}$. Note that these inequalities certainly apply for $u = \kappa_N(x)$ since $(x, \kappa_N(x)) \in \mathbb{Z}$ and $f(x, \kappa_N(x)) \in \mathcal{X}_N \subseteq \mathbb{X}$. Therefore we choose the linear $\mathcal{K}_\infty$ function $\gamma(\cdot) = (\cdot)$, take $V(\cdot) = V_N^0(\cdot) + \Lambda(\cdot)$ as our candidate Lyapunov function, and obtain for all $x \in \mathcal{X}_N$

$$\begin{aligned} \overline{\alpha}_1(|x|) \leq V(x) &\leq \overline{\alpha}_2(|x|) \\ V(f(x,\kappa_N(x))) - V(x) &\leq -\rho(|x|) \end{aligned}$$

with $\mathcal{K}_\infty$ functions $\overline{\alpha}_1(\cdot) := \gamma_1(\cdot)$ and $\overline{\alpha}_2(\cdot) := \alpha_2(\cdot) + \gamma_2(\cdot)$. From Definition 2.12, $V(\cdot)$ is a Lyapunov function in $\mathcal{X}_N$ for the system $x^+ = f(x, \kappa_N(x))$. Therefore the origin is asymptotically stable in $\mathcal{X}_N$ from Theorem 2.13. Treat unbounded $\mathbb{U}$ as in the proof of Theorem 2.19. ■

Note that we have here the appearance of a Lyapunov function that is *not* the optimal value function of the MPC regulation problem. In earlier MPC literature, observability rather than detectability was often employed as the extra assumption required to establish asymptotic stability. Exercise 2.14 discusses that approach.

2.4.5 Time-Varying Systems

Most of the control problems discussed in this book are time invariant. Time-varying problems do arise in practice, however, even if the system being controlled is time invariant. One example occurs when an observer or filter is used to estimate the state of the system being controlled since bounds on the state estimation error are often time varying. In the deterministic case, for example, state estimation error decays exponentially to zero. Another example occurs when the desired equilibrium is not a state-control pair (x_s, u_s) but a periodic trajectory. In this section, which may be omitted in the first reading, we show how MPC may be employed for a class of time-varying systems.

The problem. The time-varying nonlinear system is described by

$$x^+ = f(x, u, i)$$

where x is the current state at time i, u the current control, and x^+ the successor state at time $i + 1$. For each integer i, the function $f(\cdot, i)$ is assumed to be continuous. The solution of this system at time $k \geq i$ given that the initial state is x at time i is denoted by $\phi(k; x, \mathbf{u}, i)$; the solution now depends on both the time i and current time k rather than merely on the difference $k - i$ as in the time-invariant case. The cost $V_N(x, \mathbf{u}, i)$ also depends on time i and is defined by

$$V_N(x, \mathbf{u}, i) := \sum_{k=i}^{i+N-1} \ell(x(k), u(k), k) + V_f(x(i+N), i+N)$$

in which $x(k) := \phi(k; x, \mathbf{u}, i)$, $\mathbf{u} = (u(i), u(i+1), \ldots, u(i+N-1))$, and the stage cost $\ell(\cdot)$ and terminal cost $V_f(\cdot)$ are time varying. The state and control constraints are also time varying

$$x(i) \in \mathbb{X}(i) \qquad u(i) \in \mathbb{U}(i)$$

for all i. In addition, there is a time-varying terminal constraint

$$x(i+N) \in \mathbb{X}_f(i+N)$$

in which i is the current time. The time-varying optimal control problem at event (x,i) is $\mathbb{P}_N(x,i)$ defined by

$$\mathbb{P}_N(x,i): \quad V_N^0(x,i) = \min_{\mathbf{u}} \{V_N(x,\mathbf{u},i) \mid \mathbf{u} \in \mathcal{U}_N(x,i)\}$$

in which $\mathcal{U}_N(x,i)$ is the set of control sequences $\mathbf{u} = \big((u(i), u(i+1), \ldots, u(i+N-1)\big)$ satisfying the state, control and terminal constraints, i.e.,

$$\mathcal{U}_N(x,i) := \{\mathbf{u} \mid (x,\mathbf{u}) \in \mathbb{Z}_N(i)\}$$

in which, for each i, $\mathbb{Z}_N(i) \subset \mathbb{R}^n \times \mathbb{R}^{Nm}$ is defined by

$$\mathbb{Z}_N(i) := \big\{(x,\mathbf{u}) \mid u(k) \in \mathbb{U}(k), \quad \phi(k;x,\mathbf{u},i) \in \mathbb{X}(k), \forall k \in \mathbb{I}_{i:i+N-1}, \\ \phi(i+N;x,\mathbf{u},i) \in \mathbb{X}_f(i+N)\big\}$$

For each time i, the domain of $V_N^0(\cdot,i)$ is $\mathcal{X}_N(i)$ where

$$\begin{aligned}\mathcal{X}_N(i) &:= \{x \in \mathbb{X}(i) \mid \mathcal{U}_N(x,i) \neq \emptyset\} \\ &= \{x \in \mathbb{X}(i) \mid \exists \mathbf{u} \text{ such that } (x,\mathbf{u}) \in \mathbb{Z}_N(i)\}\end{aligned}$$

which is the projection of $\mathbb{Z}_N(i)$ onto $\mathbb{X}(i)$. Our standing assumptions (2.2 and 2.3) are replaced, in the time-varying case, by

Assumption 2.25 (Continuity of system and cost; time-varying case)**.** The functions $(x,u) \mapsto f(x,u,i)$, $(x,u) \mapsto \ell(x,u,i)$ and $x \mapsto V_f(x,i)$ are continuous for all $i \in \mathbb{I}_{\geq 0}$. Also, for all $i \in \mathbb{I}_{\geq 0}$, $f(0,0,i) = 0$, $\ell(0,0,i) = 0$ and $V_f(0,i) = 0$.

Assumption 2.26 (Properties of constraint sets; time-varying case)**.** For each $i \in \mathbb{I}_{\geq 0}$, $\mathbb{X}(i)$ and $\mathbb{X}_f(i)$ are closed, $\mathbb{X}_f(i) \subset \mathbb{X}(i)$ and $\mathbb{U}(i)$ are compact; the sets $\mathbb{U}(i), i \in \mathbb{I}_{\geq 0}$ are uniformly bounded by the compact set $\bar{\mathbb{U}}$. Each set contains the origin.

In making these assumptions we are implicitly assuming, as before, that the desired setpoint has been shifted to the origin, but in this case, it need not be constant in time. For example, letting $\bar{x}$ and $\bar{u}$ be the original positional variables, we can consider a time-varying reference trajectory $(\bar{x}_r(i), \bar{u}_r(i))$ by defining $x(i) := \bar{x}(i) - \bar{x}_r(i)$ and $u(i) := \bar{u}(i) - \bar{u}_r(i)$. Depending on the application, $\bar{x}_r(i)$ and $\bar{u}_r(i)$ could be constant, periodic, or generally time varying. In any case, because of the time-varying nature of the problem, we need to extend our definitions of invariance and control invariance.

Definition 2.27 (Sequential positive invariance and sequential control invariance)**.**

(a) A sequence of sets $(X(i))_{i\geq 0}$ is *sequentially positive invariant* for the system $x^+ = f(x,i)$ if for any $i \geq 0$, $x \in X(i)$ implies $f(x,i) \in X(i+1)$.

(b) A sequence of sets $(X(i))_{i\geq 0}$ is *sequentially control invariant* for the system $x^+ = f(x,u,i)$ if for any $i \geq 0$ and $x \in X(i)$, there exists a $u \in \mathbb{U}(i)$ such that $x^+ = f(x,u,i) \in X(i+1)$.

Let $(\mathcal{X}(i))_{i\geq 0}$ be sequentially positive invariant. If $x \in \mathcal{X}(i_0)$ for some $i_0 \geq 0$, then $\phi(i;x,i_0) \in \mathcal{X}(i)$ for all $i \geq i_0$.

The following results, which are analogs of the results for time-invariant systems given previously, are stated without proof.

Proposition 2.28 (Continuous system solution; time-varying case)**.** *Suppose Assumptions 2.25 and 2.26 are satisfied. For each initial time $i_0 \geq 0$ and final time $i \geq i_0$, the function $(x,\mathbf{u}) \mapsto \phi(i;x,\mathbf{u},i_0)$ is continuous.*

Proposition 2.29 (Existence of solution to optimal control problem; time-varying case)**.** *Suppose Assumptions 2.25 and 2.26 are satisfied. Then for each time $i \in \mathbb{I}_{\geq 0}$*

(a) The function $(x,\mathbf{u}) \mapsto V_N(x,\mathbf{u},i)$ is continuous in $\mathbb{Z}_N(i)$.

(b) For each $x \in \mathcal{X}_N(i)$, the control constraint set $\mathcal{U}_N(x,i)$ is compact.

(c) For each $x \in \mathcal{X}_N(i)$, a solution to $\mathbb{P}_N(x,i)$ exists.

(d) $\mathcal{X}_N(i)$ is closed and $x = 0 \in \mathcal{X}_N(i)$.

(e) If $\left(\mathbb{X}_f(i)\right)_{i\in\mathbb{I}_{\geq 0}}$ is sequentially control invariant for $x^+ = f(x,u,i)$, then $(\mathcal{X}_N(i))_{i\in\mathbb{I}_{\geq 0}}$ is sequentially control invariant for $x^+ = f(x,u,i)$ and sequentially positive invariant for $x^+ = f(x,\kappa_N(x,i),i)$.

Stability. Our definitions of AS (asymptotic stability) and GAS (global asymptotic stability) also require slight modification for the time-varying case.

Definition 2.30 (Asymptotically stable and GAS for time-varying systems)**.** Suppose that the sequence $(X(i))_{i\geq 0}$ is sequentially positive invariant for $x^+ = f(x,i)$. The origin is *asymptotically stable* in the sequence $(X(i))_{i\geq 0}$ for $x^+ = f(x,i)$ if the following holds for all $i \geq i_0 \geq 0$, and $x \in X(i_0)$

$$\left|\phi(i;x,i_0)\right| \leq \beta(|x|, i-i_0) \tag{2.19}$$

in which $\beta \in \mathcal{KL}$ and $\phi(i; x, i_0)$ is the solution to $x^+ = f(x, i)$ at time $i \geq i_0$ with initial condition x at time $i_0 \geq 0$. If $X(i) = \mathbb{R}^n$, the origin is *globally asymptotically stable* (GAS).

This definition is somewhat restrictive because $|\phi(i, x, i_0)|$ depends on $i - i_0$ rather than on i.

Definition 2.31 (Lyapunov function: time-varying, constrained)**.** Let the sequence $(X(i))_{i\geq 0}$ be sequentially positive invariant, and let $V(\cdot, i) : X(i) \to \mathbb{R}_{\geq 0}$ satisfy for all $x \in X(i), i \in \mathbb{I}_{\geq 0}$

$$\begin{aligned} \alpha_1(|x|_{\mathcal{A}}) \leq V(x, i) \leq \quad & \alpha_2(|x|_{\mathcal{A}}) \\ \Delta V(x, i) \leq & -\alpha_3(|x|_{\mathcal{A}}) \end{aligned}$$

with $\Delta V(x, i) := V(f(x, i), i + 1) - V(x, i)$, $\alpha_1, \alpha_2, \alpha_3 \in \mathcal{K}_\infty$. Then the function $V(\cdot, \cdot)$ is a time-varying Lyapunov function in the sequence $(X(i))_{i\geq 0}$ for $x^+ = f(x, i)$.

This definition requires a single, time-invariant bound for each $\alpha_j(\cdot)$, $j \in \{1, 2, 3\}$, which is not overly restrictive. For example, supposing there is a sequence of lower bounds $\left(\alpha_1^i(\cdot)\right)_{\mathbb{I}_{\geq 0}}$, it is necessary only that the infimum

$$\alpha_1(\cdot) := \inf_{i \in \mathbb{I}_{\geq 0}} \alpha_1^i(\cdot)$$

is class $\mathcal{K}_\infty$. If the system is time invariant or periodic, this property is satisfied (as the inf becomes a min over a finite set), but it does preclude bounds that become arbitrarily flat, such as $\alpha_1^i(s) = \frac{1}{i+1}s^2$. A similar argument holds for $j \in \{2, 3\}$ (using sup instead of inf for $j = 2$). We can now state a stability definition that we employ in this chapter

Theorem 2.32 (Lyapunov theorem for asymptotic stability (time-varying, constrained))**.** *Let the sequence $(X(i))_{\geq 0}$ be sequentially positive invariant for the system $x^+ = f(x, i)$, and $V(\cdot, \cdot)$ be a time-varying Lyapunov function in the sequence $(X(i))_{\geq 0}$ for $x^+ = f(x, i)$. Then the origin is asymptotically stable in $X(i)$ at each time $i \geq 0$ for $x^+ = f(x, i)$.*

The proof of this theorem is given in Appendix B (see Theorem B.24).

Model predictive control of time-varying systems. As before, the receding horizon control law $\kappa_N(\cdot)$, which is now time varying, is not necessarily optimal or stabilizing. By choosing the time-varying ingredients $V_f(\cdot)$ and $\mathbb{X}_f$ in the optimal control problem appropriately, however, stability can be ensured, as we now show. We replace the basic stability assumption 2.14 by its time-varying extension.

Assumption 2.33 (Basic stability assumption; time-varying case)**.**
(a) For all $i \in \mathbb{I}_{\geq 0}$, all $x \in \mathbb{X}_f(i)$, there exists a $u \in \mathbb{U}(i)$ such that

$$\begin{aligned} &f(x,u,i) \in \mathbb{X}_f(i+1) \\ &V_f(f(x,u,i),i+1) - V_f(x,i) \leq -\ell(x,u,i) \end{aligned}$$

(b) There exist $\mathcal{K}_\infty$ functions $\alpha_1(\cdot)$ and $\alpha_f(\cdot)$ satisfying

$$\begin{aligned} &\ell(x,u,i) \geq \alpha_1(|x|) \ \forall x \in \mathcal{X}_N(i), \forall u \text{ such that } (x,u) \in \mathbb{Z}_N(i), \forall i \in \mathbb{I}_{\geq 0} \\ &V_f(x,i) \leq \alpha_f(|x|), \ \forall x \in \mathbb{X}_f(i), \forall i \in \mathbb{I}_{\geq 0} \end{aligned}$$

As in the case of the time-varying Lyapunov function, requiring time-invariant bounds is typically not restrictive. A direct consequence of Assumption 2.33 is the descent property given in the following proposition.

Proposition 2.34 (Optimal cost decrease; time-varying case)**.** *Suppose Assumptions 2.25, 2.26, and 2.33 hold. Then*

$$V_N^0(f(x,\kappa_N(x,i),i),i+1) \leq V_N^0(x,i) - \ell(x,\kappa_N(x,i),i) \qquad (2.20)$$

for all $x \in \mathcal{X}_N(i)$, all $i \in \mathbb{I}_{\geq 0}$.

Proposition 2.35 (MPC cost is less than terminal cost)**.** *Suppose Assumptions 2.25, 2.26, and 2.33 hold. Then*

$$V_N^0(x,i) \leq V_f(x,i) \qquad \forall x \in \mathbb{X}_f(i), \quad \forall i \in \mathbb{I}_{\geq 0}$$

The proofs of Propositions 2.34 and 2.35 are left as Exercises 2.9 and 2.10.

Proposition 2.36 (Optimal value function properties; time-varying case)**.** *Suppose Assumptions 2.25, 2.26, and 2.33 are satisfied. Then there exist two $\mathcal{K}_\infty$ functions $\alpha_1(\cdot)$ and $\alpha_2(\cdot)$ such that, for all $i \in \mathbb{I}_{\geq 0}$*

$$\begin{aligned} V_N^0(x,i) &\geq \alpha_1(|x|) && \forall x \in \mathcal{X}_N(i) \\ V_N^0(x,i) &\leq \alpha_2(|x|) && \forall x \in \mathbb{X}_f(i) \\ V_N^0(f(x,\kappa_N(x,i),i+1)) - V_N^0(x,i) &\leq -\alpha_1(|x|) && \forall x \in \mathcal{X}_N(i) \end{aligned}$$

We can deal with the obstacle posed by the fact that the upper bound on $V_N^0(\cdot)$ holds only in $\mathbb{X}_f(i)$ in much the same way as we did previously for the time-invariant case. In general, we invoke the following assumption.

Assumption 2.37 (Uniform weak controllability)**.** There exists a $\mathcal{K}_\infty$ function $\alpha(\cdot)$ such that

$$V_N^0(x,i) \leq \alpha(|x|) \qquad \forall x \in \mathcal{X}_N(i),\ \forall i \in \mathbb{I}_{\geq 0}$$

It can be shown that Assumption 2.37 holds in a variety of other circumstances as described in the following proposition.

Proposition 2.38 (Conditions for uniform weak controllability)**.** *Suppose the functions $f(\cdot)$, $\ell(\cdot)$, and $V_f(\cdot)$ are uniformly bounded for all $i \in \mathbb{I}_{\geq 0}$, i.e., on any compact set $Z \subset \mathbb{R}^n \times \bar{\mathbb{U}}$, the set*

$$\{(f(x,u,i), \ell(x,u,i), V_f(x,i)) \mid (x,u) \in Z, i \in \mathbb{I}_{\geq 0}\}$$

is bounded. Assumption 2.37 is satisfied if any of the following conditions holds:

(a) There exists a neighborhood of the origin X such that $X \subseteq \mathbb{X}_f(i)$ for each $i \in \mathbb{I}_{\geq 0}$

(b) For $i \in \mathbb{I}_{\geq 0}$, the optimal value function $V_N^0(x,i)$ is uniformly continuous in x at $x = 0$

(c) There exists a neighborhood of the origin X and a $\mathcal{K}$ function $\alpha(\cdot)$ such that $V_N^0(x,i) \leq \alpha(|x|)$ for all $i \in \mathbb{I}_{\geq 0}$ and $x \in X \cap \mathcal{X}_N(i)$

(d) The functions $f(\cdot)$ and $\ell(\cdot)$ are uniformly continuous at the origin $(x,u) = (0,0)$ for all $i \in \mathbb{I}_{\geq 0}$, and the system is stabilizable with small inputs, i.e., there exists a $\mathcal{K}_\infty$ function $\gamma(\cdot)$ such that for all $i \in \mathbb{I}_{\geq 0}$ and $x \in \mathcal{X}_N(i)$, there exists $\mathbf{u} \in \mathcal{U}_N(x,i)$ with $|\mathbf{u}| \leq \gamma(|x|)$.

Proof.

(a) Similar to Proposition 2.16, one can show that the optimal cost

$$V_N^0(x,i) \leq V_f(x,i) \leq \alpha_2(|x|) \qquad \text{for all } x \in X$$

Thus, condition (c) is implied.

(b) From uniform continuity, we know that for each $\varepsilon > 0$, there exists $\delta > 0$ such that

$$|x| \leq \delta \quad \text{implies} \quad V_N^0(x,i) \leq \varepsilon \quad \text{for all } i \in \mathbb{I}_{\geq 0}$$

recalling that $V_N^0(\cdot)$ is nonnegative and zero at the origin. By Rawlings and Risbeck (2015, Proposition 13), this is equivalent to the existence of a $\mathcal{K}$ function $\gamma(\cdot)$ defined on $[0, b]$ (with $b > 0$) such that

$$V_N^0(x, i) \leq \gamma(|x|) \quad \text{for all } x \in X$$

with $X := \{x \in \mathbb{R}^n \mid |x| \leq b\}$ a neighborhood of the origin. Thus, condition (c) is also implied.

(c) First, we know that $V_N(\cdot)$ is uniformly bounded because it is the finite sum and composition of the uniformly bounded functions $f(\cdot)$, $\ell(\cdot)$, and $V_f(\cdot)$. Thus, $V_N^0(\cdot)$ is also uniformly bounded, because

$$0 \leq V_N^0(x, i) \leq V_N(x, \mathbf{u}, i) \quad \text{for all } \mathbf{u} \in \mathcal{U}_N(x, i)$$

and $V_N(\cdot)$ is uniformly bounded. Next, because X is a neighborhood of the origin, there exists $b_0 > 0$ such that $V_N^0(x, i) \leq \alpha(|x|)$ whenever $x \in \mathcal{X}_N(i)$ and $|x| \leq b_0$. Following Rawlings and Risbeck (2015, Proposition 14), we choose any strictly increasing and unbounded sequence $(b_k)_{k=0}^{\infty}$ and define

$$B_k(i) := \{x \in \mathcal{X}_N(i) \mid |x| \leq b_k\}$$

We then compute a new sequence $(\beta_k)_{k=0}^{\infty}$ as

$$\beta_k := k + \sup_{\substack{i \in \mathbb{I}_{\geq 0} \\ x \in B_k(i)}} V_N^0(x, i)$$

We know that this sequence is well-defined because $V_N^0(x, i)$ is uniformly bounded for $i \in \mathbb{I}_{\geq 0}$ on $\bigcup_{i \in \mathbb{I}_{\geq 0}} B_k(i)$. We then define

$$\alpha(s) := \begin{cases} \dfrac{\beta_1}{\gamma(b_0)} \gamma(s) & s \in [0, b_0) \\ \beta_{k+1} + (\beta_{k+2} - \beta_{k+1}) \dfrac{s - b_i}{b_{i+1} - b_i} & s \in [b_k, b_{k+1}) \text{ for all } k \in \mathbb{I}_{\geq 0} \end{cases}$$

which is a $\mathcal{K}_\infty$ function that satisfies

$$V_N^0(x, i) \leq \alpha(|x|) \quad \text{for all } i \in \mathbb{I}_{\geq 0}$$

as desired.

(d) See Exercise 2.22. Note that the uniform continuity of $f(\cdot)$ and $\ell(\cdot)$ implies the existence of $\mathcal{K}$ function upper bounds of the form

$$\begin{aligned} |f(x, u, i)| &\leq \alpha_{fx}(|x|) + \alpha_{fu}(|u|) \\ \ell(x, u, i) &\leq \alpha_{\ell x}(|x|) + \alpha_{\ell u}(|u|) \end{aligned}$$

for all $i \in \mathbb{I}_{\geq 0}$. ■

Assumption	Title	Page
2.2	Continuity of system and cost	97
2.3	Properties of constraint sets	98
2.14	Basic stability assumption	114
2.17	Weak controllability	116

Table 2.1: Stability assumptions; time-invariant case.

Hence, if Assumptions 2.25, 2.26, 2.33, and 2.37 hold it follows from Proposition 2.36 that, for all $i \in \mathbb{I}_{\geq 0}$, all $x \in \mathcal{X}_N(i)$

$$\begin{aligned} \alpha_1(|x|) \leq V_N^0(x,i) \leq \alpha_2(|x|) \\ V_N^0(f(x,\kappa_N(x,i),i+1)) - V_N^0(x,i) \leq -\alpha_1(|x|) \end{aligned} \tag{2.21}$$

so that, by Definition 2.31, $V_N^0(\cdot)$ is a time-varying Lyapunov function in the sequence $(X(i))_{i\geq 0}$ for $x^+ = f(x,\kappa_N(x,i),i)$. It can be shown, by a slight extension of the arguments employed in the time-invariant case, that problem $\mathbb{P}_N(\cdot)$ is recursively feasible and that $(\mathcal{X}_N(i))_{i\in\mathbb{I}_{\geq 0}}$ is sequentially positive invariant for the system $x^+ = f(x,\kappa_N(x,i),i)$. The sequence $(\mathcal{X}_N(i))_{i\geq 0}$ in the time-varying case replaces the set $\mathcal{X}_N$ in the time-invariant case.

Theorem 2.39 (Asymptotic stability of the origin: time-varying MPC)**.** *Suppose Assumptions 2.25, 2.26, 2.33, and 2.37 holds. Then,*

(a) There exist $\mathcal{K}_\infty$ functions $\alpha_1(\cdot)$ and $\alpha_2(\cdot)$ such that, for all $i \in \mathbb{I}_{\geq 0}$ and all $x \in \mathcal{X}_{\mathcal{N}}(i)$, inequalities (2.21) *are satisfied.*

(b) The origin is asymptotically stable in $\mathcal{X}_N(i)$ at each time $i \geq 0$ for the time-varying system $x^+ = f(x,\kappa_N(x,i),i)$.

Proof.

(a) It follows from Assumptions 2.25, 2.26, 2.33, and 2.37 and Proposition 2.36 that $V_N^0(\cdot)$ satisfies the inequalities (2.21).

(b) It follows from (a) and definition 2.31 that $V_N^0(\cdot)$ is a time-varying Lyapunov function. It follows from Theorem 2.32 that the origin is asymptotically stable in $\mathcal{X}_N(i)$ at each time $i \geq 0$ for the time-varying system $x^+ = f(x,\kappa_N(x,i),i)$.

■

Assumption	Title	Page
2.25	Continuity of system and cost	124
2.26	Properties of constraint sets	124
2.33	Basic stability assumption	127
2.37	Uniform weak controllability	128

Table 2.2: Stability assumptions; time-varying case.

2.5 Examples of MPC

We already have discussed the general principles underlying the design of stabilizing model predictive controllers. The stabilizing conditions on $\mathbb{X}_f$, $\ell(\cdot)$, and $V_f(\cdot)$ that guarantee stability can be implemented in a variety of ways so that MPC can take many different forms. We present the most useful forms of MPC for applications. These examples also display the roles of the three main assumptions used to guarantee closed-loop asymptotic stability. These assumptions are summarized in Table 2.1 for the time-invariant case, and Table 2.2 for the time-varying case. Referring back to these tables may prove helpful while reading this section and comparing the various forms of MPC.

One question that is often asked is whether or not the terminal constraint is necessary. Since the conditions given previously are sufficient, necessity cannot be claimed. We discuss this further later. It is evident that the constraint arises because one often has a local, rather than a global, control Lyapunov function (CLF) for the system being controlled. In a few situations, a global CLF *is* available, in which case a terminal constraint is not necessary.

All model predictive controllers determine the control action to be applied to the system being controlled by solving, at each state, an optimal control problem that is usually constrained. If the constraints in the optimal control problem include hard state constraints, then the feasible region $\mathcal{X}_N$ is a subset of $\mathbb{R}^n$. The analysis given previously shows that if the initial state $x(0)$ lies in $\mathcal{X}_N$, so do all subsequent states, a property known as *recursive feasibility*. It is always possible, however, for unanticipated events to cause the state to become infeasible. In this case, the optimal control problem, as stated, cannot be solved, and the controller fails. It is therefore desirable, if this does not conflict with design aims, to employ soft state constraints in place of hard constraints. Otherwise, any implementation of the algorithms

described subsequently should be modified to include a feature that enables recovery from faults that cause infeasibility. One remedy is to replace the hard constraints by soft constraints when the current state is infeasible, thereby restoring feasibility, and to revert back to the hard constraints as soon as they can be satisfied at the current state.

In establishing stability of the examples of MPC presented below, we make use of Theorem 2.19 (or Theorem 2.24) for time-invariant systems and Theorem 2.39 for time-varying systems. We must therefore establish that Assumptions 2.2, 2.3, and 2.14 are satisfied in the time-invariant case and that Assumptions 2.25, 2.26, and 2.33 are satisfied in the time-varying case. We normally assume that 2.2, 2.3, and 2.14(b) or 2.25, 2.26, and 2.33(b) are satisfied, so our main task below in each example is establishing satisfaction of the basic stability assumption (cost decrease) 2.14(a) or 2.33(a).

2.5.1 The Unconstrained Linear Quadratic Regulator

Consider the linear, time-invariant model $x^+ = Ax + Bu, y = Cx$ with quadratic penalties on output and state $\ell(y,u) = (1/2)(y'Q_y y + u'Ru)$ for both the finite and infinite horizon cases. We first consider what the assumptions of Theorem 2.24 imply in this case, and compare these assumptions to the standard LQR assumptions (listed in Exercise 1.20(b)).

Assumptions 2.2 is satisfied for $f(x,u) = Ax + Bu$ and $\ell(x,u) = (1/2)(x'C'Q_yCx + u'Ru)$ for all A, B, C, Q_y, R. Assumption 2.3 is satisfied with $\mathbb{Z} = \mathbb{R}^n \times \mathbb{R}^m$ and $R > 0$. Assumption 2.23 implies that $Q_y > 0$ as well. The system being IOSS implies that (A,C) is detectable (see Exercise 4.5). We can choose $\mathbb{X}_f$ to be the stabilizable subspace of (A,B) and Assumption 2.23(a) is satisfied. The set $\mathcal{X}_N$ contains the system controllability information. The set $\mathcal{X}_N$ is the stabilizable subspace of (A,B), and we can satisfy Assumption 2.23(b) by choosing $V_f(x) = (1/2)x'\Pi x$ in which Π is the solution to the steady-state Riccati equation for the stabilizable modes of (A,B).

In particular, if (A,B) is stabilizable, then $V_f(\cdot)$ can be chosen to be $V_f(x) = (1/2)x'\Pi x$ in which Π is the solution to the steady-state Riccati equation (1.18), which is positive definite. The terminal set can be taken as any (arbitrarily large) sublevel set of the terminal penalty, $\mathbb{X}_f = \text{lev}_a V_f$, $a > 0$, so that any point in $\mathbb{R}^n$ is in $\mathbb{X}_f$ for large enough a. We then have $\mathcal{X}_N = \mathbb{R}^n$ for all $N \in \mathbb{I}_{0:\infty}$. The horizon N can be finite or infinite with this choice of $V_f(\cdot)$ and the control law is invariant with

respect to the horizon length, $\kappa_N(x) = Kx$ in which K is the steady-state linear quadratic regulator gain given in (1.18). Theorem 2.24 then gives that the origin of the closed-loop system $x^+ = f(x, \kappa_N(x)) = (A + BK)x$ is globally, asymptotically stable. This can be strengthened to globally, *exponentially* stable because of the choice of quadratic stage cost and form of the resulting Lyapunov function in Theorem 2.24.

The standard assumptions for the LQR with stage cost $l(y, u) = (1/2)(y'Q_y y + u'Ru)$ are

$$Q_y > 0 \quad R > 0 \quad (A, C) \text{ detectable} \quad (A, B) \text{ stabilizable}$$

and we see that LQ theory establishes that the standard steady-state LQR is covered by Theorem 2.24. Summarizing we have

> Given the standard LQR problem, Assumptions 2.2, 2.3, and 2.23 are satisfied and $\mathcal{X}_N = \mathbb{X}_f = \mathbb{R}^n$. It follows from Theorems 2.24 and 2.21 that the origin is globally, exponentially stable for the controlled system $x^+ = Ax + B\kappa_N(x) = (A + BK)x$.

2.5.2 Unconstrained Linear Periodic Systems

In the special case where the system is time varying but periodic, a global CLF can be determined as in the LQR case. Suppose the objective function is

$$\ell(x, u, i) := \frac{1}{2}\left(x'Q(i)x + u'R(i)u\right)$$

with each $Q(i)$ and $R(i)$ positive definite. To start, choose a sequence of linear control laws

$$\kappa_f(x, i) := K(i)x$$

and let

$$A_K(i) := A(i) + B(i)K(i)$$
$$Q_K(i) := Q(i) + K(i)'R(i)K(i)$$

For integers m and n satisfying $m \geq n \geq 0$, let

$$\mathcal{A}(m, n) := A_K(m-1)A_K(m-2)\cdots A_K(n+1)A_K(n)$$

Given these matrices, the closed-loop evolution of the system under the terminal control law is

$$x(m) = \mathcal{A}(m, n)x(n)$$

for $f(x,u,i) = f(x,\kappa_f(u,i),i) = A_K(i)x$.

Suppose the periodic system $(A(i),B(i))$ is stabilizable. It follows that the control laws $K(i)$ can be chosen so that each $\mathcal{A}(i+T,i)$ is stable. Such control laws can be found, e.g., by iterating the periodic discrete algebraic Riccati equation or by solving the Riccati equation for a larger, time-invariant system (see Exercise 2.23).

For a terminal cost, we require matrices $P(i)$ that satisfy

$$A_K(i)'P(i+1)A_K(i) + Q_K(i) = P(i)$$

Summing this relationship for $i \in \mathbb{I}_{0:T-1}$ gives

$$\mathcal{A}(T,0)'P(T)\mathcal{A}(T,0) + \sum_{i=0}^{T-1} \mathcal{A}(k,0)'Q(k)\mathcal{A}(k,0) = P(0)$$

and by periodicity, $P(T) = P(0)$. Noting that $\mathcal{A}(T,0)$ is stable and the summation is positive definite (recall that the first term $|\mathcal{A}(0,0)|^2_{Q(0)} = Q(0)$ is positive definite), there exists a unique solution to this equation, and the remaining $P(i)$ are determined by the recurrence relationship. Thus, taking

$$V_f(x,i) = \frac{1}{2}x'P(i)x$$

we have, for $u = \kappa_f(x,i) = K(i)x$

$$V_f(f(x,u,i),i+1) + \ell(x,u,i) = \frac{1}{2}x'A_K(i)'P(i+1)A_K(i)x + \frac{1}{2}x'Q_K(i)x = \frac{1}{2}x'P(i)x \leq V_f(x,i)$$

as required. The terminal region can then be taken as $\mathbb{X}_f(i) = \mathbb{R}^n$. Summarizing we have

> If the periodic system is stabilizable, there exists a periodic sequence of controller gains and terminal penalties such that $\mathcal{X}_N(i) = \mathbb{X}_f(i) = \mathbb{R}^n$ for all $i \geq 0$. The origin is globally asymptotically stable by Theorem 2.39, which can be strengthened to globally exponentially stable due to the quadratic stage cost. The function $V_f(\cdot,i)$ is a global, time-varying CLF.

2.5.3 Stable Linear Systems with Control Constraints

Usually, when constraints and/or nonlinearities are present, it is impossible to obtain a *global* CLF to serve as the terminal cost function $V_f(\cdot)$. There are, however, a few special cases where this is possible, such as the stable linear system.

The system to be controlled is $x^+ = Ax + Bu$ where A is stable (its eigenvalues lie strictly inside the unit circle) and the control u is subject to the constraint $u \in \mathbb{U}$ where $\mathbb{U}$ is compact and contains the origin in its interior. The stage cost is $\ell(x,u) = (1/2)(x'Qx + u'Ru)$ where Q and R are positive definite. To establish stability of the systems under MPC, we wish to obtain a *global* CLF to serve as the terminal cost function $V_f(\cdot)$. This is usually difficult because any linear control law $u = Kx$, say, transgresses the control constraint for x sufficiently large. In other words, it is usually impossible to find a $V_f(\cdot)$ such that there exists a $u \in \mathbb{U}$ satisfying $V_f(Ax + Bu) \le V_f(x) - \ell(x,u)$ for all x in $\mathbb{R}^n$. Since A is stable, however, it is possible to obtain a Lyapunov function for the autonomous system $x^+ = Ax$ that is a suitable candidate for $V_f(\cdot)$; in fact, for all $Q > 0$, there exists a $P > 0$ such that

$$A'PA + Q = P$$

Let $V_f(\cdot)$ be defined by

$$V_f(x) = (1/2)x'Px$$

With $f(\cdot)$, $\ell(\cdot)$, and $V_f(\cdot)$ defined thus, $\mathbb{P}_N(x)$ is a parametric quadratic problem if the constraint set $\mathbb{U}$ is polyhedral and global solutions may be computed online. The terminal cost function $V_f(\cdot)$ satisfies

$$V_f(Ax) + (1/2)x'Qx - V_f(x) = (1/2)x'(A'PA + Q - P)x = 0$$

for all $x \in \mathbb{X}_f := \mathbb{R}^n$. We see that for all $x \in \mathbb{X}_f$, there exists a u, namely $u = 0$, such that $V_f(Ax + Bu) \le V_f(x) - \ell(x,u)$; $\ell(x,u) = (1/2)x'Qx$ when $u = 0$. Since there are no state or terminal constraints, $\mathcal{X}_N = \mathbb{R}^n$. It follows that there exist positive constants c_1 and c_2 such that

$$c_1 |x|^2 \le V_N^0(x) \le c_2 |x|^2$$
$$V_N^0(f(x, \kappa_N(x))) - V_N^0(x) \le -c_1 |x|^2$$

for all $x \in \mathcal{X}_N = \mathbb{R}^n$. Summarizing, we have

> Assumptions 2.2, 2.3, and 2.14 are satisfied and $\mathcal{X}_N = \mathbb{X}_f = \mathbb{R}^n$. It follows from Theorems 2.19 and 2.21 that the origin

is globally, exponentially stable for the controlled system $x^+ = Ax + B\kappa_N(x)$.

An extension of this approach for unstable A is used in Chapter 6.

2.5.4 Linear Systems with Control and State Constraints

We turn now to the consideration of systems with control and state constraints. In this situation determination of a global CLF is usually difficult if not impossible. Hence we show how local CLFs may be determined together with an invariant region in which they are valid.

The system to be controlled is $x^+ = Ax + Bu$ where A is not necessarily stable, the control u is subject to the constraint $u \in \mathbb{U}$ where $\mathbb{U}$ is compact and contains the origin in its interior, and the state x is subject to the constraint $x \in \mathbb{X}$ where $\mathbb{X}$ is closed and contains the origin in its interior. The stage cost is $\ell(x,u) = (1/2)(x'Qx + u'Ru)$ where Q and R are positive definite. Because of the constraints, it is difficult to obtain a global CLF. Hence we restrict ourselves to the more modest goal of obtaining a local CLF and proceed as follows. If (A,B) is stabilizable, the solution to the infinite horizon *unconstrained* optimal control problem $\mathbb{P}_\infty^{\text{uc}}(x)$ is known. The value function for this problem is $V_\infty^{\text{uc}}(x) = (1/2)x'Px$ where P is the unique (in the class of positive semidefinite matrices) solution to the discrete algebraic Riccati equation

$$P = A_K'PA_K + Q_K$$

in which $A_K := A + BK$, $Q_K := Q + K'RK$, and $u = Kx$, in which K is defined by

$$K := -(B'PB + R)^{-1}B'PA$$

is the optimal controller. The value function $V_\infty^{\text{uc}}(\cdot)$ for the infinite horizon unconstrained optimal control problem $\mathbb{P}_\infty^{\text{uc}}(x)$ satisfies

$$V_\infty^{\text{uc}}(x) = \min_u\{\ell(x,u) + V_\infty^{\text{uc}}(Ax + Bu)\} = \ell(x,Kx) + V_\infty^{\text{uc}}(A_Kx)$$

It is known that P is positive definite. We define the terminal cost $V_f(\cdot)$ by

$$V_f(x) := V_\infty^{\text{uc}}(x) = (1/2)x'Px$$

If $\mathbb{X}$ and $\mathbb{U}$ are polyhedral, problem $\mathbb{P}_N(x)$ is a parametric quadratic program that may be solved online using standard software. The terminal cost function $V_f(\cdot)$ satisfies

$$V_f(A_Kx) + (1/2)x'Q_Kx - V_f(x) \le 0 \ \forall x \in \mathbb{R}^n$$

The controller $u = Kx$ does not necessarily satisfy the control and state constraints, however. The terminal constraint set $\mathbb{X}_f$ must be chosen with this requirement in mind. We may choose $\mathbb{X}_f$ to be the maximal invariant constraint admissible set for $x^+ = A_K x$; this is the largest set W with respect to inclusion[5] satisfying: (a) $W \subseteq \{x \in \mathbb{X} \mid Kx \in \mathbb{U}\}$, and (b) $x \in W$ implies $x(i) = A_K^i x \in W$ for all $i \geq 0$. Thus $\mathbb{X}_f$, defined this way, is control invariant for $x^+ = Ax + Bu$, $u \in \mathbb{U}$. If the initial state x of the system is in $\mathbb{X}_f$, the controller $u = Kx$ maintains the state in $\mathbb{X}_f$ and satisfies the state and control constraints for all future time ($x(i) = A_K^i x \in \mathbb{X}_f \subset \mathbb{X}$ and $u(i) = Kx(i) \in \mathbb{U}$ for all $i \geq 0$). Hence, with $V_f(\cdot)$, $\mathbb{X}_f$, and $\ell(\cdot)$ as defined previously, Assumptions 2.2, 2.3, and 2.14 are satisfied. Summarizing, we have

> Assumptions 2.2, 2.3, and 2.14 are satisfied, and $\mathbb{X}_f$ contains the origin in its interior. Hence, by Theorems 2.19 and 2.21, the origin is exponentially stable in $\mathcal{X}_N$.

It is, of course, not necessary to choose K and $V_f(\cdot)$ as above. Any K such that $A_K = A + BK$ is stable may be chosen, and P may be obtained by solving the Lyapunov equation $A_K' P A_K + Q_K = P$. With $V_f(x) := (1/2)x'Px$ and $\mathbb{X}_f$ the maximal constraint admissible set for $x^+ = A_K x$, the origin may be shown, as above, to be asymptotically stable with a region of attraction $\mathcal{X}_N$ for $x^+ = Ax + B\kappa_N(x)$, and exponentially stable with a region of attraction any sublevel set of $V_N^0(\cdot)$. The optimal control problem is, again, a quadratic program. The terminal set $\mathbb{X}_f$ may be chosen, as above, to be the maximal invariant constraint admissible set for $x^+ = A_K x$, or it may be chosen to be a suitably small sublevel set of $V_f(\cdot)$; by suitably small, we mean small enough to ensure $\mathbb{X}_f \subseteq \mathbb{X}$ and $K\mathbb{X}_f \subseteq \mathbb{U}$. The set $\mathbb{X}_f$, if chosen this way, is ellipsoidal, a subset of the maximal constraint admissible set, and is positive invariant for $x^+ = A_K x$. The disadvantage of this choice is that $\mathbb{P}_N(x)$ is no longer a quadratic program, though it remains a convex program for which software exists.

The choice $V_f(\cdot) = V_\infty^{\mathrm{uc}}(\cdot)$ results in an interesting property of the closed-loop system $x^+ = Ax + B\kappa_N(x)$. Generally, the terminal constraint set $\mathbb{X}_f$ is *not* positive invariant for the controlled system $x^+ = Ax + B\kappa_N(x)$. Thus, in solving $\mathbb{P}_N(x)$ for an initial state $x \in \mathbb{X}_f$, the "predicted" state sequence $\mathbf{x}^0(x) = (x^0(0;x), x^0(1;x), \ldots, x^0(N;x))$ starts and ends in $\mathbb{X}_f$ but does not necessarily remain in $\mathbb{X}_f$. Thus

[5] $W \in \mathcal{W}$ is the largest set in $\mathcal{W}$ with respect to inclusion if $W' \subseteq W$ for any $W' \in \mathcal{W}$.

$x^0(0;x) = x \in \mathbb{X}_f$ and $x^0(N;x) \in \mathbb{X}_f$, because of the terminal constraint in the optimal control problem, but, for any $i \in \mathbb{I}_{1:N-1}$, $x^0(i;x)$ may lie outside of $\mathbb{X}_f$. In particular, $x^+ = Ax + B\kappa_N(x) = x^0(1;x)$ may lie outside of $\mathbb{X}_f$; $\mathbb{X}_f$ is *not* necessarily positive invariant for the controlled system $x^+ = Ax + B\kappa_N(x)$.

Consider now the problem $\mathbb{P}_N^{\mathrm{uc}}(x)$ defined in the same way as $\mathbb{P}_N(x)$ except that *all* constraints are omitted so that $\mathcal{U}_N(x) = \mathbb{R}^{Nm}$

$$\mathbb{P}_N^{\mathrm{uc}}(x): \quad V_N^{\mathrm{uc}}(x) = \min_{\mathbf{u}} V_N(x, \mathbf{u})$$

in which $V_N(\cdot)$ is defined as previously by

$$V_N(x, \mathbf{u}) := \sum_{i=0}^{N-1} \ell(x(i), u(i)) + V_f(x(N))$$

with $V_f(\cdot)$ the value function for the infinite horizon unconstrained optimal control problem, i.e., $V_f(x) := V_\infty^{\mathrm{uc}}(x) = (1/2)x'Px$. With these definitions, it follows that

$$V_N^{\mathrm{uc}}(x) = V_\infty^{\mathrm{uc}}(x) = V_f(x) = (1/2)x'Px$$
$$\kappa_N^{\mathrm{uc}}(x) = Kx, \quad K = -(B'PB + R)^{-1}B'PA$$

for all $x \in \mathbb{R}^n$; $u = Kx$ is the optimal controller for the unconstrained infinite horizon problem. But $\mathbb{X}_f$ *is* positive invariant for $x^+ = A_K x$.

We now claim that with $V_f(\cdot)$ chosen to equal to $V_\infty^{\mathrm{uc}}(\cdot)$, the terminal constraint set $\mathbb{X}_f$ *is* positive invariant for $x^+ = Ax + B\kappa_N(x)$. We do this by showing that $V_N^0(x) = V_N^{\mathrm{uc}}(x) = V_\infty^{\mathrm{uc}}(x)$ for all $x \in \mathbb{X}_f$, so that the associated control laws are the same, i.e., $\kappa_N(x) = Kx$. First, because $\mathbb{P}_N^{\mathrm{uc}}(x)$ is identical with $\mathbb{P}_N(x)$ except for the absence of all constraints, we have

$$V_N^{\mathrm{uc}}(x) = V_f(x) \le V_N^0(x) \quad \forall x \in \mathcal{X}_N \supseteq \mathbb{X}_f$$

Second, from Lemma 2.18

$$V_N^0(x) \le V_f(x) \quad \forall x \in \mathbb{X}_f$$

Hence $V_N^0(x^+) = V_N^{\mathrm{uc}}(x) = V_f(x)$ for all $x \in \mathbb{X}_f$. That $\kappa_N(x) = Kx$ for all $x \in \mathbb{X}_f$ follows from the uniqueness of the solutions to the problems $\mathbb{P}_N(x)$ and $\mathbb{P}_N^{\mathrm{uc}}(x)$. Summarizing, we have

> If $V_f(\cdot)$ is chosen to be the value function for the unconstrained infinite horizon optimal control problem, if $u = Kx$ is the associated controller, and if $\mathbb{X}_f$ is invariant for

$x^+ = A_K x$, then $\mathbb{X}_f$ is also positive invariant for the controlled system $x^+ = Ax + B\kappa_N(x)$. Also $\kappa_N(x) = Kx$ for all $x \in \mathbb{X}_f$.

2.5.5 Constrained Nonlinear Systems

The system to be controlled is

$$x^+ = f(x, u)$$

in which $f(\cdot)$ is assumed to be twice continuously differentiable. The system is subject to state and control constraints

$$x \in \mathbb{X} \qquad u \in \mathbb{U}$$

in which $\mathbb{X}$ is closed and $\mathbb{U}$ is compact; each set contains the origin in its interior. The cost function is defined by

$$V_N(x, \mathbf{u}) = \sum_{i=0}^{N-1} \ell(x(i), u(i)) + V_f(x(N))$$

in which, for each i, $x(i) := \phi(i; x, \mathbf{u})$, the solution of $x^+ = f(x, u)$ at time i if the initial state is x at time zero and the control is $\mathbf{u}$. The stage cost $\ell(\cdot)$ is defined by

$$\ell(x, u) := (1/2)(|x|_Q^2 + |u|_R^2)$$

in which Q and R are positive definite. The optimal control problem $\mathbb{P}_N(x)$ is defined by

$$\mathbb{P}_N(x): \quad V_N^0(x) = \min_{\mathbf{u}} \{V_N(x, \mathbf{u}) \mid \mathbf{u} \in \mathcal{U}_N(x)\}$$

in which $\mathcal{U}_N(x)$ is defined by (2.5) and includes the terminal constraint $x(N) = \phi(N; x, \mathbf{u}) \in \mathbb{X}_f$ (in addition to the state and control constraints).

Our first task is to choose the ingredients $V_f(\cdot)$ and $\mathbb{X}_f$ of the optimal control problem to ensure asymptotic stability of the origin for the controlled system. We obtain a terminal cost function $V_f(\cdot)$ and a terminal constraint set $\mathbb{X}_f$ by linearization of the nonlinear system $x^+ = f(x, u)$ at the origin. Hence we assume $f(\cdot)$ and $\ell(\cdot)$ are twice continuously differentiable so that Assumption 2.2 is satisfied . Suppose then that the linearized system is

$$x^+ = Ax + Bu$$

where $A := f_x(0,0)$ and $B := f_u(0,0)$. We assume that (A,B) is stabilizable and we choose any controller $u = Kx$ such that the origin is globally exponentially stable for the system $x^+ = A_K x$, $A_K := A + BK$, i.e., such that A_K is stable. Suppose also that the stage cost $\ell(\cdot)$ is defined by $\ell(x,u) := (1/2)(|x|_Q^2 + |u|_R^2)$ where Q and R are positive definite; hence $\ell(x,Kx) = (1/2)x'Q_K x$ where $Q_K := (Q + K'RK)$. Let P be defined by the Lyapunov equation

$$A_K' P A_K + \mu Q_K = P$$

for some $\mu > 1$ The reason for the factor μ will become apparent soon. Since Q_K is positive definite and A_K is stable, P is positive definite. Let the terminal cost function $V_f(\cdot)$ be defined by

$$V_f(x) := (1/2)x'Px$$

Clearly $V_f(\cdot)$ is a global CLF for the linear system $x^+ = Ax+Bu$. Indeed, it follows from its definition that $V_f(\cdot)$ satisfies

$$V_f(A_K x) + (\mu/2)x'Q_K x - V_f(x) = 0 \quad \forall x \in \mathbb{R}^n \tag{2.22}$$

Consider now the nonlinear system $x^+ = f(x,u)$ with linear control $u = Kx$. The controlled system satisfies

$$x^+ = f(x,Kx)$$

We wish to show that $V_f(\cdot)$ is a local CLF for $x^+ = f(x,u)$ in some neighborhood of the origin; specifically, we wish to show there exists an $a \in (0,\infty)$ such that

$$V_f(f(x,Kx)) + (1/2)x'Q_K x - V_f(x) \le 0 \quad \forall x \in \text{lev}_a V_f \tag{2.23}$$

in which, for all $a > 0$, $\text{lev}_a V_f := \{x \mid V_f(x) \le a\}$ is a sublevel set of V_f. Since P is positive definite, $\text{lev}_a V_f$ is an ellipsoid with the origin as its center. Comparing inequality (2.23) with (2.22), we see that (2.23) is satisfied if

$$V_f(f(x,Kx)) - V_f(A_K x) \le ((\mu-1)/2)x'Q_k x \quad \forall x \in \text{lev}_a V_f \tag{2.24}$$

Let $e(\cdot)$ be defined as follows

$$e(x) := f(x,Kx) - A_K x$$

so that

$$V_f(f(x,Kx)) - V_f(A_K x) = (A_K x)' P e(x) + (1/2) e(x)' P e(x) \quad (2.25)$$

By definition, $e(0) = f(0,0) - A_K 0 = 0$ and $e_x(x) = f_x(x,Kx) + f_u(x, Kx)K - A_K$. It follows that $e_x(0) = 0$. Since $f(\cdot)$ is twice continuously differentiable, for any $\delta > 0$, there exists a $c_\delta > 0$ such that $|e_{xx}(x)| \le c_\delta$ for all x in $\delta\mathcal{B}$. From Proposition A.11 in Appendix A

$$\begin{aligned} |e(x)| &= \left| e(0) + e_x(0)x + \int_0^1 (1-s)x' e_{xx}(sx)x\,ds \right| \\ &\le \int_0^1 (1-s)c_\delta |x|^2 \, ds \le (1/2)c_\delta |x|^2 \end{aligned}$$

for all x in $\delta\mathcal{B}$. From (2.25), we see that there exists an $\varepsilon \in (0, \delta]$ such that (2.24), and, hence, (2.23), is satisfied for all $x \in \varepsilon\mathcal{B}$. Because of our choice of $\ell(\cdot)$, there exists a $c_1 > 0$ such that $V_f(x) \ge \ell(x,Kx) \ge c_1 |x|^2$ for all $x \in \mathbb{R}^n$. It follows that $x \in \text{lev}_a V_f$ implies $|x| \le \sqrt{a/c_1}$. We can choose a to satisfy $\sqrt{a/c_1} = \varepsilon$. With this choice, $x \in \text{lev}_a V_f$ implies $|x| \le \varepsilon \le \delta$, which, in turn, implies (2.23) is satisfied.

We conclude that there exists an $a > 0$ such that $V_f(\cdot)$ and $\mathbb{X}_f := \text{lev}_a V_f$ satisfy Assumptions 2.2 and 2.3. For each $x \in \mathbb{X}_f$ there exists a $u = \kappa_f(x) := Kx$ such that $V_f(x,u) \le V_f(x) - \ell(x,u)$ since $\ell(x,Kx) = (1/2)x' Q_K x$ so that our assumption that $\ell(x,u) = (1/2)(x'Qx + u'Ru)$ where Q and R are positive definite, and our definition of $V_f(\cdot)$ ensure the existence of positive constants c_1, c_2 and c_3 such that $V_N^0(x) \ge c_1 |x|^2$ for all $\mathbb{R}^n$, $V_f(x) \le c_2 |x|^2$ and $V_N^0(f(x,\kappa_f(x))) \le V_N^0(x) - c_3 |x|^2$ for all $x \in \mathbb{X}_f$ thereby satisfying Assumption 2.14. Finally, by definition, the set $\mathbb{X}_f$ contains the origin in its interior. Summarizing, we have

> Assumptions 2.2, 2.3, and 2.14 are satisfied, and $\mathbb{X}_f$ contains the origin in its interior. In addition $\alpha_1(\cdot)$, $\alpha_2(\cdot)$, and $\alpha_3(\cdot)$ satisfy the hypotheses of Theorem 2.21. Hence, by Theorems 2.19 and 2.21, the origin is exponentially stable for $x^+ = f(x, \kappa_N(x))$ in $\mathcal{X}_N$.

Asymptotic stability of the origin in $\mathcal{X}_N$ also may be established when $\mathbb{X}_f := \{0\}$ by assuming a $\mathcal{K}_\infty$ bound on $V_N^0(\cdot)$ as in Assumption 2.17.

2.5.6 Constrained Nonlinear Time-Varying Systems

Although Assumption 2.33 (the basic stability assumption) for the time-varying case suffices to ensure that $V_N^0(\cdot)$ has sufficient cost decrease,

it can be asked if there exist a $V_f(\cdot)$ and $\mathbb{X}_f$ satisfying the hypotheses of this assumption, as well as Assumption 2.37. We give a few examples below.

Terminal equality constraint. Consider a linear time-varying system described by $x^+ = f(x(i), u(i), i) = A(i)x(i) + B(i)u(i)$ with $\ell(x, u, i) = (1/2)(x'Q(i)x(i) + u'R(i)u)$. Clearly $(\bar{x}, \bar{u}) = (0, 0)$ is an equilibrium pair since $f(0, 0, i) = 0$ for all $i \in \mathbb{I}_{\geq 0}$. The terminal constraint set is $\mathbb{X}_f(i) = \{0\}$ for all $i \in \mathbb{I}_{\geq 0}$, and the cost can be taken as $V_f(x, i) \equiv 0$. Assumption 2.33(a) is clearly satisfied. If, in addition, the matrices $A(i)$, $B(i)$, $Q(i)$, and $R(i)$ can be uniformly bounded from above, and the system is stabilizable (with $\mathbb{U}$ containing a neighborhood of the origin), then the weak controllability hypothesis implies that Assumption 2.37 is satisfied as well.

If $f(\cdot)$ is nonlinear, assumption 2.33(a) is satisfied if $f(0, 0, i) = 0$ for all $i \in \mathbb{I}_{\geq 0}$. Verifying Assumption 2.37 requires more work in the nonlinear case, but weak controllability is often the easiest way. In summary we have

> Given the terminal equality constraint and Assumption 2.37, Theorem 2.39 applies and the origin is asymptotically stable in $\mathcal{X}_N(i)$ at each time $i \geq 0$ for the time-varying system $x^+ = f(x, \kappa_N(x, i), i)$.

Periodic target tracking. If the target is a periodic reference signal and the system is periodic with period T as in Limon, Alamo, de la Peña, Zeilinger, Jones, and Pereira (2012), Falugi and Mayne (2013b), and Rawlings and Risbeck (2017), it is possible, under certain conditions, to obtain terminal ingredients that satisfy Assumptions 2.33(a) and 2.37.

In the general case, terminal region synthesis is challenging. But given sufficient smoothness in the system model, we can proceed as follows. First we subtract the periodic state and input references and work in deviation variables so that the origin is again the target. Assuming $f(\cdot)$ is twice continuously differentiable in x and u at $(0, 0, i)$, we can linearize the system to determine

$$A(i) := \frac{\partial f}{\partial x}(0, 0, i) \qquad B(i) := \frac{\partial f}{\partial u}(0, 0, i)$$

Assuming the origin is in the interior of each $\mathbb{X}(i)$ (but not necessarily each $\mathbb{U}(i)$), we determine a subspace of unsaturated inputs $\widetilde{u}$ such that (i) $u(i) = F(i)\widetilde{u}(i)$, (ii) there exists $\epsilon > 0$ such that $F(i)\widetilde{u}(i) \in \mathbb{U}(i)$

for all $|\widetilde{u}| \leq \epsilon$, and (iii) the reduced linear system $(A(i), B(i)F(i))$ is stabilizable. These conditions ensure that the reduced linear system is locally unconstrained. Taking a positive definite stage cost

$$\ell(x, u, i) := \frac{1}{2}\left(x'Q(i)x + u'R(i)u\right)$$

we chose $\mu > 1$ and proceed as in the linear unconstrained case (Section 2.5.2) using the reduced model $(A(i), B(i)F(i))$ and adjusted cost matrices $\mu Q(i)$ and $\mu R(i)$. We thus have the relationship

$$V_f(A(i)x + B(i)u, i+1) \leq V_f(x, i) - \mu\ell(x, u, i)$$

with $u = \kappa_f(x, i) := K(i)x$ and $V_f(x, i) := (1/2)x'P(i)x$. Two issues remain: first, it is unlikely that $K(i)x \in \mathbb{U}(i)$ for all x and i; and second, the cost decrease holds only for the (approximate) linearized system.

To address the first issue, we start by defining the set

$$X(i) := \{x \in \mathbb{X}(i) \mid \kappa_f(x, i) \in \mathbb{U}(i) \text{ and } f(x, \kappa_f(u, i), i) \in \mathbb{X}(i+1)\}$$

on which $\kappa_f(\cdot)$ is valid. We require $\mathbb{X}_f(i) \subseteq X(i)$ for all $i \in \mathbb{I}_{\geq 0}$. By assumption, $X(i)$ contains a neighborhood of the origin, and so we can determine constants $a(i) > 0$ sufficiently small such that

$$\operatorname{lev}_{a(i)} V_f(\cdot, i) \subseteq X(i) \quad i \geq 0$$

For the second issue, we can appeal to Taylor's theorem as in Section 2.5.5 to find constants $b(i) \in (0, a(i)]$ such that

$$V_f(f(x, u, i), i+1) - V_f(A(i)x + B(i)u, i+1) \leq (\mu - 1)\ell(x, u, i)$$

for all $x \in \operatorname{lev}_{b(i)} V_f(\cdot, i)$ and $i \in \mathbb{I}_{\geq 0}$. That is, the approximation error of the linear system is sufficiently small. Thus, adding this inequality to the approximate cost decrease condition, we recover

$$V_f(f(x, u, i), i+1) - V_f(x, i) \leq -\ell(x, u, i)$$

on terminal regions $\mathbb{X}_f(i) = \operatorname{lev}_{b(i)} V_f(\cdot, i)$. That these terminal regions are positive invariant follows from the cost decrease condition. Note also that these sets $\mathbb{X}_f(i)$ contain the origin in their interiors, and thus Assumption 2.37 is satisfied. Summarizing we have

> Given sufficient smoothness in $f(x, u, i)$, terminal region synthesis can be accomplished for tracking a periodic reference. Then the assumptions of Theorem 2.39 are satisfied,

and the origin (in deviation variables; hence, the periodic reference in the original variables) is asymptotically stable in $\mathcal{X}_N(i)$ at each time $i \geq 0$ for the time-varying system $x+ = f(x, \kappa_N(x, i), i)$.

2.6 Is a Terminal Constraint Set $\mathbb{X}_f$ Necessary?

While addition of a terminal cost $V_f(\cdot)$ does not materially affect the optimal control problem, addition of a terminal constraint $x(N) \in \mathbb{X}_f$, which is a state constraint, may have a significant effect. In particular, problems with only control constraints are usually easier to solve. So if state constraints are not present or if they are handled by penalty functions (soft constraints), it is highly desirable to avoid the addition of a terminal constraint. Moreover, it is possible to establish continuity of the value function for a range of optimal control problems *if* there are no state constraints; continuity of the value function ensures a degree of robustness (see Chapter 3). It is therefore natural to ask if the terminal constraint can be omitted without affecting stability.

A possible procedure is merely to omit the terminal constraint and to require that the initial state lies in a subset of $\mathcal{X}_N$ that is sufficiently small. We examine this alternative here and assume that $V_f(\cdot)$, $\mathbb{X}_f$ and $\ell(\cdot)$ satisfy Assumptions 2.2, 2.3, and 2.14, and that $\mathbb{X}_f := \{x \mid V_f(x) \leq a\}$ for some $a > 0$.

We assume, as in the examples of MPC discussed in Section 2.5, that the terminal cost function $V_f(\cdot)$, the constraint set $\mathbb{X}_f$, and the stage cost $\ell(\cdot)$ for the optimal control problem $\mathbb{P}_N(x)$ are chosen to satisfy Assumptions 2.2, 2.3, and 2.14 so that there exists a local control law $\kappa_f : \mathbb{X}_f \to \mathbb{U}$ such that $\mathbb{X}_f \subseteq \{x \in \mathbb{X} \mid \kappa_f(x) \in \mathbb{U}\}$ is positive invariant for $x^+ = f(x, \kappa_f(x))$ and $V_f(f(x, \kappa_f(x))) + \ell(x, \kappa_f(x)) \leq V_f(x)$ for all $x \in \mathbb{X}_f$. We assume that the function $V_f(\cdot)$ is defined on $\mathbb{X}$ even though it possesses the property $V_f(f(x, \kappa_f(x))) + \ell(x, \kappa_f(x)) \leq V_f(x)$ only in $\mathbb{X}_f$. In many cases, even if the system being controlled is nonlinear, $V_f(\cdot)$ is quadratic and positive definite, and $\kappa_f(\cdot)$ is linear. The set $\mathbb{X}_f$ may be chosen to be a sublevel set of $V_f(\cdot)$ so that $\mathbb{X}_f = W(a) := \{x \mid V_f(x) \leq a\}$ for some $a > 0$. We discuss in the sequel a modified form of the optimal control problem $\mathbb{P}_N(x)$ in which the terminal cost $V_f(\cdot)$ is replaced by $\beta V_f(\cdot)$ and the terminal constraint $\mathbb{X}_f$ is omitted, and show that if β is sufficiently large the solution of the modified optimal control problem is such that the optimal terminal state nevertheless lies in $\mathbb{X}_f$ so that terminal constraint is implicitly satisfied.

For all $\beta \geq 1$, let $\mathbb{P}_N^\beta(x)$ denote the modified optimal control problem defined by

$$\hat{V}_N^\beta(x) = \min_{\mathbf{u}} \{V_N^\beta(x, \mathbf{u}) \mid \mathbf{u} \in \hat{\mathcal{U}}_N(x)\}$$

in which the cost function to be minimized is now

$$V_N^\beta(x, \mathbf{u}) := \sum_{i=0}^{N-1} \ell(x(i), u(i)) + \beta V_f(x(N))$$

in which, for all i, $x(i) = \phi(i; x, \mathbf{u})$, the solution at time i of $x^+ = f(x, u)$ when the initial state is x and the control sequence is $\mathbf{u}$. The control constraint set $\hat{\mathcal{U}}_N(x)$ ensures satisfaction of the state and control constraints, but not the terminal constraint, and is defined by

$$\hat{\mathcal{U}}_N(x) := \{\mathbf{u} \mid (x(i), u(i)) \in \mathbb{Z}, i \in \mathbb{I}_{0:N-1}, x(N) \in \mathbb{X}\}$$

The cost function $V_N^\beta(\cdot)$ with $\beta = 1$ is identical to the cost function $V_N(\cdot)$ employed in the standard problem $\mathbb{P}_N$ considered previously. Let $\hat{\mathcal{X}}_N := \{x \in \mathbb{X} \mid \hat{\mathcal{U}}_N(x) \neq \emptyset\}$ denote the domain of $\hat{V}_N^\beta(\cdot)$; let $\mathbf{u}^\beta(x)$ denote the solution of $\mathbb{P}_N^\beta(x)$; and let $\mathbf{x}^\beta(x)$ denote the associated optimal state trajectory. Thus

$$\mathbf{u}^\beta(x) = \left(u^\beta(0; x), u^\beta(1; x), \ldots, u^\beta(N-1; x)\right)$$
$$\mathbf{x}^\beta(x) = \left(x^\beta(0; x), x^\beta(1; x), \ldots, x^\beta(N; x)\right)$$

where $x^\beta(i; x) := \phi(i; x, \mathbf{u}^\beta(x))$ for all i. The implicit MPC control law is $\kappa_N^\beta(\cdot)$ where $\kappa_N^\beta(x) := u^\beta(0; x)$. Neither $\hat{\mathcal{U}}_N(x)$ nor $\hat{\mathcal{X}}_N$ depend on the parameter β. It can be shown (Exercise 2.11) that *the pair* $(\beta V_f(\cdot), \mathbb{X}_f)$ *satisfies Assumptions 2.2–2.14* if $\beta \geq 1$, since these assumptions are satisfied by the pair $(V_f(\cdot), \mathbb{X}_f)$. The absence of the terminal constraint $x(N) \in \mathbb{X}_f$ in problem $\mathbb{P}_N^\beta(x)$, which is otherwise the same as the normal optimal control problem $\mathbb{P}_N(x)$ when $\beta = 1$, ensures that $\hat{V}_N^1(x) \leq V_N^0(x)$ for all $x \in \mathcal{X}_N$ and that $\mathcal{X}_N \subseteq \hat{\mathcal{X}}_N$ where $V_N^0(\cdot)$ is the value function for $\mathbb{P}_N(x)$ and $\mathcal{X}_N$ is the domain of $V_N^0(\cdot)$.

Problem $\mathbb{P}_N^\beta(x)$ and the associated MPC control law $\kappa_N^\beta(\cdot)$ are defined below. Suppose $\mathbf{u}^\beta(x)$ is optimal for the terminally unconstrained problem $\mathbb{P}_N^\beta(x)$, $\beta \geq 1$, and that $\mathbf{x}^\beta(x)$ is the associated optimal state trajectory.

That the origin is asymptotically stable for $x^+ = f(x, \kappa_N^\beta(x))$ and each $\beta \geq 1$, with a region of attraction that depends on the parameter β is established by Limon, Alamo, Salas, and Camacho (2006) via the following results.

Lemma 2.40 (Entering the terminal region)**.** *Suppose* $\mathbf{u}^\beta(x)$ *is optimal for the terminally unconstrained problem* $\mathbb{P}_N^\beta(x)$*, with* $\beta \geq 1$*, and that* $\mathbf{x}^\beta(x)$ *is the associated optimal state trajectory. If* $x^\beta(N;x) \notin \mathbb{X}_f$*, then* $x^\beta(i;x) \notin \mathbb{X}_f$ *for all* $i \in \mathbb{I}_{0:N-1}$*.*

Proof. Since, as shown in Exercise 2.11, $\beta V_f(x) \geq \beta V_f(f(x,\kappa_f(x))) + \ell(x,\kappa_f(x))$ and $f(x,\kappa_f(x)) \in \mathbb{X}_f$ for all $x \in \mathbb{X}_f$, all $\beta \geq 1$, it follows that for all $x \in \mathbb{X}_f$ and all $i \in \mathbb{I}_{0:N-1}$

$$\beta V_f(x) \geq \sum_{j=i}^{N-1} \ell(x^f(j;x,i), u^f(j;x,i)) + \beta V_f(x^f(N;x,i)) \geq \hat{V}_{N-i}^\beta(x)$$

in which $x^f(j;x,i)$ is the solution of $x^+ = f(x,\kappa_f(x))$ at time j if the initial state is x at time i, $u^f(j;x,i) = \kappa_f(x^f(j;x,i))$, and $\kappa_f(\cdot)$ is the local control law that satisfies the stability assumptions. The second inequality follows from the fact that the control sequence $\left(u^f(j;x,i)\right)$, $j \in \mathbb{I}_{i:N-1}$ is feasible for $\mathbb{P}_N^\beta(x)$ if $x \in \mathbb{X}_f$. Suppose contrary to what is to be proved, that there exists a $i \in \mathbb{I}_{0:N-1}$ such that $x^\beta(i;x) \in \mathbb{X}_f$. By the principle of optimality, the control sequence $(u^\beta(i;x), u^\beta(i+1;x), \ldots, u^\beta(N-1;x))$ is optimal for $\mathbb{P}_{N-i}^\beta(x^\beta(i;x))$. Hence

$$\beta V_f(x^\beta(i;x)) \geq \hat{V}_{N-i}^\beta(x^\beta(i;x)) \geq \beta V_f(x^\beta(N;x)) > \beta a$$

since $x^\beta(N;x) \notin \mathbb{X}_f$ contradicting the fact that $x^\beta(i;x) \in \mathbb{X}_f$. This proves the lemma. ∎

For all $\beta \geq 1$, let the set Γ_N^β be defined by

$$\Gamma_N^\beta := \{x \mid \hat{V}_N^\beta(x) \leq Nd + \beta a\}$$

We assume in the sequel that there exists a $d > 0$ such $\ell(x,u) \geq d$ for all $x \in \mathbb{X} \setminus \mathbb{X}_f$ and all $u \in \mathbb{U}$. The following result is due to Limon et al. (2006).

Theorem 2.41 (MPC stability; no terminal constraint)**.** *The origin is asymptotically or exponentially stable for the closed-loop system* $x^+ = f(x,\kappa_N^\beta(x))$ *with a region of attraction* Γ_N^β*. The set* Γ_N^β *is positive invariant for* $x^+ = f(x,\kappa_N^\beta(x))$*.*

Proof. From the Lemma, $x^\beta(N;x) \notin \mathbb{X}_f$ implies $x^\beta(i;x) \notin \mathbb{X}_f$ for all $i \in \mathbb{I}_{0:N}$. This, in turn, implies

$$\hat{V}_N^\beta(x) > Nd + \beta a$$

so that $x \notin \Gamma_N^\beta$. Hence $x \in \Gamma_N^\beta$ implies $x^\beta(N;x) \in \mathbb{X}_f$. It then follows, since $\beta V_f(\cdot)$ and $\mathbb{X}_f$ satisfy Assumptions 2.2 and 2.3, that the origin is asymptotically or exponentially stable for $x^+ = f(x, \kappa_N^\beta(x))$ with a region of attraction Γ_N^β. It also follows that $x \in \Gamma_N^\beta(x)$ implies

$$\hat{V}_N^\beta(x^\beta(1;x)) \le \hat{V}_N^\beta(x) - \ell(x, \kappa_N^\beta(x)) \le \hat{V}_N^\beta(x) \le Nd + \beta a$$

so that $x^\beta(1;x) = f(x, \kappa_N^\beta(x)) \in \Gamma_N^\beta$. Hence Γ_N^β is positive invariant for $x^+ = f(x, \kappa_N^\beta(x))$. ■

Limon et al. (2006) then proceed to show that Γ_N^β increases with β or, more precisely, that $\beta_1 \le \beta_2$ implies that $\Gamma_N^{\beta_1} \subseteq \Gamma_N^{\beta_2}$. They also show that for any x steerable to the interior of $\mathbb{X}_f$ by a feasible control, there exists a β such that $x \in \Gamma_N^\beta$. We refer to requiring the initial state x to lie in Γ_N^β as an *implicit terminal constraint*.

If it is desired that the feasible sets for $\mathbb{P}_i(x)$ be nested ($\mathcal{X}_i \subset \mathcal{X}_{i+1}$, $i = 1, 2, \ldots N-1$) (thereby ensuring recursive feasibility), it is *necessary*, as shown in Mayne (2013), that $\mathbb{P}_N(x)$ includes a terminal constraint that is control invariant.

2.7 Suboptimal MPC

Overview. There is a significant practical problem that we have not yet addressed, namely that if the optimal control problem $\mathbb{P}_N(x)$ solved online is not convex, which is usually the case when the system is nonlinear, the global minimum of $V_N(x, \mathbf{u})$ in $\mathcal{U}_N(x)$ cannot usually be determined. Since we assume, in the stability theory given previously, that the global minimum *is* achieved, we have to consider the impact of this unpalatable fact. It is possible, as shown in Scokaert, Mayne, and Rawlings (1999); Pannocchia, Rawlings, and Wright (2011) to achieve stability *without* requiring globally optimal solutions of $\mathbb{P}_N(x)$. The basic idea behind the suboptimal model predictive controller is simple. Suppose the current state is x and that $\mathbf{u} = (u(0), u(1), \ldots, u(N-1)) \in \mathcal{U}_N(x)$ is a feasible control sequence for $\mathbb{P}_N(x)$. The first element $u(0)$ of $\mathbf{u}$ is applied to the system $x^+ = f(x, u)$; let $\kappa_N(x, \mathbf{u})$ denote this control. In the absence of uncertainty, the next state is equal to the predicted state $x^+ = f(x, u(0))$.

Consider the control sequence $\tilde{\mathbf{u}}$ defined by

$$\tilde{\mathbf{u}} = \big(u(1), u(2), \ldots, u(N-1), \kappa_f(x(N))\big) \tag{2.26}$$

in which $x(N) = \phi(N; x, \mathbf{u})$ and $\kappa_f(\cdot)$ is a local control law with the property that $u = \kappa_f(x)$ satisfies Assumption 2.2 for all $x \in \mathbb{X}_f$. The existence of such a $\kappa_f(\cdot)$, which is often of the form $\kappa_f(x) = Kx$, is implied by Assumption 2.2. Then, since $x(N) \in \mathbb{X}_f$ and since the stabilizing conditions 2.14 are satisfied, the control sequence $\widetilde{\mathbf{u}} \in \mathcal{U}_N(x)$ satisfies

$$V_N(x^+, \widetilde{\mathbf{u}}) \le V_N(x, \mathbf{u}) - \ell(x, u(0)) \le V_N(x, \mathbf{u}) - \alpha_1(|x|) \tag{2.27}$$

with $x^+ := f(x, u(0))$.

No optimization is required to get the cost reduction $\ell(x, u(0))$ given by (2.27); in practice the control sequence $\widetilde{\mathbf{u}}$ can be improved by several iterations of an optimization algorithm. Inequality (2.27) is reminiscent of the inequality $V_N^0(x^+) \le V_N^0(x) - \alpha_1(|x|)$ that provides the basis for establishing asymptotic stability of the origin for the controlled systems previously analyzed. This suggests that the simple algorithm described previously, which places very low demands on the online optimization algorithm, may also ensure asymptotic stability of the origin.

This is almost true. The obstacle to applying standard Lyapunov theory is that there is no obvious Lyapunov function $V : \mathbb{R}^n \to \mathbb{R}_{\ge 0}$ because, at each state x^+, there exist many control sequences $\mathbf{u}^+$ satisfying $V_N(x^+, \mathbf{u}^+) \le V_N(x, \mathbf{u}) - \alpha_1(|x|)$. The function $(x, \mathbf{u}) \mapsto V_N(x, \mathbf{u})$ is *not* a function of x only and may have many different values for each x; therefore it cannot play the role of the function $V_N^0(x)$ used previously. Moreover, the controller can generate, for a given initial state, many different trajectories, all of which have to be considered. We address these issues next following the recent development in Allan, Bates, Risbeck, and Rawlings (2017).

A key step is to consider suboptimal MPC as an evolution of an *extended state* consisting of the state and warm-start pair. Given a feasible warm start, optimization algorithms can produce an improved feasible sequence or, failing even that, simply return the warm start. The first input is injected and a new warm start can generated from the returned control sequence and terminal control law.

Warm start. An admissible warm start $\widetilde{\mathbf{u}}$, must steer the current state x to the terminal region subject to the input constraints, i.e., $\widetilde{\mathbf{u}} \in \mathcal{U}_N(x)$. It also must satisfy $V_N(x, \widetilde{\mathbf{u}}) \le V_f(x)$ if $x \in \mathbb{X}_f$, which ensures that $|x| \to 0$ implies $|\mathbf{u}| \to 0$. These two conditions define the set of admissible warm starts

$$\widetilde{\mathcal{U}}_N(x) := \{\widetilde{\mathbf{u}} \in \mathcal{U}_N(x) \mid V_N(x, \widetilde{\mathbf{u}}) \le V_f(x) \text{ if } x \in \mathbb{X}_f\} \tag{2.28}$$

When $x \in \mathbb{X}_f$ and $\tilde{\mathbf{u}} \in \mathcal{U}_N(x)$ but $V_N(x, \tilde{\mathbf{u}}) > V_f(x)$, an admissible warm start $\tilde{\mathbf{u}}_f(x)$ can be recovered using the terminal control law.

Proposition 2.42 (Admissible warm start in $\mathbb{X}_f$)**.** *For any $x \in \mathbb{X}_f$, the following warm start is feasible*

$$\tilde{\mathbf{u}}_f(x) := (\kappa_f(x), \kappa_f(f(x, \kappa_f(x))), \dots) \in \tilde{\mathcal{U}}_N(x)$$

The proof of this proposition is discussed in Exercise 2.24.

We define the set of admissible control sequences $\breve{\mathcal{U}}_N(x, \tilde{\mathbf{u}})$ as those feasible control sequences $\mathbf{u}$ that result in a lower cost than the warm start; the suboptimal control law is the set of first elements of admissible control sequences

$$\begin{aligned} \breve{\mathcal{U}}_N(x, \tilde{\mathbf{u}}) &= \{\mathbf{u} \mid \mathbf{u} \in \tilde{\mathcal{U}}_N(x),\ V_N(x, \mathbf{u}) \le V_N(x, \tilde{\mathbf{u}})\} \\ \kappa_N(x, \tilde{\mathbf{u}}) &= \{u(0) \mid \mathbf{u} \in \breve{\mathcal{U}}_N(x, \tilde{\mathbf{u}})\} \end{aligned}$$

From its definition, the suboptimal control law is a function of both the state x and the warm start $\tilde{\mathbf{u}} \in \tilde{\mathcal{U}}_N(x)$.

To complete the algorithm we require a successor warm start for the successor state $x^+ = f(x, u(0))$. First defining

$$\tilde{\mathbf{u}}_w(x, \mathbf{u}) := \big(u(1), u(2), \dots, u(N-1), \kappa_f(\phi(N; x, \mathbf{u}))\big)$$

we choose the successor warm start $\tilde{\mathbf{u}}^+ \in \tilde{\mathcal{U}}_N(x^+)$ as follows

$$\tilde{\mathbf{u}}^+ := \begin{cases} \tilde{\mathbf{u}}_f(x^+) & \text{if } x^+ \in \mathbb{X}_f \text{ and} \\ & V_N(x^+, \tilde{\mathbf{u}}_f(x^+)) \le V_N(x^+, \tilde{\mathbf{u}}_w(x, \mathbf{u})) \\ \tilde{\mathbf{u}}_w(x, \mathbf{u}) & \text{else} \end{cases} \tag{2.29}$$

This mapping in (2.29) is denoted $\tilde{\mathbf{u}}^+ = \zeta(x, \mathbf{u})$, and Proposition 2.42 ensures that the warm start generated by $\zeta(x, \mathbf{u})$ is admissible for x^+. We have the following algorithm for suboptimal MPC.

Algorithm 2.43 (Suboptimal MPC)**.** First, choose $\mathbb{X}_f$ and $V_f(\cdot)$ satisfying Assumption 2.14 and obtain the initial state $x \in \mathcal{X}_N$ and any initial warm start $\tilde{\mathbf{u}} \in \tilde{\mathcal{U}}_N(x)$. Then repeat

1. Obtain current measurement of state x.
2. Compute any input $\mathbf{u} \in \breve{\mathcal{U}}_N(x, \tilde{\mathbf{u}})$.
3. Inject the first element of the input sequence $\mathbf{u}$.
4. Compute the next warm start $\tilde{\mathbf{u}}^+ = \zeta(x, \mathbf{u})$.

Because the control law $\kappa_N(x, \tilde{\mathbf{u}})$ is a function of the warm start $\tilde{\mathbf{u}}$ as well as the state x, we extend the meaning of state to include the warm start.

2.7.1 Extended State

In Algorithm 2.43 we begin with a state and warm-start pair and proceed from this pair to the next at the start of each time step. We denote this extended state as $z := (x, \tilde{\mathbf{u}})$ for $x \in \mathcal{X}_N$ and $\tilde{\mathbf{u}} \in \tilde{\mathcal{U}}_N(x)$. The extended state evolves according to

$$z^+ \in H(z) := \{(x^+, \tilde{\mathbf{u}}^+) \mid x^+ = f(x, u(0)), \\ \tilde{\mathbf{u}}^+ = \zeta(x, \mathbf{u}),\ \mathbf{u} \in \breve{\mathcal{U}}_N(z)\} \tag{2.30}$$

in which $u(0)$ is the first element of $\mathbf{u}$. We denote by $\psi(k; z)$ any solution of (2.30) with initial extended state z and denote by $\phi(k; z)$ the accompanying x trajectory. We restrict $\mathcal{Z}_N$ to the set of z for which $\tilde{\mathbf{u}} \in \tilde{\mathcal{U}}_N(x)$.

$$\tilde{\mathcal{Z}}_N := \{(x, \tilde{\mathbf{u}}) \mid x \in \mathcal{X}_N \text{ and } \tilde{\mathbf{u}} \in \tilde{\mathcal{U}}_N(x)\}$$

To directly link the asymptotic behavior of z with that of x, the following proposition is necessary.

Proposition 2.44 (Linking warm start and state)**.** *There exists a function $\alpha_r(\cdot) \in \mathcal{K}_\infty$ such that $|\tilde{\mathbf{u}}| \leq \alpha_r(|x|)$ for any $(x, \tilde{\mathbf{u}}) \in \tilde{\mathcal{Z}}_N$.*

A proof is given in (Allan et al., 2017, Proposition 10).

2.7.2 Asymptotic Stability of Difference Inclusions

Because the extended state evolves as the difference inclusion (2.30), we present the following definitions of asymptotic stability and the associated Lyapunov functions. Consider the difference inclusion $z^+ \in H(z)$, such that $H(0) = \{0\}$.

Definition 2.45 (Asymptotic stability (difference inclusion))**.** We say the origin of the difference inclusion $z^+ \in H(z)$ is asymptotically stable in a positive invariant set $\mathcal{Z}$ if there exists a function $\beta(\cdot) \in \mathcal{KL}$ such that for any $z \in \mathcal{Z}$ and for all $k \in \mathbb{I}_{\geq 0}$, all solutions $\psi(k; z)$ satisfy

$$|\psi(k; z)| \leq \beta(|z|, k)$$

Definition 2.46 (Lyapunov function (difference inclusion))**.** $V(\cdot)$ is a Lyapunov function in the positive invariant set $\mathcal{Z}$ for the difference inclusion $z^+ \in H(z)$ if there exist functions $\alpha_1(\cdot), \alpha_2(\cdot), \alpha_3(\cdot) \in \mathcal{K}_\infty$ such that for all $z \in \mathcal{Z}$

$$\alpha_1(|z|) \leq V(z) \leq \alpha_2(|z|) \tag{2.31}$$

$$\sup_{z^+ \in H(z)} V(z^+) \leq V(z) - \alpha_3(|z|) \tag{2.32}$$

Although $V(\cdot)$ is not required to be continuous everywhere, (2.31) implies that it is continuous at the origin.

Proposition 2.47 (Asymptotic stability (difference inclusion))**.** *If the set $\mathcal{Z}$ contains the origin, is positive invariant for the difference inclusion $z^+ \in H(z)$, $H(0) = \{0\}$, and it admits a Lyapunov function $V(\cdot)$ in $\mathcal{Z}$, then the origin is asymptotically stable in $\mathcal{Z}$.*

A proof of this proposition is given in (Allan et al., 2017, Proposition 13); it is similar to the proof of Theorem B.15 in Appendix B.

Theorem 2.48 (Asymptotic stability of suboptimal MPC)**.** *Suppose Assumptions 2.2, 2.3, and 2.14 are satisfied, and that $\ell(x,u) \geq \alpha_\ell(|(x,u)|)$ for all $(x,u) \in \mathbb{Z}$, and $\mathbb{X}_f = \operatorname{lev}_b V_f = \{x \in \mathbb{R}^n \mid V_f(x) \leq b\}$, for some $b > 0$. Then the function $V_N(z)$ is a Lyapunov function in the set $\widetilde{\mathcal{Z}}_N$ for the closed-loop system* (2.30) *under Algorithm 2.43. Therefore the origin is asymptotically stable in $\widetilde{\mathcal{Z}}_N$.*

Proof. First we show that $V_N(z)$ is a Lyapunov function for (2.30) on the positive invariant set $\widetilde{\mathcal{Z}}_N$. Because $\mathbf{u} \in \breve{\mathcal{U}}_N(z)$ and, by construction, $\widetilde{\mathbf{u}}^+ \in \widetilde{\mathcal{U}}_N(x^+)$, we have that $z^+ \in \widetilde{\mathcal{Z}}_N$, so that $\widetilde{\mathcal{Z}}_N$ is positive invariant. From the definition of the control law and the warm start, we have that for all $z \in \widetilde{\mathcal{Z}}_N$

$$V_N(z) \geq V_N(x, \mathbf{u}) \geq \sum_{i=0}^{N-1} \ell(x(i), u(i)) \geq \sum_{i=0}^{N-1} \alpha_\ell(|(x(i), u(i))|)$$

Next we use (B.1) from Appendix B and the triangle inequality to obtain

$$\sum_{i=0}^{N-1} \alpha_\ell(|(x(i), u(i))|) \geq \alpha_\ell\Big(\frac{1}{N} \sum_{i=0}^{N-1} |(x(i), u(i))|\Big) \geq \alpha_\ell(\,|(\mathbf{x}, \mathbf{u})|\,/N)$$

Finally using the ℓ_p-norm property that for all vectors a, b, $|(a,b)| \geq |b|$, and noting that $x(0) = x$, so we have that

$$\alpha_\ell(\,|(\mathbf{x}, \mathbf{u})|\,/N) \geq \alpha_\ell(\,|(x, \mathbf{u})|\,/N) := \alpha_1(\,|(x, \mathbf{u})|\,) = \alpha_1(z)$$

with $\alpha_1 \in \mathcal{K}_\infty$. So we have established the lower bound $V_N(z) \geq \alpha_1(z)$ for all $z \in \widetilde{\mathcal{Z}}_N$.

Because of Assumptions 2.2 and 2.3, the set $\mathcal{Z}_N$ is closed as shown in Proposition 2.10(c). The cost function $V_N(z)$ is continuous on $\mathcal{Z}_N$, which includes $z = 0$, so from Proposition B.25 we conclude that there exists $\alpha_2(\cdot) \in \mathcal{K}_\infty$ such that $V_N(z) \leq \alpha_2(|z|)$ for all $z \in \widetilde{\mathcal{Z}}_N \subset \mathcal{Z}_N$, and the upper-bound condition of Definition 6.2 is satisfied.

As in standard MPC analysis, we have for all $z \in \widetilde{\mathcal{Z}}_N$ that

$$V_N(z^+) \leq V_N(x, \mathbf{u}) - \ell(x, u(0)) \leq V_N(x, \mathbf{u}) - \alpha_\ell(|x, u(0)|)$$

Because $\widetilde{\mathbf{u}} \in \widetilde{\mathcal{U}}_N(x)$, from Proposition 2.44 we have that

$$\left|(x, \widetilde{\mathbf{u}})\right| \leq |x| + \left|\widetilde{\mathbf{u}}\right| \leq |x| + \alpha_r(|x|) := \alpha_{r'}(|x|) \leq \alpha_{r'}(|(x, u(0))|)$$

Therefore, $\alpha_\ell \circ \alpha_{r'}^{-1}(\left|(x, \widetilde{\mathbf{u}})\right|) \leq \alpha_\ell(|(x, u(0))|)$. Defining $\alpha_3(\cdot) := \alpha_\ell \circ \alpha_{r'}^{-1}(\cdot)$ and because $V_N(x, \mathbf{u}) \leq V_N(x, \widetilde{\mathbf{u}})$, we have that

$$V_N(z^+) \leq V_N(x, \widetilde{\mathbf{u}}) - \alpha_3(|z|) = V_N(z) - \alpha_3(|z|)$$

for all $z \in \widetilde{\mathcal{Z}}_N$ and $z^+ \in H(z)$. We conclude that $V_N(z)$ is a Lyapunov function for (2.30) in $\widetilde{\mathcal{Z}}_N$. Asymptotic stability follows directly from Proposition 2.47. ■

From this result, a bound on just $x(k)$ rather than $z(k) = (x(k), \widetilde{\mathbf{u}}(k))$ can also be derived. First we have that for all $k \geq 0$ and $z \in \widetilde{\mathcal{Z}}_N$

$$|z(k; z)| \leq \beta(|z|, k) = \beta(\left|(x, \widetilde{\mathbf{u}})\right|, k) \leq \beta(|x| + \left|\widetilde{\mathbf{u}}\right|, k)$$

From Proposition 2.44 we then have that

$$\beta(|x| + \left|\widetilde{\mathbf{u}}\right|, k) \leq \beta(|x| + \alpha_r(|x|), k) := \widetilde{\beta}(|x|, k)$$

with $\widetilde{\beta}(\cdot) \in \mathcal{KL}$. Combining these we have that

$$|z(k; z)| = \left|(x(k; z), \widetilde{\mathbf{u}}(k; z))\right| \leq |x(k; z)| + \left|\widetilde{\mathbf{u}}(k; z)\right| \leq \widetilde{\beta}(|x|, k)$$

which implies $|x(k; z)| \leq \widetilde{\beta}(|x|, k)$. So we have a bound on the evolution of $x(k)$ depending on *only* the x initial condition. Note that the evolution of $x(k)$ depends on the initial condition of $z = (x, \widetilde{\mathbf{u}})$, so it depends on initial warm start $\widetilde{\mathbf{u}}$ as well as initial x. We cannot ignore this dependence, which is why we had to analyze the extended state in the first place. For the same reason we also cannot define the invariant set in which the $x(k)$ evolution takes place without referring to $\widetilde{\mathcal{Z}}_N$.

2.8 Economic Model Predictive Control

Many applications of control are naturally posed as tracking problems. Vehicle guidance, robotic motion guidance, and low-level objectives such as maintaining pressures, temperatures, levels, and flows in industrial processes are typical examples. MPC can certainly provide feedback control designs with excellent tracking performance for challenging multivariable, constrained, and nonlinear systems as we have explored thus far in the text. But feedback control derived from repeated online optimization of a process model enables other, higher-level goals to be addressed as well. In this section we explore using MPC for *optimizing economic performance* of a process rather than a simple tracking objective. As before, we assume the system dynamics are described by the model

$$x^+ = f(x,u)$$

But here the stage cost is some general function $\ell(x,u)$ that measures economic performance of the process. The stage cost is not positive definite with respect to some target equilibrium point of the model as in a tracking problem. We set up the usual MPC objective function as a sum of stage costs over some future prediction horizon

$$V_N(x,\mathbf{u}) = \sum_{k=0}^{N-1} \ell(x(k),u(k)) + V_f(x(N))$$

subject to the system model with $x(0) = x$, the initial condition. As before, we consider constraints on the states and inputs, $(x,u) \in \mathbb{Z}$. So the only significant change in the MPC problem has been the redefinition of the stage cost $\ell(x,u)$ to reflect the economics of the process. The terminal penalty $V_f(x)$ may be changed for the same reason. Typical stage-cost functions would be composed of a sum of prices of the raw materials and utilities, and the values of the products being manufactured.

We can also define the best steady-state solution of the system from the economic perspective. This optimal steady-state pair (x_s,u_s) is defined as the solution to the optimization problem $\mathbb{P}_s$

$$(x_s,u_s) := \arg\min_{(x,u)\in\mathbb{Z}} \{\ell(x,u) \mid x = f(x,u)\}$$

The standard industrial approach to addressing economic performance is to calculate this best economic steady state (often on a slower time

scale than the process sample time), and then design an MPC controller with a different, tracking stage cost to reject disturbances and *track* this steady state. In this approach, a typical tracking stage cost would be the types considered thus far, e.g., $\ell_t(x,u) = (1/2)(|x-x_s|_Q^2 + |u-u_s|_R^2)$.

In economic MPC, we instead use the same economic stage cost directly in the dynamic MPC problem. Some relevant questions to be addressed with this change in design philosophy are: (i) how much economic performance improvement is possible, and (ii) how different is the closed-loop dynamic behavior. For example, we are not even guaranteed for a nonlinear system that operating at the steady state is the best possible dynamic behavior of the closed-loop system.

As an introduction to the topic, we next set up the simplest version of an economic MPC problem, in which we use a terminal constraint. In the Notes section, we comment on what generalizations are available in the literature. We now modify the basic assumptions given previously.

Assumption 2.49 (Continuity of system and cost)**.** The functions $f : \mathbb{Z} \to \mathbb{R}^n$ and $\ell : \mathbb{Z} \to \mathbb{R}_{\geq 0}$ are continuous. $V_f(\cdot) = 0$. There exists at least one point $(x_s, u_s) \in \mathbb{Z}$ satisfying $x_s = f(x_s, u_s)$.

Assumption 2.50 (Properties of constraint sets)**.** The set $\mathbb{Z}$ is closed. If there are control constraints, the set $\mathbb{U}(x)$ is compact and is uniformly bounded in $\mathbb{X}$.

Assumption 2.51 (Cost lower bound)**.**

(a) The terminal set is a single point, $\mathbb{X}_f = \{x_s\}$.

(b) The stage cost $\ell(x,u)$ is lower bounded for $(x,u) \in \mathbb{Z}$.

Note that since we are using a terminal equality constraint, we do not require the terminal penalty $V_f(\cdot)$, so it is set to zero. For clarity in this discussion, we do not assume that (x_s, u_s) has been shifted to the origin. The biggest change is that we do not assume here that the stage cost $\ell(x,u)$ is positive definite with respect to the optimal steady state, only that it is lower bounded.

Note that the set of steady states, $\mathbb{Z}_s := \{(x,u) \in \mathbb{Z} \mid x = f(x,u)\}$, is nonempty due to Assumption 2.49. It is closed because $\mathbb{Z}$ is closed (Assumption 2.50) and $f(\cdot)$ is continuous. But it may not be bounded so we are not guaranteed that the solution to $\mathbb{P}_s$ exists. So we consider (x_s, u_s) to be any element of $\mathbb{Z}_s$. We may want to choose (x_s, u_s) to be an element of the solution to $\mathbb{P}_s$, when it exists, but this is not necessary to the subsequent development.

The economic optimal control problem $\mathbb{P}_N(x)$, is the same as in (2.7)

$$\mathbb{P}_N(x): \qquad V_N^0(x) := \min_{\mathbf{u}} \{V_N(x,\mathbf{u}) \mid \mathbf{u} \in \mathcal{U}_N(x)\}$$

Due to Assumptions 2.49 and 2.50, Proposition 2.4 holds, and the solution to the optimal control problem exists. The control law, $\kappa_N(\cdot)$ is therefore well defined; if it is not unique, we consider as before a fixed selection map, and the closed-loop system is again given by

$$x^+ = f(x, \kappa_N(x)) \tag{2.33}$$

2.8.1 Asymptotic Average Performance

We already have enough structure in this simple problem to establish that the average cost of economic MPC is better, i.e., not worse, than any steady-state performance $\ell(x_s, u_s)$.

Proposition 2.52 (Asymptotic average performance)**.** *Let Assumptions 2.49, 2.50, and 2.51 hold. Then for every $x \in \mathcal{X}_N$, the following holds*

$$\limsup_{t\to\infty} \sum_{k=0}^{t-1} \frac{\ell(x(k), u(k))}{t} \leq \ell(x_s, u_s)$$

in which $x(k)$ is the closed-loop solution to (2.33) *with initial condition x, and $u(k) = \kappa_N(x(k))$.*

Proof. Because of the terminal constraint, we have that

$$V_N^0(f(x, \kappa_N(x))) \leq V_N^0(x) - \ell(x, \kappa_N(x)) + \ell(x_s, u_s) \tag{2.34}$$

Performing a sum on this inequality gives

$$\sum_{k=0}^{t=1} \frac{\ell(x(k), u(k))}{t} \leq \ell(x_s, u_s) + (1/t)(V_N^0(x(0)) - V_N^0(x(t)))$$

The left-hand side may not have a limit, so we take lim sup of both sides. Note that from Assumption 2.51(b), $\ell(x,u)$ is lower bounded for $(x,u) \in \mathbb{Z}$, hence so is $V_N(x,u)$ for $(x,u) \in \mathbb{Z}$, and $V_N^0(x)$ for $x \in \mathcal{X}_N$. Denote this bound by M. Then $\lim_{t\to\infty} -(1/t)V_N^0(x(t)) \leq \lim_{t\to\infty} -M/t = 0$ and we have that

$$\limsup_{t\to\infty} \sum_{k=0}^{t=1} \frac{\ell(x(k), u(k))}{t} \leq \ell(x_s, u_s) \qquad \blacksquare$$

This result does not imply that the economic MPC controller stabilizes the steady state (x_s, u_s), only that the *average* closed-loop performance is better than the best steady-state performance. There are many examples of nonlinear systems for which the time-average of an oscillation is better than the steady state. For such systems, we would expect an optimizing controller to destabilize even a stable steady state to obtain the performance improvement offered by cycling the system.

Note also that the appearance in (2.34) of the term $-\ell(x, \kappa_N(x)) + \ell(x_s, u_s)$, which is sign indeterminate, destroys the cost decrease property of $V_N^0(\cdot)$ so it no longer can serve as a Lyapunov function in a closed-loop stability argument. We next examine the stability question.

2.8.2 Dissipativity and Asymptotic Stability

The idea of dissipativity proves insightful in understanding when economic MPC is stabilizing (Angeli, Amrit, and Rawlings, 2012). The basic idea is motivated by considering a thermodynamic system, mechanical energy, and work. Imagine we *supply* mechanical energy to a system by performing work on the system at some rate. We denote the mechanical energy as a *storage* function, i.e., as the way in which the work performed on the system is stored by the system. If the system has no *dissipation*, then the rate of change in storage function (mechanical energy) is equal to the supply rate (work). However, if the system also dissipates mechanical energy into heat, through friction for example, then the change in the storage function is strictly less than the work supplied. We make this physical idea precise in the following definition.

Definition 2.53 (Dissipativity)**.** The system $x^+ = f(x, u)$ is dissipative with respect to supply rate $s : \mathbb{Z} \to \mathbb{R}$ if there exists a storage function $\lambda : \mathbb{X} \to \mathbb{R}$ such that for all $(x, u) \in \mathbb{Z}$

$$\lambda(f(x, u)) - \lambda(x) \leq s(x, u) \tag{2.35}$$

The system is strictly dissipative with respect to supply rate s and steady-state x_s if there exists $\alpha(\cdot) \in \mathcal{K}_\infty$ such that for all $(x, u) \in \mathbb{Z}$

$$\lambda(f(x, u)) - \lambda(x) \leq s(x, u) - \alpha(|x - x_s|) \tag{2.36}$$

Note that we do *not* assume that $\lambda(\cdot)$ is continuous, and we define strict dissipativity with $\alpha(\cdot)$ a $\mathcal{K}_\infty$ function. In other literature, $\alpha(\cdot)$ is sometimes assumed to be a continuous, positive definite function.

We require one technical assumption; its usefulness will be apparent shortly.

Assumption 2.54 (Continuity at the steady state)**.** The function $V_N^0(\cdot) + \lambda(\cdot) : \mathcal{X}_N \to \mathbb{R}$ is continuous at x_s.

The following assumption is then sufficient to guarantee that economic MPC is stabilizing.

Assumption 2.55 (Strict dissipativity)**.** The system $x^+ = f(x,u)$ is strictly dissipative with supply rate

$$s(x,u) = \ell(x,u) - \ell(x_s,u_s)$$

Theorem 2.56 (Asymptotic stability of economic MPC)**.** *Let Assumptions 2.49, 2.50, 2.51, 2.54, and 2.55 hold. Then x_s is asymptotically stable in $\mathcal{X}_N$ for the closed-loop system $x^+ = f(x, \kappa_N(x))$.*

Proof. We know that $V_N^0(\cdot)$ is *not* a Lyapunov function for the given stage cost $\ell(\cdot)$, so our task is to construct one. We first introduce a *rotated* stage cost as follows (Diehl, Amrit, and Rawlings, 2011)

$$\widetilde{\ell}(x,u) = \ell(x,u) - \ell(x_s,u_s) + \lambda(x) - \lambda(f(x,u))$$

Note from (2.36) and Assumption 2.55 that this stage cost then satisfies for all $(x,u) \in \mathbb{Z}$

$$\widetilde{\ell}(x,u) \geq \alpha(|x - x_s|) \qquad \widetilde{\ell}(x_s,u_s) = 0 \tag{2.37}$$

and we have the kind of stage cost required for a Lyapunov function. Next define an N-stage sum of this new stage cost as $\widetilde{V}_N(x,\mathbf{u}) := \sum_{k=0}^{N-1} \widetilde{\ell}(x(k),u(k))$ and perform the sum to obtain

$$\begin{aligned}\widetilde{V}_N(x,\mathbf{u}) &= \Big(\sum_{k=0}^{N-1} \ell(x(k),u(k))\Big) - N\ell(x_s,u_s) + \lambda(x) - \lambda(x_s) \\ &= V_N(x,\mathbf{u}) - N\ell(x_s,u_s) + \lambda(x) - \lambda(x_s)\end{aligned} \tag{2.38}$$

Notice that $\widetilde{V}_N(\cdot)$ and $V_N(\cdot)$ differ only by constant terms involving the steady state, (x_s,u_s), and the initial condition, x. Therefore because the optimization of $V_N(x,\mathbf{u})$ over $\mathbf{u}$ has a solution, so does the optimization of $\widetilde{V}_N(x,\mathbf{u})$, and they are the *same* solution, giving the same control law $\kappa_N(x)$.

Because of the terminal constraint, we know that $\mathcal{X}_N$ is positive invariant for the closed-loop system. Next we verify that $\widetilde{V}_N^0(x)$ is a

Lyapunov function for the closed-loop system. Since $\widetilde{\ell}(x,u)$ is nonnegative, we have from (2.37) and the definition of $\widetilde{V}_N$ as a sum of stage costs, that

$$\widetilde{V}_N^0(x) \geq \alpha(|x - x_s|)$$

for all $x \in \mathcal{X}_N$, and we have established the required lower bound. The cost difference can be calculated to establish the required cost decrease

$$\widetilde{V}_N^0(f(x,\kappa_N(x))) \leq \widetilde{V}_N^0(x) - \widetilde{\ell}(x,\kappa_N(x)) \leq \widetilde{V}_N^0(x) - \alpha(|x - x_s|)$$

for all $x \in \mathcal{X}_N$. The remaining step is to verify the upper-bounding inequality. From Assumption 2.54 and (2.38), we know that $\widetilde{V}_N^0(\cdot)$ is also continuous at x_s. Therefore, from Proposition 2.38, we have existence of $\alpha_2(\cdot) \in \mathcal{K}_\infty$ such that for all $x \in \mathcal{X}_N$

$$\widetilde{V}_N^0(x) \leq \alpha_2(|x - x_s|)$$

We have established the three inequalities and $\widetilde{V}_N^0(\cdot)$ is therefore a Lyapunov function in $\mathcal{X}_N$ for the system $x^+ = f(x,\kappa_N(x))$ and x_s. Theorem 2.13 then establishes that x_s is asymptotically stable in $\mathcal{X}_N$ for the closed-loop system. ■

These stability results can also be extended to time-varying and periodic systems.

Example 2.57: Economic MPC versus tracking MPC

Consider the linear system

$$f(x,u) = Ax + Bu \qquad A = \begin{bmatrix} 1/2 & 1 \\ 0 & 3/4 \end{bmatrix} \qquad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

with economic cost function

$$\ell_{\text{econ}}(x,u) = q'x + r'u \qquad q = \begin{bmatrix} -2 \\ 2 \end{bmatrix} \qquad r = -10$$

and sets $\mathbb{X} = [-10,10]^2$, $\mathbb{U} = [-1,1]$. The economically optimal steady state is $x_s = (8,4)$, $u_s = 1$. We compare economic MPC to tracking MPC with

$$\ell_{\text{track}}(x,u) = |x - x_s|_{10I}^2 + |u - u_s|_I^2$$

Figure 2.5 shows a phase plot of the closed-loop evolution starting from $x = (-8,8)$. Both controllers use the terminal constraint $\mathbb{X}_f = \{x_s\}$.

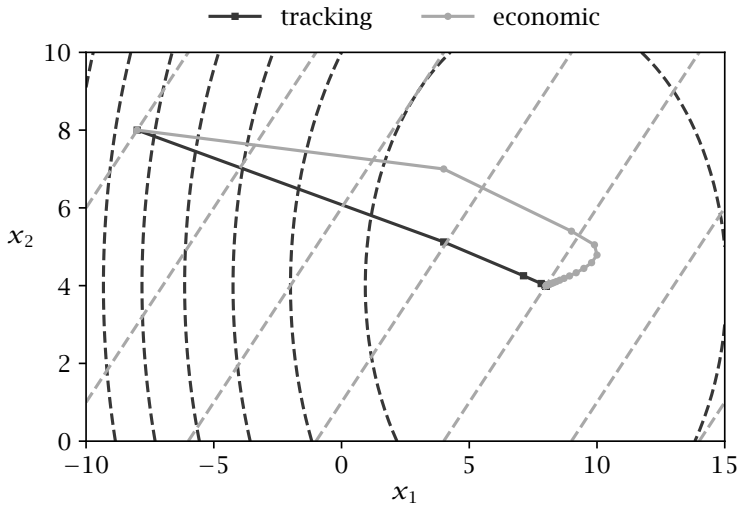


Figure 2.5: Closed-loop economic MPC versus tracking MPC starting at $x = (-8, 8)$ with optimal steady state $(8, 4)$. Both controllers asymptotically stabilize the steady state. Dashed contours show cost functions for each controller.

While tracking MPC travels directly to the setpoint, economic MPC takes a detour to achieve lower economic costs.

To prove that the economic MPC controller is stabilizing, we find a storage function. As a candidate storage function, we take

$$\lambda(x) = \mu'(x - x_s) + (x - x_s)'M(x - x_s)$$

which gives the rotated cost function

$$\widetilde{\ell}(x, u) = \ell_{\text{econ}}(x, u) + \lambda(x) - \lambda(f(x, u))$$

To start, we take $\mu = (4, 8)$ from the Lagrange multiplier of the steady-state problem. With $M = 0$, $\widetilde{\ell}(\cdot)$ is nonnegative but not positive definite, indicating that the system is dissipative but not strictly dissipative. To achieve strict dissipativity, we choose M such that $M - A'MA = 0.01I$. Although the resulting $\widetilde{\ell}(\cdot)$ function is nonconvex,

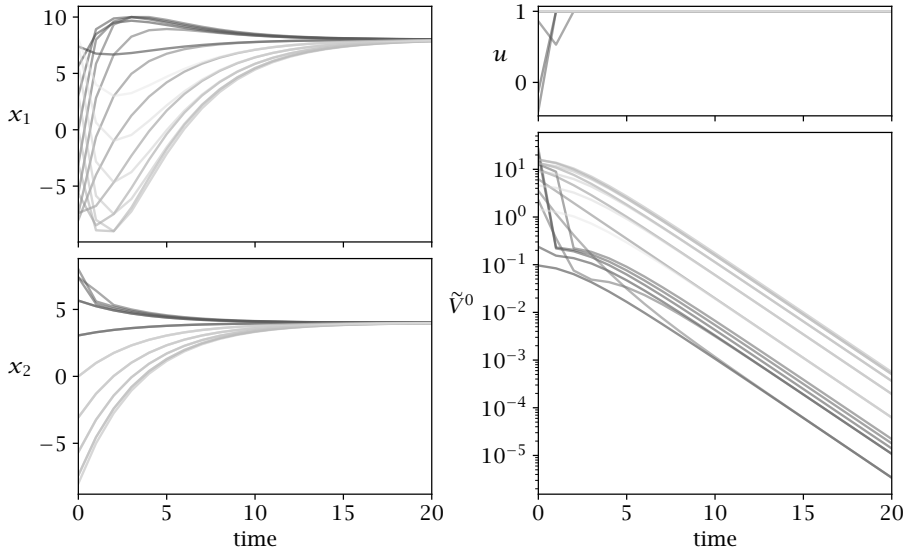


Figure 2.6: Closed-loop evolution under economic MPC. The rotated cost function $\tilde{V}^0$ is a Lyapunov function for the system.

it is nevertheless positive definite on $\mathbb{Z}$, indicating strict dissipativity. To illustrate, we simulate a variety of initial conditions in Figure 2.6. Plotting the rotated cost function $\tilde{V}^0(\cdot)$, we see that it is indeed a Lyapunov function for the system. □

2.9 Discrete Actuators

Discrete-valued actuators appear in nearly all large-scale industrial processes. These obviously include the on/off equipment switches. But, as discussed in Chapter 1, processes are often designed with multiple similar units such as furnaces, heaters, chillers, compressors, etc., operating in parallel. In these designs, an important aspect of the control problem is to choose how many and which of these several possible units to employ while the total feed flowrate to the process varies.

In industrial practice, these discrete decisions are usually removed from the MPC control layer and instead made at a different layer of the automation system using heuristics or other logical rules. If discrete inputs are chosen optimally, however, process performance can

be greatly improved, and thus we would like to treat discrete decisions directly in MPC theory.

There are two basic issues brought about by including the discrete actuators in the control decision u. The first is theoretical: how much does the established MPC theory have to change to accommodate this class of decision variables? The second is computational: is it practical to solve the modified MPC optimal control problem in the available sample time? We address the theory question here, and find that the required changes to the existing theory are surprisingly minimal. The computational question is being addressed by the rapid development of mixed-integer solvers. It is difficult to predict what limits might emerge to slow this progress, but current mixed-integer solvers are already capable of addressing a not uninteresting class of industrial applications.

Figure 1.2 provides a representative picture of the main issue. From this perspective, if we embed the discrete decisions in the field of reals, we are merely changing the feasible region $\mathbb{U}$, from a simply connected set with an interior when describing only continuous actuators, to a disconnected set that may not have an interior when describing mixed continuous/discrete actuators. So one theoretical approach to the problem is to adjust the MPC theory to accommodate these types of $\mathbb{U}$ regions.

A careful reading of the assumptions made for the results presented thus far reveals that we have little work to do. We have not assumed that the equilibrium of interest lies in the interior of $\mathbb{U}$, or even that $\mathbb{U}$ has an interior. The main assumption about $\mathbb{U}$ are Assumption 2.3 for the time-invariant case, Assumption 2.26 for the time-varying case, and Assumption 2.50 for the economic MPC problem. The main restrictions are that $\mathbb{U}$ is closed, and sometimes compact, so that the optimization of $V_N(x, \mathbf{u})$ over $\mathbf{u}$ has a solution. All of these assumptions admit $\mathbb{U}$ regions corresponding to discrete variables. The first conclusion is that the results governing nominal closed-loop stability for various forms of MPC all pass through. These include Theorem 2.19 (time-invariant case), Theorem 2.39 (time-varying case), Theorem 2.24 ($\ell(y, u)$ stage cost), and Theorem 2.56 (economic MPC).

That does not mean that *nothing* has changed. The admissible region $\mathcal{X}_N$ in which the system is stabilized may change markedly, for example. Proposition 2.10 also passes through in the discrete-actuator case, so we know that the admissible sets are still nested, $\mathcal{X}_j \subseteq \mathcal{X}_{j+1}$ for all $j \geq 0$. But it is not unusual for systems with even linear dynamics to have *disconnected* admissible regions, which is not possible

for linear systems with only continuous actuators and convex $\mathbb{U}$. When tracking a constant setpoint, the design of terminal regions and penalties must account for the fact that the discrete actuators usually remain at fixed values in a small neighborhood of the steady state of interest, and can be used only for rejecting larger disturbances and enhancing transient performance back to the steady state. Fine control about the steady state must be accomplished by the continuous actuators that are unconstrained in a neighborhood of the steady state. But this is the same issue that is faced when some subset of the continuous actuators are saturated at the steady state of interest (Rao and Rawlings, 1999), which is a routine situation in process control problems. We conclude the chapter with an example illustrating these issues.

Example 2.58: MPC with mixed continuous/discrete actuators

Consider a constant-volume tank that needs to be cooled. The system is diagrammed in Figure 2.7. The two cooling units operate such that they can be either on or off, and if on, the heat duty must be between $\dot{Q}_{\min}$ and $\dot{Q}_{\max}$. After nondimensionalizing, the system evolves according to

$$\frac{dT_1}{dt} = -\alpha(T_1 - T_0) - \rho_1(T_1 - T_2)$$
$$\frac{dT_2}{dt} = -\rho_2(T_2 - T_1) - \beta\dot{Q}$$

with $\alpha = 2$ and $\beta = \rho_1 = \rho_2 = 1$. The system states are (T_1, T_2), and the inputs are $(\dot{Q}, n_q)$ with

$$\mathbb{U} = \left\{(\dot{Q}, n_q) \in \mathbb{R} \times \{0, 1, 2\} \mid n_q\dot{Q}_{\min} \le \dot{Q} \le n_q\dot{Q}_{\max}\right\}$$

in which $\dot{Q}$ is the total cooling duty and n_q chooses the number of cooling units that are on at the given time. For $T_0 = 40$ and $\dot{Q}_{\max} = 10$, we wish to control the system to the steady state $x_s = (35, 25)$, $u_s = (10, 1)$, using costs $Q = I$ and $R = 10^{-3}I$. The system is discretized with $\Delta = 0.25$.

To start, we choose a terminal region and control law. Assuming $\dot{Q}_{\min} > 0$, both components of u are at constraints at the steady state, and thus we cannot use them in a linear terminal control law. The system is stable for $\kappa_f(x) = u_s$, however, and a valid terminal cost is $V_f(x) = (x - x_s)'P(x - x_s)$ with P satisfying $A'PA - P = -Q$. As a terminal set we take $\mathbb{X}_f = \{x \mid V_f(x) \le 1\}$, although any level set would suffice. With this terminal region, Figure 2.8 shows the feasible sets for $\dot{Q}_{\min} = 0$ and $\dot{Q}_{\min} = 9$. Note that for $\dot{Q}_{\min} > 0$, the projection

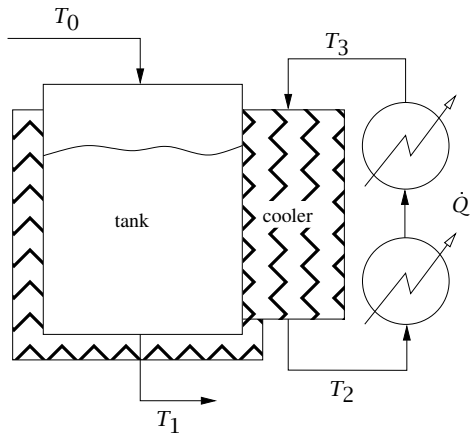


Figure 2.7: Diagram of tank/cooler system. Each cooling unit can be either on or off, and if on, it must be between its (possibly nonzero) minimum and maximum capacities.

of $\mathbb{U}$ onto the total heat duty $\dot{Q}$ is a disconnected set of possible heat duties, leading to disconnected sets $\mathcal{X}_N$ for $N \leq 5$. (The sets $\mathcal{X}_N$ for $N \geq 6$ are connected.)

To control the system, we solve the standard MPC problem with horizon $N = 8$. Figure 2.9 shows a phase portrait of closed-loop evolution for various initial conditions with $\dot{Q}_{\min} = 9$. Each evaluation of the control law requires solving a mixed-integer, quadratically constrained QP (with the quadratic constraint due to the terminal region). In general, the controller chooses $u_2 = 1$ near the setpoint and $u_2 \in \{0, 2\}$ far from it, although this behavior is not global. Despite the disconnected nature of $\mathbb{U}$, all initial conditions are driven asymptotically to the setpoint. □

2.10 Concluding Comments

MPC is an implementation, for practical reasons, of receding horizon control (RHC), in which offline determination of the RHC law $\kappa_N(\cdot)$ is replaced by online determination of its value $\kappa_N(x)$, the control action, at each state x encountered during its operation. Because the optimal control problem that defines the control is a finite horizon problem, neither stability nor optimality of the cost function is necessarily achieved

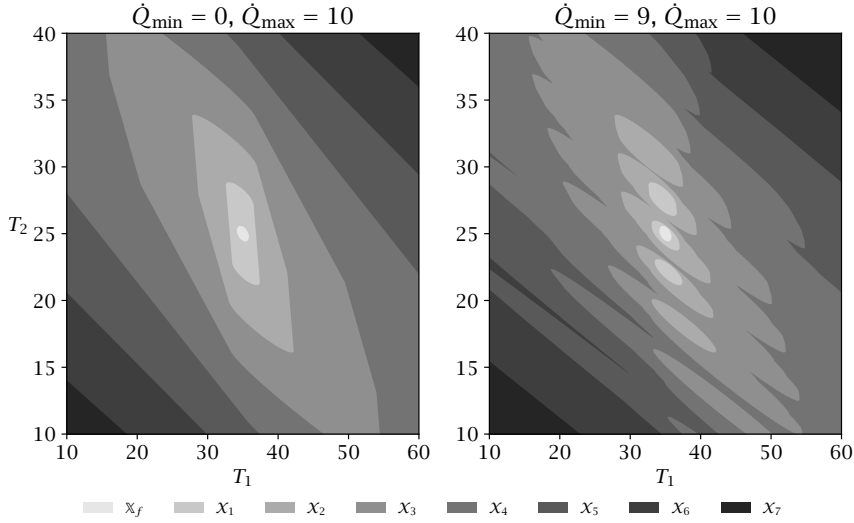


Figure 2.8: Feasible sets $\mathcal{X}_N$ for two values of $\dot{Q}_{\min}$. Note that for $\dot{Q}_{\min} = 9$ (right-hand side), $\mathcal{X}_N$ for $N \leq 4$ are disconnected sets.

by a receding horizon or model predictive controller.

This chapter shows how stability may be achieved by adding a terminal cost function and a terminal constraint to the optimal control problem. Adding a terminal cost function adds little or no complexity to the optimal control problem that has to be solved online, and usually improves performance. Indeed, the infinite horizon value function $V_\infty^0(\cdot)$ for the constrained problem would be an ideal choice for the terminal penalty because the value function $V_N^0(\cdot)$ for the online optimal control problem would then be equal to $V_\infty^0(\cdot)$, and the controller would inherit the performance advantages of the infinite horizon controller. In addition, the actual trajectories of the controlled system would be precisely equal, in the absence of uncertainty, to those predicted by the online optimizer. Of course, if we knew $V_\infty^0(\cdot)$, the optimal infinite horizon controller $\kappa_\infty(\cdot)$ could be determined and there would be no reason to employ MPC.

The infinite horizon cost $V_\infty^0(\cdot)$ is known globally only for special cases, however, such as the linear quadratic (LQ) unconstrained problem. For more general problems in which constraints and/or nonlin-

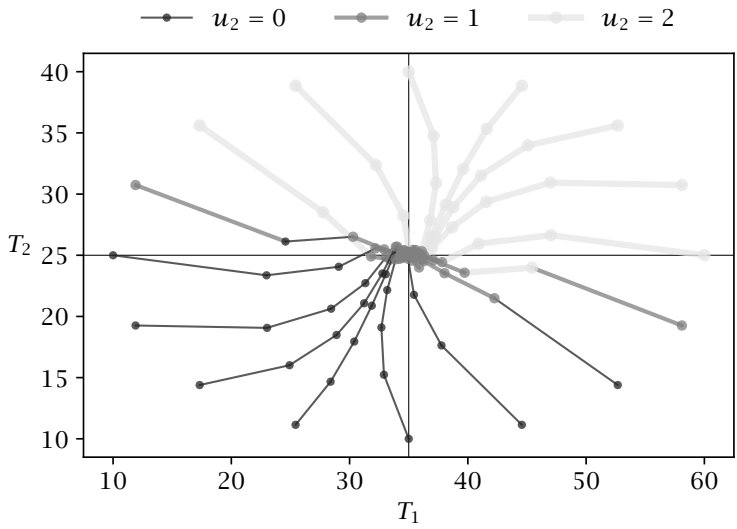


Figure 2.9: Phase portrait for closed-loop evolution of cooler system with $\dot{Q}_{\text{min}} = 9$. Line colors show value of discrete actuator u_2.

earity are present, its value—or approximate value—in a neighborhood of the setpoint can usually be obtained and the use of this local control Lyapunov function (CLF) should, in general, enhance performance. Adding a terminal cost appears to be generally advantageous.

The reason for the terminal constraint is precisely the fact that the terminal penalty is usually merely a local CLF defined in the set $\mathbb{X}_f$; to benefit from the terminal cost, the terminal state must be constrained to lie in $\mathbb{X}_f$. Unlike the addition of a terminal penalty, however, addition of a terminal constraint may increase complexity of the optimal control problem considerably. Because efficient programs exist for solving quadratic programs (QPs), in which the cost function to be minimized is quadratic and the constraints polyhedral, there is an argument for using polyhedral constraints. Indeed, a potential terminal constraint set for the constrained LQ optimal control problem is the maximal constraint admissible set, which is polyhedral. This set is complex, however, i.e., defined by many linear inequalities, and would appear to be

unsuitable for the complex control problems routinely encountered in industry.

A terminal constraint set that is considerably simpler is a suitable sublevel set of the terminal penalty, which is often a simple positive definite quadratic function resulting in a convex terminal constraint set. A disadvantage is that the terminal constraint set is now ellipsoidal rather than polytopic, and conventional QPs cannot be employed for the LQ constrained optimal control problem. This does not appear to be a serious disadvantage, however, because the optimal control problem remains convex, so interior point methods may be readily employed.

In the nonlinear case, adding an ellipsoidal terminal constraint set does not appreciably affect the complexity of the optimal control problem. A more serious problem, when the system is nonlinear, is that the optimal control problem is then usually nonconvex so that global solutions, on which many theoretical results are predicated, are usually too difficult to obtain. A method for dealing with this difficulty, which also has the advantage of reducing online complexity, is suboptimal MPC, described in this chapter and also in Chapter 6.

The current chapter also presents some results that contribute to an understanding of the subject but do not provide practical tools. For example, it is useful to know that the domain of attraction for many of the controllers described here is $\mathcal{X}_N$, the set of initial states controllable to the terminal constraint set, but this set cannot usually be computed. The set is, in principle, computable using the dynamic programming (DP) equations presented in this chapter, and may be computed if the system is linear and the constraints, including the terminal constraint, are polyhedral, provided that the state dimension and the horizon length are suitably small—considerably smaller than in problems routinely encountered in industry. In the nonlinear case, this set cannot usually be computed. Computation difficulties are not resolved if $\mathcal{X}_N$ is replaced by a suitable sublevel set of the value function $V_N^0(\cdot)$. Hence, in practice, both for linear and nonlinear MPC, this set has to be estimated by simulation.

2.11 Notes

MPC has an unusually rich history, making it impossible to summarize here the many contributions that have been made. Here we restrict attention to a subset of this literature that is closely related to the approach adopted in this book. A fuller picture is presented in the review

paper (Mayne, Rawlings, Rao, and Scokaert, 2000).

The success of conventional MPC derives from the fact that for deterministic problems (no uncertainty), feedback is not required so the solution to the open-loop optimal control problem solved online for a particular initial state is the same as that obtained by solving the feedback problem using DP, for example. Lee and Markus (1967) pointed out the possibility of MPC in their book on optimal control.

> One technique for obtaining a feedback controller synthesis is to measure the current control process state and then compute very rapidly the open-loop control function. The first portion of this function is then used during a short time interval after which a new measurement of the process state is made and a new open-loop control function is computed for this new measurement. The procedure is then repeated.

Even earlier, Propoi (1963) proposed a form of MPC utilizing linear programming, for the control of linear systems with hard constraints on the control. A big surge in interest in MPC occurred when Richalet, Rault, Testud, and Papon (1978b) advocated its use for process control. A whole series of papers, such as (Richalet, Rault, Testud, and Papon, 1978a), (Cutler and Ramaker, 1980), (Prett and Gillette, 1980), (García and Morshedi, 1986), and (Marquis and Broustail, 1988) helped cement its popularity in the process control industries, and MPC soon became the most useful method in modern control technology for control problems with hard constraints—with thousands of applications to its credit.

The basic question of stability, an important issue since optimizing a finite horizon cost does not necessarily yield a stabilizing control, was not resolved in this early literature. Early academic research in MPC, reviewed in García, Prett, and Morari (1989), did not employ Lyapunov theory and therefore restricted attention to control of unconstrained linear systems, studying the effect of control and cost horizons on stability. Similar studies appeared in the literature on generalized predictive control (GPC) (Ydstie, 1984; Peterka, 1984; De Keyser and Van Cauwenberghe, 1985; Clarke, Mohtadi, and Tuffs, 1987) that arose to address deficiencies in minimum variance control. Interestingly enough, earlier research on RHC (Kleinman, 1970; Thomas, 1975; Kwon and Pearson, 1977) had shown indirectly that the imposition of a terminal equality constraint in the finite horizon optimal control problem ensured closed-loop stability for linear unconstrained

systems. That a terminal equality constraint had an equally beneficial effect for constrained nonlinear discrete time systems was shown by Keerthi and Gilbert (1988) and for constrained nonlinear continuous time systems by Chen and Shaw (1982) and Mayne and Michalska (1990). In each of these papers, Lyapunov stability theory was employed in contrast to the then current literature on MPC and GPC.

The next advance showed that incorporation of a suitable terminal cost and terminal constraint in the finite horizon optimal control problem ensured closed-loop stability; the terminal constraint set is required to be control invariant, and the terminal cost function is required to be a local CLF. Perhaps the earliest proposal in this direction is the brief paper by Sznaier and Damborg (1987) for linear systems with polytopic constraints; in this prescient paper the terminal cost is chosen to be the value function for the *unconstrained* infinite horizon optimal control problem, and the terminal constraint set is the maximal constraint admissible set (Gilbert and Tan, 1991) for the optimal controlled system.[6] A suitable terminal cost and terminal constraint set for constrained nonlinear continuous time systems was proposed in Michalska and Mayne (1993) in the context of dual-mode MPC. In a paper that has had considerable impact, Chen and Allgöwer (1998) showed that similar ingredients may be employed to stabilize constrained nonlinear continuous time systems when conventional MPC is employed. Related results were obtained by Parisini and Zoppoli (1995), and De Nicolao, Magni, and Scattolini (1996).

Stability proofs for the form of MPC proposed, but not analyzed, in Sznaier and Damborg (1987) were finally provided by Chmielewski and Manousiouthakis (1996) and Scokaert and Rawlings (1998). These papers also showed that optimal control for the *infinite* horizon constrained optimal control problem for a specified initial state is achieved if the horizon is chosen sufficiently long. A terminal constraint is not required if a global, rather than a local, CLF is available for use as a terminal cost function. Thus, for the case when the system being controlled is linear and stable, and subject to a convex control constraint, Rawlings and Muske (1993) showed, in a paper that raised considerable interest, that closed-loop stability may be obtained if the terminal constraint is omitted and the infinite horizon cost using zero control is employed as the terminal cost. The resultant terminal cost is a global

[6] If the optimal infinite horizon controlled system is described by $x^+ = A_K x$ and if the constraints are $u \in \mathbb{U}$ and $x \in \mathbb{X}$, then the maximal constraint admissible set is $\{x \mid A_K^i x \in \mathbb{X},\ K A_K^i x \in \mathbb{U}\ \forall i \in \mathbb{I}_{\geq 0}\}$.

CLF.

The basic principles ensuring closed-loop stability in these and many other papers including (De Nicolao, Magni, and Scattolini, 1998), and (Mayne, 2000) were distilled and formulated as "stability axioms" in the review paper (Mayne et al., 2000); they appear as Assumptions 2.2, 2.3, and 2.14 in this chapter. These assumptions provide sufficient conditions for closed-loop stability for a given horizon. There is an alternative literature that shows that closed-loop stability may often be achieved if the horizon is chosen to be sufficiently long. Contributions in this direction include (Primbs and Nevistić, 2000), (Jadbabaie, Yu, and Hauser, 2001), as well as (Parisini and Zoppoli, 1995; Chmielewski and Manousiouthakis, 1996; Scokaert and Rawlings, 1998) already mentioned. An advantage of this approach is that it avoids addition of an explicit terminal constraint, although this may be avoided by alternative means as shown in Section 2.6. A significant development of this approach (Grüne and Pannek, 2017) gives a comprehensive investigation and extension of the conditions that ensure recursive feasibility and stability of MPC that does not have a terminal constraint. On the other hand, it has been shown (Mayne, 2013) that an explicit or implicit terminal constraint is necessary if positive invariance and the nested property $\mathcal{X}_{j+1} \supset \mathcal{X}_j$, $j \in \mathbb{I}_{\geq 0}$ of the feasible sets are required; the nested property ensures recursive feasibility.

Recently several researchers (Limon, Alvarado, Alamo, and Camacho, 2008, 2010; Fagiano and Teel, 2012; Falugi and Mayne, 2013a; Müller and Allgöwer, 2014; Mayne and Falugi, 2016) have shown how to extend the region of attraction $\mathcal{X}_N$, and how to solve the related problem of tracking a randomly varying reference—thereby alleviating the disadvantage caused by the reduction in the region of attraction due to the imposition of a terminal constraint. Attention has also been given to the problem of tracking a periodic reference using model predictive control (Limon et al., 2012; Falugi and Mayne, 2013b; Rawlings and Risbeck, 2017).

Regarding the analysis of nonpositive stage costs in Section 2.4.4, Grimm, Messina, Tuna, and Teel (2005) use a storage function like $\Lambda(\cdot)$ to compensate for a semidefinite stage cost. Cai and Teel (2008) give a discrete time converse theorem for IOSS for all $\mathbb{R}^n$. Allan and Rawlings (2018) give a converse theorem for IOSS on closed positive invariant sets and provide a lemma for changing the supply rate function.

Suboptimal MPC based on a warm start was proposed and analyzed by Scokaert et al. (1999). Pannocchia et al. (2011) establish that this

form of suboptimal MPC is robustly stable for systems without state constraints if the terminal constraint is replaced with an enlarged terminal penalty. As noted by Yu, Reble, Chen, and Allgöwer (2014), however, the assumptions used for these results are strong enough to imply that the *optimal* value function is *continuous*. Allan et al. (2017) establish robustness for systems with discontinuous feedback and discontinuous optimal value function.

Lazar and Heemels (2009) analyze robustness of suboptimal MPC with respect to state disturbances under the condition that the suboptimal controller is able to find a solution within a specific degree of suboptimality from the global solution. Roset, Heemels, Lazar, and Nijmeijer (2008), show how to extend the analysis to treat measurement disturbances as well as state disturbances. Because this type of suboptimal MPC is defined in terms of the globally optimal cost, its implementation requires, in principle, global solvers.

Economic MPC was introduced in Rawlings and Amrit (2009), but designing process controllers other than MPC to optimize process economics has been a part of industrial practice for a long time. When using an economic (as opposed to tracking) stage cost, cost inequalities and conditions for asymptotic stability have been established for time-invariant systems with a steady state (Diehl et al., 2011; Amrit, Rawlings, and Angeli, 2011; Angeli et al., 2012; Ellis, Durand, and Christofides, 2014). Such results have been extended in Zanon, Gros, and Diehl (2013) to the time-varying periodic case under the assumptions of a linear storage function and Lipschitz continuity of the model and stage cost; Rawlings and Risbeck (2017) require only continuity of the model and stage cost, and allow a more general form for the storage function.

For the case of a time-invariant system with optimal periodic operation, convergence to the optimal periodic solution can be shown using similar notions of dissipativity (Müller and Grüne, 2015); but this case is different than the case treated by Rawlings and Risbeck (2017) because clock time does not appear. In Müller, Angeli, and Allgöwer (2015), the authors establish the interesting result that a certain dissipativity condition is also *necessary* for asymptotic stability. For periodic processes, stability has been investigated by converting to deviation variables (Huang, Harinath, and Biegler, 2011; Rawlings and Risbeck, 2017).

Various results on stability of MPC with discrete actuators have appeared in the literature. In Bemporad and Morari (1999), convergence

to the origin is shown for mixed-logical-dynamical systems based on certain positive definite restrictions on the stage cost, although Lyapunov stability is not explicitly shown. For piecewise affine systems, Baotic, Christophersen, and Morari (2006) establish asymptotic stability for an infinite horizon control law via Lyapunov function arguments. In Di Cairano, Heemels, Lazar, and Bemporad (2014), a hybrid Lyapunov function is directly embedded within the optimal control problem, enforcing cost decrease as a hard constraint and ensuring closed-loop asymptotic stability. Alternatively, practical stability (i.e., boundedness) can often be shown by treating discretization of inputs as a disturbance and deriving error bounds with respect to the relaxed continuous-actuator system (Quevedo, Goodwin, and De Doná, 2004; Aguilera and Quevedo, 2013; Kobayshi, Shein, and Hiraishi, 2014). Finally, Picasso, Pancanti, Bemporad, and Bicchi (2003) shows asymptotic stability for open-loop stable linear systems with only practical stability for open-loop unstable systems. All of these results are concerned with stability of a steady state.

The approach presented in this chapter, which shows that current MPC asymptotic stability theorems based on Lyapunov functions also cover general nonlinear systems with mixed continuous/discrete actuators, was developed by Rawlings and Risbeck (2017).

2.12 Exercises

Exercise 2.1: Discontinuous MPC

In Example 2.8, compute $\mathcal{U}_3(x)$, $V_3^0(x)$, and $\kappa_3(x)$ at a few points on the unit circle.

Exercise 2.2: Boundedness of discrete time model

Consider the continuous time differential equation $\dot{x} = f_c(x,u)$, and its discrete time counterpart $x^+ = f(x,u)$. Suppose that $f_c(\cdot)$ is continuous, and there exists a positive constant c such that

$$|f_c(x',u) - f_c(x,u)| \leq c\,|x' - x| \qquad \forall x, x' \in \mathbb{R}^n, u \in \mathbb{U}$$

Show that $f(\cdot)$ is bounded on bounded sets. Moreover, if $\mathbb{U}$ is bounded, show that $f^{-1}(\cdot)$ is bounded on bounded sets.

Exercise 2.3: Destabilization with state constraints

Consider a state feedback regulation problem with the origin as the setpoint (Muske and Rawlings, 1993). Let the system be

$$A = \begin{bmatrix} 4/3 & -2/3 \\ 1 & 0 \end{bmatrix} \qquad B = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \qquad C = [-2/3 \;\; 1]$$

and the controller objective function tuning matrices be

$$Q = I \qquad R = I \qquad N = 5$$

(a) Plot the unconstrained regulator performance starting from initial condition $x(0) = \begin{bmatrix} 3 & 3 \end{bmatrix}'$.

(b) Add the output constraint $y(k) \leq 0.5$. Plot the response of the constrained regulator (both input and output). Is this regulator stabilizing? Can you modify the tuning parameters Q, R to affect stability as in Section 1.3.4?

(c) Change the output constraint to $y(k) \leq 1 + \epsilon, \epsilon > 0$. Plot the closed-loop response for a variety of ϵ. Are any of these regulators destabilizing?

(d) Set the output constraint back to $y(k) \leq 0.5$ and add the terminal constraint $x(N) = 0$. What is the solution to the regulator problem in this case? Increase the horizon N. Does this problem eventually go away?

Exercise 2.4: Computing the projection of $\mathbb{Z}$ onto $\mathcal{X}_N$

Given a polytope

$$\mathbb{Z} := \{(x,u) \in \mathbb{R}^n \times \mathbb{R}^m \mid Gx + Hu \leq \psi\}$$

write an Octave or MATLAB program to determine $\mathcal{X}$, the projection of $\mathbb{Z}$ onto $\mathbb{R}^n$

$$\mathcal{X} = \{x \in \mathbb{R}^n \mid \exists u \in \mathbb{R}^m \text{ such that } (x,u) \in \mathbb{Z}\}$$

Use algorithms 3.1 and 3.2 in Keerthi and Gilbert (1987).

To check your program, consider a system

$$x^+ = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$$

subject to the constraints $\mathbb{X} = \{x \mid x_1 \leq 2\}$ and $\mathbb{U} = \{u \mid -1 \leq u \leq 1\}$. Consider the MPC problem with $N = 2$, $\mathbf{u} = (u(0), u(1))$, and the set $\mathbb{Z}$ given by

$$\mathbb{Z} = \{(x, \mathbf{u}) \mid x, \phi(1; x, \mathbf{u}), \phi(2; x, \mathbf{u}) \in \mathbb{X} \text{ and } u(0), u(1) \in \mathbb{U}\}$$

Verify that the set

$$\mathcal{X}_2 := \{x \in \mathbb{R}^2 \mid \exists \mathbf{u} \in \mathbb{R}^2 \text{ such that } (x, \mathbf{u}) \in \mathbb{Z}\}$$

is given by

$$\mathcal{X}_2 = \{x \in \mathbb{R}^2 \mid Px \leq p\} \qquad P = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 2 \end{bmatrix} \qquad p = \begin{bmatrix} 2 \\ 2 \\ 3 \end{bmatrix}$$

Exercise 2.5: Computing the maximal output admissible set

Write an Octave or MATLAB program to determine the maximal constraint admissible set for the system $x^+ = Fx, y = Hx$ subject to the hard constraint $y \in Y$ in which $Y = \{y \mid Ey \leq e\}$. Use algorithm 3.2 in Gilbert and Tan (1991).

To check your program, verify for the system

$$F = \begin{bmatrix} 0.9 & 1 \\ 0 & 0.09 \end{bmatrix} \qquad H = \begin{bmatrix} 1 & 1 \end{bmatrix}$$

subject to the constraint $Y = \{y \mid -1 \leq y \leq 1\}$, and that the maximal output admissible set is given by

$$O_\infty = \{x \in \mathbb{R}^2 \mid Ax \leq b\} \qquad A = \begin{bmatrix} 1 & 1 \\ -1 & -1 \\ 0.9 & 1.09 \\ -0.9 & -1.09 \\ 0.81 & 0.9981 \\ -0.81 & -0.9981 \end{bmatrix} \qquad b = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

Show that t^*, the smallest integer t such that $O_t = O_\infty$ satisfies $t^* = 2$.

What happens to t^* as F_{22} increases and approaches one? What do you conclude for the case $F_{22} \geq 1$?

Exercise 2.6: Terminal constraint and region of attraction

Consider the system

$$x^+ = Ax + Bu$$

subject to the constraints

$$x \in \mathbb{X} \qquad u \in \mathbb{U}$$

in which

$$A = \begin{bmatrix} 2 & 1 \\ 0 & 2 \end{bmatrix} \qquad B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\mathbb{X} = \{x \in \mathbb{R}^2 \mid x_1 \leq 5\} \qquad \mathbb{U} = \{u \in \mathbb{R}^2 \mid -\mathbf{1} \leq u \leq \mathbf{1}\}$$

and $\mathbf{1} \in \mathbb{R}^2$ is a vector of ones. The MPC cost function is

$$V_N(x, \mathbf{u}) = \sum_{i=0}^{N-1} \ell(x(i), u(i)) + V_f(x(N))$$

in which

$$\ell(x, u) = (1/2)(\,|x|_Q^2 + |u|^2\,) \qquad Q = \begin{bmatrix} \alpha & 0 \\ 0 & \alpha \end{bmatrix}$$

and $V_f(\cdot)$ is the terminal penalty on the final state.

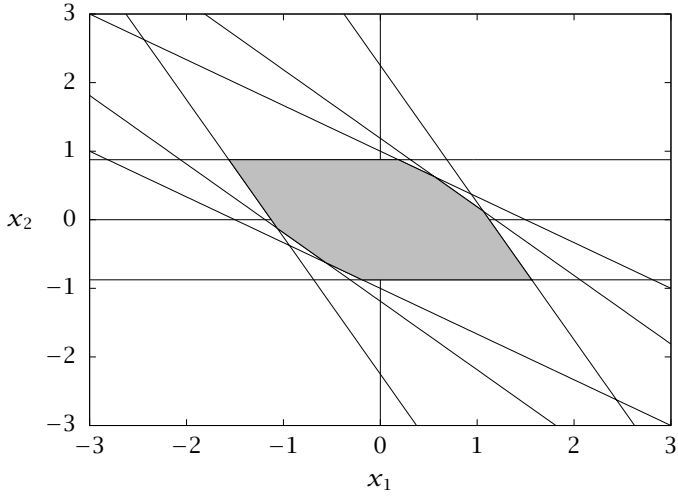


Figure 2.10: Region of attraction (shaded region) for constrained MPC controller of Exercise 2.6.

(a) Implement unconstrained MPC with no terminal cost ($V_f(\cdot) = 0$) for a few values of α. Choose a value of α for which the resultant closed loop is unstable. Try $N = 3$.

(b) Implement constrained MPC with no terminal cost or terminal constraint for the value of α obtained in the previous part. Is the resultant closed loop stable or unstable?

(c) Implement constrained MPC with terminal equality constraint $x(N) = 0$ for the same value of α. Find the region of attraction for the constrained MPC controller using the projection algorithm from Exercise 2.4. The result should resemble Figure 2.10.

Exercise 2.7: Infinite horizon cost to go as terminal penalty

Consider the system

$$x^+ = Ax + Bu$$

subject to the constraints

$$x \in \mathbb{X} \qquad u \in \mathbb{U}$$

in which

$$A = \begin{bmatrix} 2 & 1 \\ 0 & 2 \end{bmatrix} \qquad B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

and

$$\mathbb{X} = \{x \in \mathbb{R}^2 \mid -5 \le x_1 \le 5\} \qquad \mathbb{U} = \{u \in \mathbb{R}^2 \mid -\mathbf{1} \le u \le \mathbf{1}\}$$

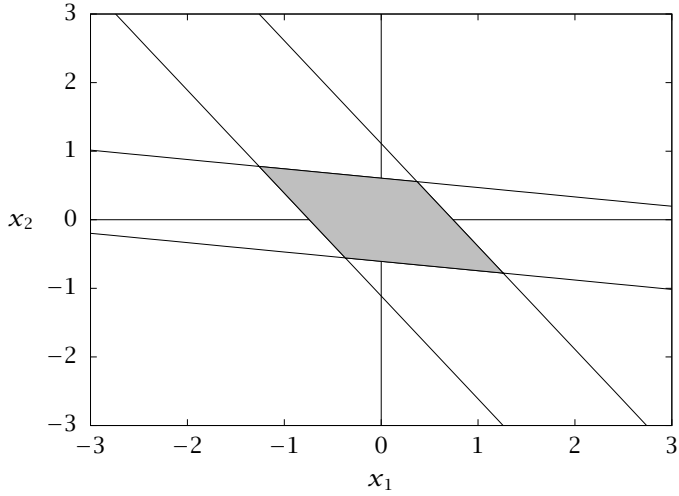


Figure 2.11: The region $\mathbb{X}_f$, in which the unconstrained LQR control law is feasible for Exercise 2.7.

The cost is

$$V_N(x, \mathbf{u}) := \sum_{i=0}^{N-1} \ell(x(i), u(i)) + V_f(x(N))$$

in which

$$\ell(x, u) = (1/2)(|x|_Q^2 + |u|^2) \qquad Q = \begin{bmatrix} \alpha & 0 \\ 0 & \alpha \end{bmatrix}$$

and $V_f(\cdot)$ is the terminal penalty on the final state and $\mathbf{1} \in \mathbb{R}^2$ is a vector of all ones. Use $\alpha = 10^{-5}$ and $N = 3$ and terminal cost $V_f(x) = (1/2)x'\Pi x$ where Π is the solution to the steady-state Riccati equation.

(a) Compute the infinite horizon optimal cost and control law for the unconstrained system.

(b) Find the region $\mathbb{X}_f$, the maximal constraint admissible set using the algorithm in Exercise 2.5 for the system $x^+ = (A+BK)x$ with constraints $x \in \mathbb{X}$ and $Kx \in \mathbb{U}$. You should obtain the region shown in Figure 2.11.

(c) Add a terminal constraint $x(N) \in \mathbb{X}_f$ and implement constrained MPC. Find $\mathcal{X}_N$, the region of attraction for the MPC problem with $V_f(\cdot)$ as the terminal cost and $x(N) \in \mathbb{X}_f$ as the terminal constraint. Contrast it with the region of attraction for the MPC problem in Exercise 2.6 with a terminal constraint $x(N) = 0$.

(d) Estimate $\bar{X}_N$, the set of initial states for which the MPC control sequence for horizon N is equal to the MPC control sequence for an infinite horizon.
Hint: $x \in \bar{X}_N$ if $x^0(N; x) \in \text{int}(\mathbb{X}_f)$. Why?

Exercise 2.8: Terminal penalty with and without terminal constraint

Consider the system

$$x^+ = Ax + Bu$$

subject to the constraints

$$x \in \mathbb{X} \qquad u \in \mathbb{U}$$

in which

$$A = \begin{bmatrix} 2 & 1 \\ 0 & 2 \end{bmatrix} \qquad B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

and

$$\mathbb{X} = \{x \in \mathbb{R}^2 \mid -15 \le x_1 \le 15\} \qquad \mathbb{U} = \{u \in \mathbb{R}^2 \mid -5 \cdot \mathbf{1} \le u \le 5 \cdot \mathbf{1}\}$$

The cost is

$$V_N(x, \mathbf{u}) = \sum_{i=0}^{N-1} \ell(x(i), u(i)) + V_f(x(N))$$

in which

$$\ell(x, u) = (1/2)(|x|_Q^2 + |u|)^2 \qquad Q = \begin{bmatrix} \alpha & 0 \\ 0 & \alpha \end{bmatrix}$$

$V_f(\cdot)$ is the terminal penalty on the final state, and $\mathbf{1} \in \mathbb{R}^2$ is a vector of ones.

Use $\alpha = 10^{-5}$ and $N = 3$ and terminal cost $V_f(x) = (1/2)x'\Pi x$ where $V_f(\cdot)$ is the infinite horizon optimal cost for the unconstrained problem.

(a) Add a terminal constraint $x(N) \in \mathbb{X}_f$, in which $\mathbb{X}_f$ is the maximal constraint admissible set for the system $x^+ = (A + BK)x$ and K is the optimal controller gain for the unconstrained problem. Using the code developed in Exercise 2.7, estimate $\mathcal{X}_N$, the region of attraction for the MPC problem with this terminal constraint and terminal cost. Also estimate $\bar{X}_N$, the region for which the MPC control sequence for horizon N is equal to the MPC control sequence for infinite horizon. Your results should resemble Figure 2.12.

(b) Remove the terminal constraint and *estimate* the domain of attraction $\hat{X}_N$ (by simulation). Compare this $\hat{X}_N$ with $\mathcal{X}_N$ and $\bar{X}_N$ obtained previously.

(c) Change the terminal cost to $V_f(x) = (3/2)x'\Pi x$ and repeat the previous part.

Exercise 2.9: Decreasing property for the time-varying case

Consider the time-varying optimal control problem specified in 2.4.5. Suppose that $V_f(\cdot)$ and $\mathbb{X}_f$ satisfy the basic stability assumption, Assumption 2.33 Prove that the value function $V_N^0(\cdot)$ has the decreasing property

$$V_N^0((x, i)^+) \le V_N^0(x, i) - \ell(x, i, \kappa_N(x, i))$$

for all $(x, i) \in \mathbb{X}\mathbb{I}$.

Exercise 2.10: Terminal cost bound for the time-varying case

Refer to Section 2.4.5. Prove that the value function $V_N^0(\cdot)$ satisfies

$$V_N^0(x, i) \le V_f(x, i + N) \quad \forall (x, i) \in \mathbb{X}_f(i + N) \times \mathbb{I}_{\ge 0}$$

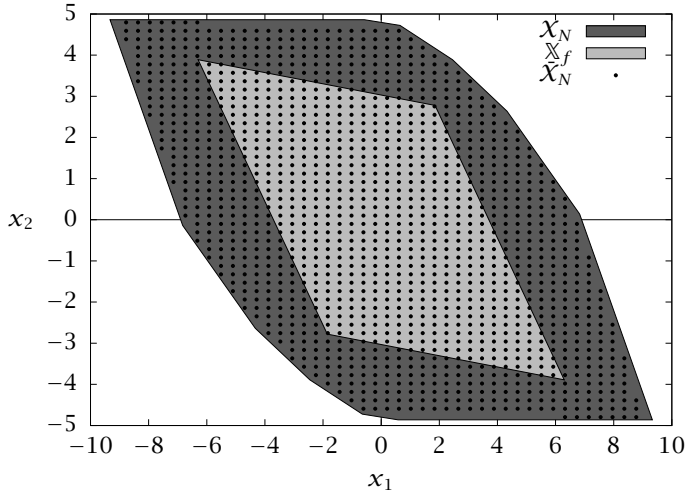


Figure 2.12: The region of attraction for terminal constraint $x(N) \in \mathbb{X}_f$ and terminal penalty $V_f(x) = (1/2)x'\Pi x$ and the estimate of $\bar{\mathcal{X}}_N$ for Exercise 2.8.

Exercise 2.11: Modification of terminal cost

Refer to Section 2.6. Show that the pair $(\beta V_f(\cdot), \mathbb{X}_f)$ satisfies Assumption 2.14 if $(V_f(\cdot), \mathbb{X}_f)$ satisfies this assumptions, $\beta \geq 1$, and $\ell(\cdot)$ satisfies Assumption 2.2.

Exercise 2.12: A Lyapunov theorem for asymptotic stability

Prove the asymptotic stability result for Lyapunov functions.

Theorem 2.59 (Lyapunov theorem for asymptotic stability)**.** *Given the dynamic system*

$$x^+ = f(x) \qquad 0 = f(0)$$

The origin is asymptotically stable if there exist $\mathcal{K}$ functions α, β, γ, and $r > 0$ such that Lyapunov function V satisfies for $x \in r\mathcal{B}$

$$\alpha(|x|) \leq V(x) \leq \beta(|x|)$$
$$V(f(x)) - V(x) \leq -\gamma(|x|)$$

Exercise 2.13: An MPC stability result

Given the following nonlinear model and objective function

$$x^+ = f(x,u), \qquad 0 = f(0,0)$$
$$x(0) = x$$
$$V_N(x,\mathbf{u}) = \sum_{k=0}^{N-1} \ell(x(k),u(k))$$

Consider the terminal constraint MPC regulator

$$\min_{\mathbf{u}} V_N(x, \mathbf{u})$$

subject to

$$x^+ = f(x,u) \qquad x(0) = x \qquad x(N) = 0$$

and denote the first move in the optimal control sequence as $u^0(x)$. Given the closed-loop system

$$x^+ = f(x, u^0(x))$$

(a) Prove that the origin is asymptotically stable for the closed-loop system. State the cost function assumption and controllability assumption required so that the control problem is feasible for some set of defined initial conditions.

(b) What assumptions about the cost function $\ell(x,u)$ are required to strengthen the controller so that the origin is exponentially stable for the closed-loop system? How does the controllability assumption change for this case?

Exercise 2.14: Stability using observability instead of IOSS

Assume that the system $x^+ = f(x,u)$, $y = h(x)$ is ℓ-observable, i.e., there exists a $\alpha \in \mathcal{K}$ and an integer $N_o \geq 1$ such that

$$\sum_{j=0}^{N_o-1} \ell(y(i), u(i)) \geq \alpha(|x|)$$

for all x and all $\mathbf{u}$; here $x(i) := \phi(i;x,\mathbf{u})$ and $y(i) := h(x(i))$. Prove the result given in Section 2.4.4 that the origin is asymptotically stable for the closed-loop system $x^+ = f(x, \kappa_N(x))$ using the assumption that $x^+ = f(x,u)$, $y = h(x)$ is ℓ-observable rather than IOSS. Assume that $N \geq N_o$.

Exercise 2.15: Input/output-to-state stability (IOSS) and convergence

Proposition 2.60 (Convergence of state under IOSS)**.** *Assume that the system $x^+ = f(x, u)$, $y = h(x)$ is IOSS and that $u(i) \to 0$ and $y(i) \to 0$ as $i \to \infty$. Then $x(i) = \phi(i;x, \mathbf{u}) \to 0$ as $i \to \infty$ for any initial state x.*

Prove Proposition 2.60. Hint: consider the solution at time $k + l$ using the state at time k as the initial state.

Exercise 2.16: Equality for quadratic functions

Prove the following result which is useful for analyzing the unreachable setpoint problem.

Lemma 2.61 (An equality for quadratic functions)**.** *Let $\mathbb{X}$ be a nonempty compact subset of $\mathbb{R}^n$, and let $\ell(\cdot)$ be a strictly convex quadratic function on $\mathbb{X}$ defined by $\ell(x) := (1/2)x'Qx + q'x + c$, $Q > 0$. Consider a sequence $(x(i))_{i \in \mathbb{I}_{1:P}}$ with mean $\bar{x}_P := (1/P)\sum_{i=1}^P x(i)$. Then the following holds*

$$\sum_{i=1}^P \ell(x(i)) = (1/2)\sum_{i=1}^P |x(i) - \bar{x}_P|_Q^2 + P\ell(\bar{x}_P)$$

It follows from this lemma that $\ell(\bar{x}_P) \leq (1/P)\sum_{i=1}^P \ell(x(i))$, which is Jensen's inequality for the special case of a quadratic function.

Exercise 2.17: Unreachable setpoint MPC and evolution in a compact set

Prove the following lemma, which is useful for analyzing the stability of MPC with an unreachable setpoint.

Lemma 2.62 (Evolution in a compact set)**.** *Suppose $x(0) = x$ lies in the set $\mathcal{X}_N$. Then the state trajectory $(x(i))$ where, for each i, $x(i) = \phi_f(i;x)$ of the controlled system $x^+ = f(x)$ evolves in a compact set.*

Exercise 2.18: MPC and multivariable, constrained systems

Consider a two-input, two-output process with the following transfer function

$$G(s) = \begin{bmatrix} \dfrac{2}{10s+1} & \dfrac{2}{s+1} \\ \dfrac{1}{s+1} & -\dfrac{4}{s+1} \end{bmatrix}$$

(a) Consider a unit setpoint change in the first output. Choose a reasonable sample time, Δ. Simulate the behavior of an offset-free discrete time MPC controller with $Q = I, S = I$ and large N.

(b) Add the constraint $-1 \le u(k) \le 1$ and simulate the response.

(c) Add the constraint $-0.1 \le \Delta u/\Delta \le 0.1$ and simulate the response.

(d) Add significant noise to both output measurements (make the standard deviation in each output about 0.1). Retune the MPC controller to obtain good performance. Describe which controller parameters you changed and why.

Exercise 2.19: LQR versus LAR

We are now all experts on the linear quadratic regulator (LQR), which employs a linear model and quadratic performance measure. Let's consider the case of a linear model but absolute value performance measure, which we call the linear absolute regulator (LAR)[7]

$$\min_{\mathbf{u}} \sum_{k=0}^{N-1} \left(q\,|x(k)| + r\,|u(k)|\right) + q(N)\,|x(N)|$$

For simplicity consider the following one-step controller, in which u and x are *scalars*

$$\min_{u(0)} V(x(0), u(0)) = |x(1)| + |u(0)|$$

subject to

$$x(1) = Ax(0) + Bu(0)$$

Draw a sketch of $x(1)$ versus $u(0)$ (recall $x(0)$ is a known parameter) and show the x-axis and y-axis intercepts on your plot. Now draw a sketch of $V(x(0), u(0))$ versus $u(0)$ in order to see what kind of optimization problem you are solving. You may want to plot both terms in the objective function individually and then add them together to make your V plot. Label on your plot the places where the cost function V

[7] Laplace would love us for making this choice, but Gauss would not be happy.

suffers discontinuities in slope. Where is the solution in your sketch? Does it exist for all $A, B, x(0)$? Is it unique for all $A, B, x(0)$?

The motivation for this problem is to change the quadratic program (QP) of the LQR to a linear program (LP) in the LAR, because the computational burden for LPs is often smaller than QPs. The absolute value terms can be converted into linear terms with the introduction of slack variables.

Exercise 2.20: Unreachable setpoints in constrained versus unconstrained linear systems

Consider the linear system with input constraint

$$x^+ = Ax + Bu \qquad u \in \mathbb{U}$$

We examine here both unconstrained systems in which $\mathbb{U} = \mathbb{R}^m$ and constrained systems in which $\mathbb{U} \subset \mathbb{R}^m$ is a convex polyhedron. Consider the stage cost defined in terms of setpoints for state and input x_{sp}, u_{sp}

$$\ell(x,u) = (1/2)\left(\left| x - x_{\text{sp}} \right|_Q^2 + \left| u - u_{\text{sp}} \right|_R^2 \right)$$

in which we assume for simplicity that $Q, R > 0$. For the setpoint to be unreachable in an unconstrained problem, the setpoint must be *inconsistent*, i.e., not a steady state of the system, or

$$x_{\text{sp}} \neq Ax_{\text{sp}} + Bu_{\text{sp}}$$

Consider also using the stage cost centered at the optimal steady state (x_s, u_s)

$$\ell_s(x,u) = (1/2)\left(|x - x_s|_Q^2 + |u - u_s|_R^2 \right)$$

The optimal steady state satisfies

$$(x_s, u_s) = \arg\min_{x,u} \ell(x,u)$$

subject to

$$\begin{bmatrix} I - A & -B \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix} = 0 \qquad u \in \mathbb{U}$$

Figure 2.13 depicts an inconsistent setpoint, and the optimal steady state for unconstrained and constrained systems.

(a) For unconstrained systems, show that optimizing the cost function with terminal constraint

$$V(x, \mathbf{u}) := \sum_{k=0}^{N-1} \ell(x(k), u(k))$$

subject to

$$x^+ = Ax + Bu \qquad x(0) = x \qquad x(N) = x_s$$

gives the same solution as optimizing the cost function

$$V_s(x, \mathbf{u}) := \sum_{k=0}^{N-1} \ell_s(x(k), u(k))$$

subject to the same model constraint, initial condition, and terminal constraint.

Therefore, there is no reason to consider the unreachable setpoint problem further for an *unconstrained* linear system. Shifting the stage cost from $\ell(x,u)$ to $\ell_s(x,u)$ provides identical control behavior and is simpler to analyze.

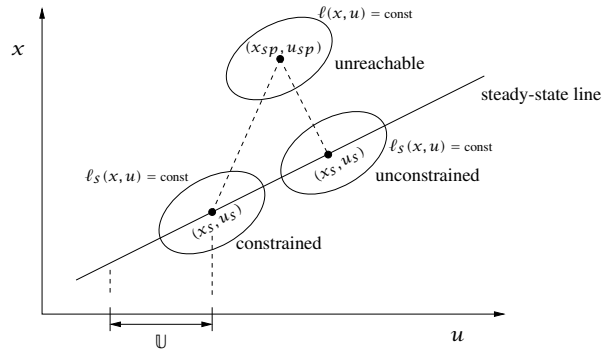


Figure 2.13: Inconsistent setpoint $(x_{\text{sp}}, u_{\text{sp}})$, unreachable stage cost $\ell(x,u)$, and optimal steady states (x_s, u_s), and stage costs $\ell_s(x,u)$ for constrained and unconstrained systems.

Hint. First define a third stage cost $l(x,u) = \ell(x,u) - \lambda'((I-A)x - Bu)$, and show, for any λ, optimizing with $l(x,u)$ as stage cost is the same as optimizing using $\ell(x,u)$ as stage cost. Then set $\lambda = \lambda_s$, the optimal Lagrange multiplier of the *steady-state* optimization problem.

(b) For *constrained* systems, provide a simple example that shows optimizing the cost function $V(x,\mathbf{u})$ subject to

$$x^+ = Ax + Bu \qquad x(0) = x \qquad x(N) = x_s \qquad u(k) \in \mathbb{U} \quad \text{for all } k \in \mathbb{I}_{0:N-1}$$

does *not* give the same solution as optimizing the cost function $V_s(x,\mathbf{u})$ subject to the same constraints. For *constrained* linear systems, these problems are different and optimizing the unreachable stage cost provides a new design opportunity.

Exercise 2.21: Filing for patent

An excited graduate student shows up at your office. He begins, "Look, I have discovered a great money-making scheme using MPC." You ask him to tell you about it. "Well," he says, "you told us in class that the optimal steady state is asymptotically stable even if you use the stage cost measuring distance from the unreachable setpoint, right?" You reply, "Yes, that's what I said." He continues, "OK, well look at this little sketch I drew," and he shows you a picture like Figure 2.14. "So imagine I use the infinite horizon cost function so the open-loop and closed-loop trajectories are identical. If the best steady state is asymptotically stable, then the stage cost asymptotically approaches $\ell(x_s,u_s)$, right?" You reply, "I guess that looks right." He then says, "OK, well if I look at the optimal cost using state x at time k and state x^+ at time $k+1$, by the principle of optimality I get the usual cost decrease"

$$V^0(x^+) \le V^0(x) - \ell(x, u^0(x)) \tag{2.39}$$

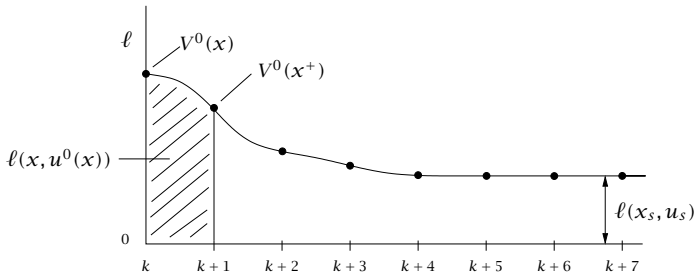


Figure 2.14: Stage cost versus time for the case of unreachable setpoint. The cost $V^0(x(k))$ is the area under the curve to the right of time k.

You interrupt, "Wait, these $V^0(\cdot)$ costs are not bounded in this case!" Unfazed, the student replies, "Yeah, I realize that, but this sketch is basically correct regardless. Say we just make the horizon *really long*; then the costs are all finite and this equation becomes closer and closer to being true as we make the horizon longer and longer." You start to feel a little queasy at this point. The student continues, "OK, so if this inequality basically holds, $V^0(x(k))$ is decreasing with k along the closed-loop trajectory, it is bounded below for all k, it converges, and, therefore, $\ell(x(k), u^0(x(k)))$ goes to zero as k goes to ∞." You definitely don't like where this is heading, and the student finishes with, "But $\ell(x, u) = 0$ implies $x = x_{\text{sp}}$ and $u = u_{\text{sp}}$, and the setpoint is *supposed* to be unreachable. But I have proven that infinite horizon MPC can reach an *unreachable* setpoint. We should patent this!"

How do you respond to this student? Here are some issues to consider.

1. Does the principle of optimality break down in the unreachable setpoint case?
2. Are the open-loop and closed-loop trajectories identical in the limit of an infinite horizon controller with an unreachable setpoint?
3. Does inequality (2.39) hold as $N \to \infty$? If so, how can you put it on solid footing? If not, why not, and with what do you replace it?
4. Do you file for patent?

Exercise 2.22: Stabilizable with small controls

Consider a time-varying system $x(i+1) = f(x, u, i)$ with stage cost $\ell(x, u, i)$ and terminal cost $V_f(x, i)$ satisfying Assumptions 2.25, 2.26, and 2.33. Suppose further that functions $f(\cdot)$ and $\ell(\cdot)$ are uniformly bounded by $\mathcal{K}_\infty$ functions α_f and α_ℓ, i.e.,

$$|f(x, u, i)| \leq \alpha_{fx}(|x|) + \alpha_{fu}(|u|)$$
$$\ell(x, u, i) \leq \alpha_{\ell x}(|x|) + \alpha_{\ell u}(|u|)$$

for all $i \in \mathbb{I}_{\geq 0}$. Prove that if there exists a $\mathcal{K}_\infty$ function $\gamma(\cdot)$ such that for each $x \in \mathcal{X}_N(i)$, there exists $\mathbf{u} \in \mathcal{U}_N(x, i)$ with $|\mathbf{u}| \leq \gamma(|x|)$, then there exists a $\mathcal{K}_\infty$ function $\alpha(\cdot)$ such that

$$V^0(x, i) \leq \alpha(|x|)$$

for all $i \in \mathbb{I}_{\geq 0}$ and $x \in \mathcal{X}_N(i)$.

Hint: the following inequality may prove useful: for any $\mathcal{K}_\infty$ function α (see (B.1))

$$\alpha(s_1 + s_2 + \cdots + s_N) \leq \alpha(Ns_1) + \alpha(Ns_2) + \cdots + \alpha(Ns_N)$$

Exercise 2.23: Power lifting

Consider a stabilizable T-periodic linear system

$$x(i+1) = A(i)x(i) + B(i)u(i)$$

with positive definite stage cost

$$\ell(x,u,i) := \frac{1}{2}\left(x'Q(i)x + u'R(i)u\right)$$

Suppose there exist periodic control laws $K(i)$ and cost matrices $P(i)$ satisfying the periodic Riccati equation

$$\begin{aligned} P(i) &= Q(i) + A(i)'P(i+1)\left(A(i) + B(i)K(i)\right) \\ K(i) &= -\left(B(i)'P(i+1)B(i) + R(i)\right)^{-1} B(i)'P(i+1)A(i) \end{aligned}$$

Show that the control law $\mathbf{K} := \operatorname{diag}(K(0), \ldots, K(T-1))$ and cost $\mathbf{P} := \operatorname{diag}(P(0), \ldots, P(T-1))$ satisfy the Riccati equation for the time-invariant lifted system

$$\mathbf{A} := \begin{bmatrix} 0 & 0 & \cdots & 0 & A(T-1) \\ A(0) & 0 & \cdots & 0 & 0 \\ 0 & A(1) & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & A(T-2) & 0 \end{bmatrix}$$

$$\mathbf{B} := \begin{bmatrix} 0 & 0 & \cdots & 0 & B(T-1) \\ B(0) & 0 & \cdots & 0 & 0 \\ 0 & B(1) & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & B(T-2) & 0 \end{bmatrix}$$

$$\mathbf{Q} := \operatorname{diag}(Q(0), \ldots, Q(T-1))$$

$$\mathbf{R} := \operatorname{diag}(R(0), \ldots, R(T-1))$$

By uniqueness of solutions to the Riccati equation, this system can be used to synthesize control laws for periodic systems.

Exercise 2.24: Feasible warm start in $\mathbb{X}_f$

Establish Proposition 2.42, which states that for any $x \in \mathbb{X}_f$, the following warm start is feasible

$$\widetilde{\mathbf{u}}_f(x) := (\kappa_f(x), \kappa_f(f(x, \kappa_f(x))), \ldots) \in \widetilde{\mathcal{U}}_N(x)$$

Recall that a warm start $\widetilde{\mathbf{u}}$ is a member of $\widetilde{\mathcal{U}}_N(x)$ if all elements of the sequence of controls are members of $\mathbb{U}$, the state trajectory $\phi(k; x, \widetilde{\mathbf{u}})$ terminates in $\mathbb{X}_f$, and $V_N(x, \widetilde{\mathbf{u}})$ is less than $V_f(x)$.

Exercise 2.25: The geometry of cost rotation

Let's examine the rotated cost function in the simplest possible setting to understand what "rotation" means in this context. Consider the discrete time dynamic model and strictly convex quadratic cost function

$$x^+ = f(x,u) \qquad \ell(x,u) = (1/2)(|x - x_{\mathrm{sp}}|_Q^2 + |u - u_{\mathrm{sp}}|_R^2)$$

with $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$, $\ell\colon \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}_{\geq 0}$, $Q \in \mathbb{R}^{n\times n}, R \in \mathbb{R}^{m\times m}$ with $Q, R > 0$. We define the feasible control region as $u \in \mathbb{U}$ for some nonempty set $\mathbb{U}$. We wish to illustrate the ideas with the following simple linear system

$$f(x,u) = Ax + Bu \qquad A = 1/2 \qquad B = 1/2$$

subject to polyhedral constraint

$$\mathbb{U} = \{u \mid -1 \leq u \leq 1\}$$

We choose an *unreachable* setpoint that is not a steady state, and cost matrices as follows

$$(u_{\mathrm{sp}}, x_{\mathrm{sp}}) = (2,3) \qquad Q = R = 2$$

The optimal steady state (u_s, x_s) is given by the solution to the following optimization

$$(u_s, x_s) = \arg\min_{u,x} \{\ell(x,u) \mid u \in \mathbb{U},\ x = f(x,u)\} \tag{2.40}$$

(a) Solve this quadratic program and show that the solution is $(x_s, u_s) = (1,1)$. What is the Lagrange multiplier for the equality constraint?

(b) Next we define the rotated cost function following Diehl et al. (2011)

$$\tilde{\ell}(x,u) = \ell(x,u) - \lambda'(x - f(x,u)) - \ell(x_s,u_s)$$

Plot the contour of zero rotated cost $\tilde{\ell}(x,u) = 0$ for three λ values, $\lambda = 0, -8, -12$. Compare your contours to those shown in Figure 2.15.

Notice that as you decrease λ, you rotate (and enlarge) the zero cost contour of $\ell(x,u)$ about the point (x_s, u_s), hence the name *rotated* stage cost.

(c) Notice that the original cost function, which corresponds to $\lambda = 0$, has negative cost values (interior of the circle) that are in the feasible region. The zero contour for $\lambda = -8$ has become tangent to the feasible region, so the cost is nonnegative in the feasible region. But for $\lambda = -12$, the zero contour has been *over rotated* so that it again has negative values in the feasible region.

How does the value $\lambda = -8$ compare to the Lagrange multiplier of the optimal steady-state problem?

(d) Explain why MPC based on the rotated stage cost is a Lyapunov function for the closed-loop system.

Exercise 2.26: Strong duality implies dissipativity

Consider again the steady-state economic problem $\mathbb{P}_s$ for the optimal steady state (x_s, u_s)

$$\ell(x_s,u_s) := \min_{(x,u)\in\mathbb{Z}} \{\ell(x,u) \mid x = f(x,u)\}$$

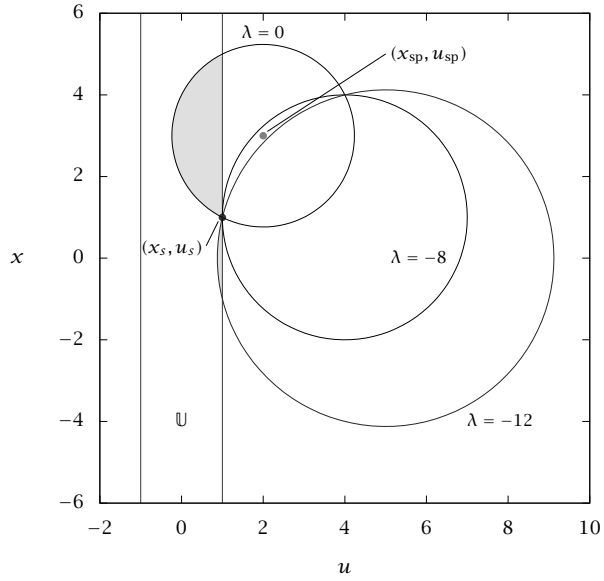


Figure 2.15: Rotated cost-function contour $\tilde{\ell}(x,u) = 0$ (circles) for $\lambda = 0, -8, -12$. Shaded region shows feasible region where $\tilde{\ell}(x,u) < 0$.

Form the Lagrangian and show that the solution is given also by

$$\ell(x_s,u_s) = \min_{(x,u)\in\mathbb{Z}} \max_{\lambda} \ell(x,u) - \lambda'(x - f(x,u))$$

Switching the order of min and max gives

$$\min_{(x,u)\in\mathbb{Z}} \max_{\lambda} \ell(x,u) - \lambda'(x - f(x,u)) \geq \max_{\lambda} \min_{(x,u)\in\mathbb{Z}} \ell(x,u) - \lambda'(x - f(x,u))$$

due to weak duality. The strong duality assumption states that equality is achieved in this inequality above, so that

$$\ell(x_s,u_s) = \max_{\lambda} \min_{(x,u)\in\mathbb{Z}} \ell(x,u) - \lambda'(x - f(x,u))$$

Let λ_s denote the optimal Lagrange multiplier in this problem. (For a brief review of these concepts, see also Exercises C.4, C.5, and C.6 in Appendix C.)

Show that the strong duality assumption implies that the system $x^+ = f(x,u)$ is dissipative with respect to the supply rate $s(x,u) = \ell(x,u) - \ell(x_s,u_s)$.

Bibliography

R. P. Aguilera and D. E. Quevedo. Stability analysis of quadratic MPC with a discrete input alphabet. *IEEE Trans. Auto. Cont.*, 58(12):3190–3196, 2013.

D. A. Allan and J. B. Rawlings. An input/output-to-state stability converse theorem for closed positive invariant sets. Technical Report 2018–01, TWCCC Technical Report, December 2018.

D. A. Allan, C. N. Bates, M. J. Risbeck, and J. B. Rawlings. On the inherent robustness of optimal and suboptimal nonlinear MPC. *Sys. Cont. Let.*, 106: 68–78, August 2017.

R. Amrit, J. B. Rawlings, and D. Angeli. Economic optimization using model predictive control with a terminal cost. *Annual Rev. Control*, 35:178–186, 2011.

D. Angeli, R. Amrit, and J. B. Rawlings. On average performance and stability of economic model predictive control. *IEEE Trans. Auto. Cont.*, 57(7):1615–1626, 2012.

M. Baotic, F. J. Christophersen, and M. Morari. Constrained optimal control of hybrid systems with a linear performance index. *IEEE Trans. Auto. Cont.*, 51 (12):1903–1919, 2006.

A. Bemporad and M. Morari. Control of systems integrating logic, dynamics, and constraints. *Automatica*, 35:407–427, 1999.

F. Blanchini and S. Miani. *Set-Theoretic methods in Control*. Systems & Control: Foundations and Applications. Birkhäuser, 2008.

C. Cai and A. R. Teel. Input–output-to-state stability for discrete-time systems. *Automatica*, 44(2):326 – 336, 2008.

C. C. Chen and L. Shaw. On receding horizon feedback control. *Automatica*, 16(3):349–352, 1982.

H. Chen and F. Allgöwer. A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability. *Automatica*, 34(10):1205–1217, 1998.

D. Chmielewski and V. Manousiouthakis. On constrained infinite-time linear quadratic optimal control. *Sys. Cont. Let.*, 29:121–129, 1996.

D. W. Clarke, C. Mohtadi, and P. S. Tuffs. Generalized predictive control—Part I. The basic algorithm. *Automatica*, 23(2):137–148, 1987.

C. R. Cutler and B. L. Ramaker. Dynamic matrix control—a computer control algorithm. In *Proceedings of the Joint Automatic Control Conference*, 1980.

R. M. C. De Keyser and A. R. Van Cauwenberghe. Extended prediction self-adaptive control. In H. A. Barker and P. C. Young, editors, *Proceedings of the 7th IFAC Symposium on Identification and System Parameter Estimation*, pages 1255–1260. Pergamon Press, Oxford, 1985.

G. De Nicolao, L. Magni, and R. Scattolini. Stabilizing nonlinear receding horizon control via a nonquadratic penalty. In *Proceedings IMACS Multiconference CESA*, volume 1, pages 185–187, Lille, France, 1996.

G. De Nicolao, L. Magni, and R. Scattolini. Stabilizing receding-horizon control of nonlinear time-varying systems. *IEEE Trans. Auto. Cont.*, 43(7):1030–1036, 1998.

S. Di Cairano, W. P. M. H. Heemels, M. Lazar, and A. Bemporad. Stabilizing dynamic controllers for hybrid systems: A hybrid control Lyapunov function approach. *IEEE Trans. Auto. Cont.*, 59(10):2629–2643, 2014.

M. Diehl, R. Amrit, and J. B. Rawlings. A Lyapunov function for economic optimizing model predictive control. *IEEE Trans. Auto. Cont.*, 56(3):703–707, 2011.

M. Ellis, H. Durand, and P. D. Christofides. A tutorial review of economic model predictive control methods. *J. Proc. Cont.*, 24(8):1156–1178, 2014.

L. Fagiano and A. R. Teel. On generalised terminal state constraints for model predictive control. In *Proceedings of 4th IFAC Nonlinear Model Predictive Control Conference*, pages 299–304, 2012.

P. Falugi and D. Q. Mayne. Model predictive control for tracking random references. In *Proceedings of European Control Conference (ECC)*, pages 518–523, 2013a.

P. Falugi and D. Q. Mayne. Tracking a periodic reference using nonlinear model predictive control. In *Proceedings of 52nd IEEE Conference on Decision and Control*, pages 5096–5100, Florence, Italy, December 2013b.

C. E. García and A. M. Morshedi. Quadratic programming solution of dynamic matrix control (QDMC). *Chem. Eng. Commun.*, 46:73–87, 1986.

C. E. García, D. M. Prett, and M. Morari. Model predictive control: Theory and practice—a survey. *Automatica*, 25(3):335–348, 1989.

E. G. Gilbert and K. T. Tan. Linear systems with state and control constraints: The theory and application of maximal output admissible sets. *IEEE Trans. Auto. Cont.*, 36(9):1008–1020, September 1991.

G. Grimm, M. J. Messina, S. E. Tuna, and A. R. Teel. Model predictive control: For want of a local control Lyapunov function, all is not lost. *IEEE Trans. Auto. Cont.*, 50(5):546–558, 2005.

L. Grüne and J. Pannek. *Nonlinear Model Predictive Control: Theory and Algorithms*. Communications and Control Engineering. Springer-Verlag, London, second edition, 2017.

R. Huang, E. Harinath, and L. T. Biegler. Lyapunov stability of economically oriented NMPC for cyclic processes. *J. Proc. Cont.*, 21:501–509, 2011.

A. Jadbabaie, J. Yu, and J. Hauser. Unconstrained receding horizon control of nonlinear systems. *IEEE Trans. Auto. Cont.*, 46(5):776–783, 2001.

S. S. Keerthi and E. G. Gilbert. Computation of minimum-time feedback control laws for systems with state-control constraints. *IEEE Trans. Auto. Cont.*, 32: 432–435, 1987.

S. S. Keerthi and E. G. Gilbert. Optimal infinite-horizon feedback laws for a general class of constrained discrete-time systems: Stability and moving-horizon approximations. *J. Optim. Theory Appl.*, 57(2):265–293, May 1988.

D. L. Kleinman. An easy way to stabilize a linear constant system. *IEEE Trans. Auto. Cont.*, 15(12):692, December 1970.

K. Kobayshi, W. W. Shein, and K. Hiraishi. Large-scale MPC with continuous/discrete-valued inputs: Compensation of quantization errors, stabilization, and its application. *SICE J. Cont., Meas., and Sys. Integr.*, 7(3): 152–158, 2014.

W. H. Kwon and A. E. Pearson. A modified quadratic cost problem and feedback stabilization of a linear system. *IEEE Trans. Auto. Cont.*, 22(5):838–842, October 1977.

M. Lazar and W. P. M. H. Heemels. Predictive control of hybrid systems: Input-to-state stability results for sub-optimal solutions. *Automatica*, 45(1):180–185, 2009.

E. B. Lee and L. Markus. *Foundations of Optimal Control Theory*. John Wiley and Sons, New York, 1967.

D. Limon, T. Alamo, F. Salas, and E. F. Camacho. On the stability of MPC without terminal constraint. *IEEE Trans. Auto. Cont.*, 51(5):832–836, May 2006.

D. Limon, I. Alvarado, T. Alamo, and E. F. Camacho. MPC for tracking piecewise constant references for constrained linear systems. *Automatica*, pages 2382–2387, 2008.

D. Limon, I. Alvarado, T. Alamo, and E. F. Camacho. Robust tube-based MPC for tracking of constrained linear systems with additive disturbances. *J. Proc. Cont.*, 20:248–260, 2010.

D. Limon, T. Alamo, D. M. de la Peña, M. N. Zeilinger, C. N. Jones, and M. Pereira. MPC for tracking periodic reference signals. In *4th IFAC Nonlinear Model Predictive Control Conference*, pages 490–495, 2012.

P. Marquis and J. P. Broustail. SMOC, a bridge between state space and model predictive controllers: Application to the automation of a hydrotreating unit. In T. J. McAvoy, Y. Arkun, and E. Zafiriou, editors, *Proceedings of the 1988 IFAC Workshop on Model Based Process Control*, pages 37–43. Pergamon Press, Oxford, 1988.

D. Q. Mayne. Nonlinear model predictive control: challenges and opportunities. In F. Allgöwer and A. Zheng, editors, *Nonlinear Model Predictive Control*, pages 23–44. Birkhäuser Verlag, Basel, 2000.

D. Q. Mayne. An apologia for stabilising conditions in model predictive control. *Int. J. Control*, 86(11):2090–2095, 2013.

D. Q. Mayne and P. Falugi. Generalized stabilizing conditions for model predictive control. *J. Optim. Theory Appl.*, 169:719–734, 2016.

D. Q. Mayne and H. Michalska. Receding horizon control of non-linear systems. 35(5):814–824, 1990.

D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert. Constrained model predictive control: Stability and optimality. *Automatica*, 36(6):789–814, 2000.

E. S. Meadows, M. A. Henson, J. W. Eaton, and J. B. Rawlings. Receding horizon control and discontinuous state feedback stabilization. *Int. J. Control*, 62 (5):1217–1229, 1995.

H. Michalska and D. Q. Mayne. Robust receding horizon control of constrained nonlinear systems. *IEEE Trans. Auto. Cont.*, 38(11):1623–1633, 1993.

M. A. Müller and F. Allgöwer. Distributed economic MPC: a framework for cooperative control problems. In *Proceedings of the 19th World Congress of the International Federation of Automatic Control*, pages 1029–1034, Cape Town, South Africa, 2014.

M. A. Müller and L. Grüne. Economic model predictive control without terminal constraints: Optimal periodic operation. In *2015 54th IEEE Conference on Decision and Control (CDC)*, pages 4946–4951, 2015.

M. A. Müller, D. Angeli, and F. Allgöwer. On necessity and robustness of dissipativity in economic model predictive control. *IEEE Trans. Auto. Cont.*, 60 (6):1671–1676, June 2015.

K. R. Muske and J. B. Rawlings. Model predictive control with linear models. *AIChE J.*, 39(2):262–287, 1993.

G. Pannocchia, J. B. Rawlings, and S. J. Wright. Conditions under which suboptimal nonlinear MPC is inherently robust. *Sys. Cont. Let.*, 60:747–755, 2011.

G. Pannocchia, J. B. Rawlings, D. Q. Mayne, and G. Mancuso. Whither discrete time model predictive control? *IEEE Trans. Auto. Cont.*, 60(1):246–252, January 2015.

T. Parisini and R. Zoppoli. A receding-horizon regulator for nonlinear systems and a neural approximation. *Automatica*, 31(10):1443–1451, 1995.

V. Peterka. Predictor-based self-tuning control. *Automatica*, 20(1):39–50, 1984.

B. Picasso, S. Pancanti, A. Bemporad, and A. Bicchi. Receding–horizon control of LTI systems with quantized inputs. In *Analysis and Design of Hybrid Systems 2003 (ADHS 03): A Proceedings Volume from the IFAC Conference, St. Malo, Brittany, France, 16-18 June 2003*, volume 259, 2003.

D. M. Prett and R. D. Gillette. Optimization and constrained multivariable control of a catalytic cracking unit. In *Proceedings of the Joint Automatic Control Conference*, pages WP5–C, San Francisco, CA, 1980.

J. A. Primbs and V. Nevistić. Feasibility and stability of constrained finite receding horizon control. *Automatica*, 36:965–971, 2000.

A. I. Propoi. Use of linear programming methods for synthesizing sampled-data automatic systems. *Autom. Rem. Control*, 24(7):837–844, July 1963.

D. E. Quevedo, G. C. Goodwin, and J. A. De Doná. Finite constraint set receding horizon quadratic control. *Int. J. Robust and Nonlinear Control*, 14(4):355–377, 2004.

C. V. Rao and J. B. Rawlings. Steady states and constraints in model predictive control. *AIChE J.*, 45(6):1266–1278, 1999.

J. B. Rawlings and R. Amrit. Optimizing process economic performance using model predictive control. In L. Magni, D. M. Raimondo, and F. Allgöwer, editors, *Nonlinear Model Predictive Control*, volume 384 of *Lecture Notes in Control and Information Sciences*, pages 119–138. Springer, Berlin, 2009.

J. B. Rawlings and K. R. Muske. Stability of constrained receding horizon control. *IEEE Trans. Auto. Cont.*, 38(10):1512–1516, October 1993.

J. B. Rawlings and M. J. Risbeck. On the equivalence between statements with epsilon-delta and K-functions. Technical Report 2015–01, TWCCC Technical Report, December 2015.

J. B. Rawlings and M. J. Risbeck. Model predictive control with discrete actuators: Theory and application. *Automatica*, 78:258–265, 2017.

J. Richalet, A. Rault, J. L. Testud, and J. Papon. Model predictive heuristic control: Applications to industrial processes. *Automatica*, 14:413–428, 1978a.

J. Richalet, A. Rault, J. L. Testud, and J. Papon. Algorithmic control of industrial processes. In *Proceedings of the 4th IFAC Symposium on Identification and System Parameter Estimation*, pages 1119–1167. North-Holland Publishing Company, 1978b.

B. J. P. Roset, W. P. M. H. Heemels, M. Lazar, and H. Nijmeijer. On robustness of constrained discrete-time systems to state measurement errors. *Automatica*, 44(4):1161 – 1165, 2008.

P. O. M. Scokaert and J. B. Rawlings. Constrained linear quadratic regulation. *IEEE Trans. Auto. Cont.*, 43(8):1163–1169, August 1998.

P. O. M. Scokaert, D. Q. Mayne, and J. B. Rawlings. Suboptimal model predictive control (feasibility implies stability). *IEEE Trans. Auto. Cont.*, 44(3):648–654, March 1999.

M. Sznaier and M. J. Damborg. Suboptimal control of linear systems with state and control inequality constraints. In *Proceedings of the 26th Conference on Decision and Control*, pages 761–762, Los Angeles, CA, 1987.

Y. A. Thomas. Linear quadratic optimal estimation and control with receding horizon. *Electron. Lett.*, 11:19–21, January 1975.

B. E. Ydstie. Extended horizon adaptive control. In J. Gertler and L. Keviczky, editors, *Proceedings of the 9th IFAC World Congress*, pages 911–915. Pergamon Press, Oxford, 1984.

S. Yu, M. Reble, H. Chen, and F. Allgöwer. Inherent robustness properties of quasi-infinite horizon nonlinear model predictive control. *Automatica*, 50 (9):2269 – 2280, 2014.

M. Zanon, S. Gros, and M. Diehl. A Lyapunov function for periodic economic optimizing model predictive control. In *Decision and Control (CDC), 2013 IEEE 52nd Annual Conference on*, pages 5107–5112, 2013.

3

Robust and Stochastic Model Predictive Control

3.1 Introduction

3.1.1 Types of Uncertainty

Robust and stochastic control concern control of systems that are uncertain in some sense so that predicted behavior based on a *nominal* model is not identical to actual behavior. Uncertainty may arise in different ways. The system may have an additive disturbance that is unknown, the state of the system may not be perfectly known, or the model of the system that is used to determine control may be inaccurate.

A system with additive disturbance satisfies the following difference equation

$$x^+ = f(x, u, w)$$

If the disturbance w in constrained optimal control problems is bounded it is often possible to design a model predictive controller that ensures the state and control constraints are satisfied for all possible disturbance sequences (robust MPC). If the disturbance w is unbounded, it is impossible to ensure that the usual state and control constraints are satisfied for *all* disturbance sequences. The model predictive controller is then designed to ensure that the constraints are satisfied on average or, more usually, with a prespecified probability (stochastic MPC).

The situation in which the state is not perfectly measured may be treated in several ways. For example, inherent robustness is often studied using the model $x^+ = f(x + e, u, w)$ where e denotes the error in measuring the state. In the stochastic optimal control literature, where

the measured output is $y = Cx + v$ and the disturbance w and measurement noise v are usually assumed to be Gaussian white noise processes, the state or *hyperstate* of the optimal control problem is the conditional density of the state x at time k given prior measurements $(y(0), y(1), \ldots, y(k-1))$. Because this density usually is difficult to compute and use, except in the linear case when it is provided by the Kalman filter, a suboptimal procedure often is adopted. In this suboptimal approach, the state x is replaced by its estimate $\hat{x}$ in a control law determined under the assumption that the state is accessible. This procedure is usually referred to as *certainty equivalence*, a term that was originally employed for the linear quadratic Gaussian (LQG) or similar cases when this procedure did not result in loss of optimality. When $f(\cdot)$ is linear, the evolution of the state estimate $\hat{x}$ may be expressed by a difference equation

$$\hat{x}^+ = g(\hat{x}, u) + \xi$$

in which ξ is the *innovation process*. In controlling $\hat{x}$, we should ensure that the actual state x, which lies in a bounded, possibly time-varying set if the innovation process is bounded, satisfies the constraints of the optimal control problem certainly (robust MPC). If the innovation process is not bounded, the constraints should be satisfied with a prespecified probability (stochastic MPC).

A system that has parametric uncertainty may be modeled as

$$x^+ = f(x, u, \theta)$$

in which θ represents parameters of the system that are known only to the extent that they belong to a compact set Θ. A much-studied example is

$$x^+ = Ax + Bu$$

in which $\theta := (A, B)$ may take any value in $\Theta := \text{co}\{(A_i, B_i) \mid i \in \mathcal{I}\}$ where $\mathcal{I} = \{1, 2, \ldots, I\}$, say, is an index set.

Finally the system description may not include all the dynamics. For example, fast dynamics may be ignored to simplify the system description, or a system described by a partial differential equation may be modeled by an ordinary differential equation (ODE).

It is possible, of course, for all these types of uncertainty to occur in a single application. In this chapter we focus on the effect of additive disturbance. Output MPC—in which the controller employs an estimate of the state, rather than the state itself—is discussed in Chapter 5.

3.1.2 Feedback Versus Open-Loop Control

It is well known that feedback is required only when uncertainty is present; in the absence of uncertainty, feedback control and open-loop control are equivalent. Indeed, when uncertainty is not present, as for the systems studied in Chapter 2, the optimal control for a given initial state may be computed using either dynamic programming (DP) that provides an optimal control policy or sequence of feedback control laws, or an open-loop optimal control that merely provides a sequence of control actions. A simple example illustrates this fact. Consider the deterministic linear dynamic system defined by

$$x^+ = x + u$$

The optimal control problem, with horizon $N = 3$, is

$$\mathbb{P}_3(x): \qquad V_3^0(x) = \min_{\mathbf{u}_3} V_3(x, \mathbf{u})$$

in which $\mathbf{u} = (u(0), u(1), u(2))$

$$V_3(x, \mathbf{u}) := (1/2) \sum_{i=0}^{2} [(x(i)^2 + u(i)^2)] + (1/2)x(3)^2$$

in which, for each i, $x(i) = \phi(i; x, \mathbf{u}) = x + u(0) + u(1) + \ldots + u(i-1)$, the solution of the difference equation $x^+ = x + u$ at time i if the initial state is $x(0) = x$ and the control (input) sequence is $\mathbf{u} = (u(0), u(1), u(2))$; in matrix operations $\mathbf{u}$ is taken to be the column vector $[u(0), u(1), u(2)]'$. Thus

$$\begin{aligned} V_3(x, \mathbf{u}) &= (1/2)[x^2 + (x + u(0))^2 + (x + u(0) + u(1))^2 + \\ &\qquad (x + u(0) + u(1) + u(2))^2 + u(0)^2 + u(1)^2 + u(2)^2] \\ &= (3/2)x^2 + x \begin{bmatrix} 3 & 2 & 1 \end{bmatrix} \mathbf{u} + (1/2)\mathbf{u}' P_3 \mathbf{u} \end{aligned}$$

in which

$$P_3 = \begin{bmatrix} 4 & 2 & 1 \\ 2 & 3 & 1 \\ 1 & 1 & 2 \end{bmatrix}$$

Therefore, the vector form of the optimal *open-loop* control sequence for an initial state of x is

$$\mathbf{u}^0(x) = -P_3^{-1} \begin{bmatrix} 3 & 2 & 1 \end{bmatrix}' x = -\begin{bmatrix} 0.615 & 0.231 & 0.077 \end{bmatrix}' x$$

and the optimal control and state sequences are

$$\mathbf{u}^0(x) = (-0.615x, -0.231x, -0.077x)$$
$$\mathbf{x}^0(x) = (x, 0.385x, 0.154x, 0.077x)$$

To compute the optimal *feedback* control, we use the DP recursions

$$V_i^0(x) = \min_{u\in\mathbb{R}}\{x^2/2 + u^2/2 + V_{i-1}^0(x+u)\}$$
$$\kappa_i^0(x) = \arg\min_{u\in\mathbb{R}}\{x^2/2 + u^2/2 + V_{i-1}^0(x+u)\}$$

with boundary condition

$$V_0^0(x) = (1/2)x^2$$

This procedure gives the value function $V_i^0(\cdot)$ and the optimal control law $\kappa_i^0(\cdot)$ at each i where the subscript i denotes time to go. Solving the DP recursion, for all $x \in \mathbb{R}$, all $i \in \{1, 2, 3\}$, yields

$$V_1^0(x) = (3/4)x^2 \qquad \kappa_1^0(x) = -(1/2)x$$
$$V_2^0(x) = (4/5)x^2 \qquad \kappa_2^0(x) = -(3/5)x$$
$$V_3^0(x) = (21/26)x^2 \qquad \kappa_3^0(x) = -(8/13)x$$

Starting at state x at time zero, and applying the optimal control laws iteratively to the *deterministic* system $x^+ = x + u$ (recalling that at time i the optimal control law is $\kappa_{3-i}^0(\cdot)$ since, at time i, $3 - i$ is the time to go) yields

$$x^0(0) = x \qquad u^0(0) = -(8/13)x$$
$$x^0(1) = (5/13)x \qquad u^0(1) = -(3/13)x$$
$$x^0(2) = (2/13)x \qquad u^0(2) = -(1/13)x$$
$$x^0(3;x) = (1/13)x$$

so that the optimal control and state sequences are, respectively,

$$\mathbf{u}^0(x) = (-(8/13)x, -(3/13)x, -(1/13)x)$$
$$\mathbf{x}^0(x) = (x, (5/13)x, (2/13)x, (1/13)x)$$

which are identical with the optimal open-loop values computed above.

Consider next an uncertain version of the dynamic system in which uncertainty takes the simple form of an additive disturbance w; the system is defined by

$$x^+ = x + u + w$$

in which the only knowledge of w is that it lies in the compact set $\mathbb{W} := [-1, 1]$. Let $\phi(i; x, \mathbf{u}, \mathbf{w})$ denote the solution of this system at time i if the initial state is x at time zero, and the input and disturbance sequences are, respectively, $\mathbf{u}$ and $\mathbf{w} := (w(0), w(1), w(2))$. The cost now depends on the disturbance sequence—but it also depends, in contrast to the deterministic problem discussed above, on whether the control is open-loop or feedback. To discuss the latter case, we define a feedback policy $\boldsymbol{\mu}$ to be a sequence of control laws

$$\boldsymbol{\mu} := (\mu_0(\cdot), \mu_1(\cdot), \mu_2(\cdot))$$

in which $\mu_i : \mathbb{R} \to \mathbb{R}$, $i = 0, 1, 2$; under policy $\boldsymbol{\mu}$, if the state at time i is x, the control is $\mu_i(x)$. Let $\mathcal{M}$ denote the class of *admissible* policies, for example, those policies for which each control law $\mu_i(\cdot)$ is continuous. Then, $\phi(i; x, \boldsymbol{\mu}, \mathbf{w})$ denotes the solution at time $i \in \{0, 1, 2, 3\}$ of the following difference equation

$$x(i+1) = x(i) + \mu_i(x(i)) + w(i) \qquad x(0) = x$$

An open-loop control sequence $\mathbf{u} = (u(0), u(1), u(2))$ is then merely a degenerate policy $\boldsymbol{\mu} = (\mu_0(\cdot), \mu_1(\cdot), \mu_2(\cdot))$ where each control law $\mu_i(\cdot)$ satisfies

$$\mu_i(x) = u(i)$$

for all $x \in \mathbb{R}$ and all $i \in \{0, 1, 2\}$. The cost $V_3(\cdot)$ may now be defined

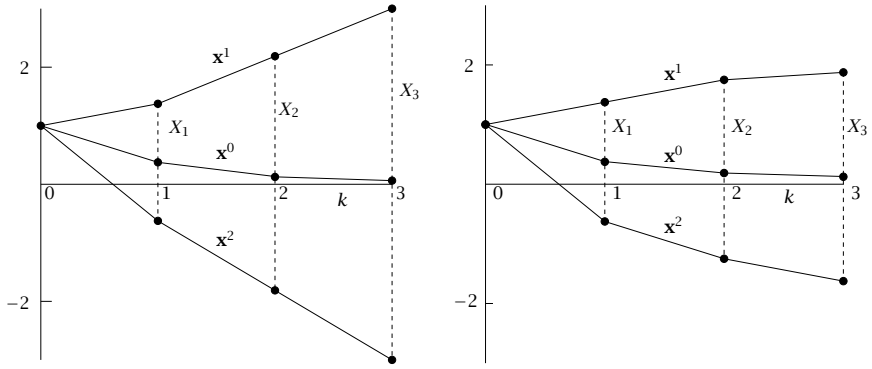
$$V_3(x, \boldsymbol{\mu}, \mathbf{w}) := (1/2) \sum_{i=0}^{2} [(x(i)^2 + u(i)^2)] + (1/2)x(3)^2$$

where, now, $x(i) = \phi(i; x, \boldsymbol{\mu}, \mathbf{w})$ and $u(i) = \mu_i(x(i))$. Since the disturbance is unpredictable, the value of $\mathbf{w}$ is not known at time zero, so the optimal control problem must "eliminate" it in some meaningful way so that the solution $\boldsymbol{\mu}^0(x)$ does not depend on $\mathbf{w}$. To eliminate $\mathbf{w}$, the optimal control problem $\mathbb{P}_3^*(x)$ is defined by

$$\mathbb{P}_3^*(x): \qquad V_3^0(x) := \inf_{\boldsymbol{\mu} \in \mathcal{M}} J_3(x, \boldsymbol{\mu})$$

in which the cost $J_3(\cdot)$ is defined in such a way that it does not depend on $\mathbf{w}$; inf is used rather than min in this definition since the minimum may not exist. The most popular choice for $J_3(\cdot)$ in the MPC literature is

$$J_3(x, \boldsymbol{\mu}) := \max_{\mathbf{w} \in \mathcal{W}} V_3(x, \boldsymbol{\mu}, \mathbf{w})$$



(a) Open-loop trajectories.

(b) Feedback trajectories.

Figure 3.1: Open-loop and feedback trajectories.

in which the disturbance $\mathbf{w}$ is assumed to lie in $\mathcal{W}$ a bounded class of admissible disturbance sequences. Alternatively, if the disturbance sequence is random, the cost $J_3(\cdot)$ may be chosen to be

$$J_3(x, \boldsymbol{\mu}) := \mathcal{E}V_3(x, \boldsymbol{\mu}, \mathbf{w})$$

in which $\mathcal{E}$ denotes "expectation" or average, over random disturbance sequences. For our purpose here, we adopt the simple cost

$$J_3(x, \boldsymbol{\mu}) := V_3(x, \boldsymbol{\mu}, \mathbf{0})$$

in which $\mathbf{0} := (0, 0, 0)$ is the zero disturbance sequence. In this case, $J_3(x, \boldsymbol{\mu})$ is the nominal cost, i.e., the cost associated with the nominal system $x^+ = x + u$ in which the disturbance is neglected. With this cost function, the solution to $\mathbb{P}_3^*(x)$ is the DP solution, obtained previously, to the deterministic *nominal* optimal control problem.

We now compare two solutions to $\mathbb{P}_3(x)$: the open-loop solution in which $\mathcal{M}$ is restricted to be the set of control sequences, and the feedback solution in which $\mathcal{M}$ is the class of admissible policies. The solution to the first problem is the solution to the deterministic problem discussed previously; the optimal control sequence is

$$\mathbf{u}^0(x) = (-(8/13)x, -(3/13)x, -(1/13)x)$$

in which x is the initial state at time zero. The solution to the second problem is the sequence of control laws determined previously, also for

the deterministic problem, using *dynamic programming*; the optimal policy is $\boldsymbol{\mu}^0 = \left(\mu_0^0(\cdot), \mu_1^0(\cdot), \mu_2(\cdot)\right)$ where the control laws (functions) $\mu_i(\cdot)$, $i = 0, 1, 2$, are defined by

$$\begin{aligned}
\mu_0^0(x) &:= \kappa_3^0(x) = -(8/13)x \quad \forall x \in \mathbb{R} \\
\mu_1^0(x) &:= \kappa_2^0(x) = -(3/5)x \quad \forall x \in \mathbb{R} \\
\mu_2^0(x) &:= \kappa_1^0(x) = -(1/2)x \quad \forall x \in \mathbb{R}
\end{aligned}$$

The two solutions, $\mathbf{u}^0(\cdot)$ and $\boldsymbol{\mu}^0$, when applied to the uncertain system $x^+ = x + u + w$, do *not* yield the same trajectories for all disturbance sequences. This is illustrated in Figure 3.1 for the three disturbance sequences, $\mathbf{w}^0 := (0, 0, 0)$, $\mathbf{w}^1 := (1, 1, 1)$, and $\mathbf{w}^2 := (-1, -1, -1)$; and initial state $x = 1$ for which the corresponding state trajectories, denoted $\mathbf{x}^0$, $\mathbf{x}^1$, and $\mathbf{x}^2$, are

Open-loop solution.

$$\begin{aligned}
\mathbf{x}^0 &= (1, (5/13), (2/13), (1/13)) \\
\mathbf{x}^1 &= (1, (18/13), (28/13), (40/13)) \\
\mathbf{x}^2 &= (1, -(8/13), -(24/13), -(38/13))
\end{aligned}$$

Feedback solution.

$$\begin{aligned}
\mathbf{x}^0 &= (1, (5/13), (2/13), (1/13)) \\
\mathbf{x}^1 &= (1, (18/13), (101/65), (231/130)) \\
\mathbf{x}^2 &= (1, -(8/13), -(81/65), -(211/130))
\end{aligned}$$

Even for the short horizon of 3, the superiority of the feedback solution can be seen although the feedback was designed for the deterministic (nominal) system and therefore did not take the disturbance into account. For the open-loop solution $|x^2(3) - x^1(3)| = 6$, whereas for the feedback case $|x^2(3) - x^1(3)| = 3.4$; the open-loop solution does not restrain the *spread* of the trajectories resulting from the disturbance $\mathbf{w}$. If the horizon length is N, for the open-loop solution, $|x^2(N) - x^1(N)| = 2N$, whereas for the feedback case $|x^2(N) - x^1(N)| \to 3.24$ as $N \to \infty$. The obvious and well-known conclusion is that feedback control is superior to open-loop control when uncertainty is present. Feedback control requires determination of a control *policy*, however, which is a difficult task if nonlinearity and/or constraints are features of the optimal control problem.

3.1.3 Robust and Stochastic MPC

An important feature of conventional, or deterministic, MPC discussed in Chapter 2 is that the solution of the open-loop optimal control problem solved online is identical to that obtained by DP for the given initial state. When uncertainty is present and the state is known or observations of the state are available, feedback control is superior to open-loop control. The optimal control problem solved online must, therefore, permit feedback in order for its solution to coincide with the DP solution. In robust and stochastic MPC, the decision variable is $\boldsymbol{\mu}$, a sequence of control *laws*, rather than $\mathbf{u}$, a sequence of control *actions*. MPC in which the decision variable is a policy has been termed *feedback* MPC to distinguish it from conventional MPC. Both forms of MPC naturally provide feedback control since the control that is implemented depends on the current state x in both cases. But the control that is applied depends on whether the optimal control problem solved is open loop, in which case the decision variable is a control sequence, or feedback, in which case the decision variable is a feedback policy.

In feedback MPC the solution to the optimal control problem $\mathbb{P}_N^*(x)$ is the policy $\boldsymbol{\mu}^0(x) = \left(\mu_0^0(\cdot;x), \mu_1^0(\cdot;x), \ldots, \mu_{N-1}^0(\cdot;x)\right)$. The constituent control laws are restrictions of those determined by DP and therefore depend on the initial state x as implied by the notation. Thus, only the value $u^0(x) = \mu_0(x;x)$ of the control law $\mu_0(\cdot;x)$ at the initial state x need be determined, while successive laws need only be determined over a limited subset of the state space. In the example illustrated in Figure 3.1, $\mu_0(\cdot;x)$ need be determined only at the point $x = 1$, $\mu_1(\cdot;x)$ need only be determined in the subset $[-8/13, 18/13]$, and $\mu_2(\cdot;x)$ in the subset $[-81/65, 101/65]$, whereas in the DP solution these control laws are defined over the infinite interval $(-\infty, \infty)$.

While feedback MPC is superior in the presence of uncertainty, the associated optimal control problem is vastly more complex than the optimal control problem employed in deterministic MPC. The decision variable $\boldsymbol{\mu}$, being a sequence of control laws, is infinite dimensional; each law or function requires, in general, an infinite dimensional grid to specify it. The complexity is comparable to solving the DP equation, so that MPC, which in the deterministic case replaces DP with a solvable open-loop optimization problem, is not easily solved when uncertainty is present. Hence much research effort has been devoted to forms of feedback MPC that sacrifice optimality for simplicity. As in the early days of adaptive control, many different proposals have been made.

These proposals for robust MPC are all simpler to implement than the optimal solution provided by DP.

At the current stage of research it is perhaps premature to select a particular approach; we have, nevertheless, selected one approach, *tube-based* MPC that we describe here and in Chapter 5. There is a good reason for our choice. It is well known that standard mathematical optimization algorithms may be used to obtain an optimal open-loop control sequence for an optimal control problem. What is perhaps less well known is that there exist algorithms, the second variation algorithms, that provide not only an optimal control sequence but also a *local* time-varying feedback law of the form $u(k) = \bar{u}(k) + K(k)(x(k) - \bar{x}(k))$ in which $(\bar{u}(k))$ is the optimal open-loop control sequence and $(\bar{x}(k))$ the corresponding optimal open-loop state sequence. This policy provides feedback control for states $x(k)$ close to the nominal states $\bar{x}(k)$.

The second variation algorithms are perhaps too complex for routine use in MPC because they require computation of the second derivatives with respect to (x, u) of $f(\cdot)$ and $\ell(\cdot)$. When the system is linear, the cost quadratic, and the disturbance additive, however, the optimal control law for the unconstrained infinite horizon case is $u = Kx$. This result may be expressed as a time-varying control law $u(k) = \bar{u}(k) + K(x(k) - \bar{x}(k))$ in which the state and control sequences $(\bar{x}(k))$ and $(\bar{u}(k))$ satisfy the nominal difference equations $\bar{x}^+ = A\bar{x} + B\bar{u}$, $\bar{u} = Kz$, i.e., the sequences $(\bar{x}(k))$ and $(\bar{u}(k))$ are optimal open-loop solutions for zero disturbance and some initial state. The time-varying control law $u(k) = \bar{u}(k) + K(x(k) - \bar{x}(k))$ is clearly optimal in the unconstrained case; it remains optimal for the constrained case in the neighborhood of the nominal trajectory $(\bar{x}(k))$ if $(\bar{x}(k))$ and $(\bar{u}(k))$ lie in the interior of their respective constraint sets.

These comments suggest that a time-varying policy of the form $u(x, k) = \bar{u}(k) + K(x - \bar{x}(k))$ might be adequate, at least when $f(\cdot)$ is linear. The nominal control and state sequences, $(\bar{u}(k))$ and $(\bar{x}(k))$, respectively, can be determined by solving a standard open-loop optimal control problem of the form usually employed in MPC, and the feedback matrix K can be determined offline. We show that this form of robust MPC has the same order of online complexity as that conventionally used for deterministic systems. It requires a modified form of the online optimal control problem in which the constraints are simply *tightened* to allow for disturbances, thereby constraining the trajectories of the uncertain system to lie in a tube centered on the nominal trajectories. Offline computations are required to determine the mod-

ified constraints and the feedback matrix K. We also present, in the last section of this chapter, a modification of this tube-based procedure for nonlinear systems for which a *nonlinear* local feedback policy is required.

A word of caution is necessary. Just as nominal model predictive controllers presented in Chapter 2 may fail in the presence of uncertainty, the controllers presented in this chapter may fail if the actual uncertainty does not satisfy our assumptions. In robust MPC this may occur when the disturbance that we assume to be bounded exceeds the assumed bounds; the controlled systems are robust only to the specified uncertainties. As always, online fault diagnosis and safe recovery procedures may be required to protect the system from unanticipated events.

3.1.4 Tubes

The approach that we adopt is motivated by the following observation. Both open-loop and feedback control generate, in the presence of uncertainty, a *bundle* or *tube* of trajectories, each trajectory in the bundle or tube corresponding to a particular realization of the uncertainty. In Figure 3.1(a), the tube corresponding to $\mathbf{u} = \mathbf{u}^0(x)$ and initial state $x = 1$, is (X_0, X_1, X_2, X_3) where $X_0 = \{1\}$; for each i, $X_i = \{\phi(i; x, \mathbf{u}, \mathbf{w}) \mid \mathbf{w} \in \mathcal{W}\}$, the set of states at time i generated by all possible realizations of the disturbance sequence. In robust MPC the state constraints must be satisfied by every trajectory in the tube. In stochastic MPC the tube has the property that state sequences lie within this tube with a prespecified probability.

Control of uncertain systems is best viewed as control of tubes rather than trajectories; the designer chooses, for each initial state, a tube in which all realizations of the state trajectory are controlled to lie (robust MPC), or in which the realizations lie with a given probability (stochastic MPC). By suitable choice of the tube, satisfaction of state and control constraints may be guaranteed for every realization of the disturbance sequence, or guaranteed with a given probability.

Determination of a suitable tube $(X_0, X_1, \ldots)$ corresponding to a given initial state x and policy $\boldsymbol{\mu}$ is difficult even for linear systems, however, and even more difficult for nonlinear systems. Hence, in the sequel, we show for robust MPC how simple tubes that bound all realizations of the state trajectory may be constructed. For example, for linear systems with convex constraints, a tube $(X_0, X_1, \ldots,)$ may be designed to bound all realizations of the state trajectory; for each i,

$X_i = \{\bar{x}(i)\} \oplus S$, $\bar{x}(i)$ is the state at time i of a deterministic system, X_i is a polytope, and S is a positive invariant set. This construction permits robust model predictive controllers to be designed with not much more computation online than that required for deterministic systems. The stochastic MPC controllers are designed to satisfy constraints with a given probability.

3.1.5 Difference Inclusion Description of Uncertain Systems

Here we introduce some notation that will be useful in the sequel. A deterministic discrete time system is usually described by a difference equation

$$x^+ = f(x, u) \tag{3.1}$$

We use $\phi(k; x, i, \mathbf{u})$ to denote the solution of (3.1) at time k when the initial state at time i is x and the control sequence is $\mathbf{u} = (u(0), u(1), \ldots)$; if the initial time $i = 0$, we write $\phi(k; x, \mathbf{u})$ in place of $\phi(k; (x, 0), \mathbf{u})$. Similarly, an uncertain system may be described by the difference equation

$$x^+ = f(x, u, w) \tag{3.2}$$

in which the variable w that represents the uncertainty takes values in a specified set $\mathbb{W}$. We use $\phi(k; x, i, \mathbf{u}, \mathbf{w})$ to denote the solution of (3.2) when the initial state at time i is x and the control and disturbance sequences are, respectively, $\mathbf{u} = (u(0), u(1), \ldots)$ and $\mathbf{w} = (w(0), w(1), \ldots)$. The uncertain system may alternatively be described by a *difference inclusion* of the form

$$x^+ \in F(x, u)$$

in which $F(\cdot)$ is a set-valued map. We use the notation $F : \mathbb{R}^n \times \mathbb{R}^m \rightsquigarrow \mathbb{R}^n$ or[1] $F : \mathbb{R}^n \times \mathbb{R}^m \to 2^{\mathbb{R}^n}$ to denote a function that maps points in $\mathbb{R}^n \times \mathbb{R}^m$ into subsets of $\mathbb{R}^n$. If the uncertain system is described by (3.2), then

$$F(x, u) = f(x, u, \mathbb{W}) := \{f(x, u, w) \mid w \in \mathbb{W}\}$$

If x is the current state, and u the current control, the successor state x^+ lies anywhere in the set $F(x, u)$. When the control policy $\boldsymbol{\mu} := (\mu_0(\cdot), \mu_1(\cdot), \ldots)$ is employed, the state evolves according to

$$x^+ \in F(x, \mu_k(x)), \quad k^+ = k + 1 \tag{3.3}$$

[1] For any set X, 2^X denotes the set of all subsets of X.

in which x is the current state, k the current time, and x^+ the successor state at time $k^+ = k + 1$. The system described by (3.3) does not have a single solution for a given initial state; it has a solution for each possible realization $\mathbf{w}$ of the disturbance sequence. We use $S(x, i)$ to denote the set of solutions of (3.3) if the initial state is x at time i. If $\phi^*(\cdot) \in S(x, i)$ then

$$\phi^*(t) = \phi(t; x, i, \boldsymbol{\mu}, \mathbf{w})$$

for some admissible disturbance sequence $\mathbf{w}$ in which $\phi(t; x, i, \boldsymbol{\mu}, \mathbf{w})$ denotes the solution at time t of

$$x^+ = f(x, \mu_k(x), w)$$

when the initial state is x at time i and the disturbance sequence is $\mathbf{w}$. The policy $\boldsymbol{\mu}$ is defined, as before, to be the sequence of control laws $(\mu_0(\cdot), \mu_1(\cdot), \ldots, \mu_{N-1}(\cdot))$. The tube $\mathbf{X} = (X_0, X_1, \ldots)$, discussed in Section 3.5, generated when policy $\boldsymbol{\mu}$ is employed, satisfies

$$X_{k+1} = \mathbf{F}(X_k, \mu_k(\cdot)) := f(X_k, \mu_k(x), \mathbb{W})$$

3.2 Nominal (*Inherent*) Robustness

3.2.1 Introduction

Because feedback MPC is complex, it is natural to inquire if nominal MPC, i.e., MPC based on the nominal system ignoring uncertainty, is sufficiently robust to uncertainty. Before proceeding with a detailed analysis, a few comments may be helpful.

MPC uses, as a Lyapunov function, the value function of a parametric optimal control problem. Often the value function is continuous, but this is not necessarily the case, especially if state and/or terminal constraints are present. It is also possible for the value function to be continuous but the associated control law to be discontinuous; this can happen, for example, if the minimizing control is not unique.

It is important to realize that a control law may be stabilizing but not robustly stabilizing; arbitrary perturbations, no matter how small, can destabilize the system. This point is illustrated in Teel (2004) with the following discontinuous autonomous system ($n = 2$, $x = (x_1, x_2)$)

$$x^+ = f(x) \qquad f(x) = \begin{cases} (0, |x|) & x_1 \neq 0 \\ (0, 0) & \text{otherwise} \end{cases}$$

If the initial state is $x = (1, 1)$, then $\phi(1; x) = (0, \sqrt{2})$ and $\phi(2; x) = (0, 0)$, with similar behavior for other initial states. In fact, all solutions satisfy

$$\phi(k; x) \leq \beta(|x|, k)$$

in which $\beta(\cdot)$, defined by

$$\beta(|x|, k) := 2(1/2)^k |x|$$

is a $\mathcal{KL}$ function, so that the origin is *globally asymptotically stable* (GAS). Consider now a perturbed system satisfying

$$x^+ = \begin{bmatrix} \delta \\ |x| + \delta \end{bmatrix}$$

in which $\delta > 0$ is a constant perturbation that causes x_1 to remain strictly positive. If the initial state is $x = \varepsilon(1, 1)$, then $x_1(k) = \delta$ for $k \geq 1$, and $x_2(k) > \varepsilon\sqrt{2} + k\delta \to \infty$ as $k \to \infty$, no matter how small δ and ε are. Hence the origin is unstable in the presence of an arbitrarily small perturbation; global asymptotic stability is not a robust property of this system.

This example may appear contrived but, as Teel (2004) points out, a similar phenomenon can arise in receding horizon optimal control of a *continuous system*. Consider the following system

$$x^+ = \begin{bmatrix} x_1(1 - u) \\ |x| u \end{bmatrix}$$

in which the control u is constrained to lie in the set $\mathbb{U} = [-1, 1]$. Suppose we choose a horizon length $N = 2$ and choose $\mathbb{X}_f$ to be the origin. If $x_1 \neq 0$, the only feasible control sequence steering x to 0 in two steps is $\mathbf{u} = \{1, 0\}$; the resulting state sequence is $(x, (0, |x|), (0, 0))$. Since there is only one feasible control sequence, it is also optimal, and $\kappa_2(x) = 1$ for all x such that $x_1 \neq 0$. If $x_1 = 0$, then the only optimal control sequence is $\mathbf{u} = (0, 0)$ and $\kappa_2(x) = 0$. The resultant closed-loop system satisfies

$$x^+ = f(x) := \begin{bmatrix} x_1(1 - \kappa_2(x)) \\ |x| \kappa_2(x) \end{bmatrix}$$

in which $\kappa_2(x) = 1$ if $x_1 \neq 0$, and $\kappa_2(x) = 0$ otherwise. Thus

$$f(x) = \begin{cases} (0, |x|) & x_1 \neq 0 \\ (0, 0) & \text{otherwise} \end{cases} \tag{3.4}$$

The system $x^+ = f(x)$ is the discontinuous system analyzed previously. Thus, receding horizon optimal control of a continuous system has resulted in a discontinuous system that is globally asymptotically stable (GAS) but has no robustness.

3.2.2 Difference Inclusion Description of Discontinuous Systems

Consider a system

$$x^+ = f(x)$$

in which $f(\cdot)$ is not continuous. An example of such a system occurred in the previous subsection where $f(\cdot)$ satisfies (3.4). Solutions of this system are very sensitive to the value of x_1. An infinitesimal change in x_1 at time zero, say, from 0 can cause a substantial change in the subsequent trajectory resulting, in this example, in a loss of robustness. To design a robust system, one must take into account, in the design process, the system's extreme sensitivity to variations in state. This can be done by *regularizing* the system (Teel, 2004). If $f(\cdot)$ is locally bounded,[2] the *regularization* $x^+ = f(x)$ is defined to be

$$x^+ \in F(x) := \bigcap_{\delta>0} \overline{f(\{x\} \oplus \delta\mathcal{B})}$$

in which $\mathcal{B}$ is the closed unit ball so that $\{x\} \oplus \delta\bar{\mathcal{B}} = \{z \mid |z - x| \le \delta\}$ and $\overline{A}$ denotes the closure of set A. At points where $f(\cdot)$ is continuous, $F(x) = \{f(x)\}$, i.e., $F(x)$ is the single point $f(x)$. If $f(\cdot)$ is piecewise continuous, e.g., if $f(x) = x$ if $x < 1$ and $f(x) = 2x$ if $x \ge 1$, then $F(x) = \{\lim_{x_i \to x} f(x_i)\}$, the set of all limits of $f(x_i)$ as $x_i \to x$. For our example immediately above, $F(x) = \{x\}$ if $x < 1$ and $F(x) = \{2x\}$ if $x > 1$. When $x = 1$, the limit of $f(x_i)$ as $x_i \to 1$ from below is 1 and the limit of $f(x_i)$ as $x \to 1$ from above is 2, so that $F(1) = \{1, 2\}$. The regularization of $x^+ = f(x)$ where $f(\cdot)$ is defined in (3.4) is $x^+ \in F(x)$ where $F(\cdot)$ is defined by

$$F(x) = \left\{ \begin{bmatrix} 0 \\ |x| \end{bmatrix} \right\} \qquad x_1 \neq 0 \tag{3.5}$$

$$F(x) = \left\{ \begin{bmatrix} 0 \\ |x| \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \end{bmatrix} \right\} \quad x_1 = 0 \tag{3.6}$$

[2] A function $f : \mathbb{R}^p \to \mathbb{R}^n$ is locally bounded if, for every $x \in \mathbb{R}^p$, there exists a neighborhood $\mathcal{N}$ of x and a $c > 0$ such that $|f(z)| \le c$ for all $z \in \mathcal{N}$.

If the initial state is $x = (1,1)$, as before, then the difference inclusion generates the following tube

$$X_0 = \left\{ \begin{bmatrix} 1 \\ 1 \end{bmatrix} \right\}, \quad X_1 = \left\{ \begin{bmatrix} 0 \\ \sqrt{2} \end{bmatrix} \right\}, \quad X_2 = \left\{ \begin{bmatrix} 0 \\ \sqrt{2} \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \end{bmatrix} \right\}, \quad \dots$$

with $X_k = X_2$ for all $k \geq 2$. The set X_k of possible states clearly does not converge to the origin even though the trajectory generated by the original system does.

3.2.3 When Is Nominal MPC Robust?

The discussion in Section 2.4.1 shows that nominal MPC is not necessarily robust. It is therefore natural to ask under what conditions nominal MPC is robust. To answer this, we have to define robustness precisely. In Appendix B, we define robust stability, and robust asymptotic stability, of a set. We employ this concept later in this chapter in the design of robust model predictive controllers that for a given initial state in the region of attraction, steer *every* realization of the state trajectory to this set. Here, however, we address a slightly different question: when is nominal MPC that steers every trajectory in the region of attraction to the origin robust? Obviously, the disturbance will preclude the controller from steering the state of the perturbed system to the origin; the best that can be hoped for is that the controller will steer the state to some small neighborhood of the origin. Let the nominal (controlled) system be described by $x^+ = f(x)$ in which $f(\cdot)$ is not necessarily continuous, and let the perturbed system be described by $x^+ = f(x+e) + w$. Also let $S_\delta(x)$ denote the set of solutions for the perturbed system with initial state x and perturbation sequences $\mathbf{e} := (e(0), e(1), e(2), \dots)$ and $\mathbf{w} := (w(0), w(1), w(2), \dots)$ satisfying $\max\{\|\mathbf{e}\|, \|\mathbf{w}\|\} \leq \delta$ where, for any sequence $\boldsymbol{\nu}$, $\|\boldsymbol{\nu}\|$ denotes the sup norm, $\sup_{k \geq 0} |\nu(k)|$. The definition of robustness that we employ is (Teel, 2004)

Definition 3.1 (Robust global asymptotic stability)**.** Let $\mathcal{A}$ be compact, and let $d(x, \mathcal{A}) := \min_a \{|a - x| \mid a \in \mathcal{A}\}$, and $|x|_{\mathcal{A}} := d(x, \mathcal{A})$. The set $\mathcal{A}$ is robustly globally asymptotically stable (RGAS) for $x^+ = f(x)$ if there exists a class $\mathcal{KL}$ function $\beta(\cdot)$ such that for each $\varepsilon > 0$ and each compact set C, there exists a $\delta > 0$ such that for each $x \in C$ and each $\phi \in S_\delta(x)$, there holds $|\phi(k;x)|_{\mathcal{A}} \leq \beta(|x|_{\mathcal{A}}, k) + \varepsilon$ for all $k \in \mathbb{I}_{\geq 0}$.

Taking the set $\mathcal{A}$ to be the origin ($\mathcal{A} = \{0\}$) so that $|x|_{\mathcal{A}} = |x|$, we see that if the origin is robustly asymptotically stable for $x^+ = f(x)$,

then, for each $\varepsilon > 0$, there exists a $\delta > 0$ such that every trajectory of the perturbed system $x^+ = f(x+e)+w$ with $\max\{\|\mathbf{e}\|, \|\mathbf{w}\|\} \leq \delta$ converges to $\varepsilon\mathcal{B}$ ($\mathcal{B}$ is the closed unit ball); this is the attractivity property. Also, if the initial state x satisfies $|x| \leq \beta^{-1}(\varepsilon, 0)$, then $|\phi(k;x)| \leq \beta(\beta^{-1}(\varepsilon, 0), 0) + \varepsilon = 2\varepsilon$ for all $k \in \mathbb{I}_{\geq 0}$ and for all $\phi \in S_\delta$, which is the Lyapunov stability property. Here the function $\beta^{-1}(\cdot, 0)$ is the inverse of the function $\alpha \mapsto \beta(\alpha, 0)$.

We return to the question: under what conditions is asymptotic stability robust? We first define a slight extension to the definition of a Lyapunov function given in Chapter 2: A function $V : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ is defined to be a Lyapunov function for $x^+ = f(x)$ in $\mathbb{X}$ and set $\mathcal{A}$ if there exist functions $\alpha_i \in \mathcal{K}_\infty$, $i = 1, 2$ and a continuous, positive definite function $\alpha_3(\cdot)$ such that, for any $x \in \mathbb{X}$

$$\alpha_1(|x|_{\mathcal{A}}) \leq V(x) \leq \alpha_2(|x|_{\mathcal{A}})$$
$$V(f(x)) \leq V(x) - \alpha_3(|x|_{\mathcal{A}})$$

in which $|x|_{\mathcal{A}}$ is defined to be distance $d(x, \mathcal{A})$ of x from the set $\mathcal{A}$. The following important result (Teel, 2004; Kellett and Teel, 2004) answers the important question, "When is asymptotic stability robust?"

Theorem 3.2 (Lyapunov function and RGAS)**.** *Suppose $\mathcal{A}$ is compact and that $f(\cdot)$ is locally bounded.*[3] *The set $\mathcal{A}$ is RGAS for the system $x^+ = f(x)$ if and only if the system admits a* continuous *global Lyapunov function for $\mathcal{A}$.*

This result proves the existence of a $\delta > 0$ that specifies the permitted magnitude of the perturbations, but does not give a value for δ. Robustness against perturbations of a specified magnitude may be required in practice; in the following section we show how to achieve this aim if it is possible.

In MPC, the value function of the finite horizon optimal control problem that is solved online is used as a Lyapunov function. In certain cases, such as linear systems with polyhedral constraints, the value function is known to be continuous; see Proposition 7.13. Theorem 3.2, suitably modified because the region of attraction is not global, then shows that asymptotic stability is robust, i.e., that asymptotic stability is not destroyed by *small* perturbations.

Theorem 3.2 characterizes robust stability of the set $\mathcal{A}$ for the system $x^+ = f(x)$ in the sense that it shows robust stability is equivalent

[3] A function $f : \mathcal{X} \to \mathcal{Y}$ is locally bounded if, for every $x \in \mathcal{X}$, there exists a neighborhood $\mathcal{N}$ of x such that the set $f(\mathcal{N})$ in $\mathcal{Y}$ is bounded.

to the existence of a continuous global Lyapunov function for the system. It also is possible to characterize robustness of $x^+ = f(x)$ by global asymptotic stability of its regularization $x^+ \in F(x)$. It is shown in Appendix B that for the system $x^+ \in F(x)$, the set $\mathcal{A}$ is GAS if there exists a $\mathcal{KL}$ function $\beta(\cdot)$ such that for each $x \in \mathbb{R}^n$ and each solution $\phi(\cdot) \in S(x)$ of $x^+ \in F(x)$ with initial state x, $\phi(k) \leq \beta(|x|_{\mathcal{A}}, k)$ for all $k \in \mathbb{I}_{\geq 0}$. The following alternative characterization of robust stability of $\mathcal{A}$ for the system $x^+ = f(x)$ appears in (Teel, 2004).

Theorem 3.3 (Robust global asymptotic stability and regularization)**.** *Suppose $\mathcal{A}$ is compact and that $f(\cdot)$ is locally bounded. The set $\mathcal{A}$ is RGAS for the system $x^+ = f(x)$ if and only if the set $\mathcal{A}$ is GAS for $x^+ \in F(x)$, the regularization of $x^+ = f(x)$.*

We saw previously that for $f(\cdot)$ and $F(\cdot)$ defined respectively in (3.4) and (3.6), the origin is not globally asymptotically stable for the regularization $x^+ \in F(x)$ of $x^+ = f(x)$ since not every solution of $x^+ \in F(x)$ converges to the origin. Hence the origin is not RGAS for this system.

3.2.4 Robustness of Nominal MPC

If the origin is asymptotically stable for the nominal version of an uncertain system, it is sometimes possible to establish that there exists a set $\mathcal{A}$ that is asymptotically stable for the uncertain system. We consider the uncertain system described by

$$x^+ = f(x, u, w) \tag{3.7}$$

in which w is a bounded additive disturbance and $f(\cdot)$ is continuous. The system is subject to the control constraints

$$u(i) \in \mathbb{U} \quad \forall i \in \mathbb{I}_{\geq 0}$$

The set $\mathbb{U}$ is compact and contains the origin in its interior. The disturbance w may take any value in the set $\mathbb{W}$. As before, $\mathbf{u}$ denotes the control sequence $(u(0), u(1), \ldots)$ and $\mathbf{w}$ the disturbance sequence $(w(0), w(1), \ldots)$; $\phi(i; x, \mathbf{u}, \mathbf{w})$ denotes the solution of (3.7) at time i if the initial state is x, and the control and disturbance sequences are, respectively, $\mathbf{u}$ and $\mathbf{w}$. The *nominal* system is described by

$$x^+ = \bar{f}(x, u) := f(x, u, 0) \tag{3.8}$$

and $\bar{\phi}(i;x,\mathbf{u})$ denotes the solution of the nominal system (3.8) at time i if the initial state is x and the control sequence is $\mathbf{u}$. The *nominal* control problem, defined subsequently, includes, for reasons discussed in Chapter 2, a terminal constraint

$$x(N) \in \mathbb{X}_f$$

The *nominal* optimal control problem is

$$\begin{aligned}\mathbb{P}_N(x): \qquad & V_N^0(x) = \min_{\mathbf{u}}\{V_N(x,\mathbf{u}) \mid \mathbf{u} \in \mathcal{U}_N(x)\} \\ & \mathbf{u}^0(x) = \arg\min_{\mathbf{u}}\{V_N(x,\mathbf{u}) \mid \mathbf{u} \in \mathcal{U}_N(x)\}\end{aligned}$$

in which $\mathbf{u}^0 = \left(u_0^0(x), u_1^0(x), \ldots, u_{N-1}^0(x)\right)$ and the nominal cost $V_N(\cdot)$ is defined by

$$V_N(x,\mathbf{u}) := \sum_{i=0}^{N-1} \ell(x(i),u(i)) + V_f(x(N)) \tag{3.9}$$

In (3.9) and (3.10), $x(i) := \bar{\phi}(i;x,\mathbf{u})$, the state of the nominal system at time i, for all $i \in \mathbb{I}_{0:N-1} = \{0,1,2,\ldots,N-1\}$. The set of *admissible* control sequences $\mathcal{U}_N(x)$ is defined by

$$\mathcal{U}_N(x) := \{\mathbf{u} \mid u(i) \in \mathbb{U} \ \forall i \in \mathbb{I}_{0:N-1},\ x(N) \in \mathbb{X}_f\} \tag{3.10}$$

which is the set of control sequences such that the nominal system satisfies the nominal control and terminal constraints when the initial state at time zero is x. Thus, $\mathcal{U}_N(x)$ is the set of feasible controls for the nominal optimal control problem $\mathbb{P}_N(x)$. The set $\mathcal{X}_N \subset \mathbb{R}^n$, defined by

$$\mathcal{X}_N := \{x \in \mathbb{R}^n \mid \mathcal{U}_N(x) \neq \varnothing\}$$

is the domain of the value function $V_N^0(\cdot)$, i.e., the set of x for which $\mathbb{P}_N(x)$ has a solution; $\mathcal{X}_N$ is also the domain of the minimizer $\mathbf{u}^0(x)$. The value of the nominal control at state x is $u^0(0;x)$, the first control in the sequence $\mathbf{u}^0(x)$. Hence the *implicit* nominal MPC control law is $\kappa_N : \mathcal{X}_N \to \mathbb{U}$ defined by

$$\kappa_N(x) = u^0(0;x)$$

We assume, as before, that $\ell(\cdot)$ and $V_f(\cdot)$ are defined by

$$\ell(x,u) := (1/2)(x'Qx + u'Ru) \qquad V_f(x) := (1/2)x'P_f x$$

in which Q, R, and P_f are all positive definite. We also assume that $V_f(\cdot)$ and $\mathbb{X}_f := \{x \mid V_f(x) \le c_f\}$ for some $c_f > 0$ satisfy the standard stability assumption that, for all $x \in \mathbb{X}_f$, there exists a $u = \kappa_f(x) \in \mathbb{U}$ such that $V_f(\bar{f}(x,u)) \le V_f(x) - \ell(x,u)$ and $\bar{f}(x,u) \in \mathbb{X}_f$. Because $V_f(\cdot)$ is quadratic, there exist positive constants c_1^f and c_2^f such that $c_1^f |x|^2 \le V_f(x) \le c_2^f |x|^2$ and $V_f(\bar{f}(x,\kappa_f(x))) \le V_f(x) - c_1^f |x|^2$.

Under these assumptions, as shown in Chapter 2, there exist positive constants c_1 and c_2, $c_2 > c_1$, satisfying

$$c_1 |x|^2 \le V_N^0(x) \le c_2 |x|^2 \tag{3.11}$$

$$V_N^0(\bar{f}(x,\kappa_N(x))) \le V_N^0(x) - c_1 |x|^2 \tag{3.12}$$

for all $x \in \mathcal{X}_N$. It then follows that

$$V_N^0(x^+) \le \gamma V_N^0(x)$$

for all $x \in \bar{\mathcal{X}}_N$ with $x^+ := \bar{f}(x,\kappa_N(x))$ and $\gamma = (1 - c_1/c_2) \in (0, 1)$. Hence, $\bar{V}_N^0(x(i))$ decays exponentially to zero as $i \to \infty$; moreover, $V_N^0(x(i)) \le \gamma^i V_N^0(x(0))$ for all $i \in \mathbb{I}_{\ge 0}$. From (3.11), the origin is exponentially stable, with a region of attraction $\bar{\mathcal{X}}_N$ for the nominal system under MPC.

We now examine the consequences of applying the nominal model predictive controller $\kappa_N(\cdot)$ to the *uncertain* system (3.7). The controlled uncertain system satisfies the difference equation

$$x^+ = f(x,\kappa_N(x),w) \tag{3.13}$$

in which w can take any value in $\mathbb{W}$. It is obvious that the state $x(i)$ of the controlled system (3.13) cannot tend to the origin as $i \to \infty$; the best that can be hoped for is that $x(i)$ tends to and remains in some neighborhood R_b of the origin. We shall establish this, if the disturbance w is sufficiently small, using the value function $V_N^0(\cdot)$ of the nominal optimal control problem as a Lyapunov function for the controlled uncertain system (3.13).

To analyze the effect of the disturbance w we employ the following useful technical result (Allan, Bates, Risbeck, and Rawlings, 2017, Proposition 20).

Proposition 3.4 (Bound for continuous functions)**.** *Let $C \subseteq D \subseteq \mathbb{R}^n$ with C compact and D closed. If $f(\cdot)$ is continuous, there exists an $\alpha(\cdot) \in \mathcal{K}_\infty$ such that, for all $x \in C$ and $y \in D$, we have that $|f(x) - f(y)| \le \alpha(|x - y|)$.*

Since $\mathcal{X}_N$ is not necessarily robustly positive invariant (see Definition 3.6) for the uncertain system $x^+ = f(x, \kappa_N(x), w)$, we replace it by a subset, $R_c := \text{lev}_c\, V_N^0 = \{x \mid V_N^0(x) \le c\}$ for $c > 0$, a sublevel set of $V_N^0(\cdot)$ contained in $\mathcal{X}_N$. Let R_b denote $\text{lev}_b\, V_N^0 = \{x \mid V_N^0(x) \le b\}$, the smallest sublevel set containing $\mathbb{X}_f$. Because $V_N^0(\cdot)$ is lower semicontinuous (Rockafellar and Wets, 1998, Exercise 1.19) and $V_N^0(x) \ge c_1 |x|^2$, both R_b and R_c are compact. We show below, if $\mathbb{W}$ is sufficiently small, then R_b and R_c are robustly positive invariant for the uncertain system $x^+ = f(x, \kappa_N(x), w)$, $w \in \mathbb{W}$ and every trajectory of $x^+ = f(x, \kappa_N(x), w)$, commencing at a state $x \in R_c$, converges to R_b and thereafter remains in this set.

Satisfaction of the terminal constraint. Our first task is to show that the terminal constraint $x(N) \in \mathbb{X}_f$ is satisfied by the uncertain system if $\mathbb{W}$ is sufficiently small. Let $\mathbf{u}^*(x) := \left(u_1^0(x), u_2^0(x), \ldots, u_{N-1}^0(x)\right)$ and let $\tilde{\mathbf{u}}(x) := \left(\mathbf{u}^*(x), \kappa_f(x^0(N; x))\right)$. Since $V_f^*(\cdot)$ defined by

$$V_f^*(x, \mathbf{u}) := V_f(\bar{\phi}(N; x, \mathbf{u}))$$

is continuous, it follows from Proposition 3.4 that there exists a $\mathcal{K}_\infty$ function $\alpha_a(\cdot)$ such that

$$\left| V_f^*(x^+, \mathbf{u}) - V_f^*(\bar{x}^+, \mathbf{u}) \right| \le \alpha_a(|x^+ - \bar{x}^+|)$$

for all $(\bar{x}^+, \mathbf{u}) \in R_c \times \mathbb{U}^N$ and all $(x^+, \mathbf{u}) \in f(R_c, \mathbb{U}, \mathbb{W}) \times \mathbb{U}^N$. This result holds, in particular, for $\bar{x}^+ := f(x, \kappa_N(x), 0)$, $x^+ := f(x, \kappa_N(x), w)$ and $\mathbf{u} = \tilde{\mathbf{u}}(x)$ with $x \in R_c$. As shown in Chapter 2, $x^0(N; \bar{x}^+) \in \mathbb{X}_f$; we wish to show $x^0(N; x^+) \in \mathbb{X}_f$.

Since $V_f^*(\bar{x}^+, \tilde{\mathbf{u}}(x)) = V_f(f(x^0(N; x), \kappa_f(x^0(N; x)))) \le \gamma_f c_f$ and since $V_f^*(x^+, \tilde{\mathbf{u}}(x)) \le V_f^*(\bar{x}^+, \tilde{\mathbf{u}}(x)) + \alpha_a(|x^+ - \bar{x}^+|)$ it follows that $V_f(x^0(N; x^+)) \le V_f(x^0(N; \bar{x}^+)) + \alpha_a(|x^+ - \bar{x}^+|) \le$
$\gamma_f c_f + \alpha_a(|x^+ - \bar{x}^+|)$. Hence, $x^0(N; x) \in \mathbb{X}_f$ implies $x^0(N; x^+) \in \mathbb{X}_f$ if $\alpha_a(|x^+ - \bar{x}^+|) \le (1 - \gamma_f) c_f$.

Robust positive invariance of R_c for the controlled uncertain system. Suppose $x \in R_c$. Since $V_N(\cdot)$ is continuous, it follows from Proposition 3.4 that there exists a $\mathcal{K}_\infty$ function $\alpha_b(\cdot)$ such that

$$\left| V_N(x^+, \mathbf{u}) - V_N(\bar{x}^+, \mathbf{u}) \right| \le \alpha_b(|x^+ - \bar{x}^+|)$$

for all $(x^+, \mathbf{u}) \in f(R_c, \mathbb{U}, \mathbb{W}) \times \mathbb{U}^N$, all $(\bar{x}^+, \mathbf{u}) \in R_c \times \mathbb{U}^N$. This result holds in particular for $x^+ = f(x, \kappa_N(x), w)$, $\bar{x}^+ = f(x, \kappa_N(x), 0)$ and

$\mathbf{u} = \tilde{\mathbf{u}}(x)$ with $x \in R_c$. Hence, if $x \in R_c$

$$V_N(x^+, \tilde{\mathbf{u}}(x)) \le V_N(\bar{x}^+, \tilde{\mathbf{u}}(x)) + \alpha_b(|x^+ - \bar{x}^+|)$$

Since $V_N(x^+, \tilde{\mathbf{u}}(x)) \le V_N^0(x) - c_1|x|^2$ and, since the control $\tilde{\mathbf{u}}(x)$, $x \in R_c$ satisfies both the control and terminal constraints if $\alpha_a(|x^+ - \bar{x}^+|) \le (1-\gamma_f)c_f$, it follows that

$$V_N^0(x^+) \le V_N(x^+, \tilde{\mathbf{u}}(x)) \le V_N^0(x) - c_1|x|^2 + \alpha_b(|x^+ - \bar{x}^+|)$$

so that

$$V_N^0(x^+) \le \gamma V_N^0(x) + \alpha_b(|x^+ - \bar{x}^+|)$$

Hence $x \in R_c$ implies $x^+ = f(x, \kappa_N(x), w) \in R_c$ for all $w \in \mathbb{W}$ if $\alpha_a(|x^+ - \bar{x}^+|) \le (1-\gamma_f)c_f$ and $\alpha_b(|x^+ - \bar{x}^+|) \le (1-\gamma)c$.

Robust positive invariance of R_b for the controlled uncertain system. Similarly, $x \in R_b$ implies $x^+ = f(x, \kappa_N(x), w) \in R_b$ for all $w \in \mathbb{W}$ if $\alpha_a(|x^+ - \bar{x}^+|) \le (1-\gamma_f)c_f$ and $\alpha_b(|x^+ - \bar{x}^+|) \le (1-\gamma)b$.

Descent property of $V_N^0(\cdot)$ in $R_c \setminus R_b$. Suppose that $x \in R_c \setminus R_b$ and that $\alpha_a(|x^+ - \bar{x}^+|) \le (1-\gamma_f)c_f$. Then because $\tilde{\mathbf{u}}(x) \in \mathcal{U}_N(x^+)$, we have that $\mathbb{P}_N(x^+)$ is feasible and thus $V_N^0(x^+)$ is well defined. As above, we have that $V_N^0(x^+) \le \gamma V_N^0(x) + \alpha_b(|x^+ - \bar{x}^+|)$. Let $\gamma^* \in (\gamma, 1)$. If $\alpha_b(|x^+ - \bar{x}^+|) \le (\gamma^* - \gamma)b$, we have that

$$\begin{aligned} V_N^0(x^+) &\le \gamma V_N^0(x) + (\gamma^* - \gamma)b \\ &< \gamma V_N^0(x) + (\gamma^* - \gamma)V_N^0(x) \\ &= \gamma^* V_N^0(x) \end{aligned}$$

because $V_N^0(x) > b$.

Summary. These conditions can be simplified if we assume that $f(\cdot)$ is uniformly Lipschitz continuous in w with Lipschitz constant L so that $|f(x^+\kappa_N(x), w) - f(x, \kappa_N(x), 0)| \le L|w|$ for all $(x,u) \in R_c \times \mathbb{U}$. The function $f(\cdot)$ has this property with $L = 1$ if $f(x,u,w) = f'(x, u) + w$. Under this assumption, the four conditions become

1. $\alpha_a(|Lw|) \le (1-\gamma_f)c_f$
2. $\alpha_a(|Lw|) \le (1-\gamma)c$
3. $\alpha_b(|Lw| \le (1-\gamma)c$
4. $\alpha_b(|Lw|) \le (\gamma^* - \gamma)b$

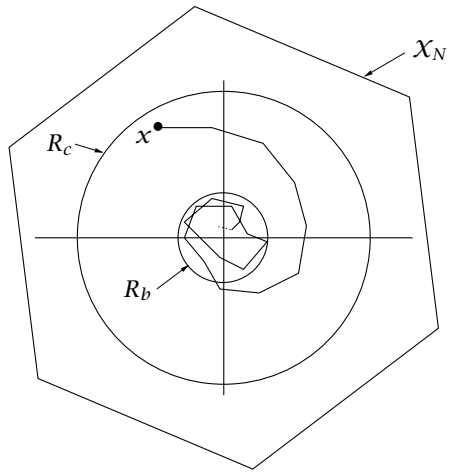


Figure 3.2: The sets $\mathcal{X}_N$, R_b, and R_c.

Let $\delta^* > 0$ denote the largest δ such that all four conditions are satisfied if $w \in \mathbb{W}$ with $|\mathbb{W}| \leq \delta$.[4] Condition 3 can be satisfied if $b \geq \delta^*/(1-\gamma)$.

Proposition 3.5 (Robustness of nominal MPC)**.** *Suppose all assumptions in Section 3.2.4 are satisfied and that $|\mathbb{W}| \leq \delta^*$ and $c > b$. Then, any initial state $x \in R_c$ of the controlled system $x^+ = f(x, \kappa_N(x), w)$ is steered to the set R_b in finite time for all admissible disturbance sequences* $\mathbf{w}$ *satisfying $w(i) \in \mathbb{W}$ for all $i \in \mathbb{I}_{\geq 0}$. Thereafter, the state remains in R_b for all admissible disturbance sequences.*

Figure 3.2 illustrates this result.

3.3 Min-Max Optimal Control: Dynamic Programming Solution

3.3.1 Introduction

In this section we show how robust control of an uncertain system may be achieved using dynamic programming (DP). Our purpose here is to use DP to gain insight. The results we obtain here are not of practical use for complex systems, but reveal the nature of the problem and show what the ideal optimal control problem solved online should be.

[4] $|\mathbb{W}| := \max_w \{|w| \mid w \in \mathbb{W}\}$

In Section 3.2 we examined the inherent robustness of an asymptotically stable system. If uncertainty is present, and it always is, it is preferable to design the controller to be *robust*, i.e., able to cope with some uncertainty. In this section we discuss the design of a robust controller for the system

$$x^+ = f(x, u, w) \tag{3.14}$$

in which a bounded disturbance input w models the uncertainty. The disturbance is assumed to satisfy $w \in \mathbb{W}$ where $\mathbb{W}$ is compact convex, and contains the origin in its interior. The controlled system is required to satisfy the same state and control constraints as above, namely $(x, u) \in \mathbb{Z}$ as well as a terminal constraint $x(N) \in \mathbb{X}_f$. The constraint $(x, u) \in \mathbb{Z}$ may be expressed equivalently as $x \in \mathbb{X}$ and $u \in \mathbb{U}(x)$ in which $\mathbb{X} = \{x \mid \exists\ u \text{ such that } (x, u) \in \mathbb{Z}\}$ and $\mathbb{U}(x) = \{u \mid (x, u) \in \mathbb{Z}\}$. Because of the disturbance, superior control may be achieved by employing feedback, in the form of a control *policy*, i.e., a sequence of control *laws* rather than employing open-loop control in the form of a sequence of control *actions*. Each control law is a function that maps states into control actions; if the control law at time i is $\mu_i(\cdot)$, then the system at time i satisfies $x(i+1) = f(x(i), \mu_i(x(i)))$. Because of uncertainty, feedback and open-loop control for a given initial state are not equivalent.

The solution at time k of (3.14) with control and disturbance sequences $\mathbf{u} = (u(0), \ldots, u(N-1))$ and $\mathbf{w} = (w(0), \ldots, w(N-1))$ if the initial state is x at time 0 is $\phi(k; x, \mathbf{u}, \mathbf{w})$. Similarly, the solution at time k due to feedback policy $\boldsymbol{\mu} = (\mu_0(\cdot), \ldots, \mu_{N-1}(\cdot))$ and disturbance sequence $\mathbf{w}$ is denoted by $\phi(k; x, \boldsymbol{\mu}, \mathbf{w})$. As discussed previously, the cost may be taken to be that of the nominal trajectory, or the average, or maximum taken over all possible realizations of the disturbance sequence. Here we employ, as is common in the literature, the maximum over all realizations of the disturbance sequence $\mathbf{w}$, and define the cost due to policy $\boldsymbol{\mu}$ with initial state x to be

$$V_N(x, \boldsymbol{\mu}) := \max_{\mathbf{w}} \{J_N(x, \boldsymbol{\mu}, \mathbf{w}) \mid \mathbf{w} \in \mathcal{W}\} \tag{3.15}$$

in which $\mathcal{W} = \mathbb{W}^N$ is the set of admissible disturbance sequences, and $J_N(x, \boldsymbol{\mu}, \mathbf{w})$ is the cost due to an individual realization $\mathbf{w}$ of the disturbance process and is defined by

$$J_N(x, \boldsymbol{\mu}, \mathbf{w}) := \sum_{i=0}^{N-1} \ell(x(i), u(i), w(i)) + V_f(x(N)) \tag{3.16}$$

in which $\boldsymbol{\mu} = (\mu_0(\cdot), \mu_1(\cdot), \ldots, \mu_{N-1}(\cdot))$, $x(i) = \phi(i; x, \mu, \mathbf{w})$, and $u(i) = \mu_i(x(i))$. Let $\mathcal{M}(x)$ denote the set of feedback policies $\boldsymbol{\mu}$ that for a given initial state x satisfy: the state and control constraints, and the terminal constraint for every admissible disturbance sequence $\mathbf{w} \in \mathcal{W}$. The first control law $\mu_0(\cdot)$ in $\boldsymbol{\mu}$ may be replaced by a control action $u_0 = \mu_0(x)$ to simplify optimization, since the initial state x is known whereas future states are uncertain. The set of admissible control policies $\mathcal{M}(x)$ is defined by

$$\mathcal{M}(x) := \{\boldsymbol{\mu} \mid \mu_0(x) \in \mathbb{U}(x),\ \phi(i; x, \boldsymbol{\mu}, \mathbf{w}) \in \mathbb{X},\ \mu_i(\phi(i; x, \boldsymbol{\mu}, \mathbf{w})) \in \mathbb{U}(x) \\ \forall i \in \mathbb{I}_{0:N-1},\ \phi(N; x, \boldsymbol{\mu}, \mathbf{w}) \in \mathbb{X}_f\ \forall \mathbf{w} \in \mathcal{W}\}$$

The robust optimal control problem is

$$\mathbb{P}_N(x): \quad \inf_{\boldsymbol{\mu}} \{V_N(x, \boldsymbol{\mu}) \mid \boldsymbol{\mu} \in \mathcal{M}(x)\} \tag{3.17}$$

The solution to $\mathbb{P}_N(x)$, if it exists, is the policy $\boldsymbol{\mu}^0(x)$

$$\boldsymbol{\mu}^0(x) = \left(\mu_0^0(\cdot; x), \mu_1^0(\cdot; x), \ldots, \mu_{N-1}^0(\cdot, x)\right)$$

and the value function is $V_N^0(x) = V_N(x, \boldsymbol{\mu}^0(x))$.

Dynamic programming solves problem $\mathbb{P}_N(x)$ with horizon N for all x such that the problem is feasible, yielding the optimal control policy $\boldsymbol{\mu}^0(\cdot) = (\mu_0(\cdot), \ldots, \mu_{N-1}(\cdot))$ for the optimal control problem with horizon N. In doing so, it also solves, for each $i \in \mathbb{I}_{1:N}$, problem $\mathbb{P}_i(x)$ yielding the optimal control policy for the problem with horizon i.

3.3.2 Properties of the Dynamic Programming Solution

As for deterministic optimal control, the value function and implicit control law may, in principle, be obtained by DP. But DP is, in most cases, impossible to use because of its large computational demands. There are, of course, important exceptions such as H_2 and H_∞ optimal control for unconstrained linear systems with quadratic cost functions. DP also can be used for low dimensional constrained optimal control problems when the system is linear, the constraints are affine, and the cost is affine or quadratic. Even when DP is computationally prohibitive, however, it remains a useful tool because of the insight it provides. Because of the cost definition, min-max DP is required. For each $i \in \{0, 1, \ldots, N\}$, let $V_i^0(\cdot)$ and $\kappa_i(\cdot)$ denote, respectively, the partial value function and the optimal solution to the optimal control

problem $\mathbb{P}_i$ defined by (3.17) with i replacing N. The DP recursion equations for computing these functions are

$$V_i^0(x) = \min_{u\in\mathbb{U}(x)} \max_{w\in\mathbb{W}} \{\ell(x,u,w) + V_{i-1}^0(f(x,u,w)) \mid f(x,u,\mathbb{W}) \subseteq \mathcal{X}_{i-1}\}$$

$$\kappa_i(x) = \arg \min_{u\in\mathbb{U}(x)} \max_{w\in\mathbb{W}} \{\ell(x,u,w) + V_{i-1}^0(f(x,u,\mathbb{W})) \mid f(x,u,\mathbb{W}) \subseteq \mathcal{X}_{i-1}\}$$

$$\mathcal{X}_i = \{x \in \mathbb{X} \mid \exists\ u \in \mathbb{U}(x) \text{ such that } f(x,u,\mathbb{W}) \subseteq \mathcal{X}_{i-1}\}$$

with boundary conditions

$$V_0^0(x) = V_f(x) \qquad \mathcal{X}_0 = \mathbb{X}_f$$

In these equations, the subscript i denotes time to go so that $\kappa_i(\cdot) := \mu_{N-i}(\cdot)$ (equivalently $\mu_i(\cdot) := \kappa_{N-i}(\cdot)$). In particular, $\kappa_N(\cdot) = \mu_0(\cdot)$. For each i, $\mathcal{X}_i$ is the domain of $V_i^0(\cdot)$ (and $\kappa_i(\cdot)$) and is therefore the set of states x for which a solution to problem $\mathbb{P}_i(x)$ exists. Thus $\mathcal{X}_i$ is the set of states that can be *robustly* steered by state feedback, i.e., by a policy $\boldsymbol{\mu} \in \mathcal{M}(x)$, to $\mathbb{X}_f$ in i steps or less satisfying all constraints for all disturbance sequences. It follows from these definitions that

$$V_i^0(x) = \max_{w\in\mathbb{W}} \{\ell(x,\kappa_i(x),w) + V_{i-1}^0(f(x,\kappa_i(x),w))\} \tag{3.18}$$

as discussed in Exercise 3.1.

As in the deterministic case studied in Chapter 2, we are interested in obtaining conditions that ensure that the optimal finite horizon control law $\kappa_0^0(\cdot)$ is stabilizing. To do this we replace the stabilizing Assumption 2.14 in Section 2.4.2 of Chapter 2 by conditions appropriate to the robust control problem. The presence of a disturbance requires us to generalize some earlier definitions; we therefore define the terms *robustly control invariant* and *robustly positive invariant* that generalize our previous definitions of *control invariant* and *positive invariant* respectively.

Definition 3.6 (Robust control invariance)**.** A set $X \subseteq \mathbb{R}^n$ is *robustly control invariant* for $x^+ = f(x,u,w)$, $w \in \mathbb{W}$ if, for every $x \in X$, there exists a $u \in \mathbb{U}(x)$ such that $f(x,u,\mathbb{W}) \subseteq X$.

Definition 3.7 (Robust positive invariance)**.** A set X is *robustly positive invariant* for $x^+ = f(x,w)$, $w \in \mathbb{W}$ if, for every $x \in X$, $f(x,\mathbb{W}) \subseteq X$.

As in Chapter 2, stabilizing conditions are imposed on the ingredients $\ell(\cdot), V_f(\cdot)$, and $\mathbb{X}_f$ of the optimal control problem to ensure that

the resultant controlled system has desirable stability properties; the solution to a finite horizon optimal control problem does not necessarily ensure stability. Our new assumption is a robust generalization of the stabilizing Assumption 2.2 employed in Chapter 2.

Assumption 3.8 (Basic stability assumption; robust case)**.**

(a) For all $x \in \mathbb{X}_f$ there exists a $u = \kappa_f(x) \in \mathbb{U}(x)$ such that

$$V_f(f(x,u,0)) \leq V_f(x) - \ell(x,u,0) \text{ and } f(x,u,w) \in \mathbb{X}_f \ \forall w \in \mathbb{W}$$

(b) $\mathbb{X}_f \subseteq \mathbb{X}$

(c) There exist $\mathcal{K}_\infty$ functions $\alpha_1(\cdot)$ and $\alpha_f(\cdot)$ satisfying

$$\ell(x,u,w) \geq \alpha_1(|x|) \ \forall (x,w) \in \mathbb{R}^n \times \mathbb{W} \ \forall u \text{ such that } (x,u) \in \mathbb{Z}$$
$$V_f(x) \leq \alpha_f(|x|), \ \forall x \in \mathbb{X}_f$$

Assumption 3.8(a) replaces the unrealistic assumption in the first edition that, for each $x \in \mathbb{X}_f$, there exists a $u \in \mathbb{U}$ such that, for all $w \in \mathbb{W}$, $V_f(f(x,u,w)) \leq V_f(x) - \ell(x,u,w)$ and $f(x,u,w) \in \mathbb{X}_f$. Let $\delta \in \mathbb{R}_{\geq 0}$ be defined by

$$\delta := \max_{(x,w) \in \mathbb{X}_f \times \mathbb{W}} \{V_f(f(x,\kappa_f(x),w)) - V_f(x) + \ell(x,\kappa_f(x),w)\}$$

so that, if Assumption 3.8 holds

$$V_f(f(x,\kappa_f(x),w)) \leq V_f(x) - \ell(x,u,w) + \delta \ \forall (x,w) \in \mathbb{X}_f \times \mathbb{W} \quad (3.19)$$

If $\delta = 0$, the controller $\kappa_f(\cdot)$ can steer any $x \in \mathbb{X}_f$ to the origin despite the disturbance.

Theorem 3.9 (Recursive feasibility of control policies)**.** *Suppose Assumption 3.8 holds. Then*

(a) $\mathcal{X}_N \supseteq \mathcal{X}_{N-1} \supseteq \ldots \supseteq \mathcal{X}_1 \supseteq \mathcal{X}_0 = \mathbb{X}_f$

(b) $\mathcal{X}_i$ *is robustly control invariant for* $x^+ = f(x,u,w) \ \forall i \in \mathbb{I}_{0:N}$

(c) $\mathcal{X}_i$ *is robustly positive invariant for* $x^+ = f(x,\kappa_i(x),w), \ \forall i \in \mathbb{I}_{0:N}$

(d) $[V^0_{i+1} - V^0_i](x) \leq \max_{w \in \mathbb{W}} \{[V^0_i - V^0_{i-1}](f(x,\kappa_i(x),w))\} \ \forall x \in \mathcal{X}_i$, $\forall i \in \mathbb{I}_{1:N-1}$. *Also* $V^0_i(x) - V^0_{i-1}(x) \leq \delta \ \forall x \in \mathcal{X}_{i-1}, \ \forall i \in \{1,\ldots,N\}$ *and* $V^0_i(x) \leq V_f(x) + i\delta \ \forall \ x \in X_f, \ \forall i \in \mathbb{I}_{1:N}$

(e) For any $x \in \mathcal{X}_N$, $\left(\kappa_N(x), \kappa_{N-1}(\cdot), \ldots, \kappa_1(\cdot), \kappa_f(\cdot)\right)$ *is a feasible policy for* $\mathbb{P}_{N+1}(x)$*, and, for any* $x \in \mathcal{X}_{N-1}$, $(\kappa_{N-1}(x), \kappa_{N-2}(\cdot), \ldots, \kappa_1(\cdot), \kappa_f(\cdot))$ *is a feasible policy for* $\mathbb{P}_N(x)$.

Proof.

(a)–(c) Suppose, for some i, $\mathcal{X}_i$ is robust control invariant so that any point $x \in \mathcal{X}_i$ can be robustly steered into $\mathcal{X}_i$. By construction, $\mathcal{X}_{i+1}$ is the set of all points x that can be robustly steered into $\mathcal{X}_i$. Also $\mathcal{X}_{i+1} \supseteq \mathcal{X}_i$ so that $\mathcal{X}_{i+1}$ is robust control invariant. But $\mathcal{X}_0 = \mathbb{X}_f$ is robust control invariant. Both (a) and (b) follow by induction. Part (c) follows from (b).

(d) From (3.18) we have

$$\begin{aligned}
[V_{i+1}^0 - V_i^0](x) &= \max_{w\in\mathbb{W}}\{\ell(x,\kappa_{i+1}(x),w) + V_i^0(f(x,\kappa_{i+1}(x),w))\} \\
&\quad - \max_{w\in\mathbb{W}}\{\ell(x,\kappa_i(x),w) + V_{i-1}^0(f(x,\kappa_i(x),w))\} \\
&\le \max_{w\in\mathbb{W}}\{\ell(x,\kappa_i(x),w) + V_i^0(f(x,\kappa_i(x),w))\} \\
&\quad - \max_{w\in\mathbb{W}}\{\ell(x,\kappa_i(x),w) + V_{i-1}^0(f(x,\kappa_i(x),w))\}
\end{aligned}$$

for all $x \in \mathcal{X}_i$ since $\kappa_i(\cdot)$ may *not* be optimal for problem $\mathbb{P}_{i+1}(x)$. We now use the fact that $\max_w\{a(w)\} - \max_w\{b(w)\} \le \max_w\{a(w) - b(w)\}$, which is discussed in Exercise 3.2, to obtain

$$[V_{i+1}^0 - V_i^0](x) \le \max_{w\in\mathbb{W}}\{[V_i^0 - V_{i-1}^0](f(x,\kappa_i(x),w))\}$$

for all $x \in X_i$. Also, for all $x \in \mathcal{X}_0 = \mathbb{X}_f$

$$[V_1^0 - V_0^0](x) = \max_{w\in\mathbb{W}}\{\ell(x,\kappa_1(x),w) + V_f(f(x,\kappa_1(x),w)) - V_f(x)\} \le \delta$$

in which the last inequality follows from Assumption 3.8. By induction, $V_i^0(x) - V_{i-1}^0(x) \le \delta \ \forall x \in \mathcal{X}_{i-1}, \ \forall i \in \{1,\ldots,N\}$. It follows that $V_i^0(x) \le V_f(x) + i\delta$ for all $x \in \mathbb{X}_f$, all $i \in \{1,\ldots,N\}$.

(e) Suppose $x \in \mathcal{X}_N$. Then $\boldsymbol{\kappa}^0(x) = (\kappa_N(x), \kappa_{N-1}(\cdot),\ldots,\kappa_1(\cdot))$ is a feasible and optimal policy for problem $\mathbb{P}_N(x)$, and steers every trajectory emanating from x into $\mathcal{X}_0 = \mathbb{X}_f$ in N time steps. Because $\mathbb{X}_f$ is robustly positive invariant for $x^+ = f(x,\kappa_f(x),w)$, $w \in \mathbb{W}$, the policy $\big(\kappa_N(x), \kappa_{N-1}(\cdot),\ldots,\kappa_1(\cdot),\kappa_f(\cdot)\big)$ is feasible for problem $\mathbb{P}_{N+1}(x)$. Similarly, the policy $(\kappa_{N-1}(x), \kappa_{N-2}(\cdot),\ldots,\kappa_1(\cdot))$ is feasible and optimal for problem $\mathbb{P}_{N-1}(x)$, and steers every trajectory emanating from $x \in \mathcal{X}_{N-1}$ into $\mathcal{X}_0 = \mathbb{X}_f$ in $N-1$ time steps. Therefore the policy $\big(\kappa_{N-1}(x), \kappa_{N-2}(\cdot),\ldots,\kappa_1(\cdot),\kappa_f(\cdot)\big)$ is feasible for $\mathbb{P}_N(x)$ for any $x \in \mathcal{X}_{N-1}$. ■

3.4 Robust Min-Max MPC

Because use of dynamic programming (DP) is usually prohibitive, obtaining an alternative, robust min-max model predictive control, is desirable. We present here an analysis that uses the improved stability condition Assumption 3.8. The system to be controlled is defined in (3.14) and the cost function $V_N(\cdot)$ in (3.15) and (3.16). The decision variable, which, in DP, is a sequence $\boldsymbol{\mu} = (\mu_0(\cdot), \mu_1(\cdot), \dots, \mu_{N-1}(\cdot))$ of control laws, each of which is an arbitrary function of the state x, is too complex for online optimization; so, we replace $\boldsymbol{\mu}$ by the simpler object $\boldsymbol{\mu}(\mathbf{v}) := (\mu(\cdot, v_0), \mu(\cdot, v_1), \dots \mu(\cdot, v_{N-1}))$ in which $\mathbf{v} = (v_0, v_1, \dots, v_{N-1})$ is a sequence of parameters with $\mu(\cdot)$ parameterized by v_i, $i \in \mathbb{I}_{0:N-1}$.

A simple parameterization is $\boldsymbol{\mu}(\mathbf{v}) = \mathbf{v} = (v_0, v_1, \dots, v_{N-1})$, a sequence of control *actions* rather than control *laws*. The decision variable $\mathbf{v}$ in this case is similar to the control sequence $\mathbf{u}$ used in deterministic MPC, and is simple enough for implementation; the disadvantage is that feedback is not allowed in the optimal control problem $\mathbb{P}_N(x)$. Hence the predicted trajectories may diverge considerably. An equally simple parameterization that has proved to be useful when the system being controlled is linear and time invariant is $\boldsymbol{\mu}(\mathbf{v}) = (\mu(\cdot, v_0) \dots, \mu(\cdot, v_{N-1}))$ in which, for each i, $\mu(x, v_i) := v_i + Kx$; if $f(x, u, w) = Ax + Bu + w$, K is chosen so that $A_K := A + BK$ is Hurwitz. More generally, $\mu(x, v_i) := \sum_{j \in J} v_i^j \theta_j(x) = \langle v_i, \theta(x) \rangle$, $\theta(x) := (\theta_1(x), \theta_2(x), \dots, \theta_J(x))$. Hence the policy sequence $\boldsymbol{\mu}(\mathbf{v})$ is parameterized by the vector sequence $\mathbf{v} = (v_0, v_1, \dots, v_{N-1})$. Choosing appropriate basis functions $\theta_j(\cdot)$, $j \in J$, is not simple. The decision variable is the vector sequence $\mathbf{v}$.

With this parameterization, the optimal control problem $\mathbb{P}_N(x)$ becomes

$$\mathbb{P}_N(x): \qquad V_N^0(x) = \min_{\mathbf{v}} \{V_N(x, \boldsymbol{\mu}(\mathbf{v})) \mid \mathbf{v} \in \mathcal{V}_N(x)\}$$

in which

$$V_N(x, \boldsymbol{\mu}(\mathbf{v})) := \max_{\mathbf{w}} \{J_N(x, \boldsymbol{\mu}(\mathbf{v}), \mathbf{w}) \mid \mathbf{w} \in \mathbb{W}^N\}$$

$$J_N(x, \boldsymbol{\mu}(\mathbf{v}), \mathbf{w}) := \sum_{i=0}^{N-1} \ell(x(i), u(i), w(i)) + V_f(x(N))$$

$$\mathcal{V}_N(x) := \{\mathbf{v} \mid (x(i), u(i)) \in \mathbb{Z},\ \forall i \in \mathbb{I}_{0:N-1},\ x(N) \in \mathbb{X}_f,\ \forall \mathbf{w} \in \mathbb{W}^N\}$$

with $x(i) := \phi(i; x, \boldsymbol{\mu}(\mathbf{v}), \mathbf{w})$ and $u(i) = \mu(x(i), v_i)$; $\phi(i; x, \boldsymbol{\mu}(\mathbf{v}), \mathbf{w})$ denotes the solution at time i of $x(i+1) = f(x(i), u(i), w(i))$ with $x(0) = x$, $u(i) = \mu(x(i), v_i)$ for all $i \in \mathbb{I}_{0:N-1}$, and disturbance sequence $\mathbf{w}$. Let $\mathbf{v}^0(x)$ denote the minimizing value of the decision variable $\mathbf{v}$, $\boldsymbol{\mu}^0(x) := \boldsymbol{\mu}(\mathbf{v}^0(x))$ the corresponding optimal control policy, and let $V_N^0(x) := V_N(x, \boldsymbol{\mu}^0(x))$ denote the value function. We implicitly assume that a solution to $\mathbb{P}_N(x)$ exists for all $x \in \mathcal{X}_N(x) := \{x \mid \mathcal{V}_N(x) \neq \emptyset\}$ and that $\mathcal{X}_N$ is not empty. The MPC action at state x is $\mu_0^0(x) = \mu(x, v_0^0(x))$, with $v_0^0(x)$ the first element of the optimal decision variable sequence $\mathbf{v}^0(x)$. The implicit MPC law is $\mu_0^0(\cdot)$. To complete the problem definition, we assume that $V_f(\cdot)$ and $\ell(\cdot)$ satisfy Assumption 3.8.

It follows from Assumption 3.8 that there exists a $\mathcal{K}_\infty$ function $\alpha_1(\cdot)$ such that $V_N^0(x) \geq \alpha_1(|x|)$ for all $x \in \mathcal{X}_N$, the domain of $V_N^0(\cdot)$. Determination of an upper bound for $V_N^0(\cdot)$ is difficult, so we *assume* that there exists a $\mathcal{K}_\infty$ function $\alpha_2(\cdot)$ such that $V_N^0(x) \leq \alpha_2(|x|)$ for all $x \in \mathcal{X}_N$. We now consider the descent condition, i.e., we determine an upper bound for $V_N^0(x^+) - V_N^0(x)$ as well as a *warm start* for obtaining, via optimization, the optimal decision sequence $\mathbf{v}^0(x^+)$ given $\mathbf{v}^0(x)$.

Suppose that, at state x, the value function $V_N^0(x)$ and the optimal decision sequence $\mathbf{v}^0(x)$ have been determined, as well as the control action $\mu_0^0(x)$. The subsequent state is $x^+ = f(x, \mu_0^0(x), w_0)$, with w_0 the value of the additive disturbance ($w(t)$ if the current time is t). Let

$$\boldsymbol{\mu}^*(x) := \boldsymbol{\mu}_{1:N-1}^0(x) = \left(\mu(\,\cdot\,, v_1^0(x)), \mu(\,\cdot\,, v_2^0(x)), \ldots, \mu(\,\cdot\,, v_{N-1}^0(x))\right)$$

denote $\boldsymbol{\mu}^0(x)$ with its first element $\mu(\,\cdot\,, v_0^0(x))$ removed; $\boldsymbol{\mu}^*(x)$ is a sequence of $N-1$ control laws. In addition let $\tilde{\mathbf{u}}(x)$ be defined by

$$\tilde{\mathbf{u}}(x) := (\boldsymbol{\mu}^*(x), \kappa_f(\cdot))$$

$\tilde{\mathbf{u}}(x)$ is a sequence of N control laws.

For any sequence $\mathbf{z}$ let $\mathbf{z}_{a:b}$ denote the subsequence $(z(a), z(a+1), \ldots, z(b))$; as above, $\mathbf{z} := \mathbf{z}_{0:N-1}$. Because $x \in \mathcal{X}_N$ is feasible for the optimal control problem $\mathbb{P}_N(x)$, every random trajectory with disturbance sequence $\mathbf{w} = \mathbf{w}_{0:N-1} \in \mathbb{W}^N$ emanating from $x \in \mathcal{X}_N$ under the control policy $\boldsymbol{\mu}^0(x)$ reaches the terminal state $x_N = \phi(N; x, \boldsymbol{\mu}^0(x), \mathbf{w}) \in \mathbb{X}_f$ in N steps. Since $w(0)$ is the first element of $\mathbf{w}$, $\mathbf{w} = (w(0), \mathbf{w}_{1:N-1})$. Hence the random trajectory with control sequence $\boldsymbol{\mu}_{1:N-1}^0(x)$ and disturbance sequence $\mathbf{w}_{1:N-1}$ emanating from $x^+ = f(x, \mu_0^0(x), w(0))$ reaches $x_N \in \mathbb{X}_f$ in $N-1$ steps. Clearly

$$J_{N-1}(x^+, \boldsymbol{\mu}_{1:N-1}^0(x), \mathbf{w}_{1:N-1}) = J_N(x, \boldsymbol{\mu}^0(x), \mathbf{w}) - \ell(x, \mu_0^0(x), w_0)$$

By Assumption 3.8, $\ell(x, \mu_0^0(x), w(0)) = \ell(x, \kappa_N(x), w(0)) \geq \alpha_1(|x|)$ and

$$J_{N-1}(x^+, \boldsymbol{\mu}_{1:N-1}^0(x), \mathbf{w}_{1:N-1}) \leq J_N(x, \boldsymbol{\mu}^0(x), \mathbf{w}) - \alpha_1(|x|)$$

The policy sequence $\widetilde{\boldsymbol{\mu}}(x)$, which appends $\kappa_f(\cdot)$ to $\boldsymbol{\mu}_{1:N-1}^0(x)$, steers x^+ to x_N in $N-1$ steps and then steers $x_N \in \mathbb{X}_f$ to $x(N+1) = f(x_N, \kappa_f(x_N), w_N)$ that lies in the interior of $\mathbb{X}_f$. Using Assumption 3.8, we obtain

$$J_N(x^+, \widetilde{\boldsymbol{\mu}}(x), \mathbf{w}_{1:N}) \leq J_N(x, \boldsymbol{\mu}^0(x), \mathbf{w}) - \alpha_1(|x|) + \delta$$

Using this inequality with $\mathbf{w}_{0:N} = (w(0), \mathbf{w}^0(x^+))$[5] so that $\mathbf{w}_{1:N} = \mathbf{w}^0(x^+)$ and $\mathbf{w} = \mathbf{w}_{0:N-1} = \left(w(0), \mathbf{w}_{0:N-2}^0(x^+)\right)$ yields

$$\begin{aligned} V_N^0(x^+) = J_N(x^+, \boldsymbol{\mu}^0(x^+), \mathbf{w}^0(x^+)) &\leq J_N(x^+, \widetilde{\boldsymbol{\mu}}(x), \mathbf{w}^0(x^+)) \\ &\leq J_N(x, \boldsymbol{\mu}^0(x), (w(0), \mathbf{w}_{0:N-2}^0(x^+)) - \alpha_1(|x|) + \delta \\ &\leq V_N^0(x) - \alpha_1(|x|) + \delta \end{aligned}$$

The last inequality follows from the fact that the disturbance sequence $\left(w(0), \mathbf{w}_{0:N-2}^0(x^+)\right)$ does not necessarily maximize $\mathbf{w} \mapsto J_N(x, \boldsymbol{\mu}^0(x), \mathbf{w})$.

Assume now that $\ell(\cdot)$ is quadratic and positive definite so that $\alpha_1(|x|) \geq c_1 |x|^2$. Assume also that $V_N^0(x) \leq c_2 |x|^2$ so that for all $x \in \mathcal{X}_N$

$$V_N^0(x^+) \leq \gamma V_N^0(x) + \delta$$

with $\gamma = 1 - c_1/c_2 \in (0,1)$. Let $\varepsilon > 0$. It follows that for all $x \in \mathcal{X}_N$ such that $V_N^0(x) \geq c := (\delta + \varepsilon)/(1-\gamma)$

$$V_N^0(x^+) \leq \gamma V_N^0(x) + \delta \leq V_N^0(x) - (1-\gamma)c + \delta \leq V_N^0(x) - \varepsilon$$

since $V_N^0(x) \geq c$ and, by definition, $(1-\gamma)c = \delta + \varepsilon$. Secondly, if x lies in $\operatorname{lev}_c V_N^0$, then

$$V_N^0(x^+) \leq \gamma c + \delta \leq c - \varepsilon$$

since $V_N^0(x) \leq c$ and, by definition, $c = \gamma c + \delta + \varepsilon$. Hence $x \in \operatorname{lev}_c V_N^0$ implies $x^+ \in f(x, \mu_0^0(x), \mathbb{W}) \subset \operatorname{lev}_c V_N^0$.

[5] $\mathbf{w}^0(x^+) := \arg\max_{\mathbf{w} \in \mathbb{W}^N} J_N(x^+, \boldsymbol{\mu}^0(x^+), \mathbf{w})$.

Summary. If $\delta < (1-\gamma)c$ ($c > \delta/(1-\gamma)$) and $\text{lev}_c V_N^0 \subset \mathcal{X}_N$, every initial state $x \in \mathcal{X}_N$ of the closed-loop system $x^+ = f(x, \mu_0^0(x), w)$ is steered to the sublevel set $\text{lev}_c V_N^0$ in finite time for all disturbance sequences $\mathbf{w}$ satisfying $w(i) \in \mathbb{W}$, all $i \geq 0$, and thereafter remains in this set; the set $\text{lev}_c V_N^0$ is positive invariant for $x^+ = f(x, \mu_0^0(x), w)$, $w \in \mathbb{W}$. The policy sequence $\tilde{\mathbf{u}}(x)$, easily obtained from $\boldsymbol{\mu}^0(x)$, is feasible for $\mathbb{P}_N(x^+)$ and is a suitable warm start for computing $\boldsymbol{\mu}^0(x^+)$.

3.5 Tube-Based Robust MPC

3.5.1 Introduction

It was shown in Section 3.4 that it is possible to control an uncertain system robustly using a version of MPC that requires solving *online* an optimal control problem of minimizing a cost subject to satisfaction of state and control constraints for *all* possible disturbance sequences. For MPC with horizon N and q_x state constraints, the number of state constraints in the optimal control problem is Nq_x. Since the state constraints should be satisfied for *all* disturbance sequences, the number of state constraints for the uncertain case is MNq_x, with M equal to the number of disturbance sequences. For linear MPC, M can be as small as V^N with V equal to the number of vertices of $\mathbb{W}$ with $\mathbb{W}$ polytopic. For nonlinear MPC, Monte Carlo optimization must be employed, in which case M can easily be several thousand to achieve constraint satisfaction with high probability. The number of constraints MNq_x can thus exceed 10^6 in process control applications.

It is therefore desirable to find approaches for which the online computational requirement is more modest. We describe, in this section, a *tube-based* approach. We show that all trajectories of the uncertain system lie in a bounded neighborhood of a nominal trajectory. This bounded neighborhood is called a tube. Determination of the tube enables satisfaction of the constraints by the uncertain system for *all* disturbance sequences to be obtained by ensuring that the nominal trajectory satisfies suitably tightened constraints. If the nominal trajectory satisfies the tightened constraints, every random trajectory in the associated tube satisfies the original constraints. Computation of the tightened constraints may be computationally expensive but can be done *offline*; the *online* computational requirements are similar to those for nominal MPC.

To describe tube-based MPC, we use some concepts in set algebra. Given two subsets A and B of $\mathbb{R}^n$, we define set addition, set subtrac-

tion (sometimes called Minkowski or Pontryagin set subtraction), set multiplication, and Hausdorff distance between two sets as follows.

Definition 3.10 (Set algebra and Hausdorff distance)**.**

(a) Set addition: $A \oplus B := \{a + b \mid a \in A, b \in B\}$

(b) Set subtraction: $A \ominus B := \{x \in \mathbb{R}^n \mid \{x\} \oplus B \subseteq A\}$

(c) Set multiplication: Let $K \in \mathbb{R}^{m \times n}$; then $KA := \{Ka \mid a \in A\}$

(d) The Hausdorff distance $d_H(\cdot)$ between two subsets A and B of $\mathbb{R}^n$ is defined by

$$d_H(A, B) := \max\{\sup_{a \in A} d(a, B), \sup_{b \in B} d(b, A)\}$$

in which $d(x, S)$ denotes the distance of a point $x \in \mathbb{R}^n$ from a set $S \subset \mathbb{R}^n$ and is defined by

$$d(x, S) := \inf_{y} \{d(x, y) \mid y \in S\} \qquad d(x, y) := |x - y|$$

In these definitions, $\{x\}$ denotes the set consisting of a single point x, and $\{x\} \oplus B$ therefore denotes the set $\{x + b \mid b \in B\}$; the set $A \ominus B$ is the largest set C such that $B \oplus C \subseteq A$. A sequence $(x(i))$ is said to converge to a set S if $d(x(i), S) \to 0$ as $i \to \infty$. If $d_H(A, B) \leq \varepsilon$, then the distance of every point $a \in A$ from B is less than or equal to ε, and that the distance of every point $b \in B$ from A is less than or equal to ε. We say that the sequence of sets $(A(i))$ converges, in the Hausdorff metric, to the set B if $d_H(A(i), B) \to 0$ as $i \to \infty$.

Our first task is to generate an outer-bounding tube. An excellent background for the following discussion is provided in Kolmanovsky and Gilbert (1998).

3.5.2 Outer-Bounding Tube for a Linear System with Additive Disturbance

Consider the following linear system

$$x^+ = Ax + Bu + w$$

in which $w \in \mathbb{W}$, a compact convex subset of $\mathbb{R}^n$ containing the origin. We assume that $\mathbb{W}$ contains the origin in its interior. Let $\phi(i; x, \mathbf{u}, \mathbf{w})$ denote the solution of $x^+ = Ax + Bu + w$ at time i if the initial state at

time zero is x, and the control and disturbance sequences are, respectively, $\mathbf{u}$ and $\mathbf{w}$.

Let the nominal system be described by

$$\bar{x}^+ = A\bar{x} + B\bar{u}$$

and let $\bar{\phi}(i;\bar{x},\mathbf{u})$ denote the solution of $\bar{x}^+ = A\bar{x} + B\bar{u}$ at time i if the initial state at time zero is $\bar{x}$. Then $e := x - \bar{x}$, the deviation of the actual state x from the nominal state $\bar{x}$, satisfies the difference equation

$$e^+ = Ae + w$$

so that

$$e(i) = A^i e(0) + \sum_{j=0}^{i-1} A^{i-1-j} w(j)$$

in which $e(0) = x(0) - \bar{x}(0)$. If $e(0) = 0$, then $e(i) \in S(i)$ where the set $S(i)$ is defined by

$$S(i) := \sum_{j=0}^{i-1} A^j \mathbb{W} = \mathbb{W} \oplus A\mathbb{W} \oplus \cdots \oplus A^{i-1}\mathbb{W}$$

in which $\sum$ and $\oplus$ denote set addition. It follows from our assumptions on $\mathbb{W}$ that $S(i)$ contains the origin in its interior for all $i \geq n$.

We first consider the tube $\mathbf{X}(x,\mathbf{u})$ generated by the open-loop control sequence $\mathbf{u}$ when $x(0) = \bar{x}(0) = x$, and $e(0) = 0$. It is easily seen that $\mathbf{X}(x,\mathbf{u}) = (X(0;x), X(1;x,\mathbf{u}), \ldots, X(N;x,\mathbf{u}))$ with

$$X(i;x) := \{\bar{x}(i)\} \oplus S(i)$$

and $\bar{x}(i) = \bar{\phi}(i;x,\mathbf{u})$, the state at time i of the nominal system, is the center of the tube. So it is relatively easy to obtain the exact tube generated by an open-loop control if the system is linear and has a bounded additive disturbance, provided that one can compute the sets $S(i)$.

If A is stable, then, as shown in Kolmanovsky and Gilbert (1998), $S(\infty) := \sum_{j=0}^{\infty} A^j \mathbb{W}$ exists and is positive invariant for $x^+ = Ax + w$, i.e., $x \in S(\infty)$ implies that $Ax + w \in S(\infty)$ for all $w \in \mathbb{W}$; also $S(i) \to S(\infty)$ in the Hausdorff metric as $i \to \infty$. The set $S(\infty)$ is known to be the minimal robust positive invariant set[6] for $x^+ = Ax + w$, $w \in \mathbb{W}$. Also

[6] Every other robust positive invariant set X satisfies $X \supseteq S_\infty$.

$S(i) \subseteq S(i+1) \subseteq S(\infty)$ for all $i \in \mathbb{I}_{\geq 0}$ so that the tube $\hat{\mathbf{X}}(x, \mathbf{u})$ defined by

$$\hat{\mathbf{X}}(x, \mathbf{u}) := \left(\hat{X}(0; x), \hat{X}(1; x, \mathbf{u}), \ldots, \hat{X}(N; x, \mathbf{u})\right)$$

in which

$$\hat{X}(0; x) = \{x\} \oplus S(\infty) \qquad \hat{X}(i; x, \mathbf{u}) = \{\bar{x}(i)\} \oplus S(\infty)$$

is an outer-bounding tube with constant "cross section" $S(\infty)$ for the exact tube $\mathbf{X}(x, \mathbf{u})$ ($X(i; x, \mathbf{u}) \subseteq \hat{X}(i; x, \mathbf{u})$ for all $i \in \mathbb{I}_{\geq 0}$). It is sometimes more convenient to use the constant cross-section outer-bounding tube $\hat{\mathbf{X}}(x, \mathbf{u})$ in place of the exact tube $\mathbf{X}(x, \mathbf{u})$. If we restrict attention to the interval $[0, N]$ as we do in computing the MPC action, then replacing $S(\infty)$ by $S(N)$ yields a less conservative, constrained cross-section, outer-bounding tube for the interval $[0, N]$.

Use of the exact tube $\mathbf{X}(x, \mathbf{u})$ and the outer-bounding tube $\hat{\mathbf{X}}(x, \mathbf{u})$ may be limited for reasons discussed earlier—the sets $S(i)$ may be unnecessarily large simply because an open-loop control sequence rather than a feedback policy was employed to generate the tube. For example, if $\mathbb{W} = [-1, 1]$ and $x^+ = x + u + w$, then $S(i) = (i+1)\mathbb{W}$ increases without bound as time i increases. We must introduce feedback to contain the size of $S(i)$, but wish to do so in a simple way because optimizing over arbitrary policies is prohibitive. The feedback policy we propose is

$$u = \bar{u} + K(x - \bar{x})$$

in which x is the current state of the system $x^+ = Ax + Bu + w$, $\bar{x}$ is the current state of a nominal system defined below, and $\bar{u}$ is the current input to the nominal system. With this feedback policy, the state x satisfies the difference equation

$$x^+ = Ax + B\bar{u} + BKe + w$$

in which $e := x - \bar{x}$ is the deviation of the actual state from the nominal state. The nominal system corresponding to the uncertain system $x^+ = Ax + B\bar{u} + BKe + w$ is

$$\bar{x}^+ = A\bar{x} + B\bar{u}$$

The deviation $e = x - \bar{x}$ now satisfies the difference equation

$$e^+ = A_K e + w \qquad A_K := A + BK$$

which is the same equation used previously except that A, which is possibly unstable, is replaced by A_K, which is stable by design. If K is

chosen so that A_K is stable, then the corresponding uncertainty sets $S_K(i)$ defined by

$$S_K(i) := \sum_{j=0}^{i-1} A_K^j \mathbb{W}$$

can be expected to be smaller than the original uncertainty sets $S(i)$, $i \in \mathbb{I}_{\geq 0}$, considerably smaller if A is unstable and i is large. Our assumptions on $\mathbb{W}$ imply that $S_K(i)$, like $S(i)$, contains the origin in its interior for each i. Since A_K is stable, the set $S_K(\infty) := \sum_{j=0}^{\infty} A_K^j \mathbb{W}$ exists and is positive invariant for $e^+ = A_K e + w$. Also, $S_K(i) \to S_K(\infty)$ in the Hausdorff metric as $i \to \infty$. Since K is fixed, the feedback policy $u = \bar{u} + K(x - \bar{x})$ is simply parameterized by the open-loop control sequence $\bar{\mathbf{u}}$. If $x(0) = \bar{x}(0) = x$, the tube generated by the feedback policy $u = \bar{u} + K(x - \bar{x})$ is $\mathbf{X}(x, \bar{\mathbf{u}}) = (X(0; x), X(1; x, \bar{\mathbf{u}}), \ldots, X(N; x, \bar{\mathbf{u}}))$ in which

$$X(0; x) = \{x\} \qquad X(i; x, \bar{\mathbf{u}}) := \{\bar{x}(i)\} \oplus S_K(i)$$

and $\bar{x}(i)$ is the solution of the nominal system $\bar{x}^+ = A\bar{x} + B\bar{u}$ at time i if the initial state $\bar{x}(0) = x$, and the control sequence is $\bar{\mathbf{u}}$. For given initial state x and control sequence $\bar{\mathbf{u}}$, the solution of $x^+ = Ax + B(\bar{u} + Ke) + w$ lies in the tube $\mathbf{X}(x, \bar{\mathbf{u}})$ for *every* admissible disturbance sequence $\mathbf{w}$. As before, $S_K(i)$ may be replaced by $S_K(\infty)$ to get an outer-bounding tube. If attention is confined to the interval $[0, N]$, $S_K(i)$ may be replaced by $S_K(N)$ to obtain a less conservative outer-bounding tube. If we consider again our previous example, $\mathbb{W} = [-1, 1]$ and $x^+ = x + u + w$, and choose $K = -(1/2)$, then $A_K = 1/2$, $S_K(i) = (1 + 0.5 + \ldots + 0.5^{i-1})\mathbb{W} \subset 2\mathbb{W}$, and $S_K(\infty) = 2\mathbb{W} = [-2, 2]$. In contrast, $S(i) \to [-\infty, \infty]$ as $i \to \infty$.

In the preceding discussion, we required $x(0) = \bar{x}(0)$ so that $e(0) = 0$ in order to ensure $e(i) \in S(i)$ or $e(i) \in S_K(i)$. When A_K is stable, however, it is possible to relax this restriction. This follows from the previous statement that $S_K(\infty)$ exists and is robustly positive invariant for $e^+ = A_K e + w$, i.e., $e \in S_K(\infty)$ implies $e^+ \in S_K(\infty)$ for all $e^+ \in \{A_K e\} \oplus \mathbb{W}$. Hence, if $e(0) \in S_K(\infty)$, then $e(i) \in S_K(\infty)$ for all $i \in \mathbb{I}_{\geq 0}$, all $\mathbf{w} \in \mathbb{W}^i$.

In tube-based MPC, we ensure that $\bar{x}(i) \to 0$ as $i \to \infty$, so that $x(i)$, which lies in the sequence of sets $(\{\bar{x}(i)\} \oplus S_K(i))_{0:\infty}$, converges to the set $S_K(\infty)$ as $i \to \infty$. Figure 3.3 illustrates this result ($S := S_K(\infty)$). Even though $S_K(\infty)$ is difficult to compute, this is a useful theoretical property of the controlled system.

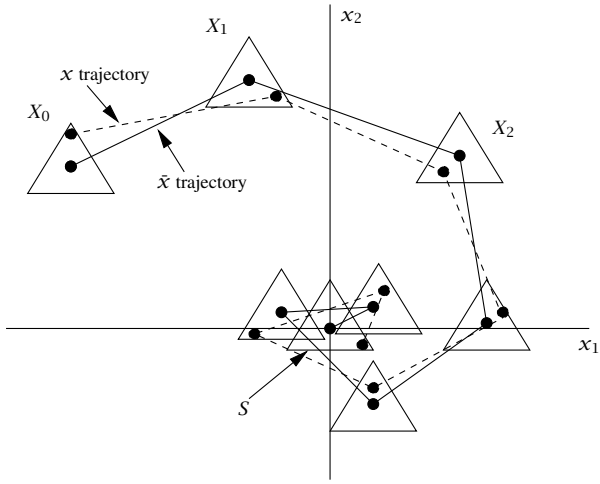


Figure 3.3: Outer-bounding tube $\mathbf{X}(\bar{x}, \bar{\mathbf{u}})$; $X_i = \{\bar{x}(i)\} \oplus S_K(\infty)$.

The controller is required to ensure that state-control constraint $(x, u) \in \mathbb{Z}$ is not transgressed. Let $\bar{\mathbb{Z}}$ be defined by

$$\bar{\mathbb{Z}} := \mathbb{Z} \ominus (S_K(\infty) \times KS_K(\infty))$$

since it follows from the definition of the set operation $\ominus$ that $\bar{\mathbb{Z}} \oplus (S_K(\infty) \times KS_K(\infty)) \subseteq \mathbb{Z}$. In the simple case when $\mathbb{Z} = \mathbb{X} \times \mathbb{U}$

$$\bar{\mathbb{Z}} = \bar{\mathbb{X}} \times \bar{\mathbb{U}} \qquad \bar{\mathbb{X}} = \mathbb{X} \ominus S_K(\infty) \qquad \bar{\mathbb{U}} = \mathbb{U} \ominus KS_K(\infty)$$

Computation of the set $S_K(\infty)$—which is known to be difficult—is not required, as we show later. It follows from the preceding discussion that if the nominal state and control trajectories $\bar{\mathbf{x}}$ and $\bar{\mathbf{u}}$ satisfy the *tightened* constraint $(\bar{x}(i), \bar{u}(i)) \in \hat{\mathbb{Z}} \subset \bar{\mathbb{Z}}$ for all $i \in \mathbb{I}_{0:N-1}$, the state and control trajectories $\mathbf{x}$ and $\mathbf{u}$ of the uncertain system then satisfy the original constraints $(x(i), u(i)) \in \mathbb{Z}$ for all $i \in \mathbb{I}_{0:N-1}$. This is the basis for tube-based robust MPC discussed next.

3.5.3 Tube-Based MPC of Linear Systems with Additive Disturbances

The tube-based controller has two components: (i) a nominal state-control trajectory $(\bar{x}(i), \bar{u}(i))_{i \in \mathbb{I}_{\geq 0}}$ that commences at the initial state x and that satisfies the tightened constraint, and (ii) a feedback controller $u = \bar{u} + K(x - \bar{x})$ that attempts to steer the uncertain state-control trajectory to the nominal trajectory. The nominal state-control

trajectory may be generated at the initial time or generated sequentially using standard MPC for deterministic systems. The latter gives more flexibility to cope with changing conditions, such as changing setpoint. Assume, then, that a controller $\bar{u} = \bar{\kappa}_N(\bar{x})$ for the nominal system $\bar{x}^+ = A\bar{x} + B\bar{u}$ has been determined using results in Chapter 2 by solving the standard optimal control problem of the form

$$\begin{aligned}
\bar{P}_N(\bar{x}): \quad & \bar{V}_N^0(\bar{x}) = \min_{\bar{\mathbf{u}}} \{\bar{V}_N(\bar{x}, \bar{\mathbf{u}}) \mid \bar{\mathbf{u}} \in \bar{\mathcal{U}}_N(\bar{x})\} \\
& \bar{V}_N(\bar{x}, \bar{\mathbf{u}}) = \sum_{i=0}^{N-1} \ell(\bar{x}(i), \bar{u}(i)) + V_f(\bar{x}(N)) \\
& \bar{\mathcal{U}}_N(\bar{x}) = \{\bar{\mathbf{u}} \mid (\bar{x}(i), \bar{u}(i)) \in \bar{\mathbb{Z}}, i \in \mathbb{I}_{0:N-1}, \bar{x}(N) \in \mathbb{X}_f\}
\end{aligned}$$

in which $\bar{x}(i) = \bar{\phi}(i; x, \bar{\mathbf{u}})$. Under usual conditions, the origin is asymptotically stable for the controlled nominal system described by

$$\bar{x}^+ = A\bar{x} + B\bar{\kappa}_N(\bar{x})$$

and the controlled system satisfies the constraint $(\bar{x}(i), \bar{u}(i)) \in \bar{\mathbb{Z}}$ for all $i \in \mathbb{I}_{\geq 0}$. Let $\bar{\mathcal{X}}_N$ denote the set $\{\bar{x} \mid \bar{\mathcal{U}}_N(\bar{x}) \neq \emptyset\}$. Of course, determination of the control $\bar{\kappa}_N(\bar{x})$ requires solving online the constrained optimal control problem $\mathbb{P}_N(\bar{x})$.

The feedback controller, given the state x of the system being controlled, and the state $\bar{x}$ of the nominal system, generates the control $u = \bar{\kappa}_N(\bar{x}) + K(x - \bar{x})$. The composite system with state $(x, \bar{x})$ satisfies

$$\begin{aligned}
x^+ &= Ax + B\bar{\kappa}_N(\bar{x}) + BK(x - \bar{x}) + w \\
\bar{x}^+ &= A\bar{x} + B\bar{\kappa}_N(\bar{x})
\end{aligned}$$

The system with state $(e, \bar{x})$, $e := x - \bar{x}$, satisfies a simpler difference equation

$$\begin{aligned}
e^+ &= A_K e + w \\
\bar{x}^+ &= A\bar{x} + B\bar{\kappa}_N(\bar{x})
\end{aligned}$$

The two states $(x, \bar{x})$ and $(e, \bar{x})$ are related by

$$\begin{bmatrix} e \\ \bar{x} \end{bmatrix} = T \begin{bmatrix} x \\ \bar{x} \end{bmatrix} \qquad T := \begin{bmatrix} I & -I \\ 0 & I \end{bmatrix}$$

Since T is invertible, the two systems with states $(x, \bar{x})$ and $(e, \bar{x})$ are equivalent. Hence, to establish robust stability it suffices to consider the simpler system with state $(e, \bar{x})$. First, we define robustly asymptotically stable (RAS).

Definition 3.11 (Robust asymptotic stability of a set)**.** Suppose the sets S_1 and S_2, $S_2 \subset S_1$, are robustly positive invariant for the system $z^+ = f(z,w)$, $w \in \mathbb{W}$. The set S_2 is RAS for $z^+ = f(z,w)$ in S_1 if there exists a $\mathcal{KL}$ function $\beta(\cdot)$ such that every solution $\phi(\,\cdot\,;z,\mathbf{w})$ of $z^+ = f(z,w)$ with initial state $z \in S_1$ and any disturbance sequence $\mathbf{w} \in \mathbb{W}^\infty$ satisfies

$$\left|\phi(i;z,\mathbf{w})\right|_{S_2} \le \beta(|z|_{S_2}, i) \;\; \forall i \in \mathbb{I}_{\ge 0}$$

In this definition, $|z|_S := d(z,S)$, the distance of z from set S.

We now assume that $\bar{\kappa}_N(\cdot)$ and $\bar{\mathbb{Z}}$ have been determined to ensure the origin is asymptotically stable in a positive invariant set $\bar{X}$ for the controlled nominal system $\bar{x}^+ = A\bar{x} + B\bar{\kappa}_N(\bar{x})$. Under this assumption we have

Proposition 3.12 (Robust asymptotic stability of tube-based MPC for linear systems)**.** *The set $S_K(\infty) \times \{0\}$ is RAS for the composite system $(e^+ = A_K e + w, \bar{x}^+ = A\bar{x} + B\bar{\kappa}_N(\bar{x}))$ in the positive invariant set $S_K(\infty) \times \bar{X}_N$.*

Proof. Because the origin is asymptotically stable for $\bar{x}^+ = A\bar{x} + B\bar{\kappa}_N(\bar{x})$, there exists a $\mathcal{KL}$ function $\beta(\cdot)$ such that every solution $\bar{\phi}(\,\cdot\,;\bar{x})$ of the controlled nominal system with initial state $\bar{x} \in \bar{X}_N$ satisfies

$$\left|\bar{\phi}(i;\bar{x})\right| \le \beta(|\bar{x}|, i) \;\; \forall i \in \mathbb{I}_{\ge 0}$$

Since $e(0) \in S_K(\infty)$ implies $e(i) \in S_K(\infty)$ for all $i \in \mathbb{I}_{\ge 0}$, it follows that

$$\left|(e(i), \bar{\phi}(i;\bar{x}))\right|_{S_K(\infty)\times\{0\}} \le |e(i)|_{S_K(\infty)} + \left|\bar{\phi}(i;\bar{x})\right| \le \beta(|\bar{x}|, i)$$

Hence the set $S_K(\infty) \times \{0\}$ is RAS in $S_K(\infty) \times \bar{X}_N$ for the composite system $(e^+ = A_K e + w, \bar{x}^+ = A\bar{x} + B\bar{\kappa}_N(\bar{x}))$. ∎

It might be of interest to note that (see Exercise 3.4)

$$d_H(\{\bar{\phi}(i;\bar{x})\} \oplus S_K(\infty), S_K(\infty)) \le \left|\bar{\phi}(i;\bar{x})\right| \le \beta(|\bar{x}|, i)$$

for every solution $\bar{\phi}(\cdot)$ of the nominal system with initial state $\bar{x} \in \mathcal{X}_N$.

Finally we show how suitable tightened constraints may be determined. It was shown above that the nominal system should satisfy the tightened constraint $(\bar{x}, \bar{u}) \in \bar{\mathbb{Z}} = \mathbb{Z} \ominus (S_K(\infty), KS_K(\infty))$. Since $S_K(\infty)$ is difficult to compute and use, impossible for many process control applications, we present an alternative. Suppose $\mathbb{Z}$ is polytopic and is described by a set of scalar inequalities of the form $c'z \le d$

($c_x'x + c_u'u \le d$). We show next how each constraint of this form may be tightened so that satisfaction of the tightened constraint by the nominal system ensures satisfaction of original constraint by the uncertain system. For all $j \in \mathbb{I}_{\ge 0}$, let

$$\theta_j := \max_e \{c'(e, Ke) \mid e \in S_K(j)\} = \max_{\mathbf{w}} \{\sum_{i=0}^{j-1} c'(I, K)A_K^i w_i \mid \mathbf{w} \in \mathbb{W}_{0:j-1}\}$$

in which $c'(e, Ke) = c_x'e + c_u'Ke$ and $c'(I, K)A_K^i w_i = c_x'A_K^i w_i + c_u'KA_K^i w_i$. Satisfaction of the constraint $c'\bar{z} \le d - \theta_\infty$ by the nominal system ensures satisfaction of $c'z \le d$, $z = \bar{z} + (e, Ke)$, by the uncertain system; however, computation of θ_∞ is impractical so we adopt the approach in (Raković, Kerrigan, Kouramas, and Mayne, 2005a). Because A_K is Hurwitz, for all $\alpha \in (0, 1)$ there exists a finite integer N such that $A_K^N \mathbb{W} \subset \alpha \mathbb{W}$ and $KA_K^N \mathbb{W} \subset \alpha K \mathbb{W}$. It follows that

$$\theta_\infty \le \theta_N + \alpha\theta_\infty$$

so that

$$\theta_\infty \le (1 - \alpha)^{-1}\theta_N$$

Hence, satisfaction of the tightened constraint $c'\bar{z} \le d - (1 - \alpha)^{-1}\theta_N$ by the nominal system ensures that the uncertain system satisfies the original constraint $c'z \le d$. The tightened constraint set $\bar{\mathbb{Z}}$ is defined by these modified constraints.

Example 3.13: Calculation of tightened constraints

Consider the system

$$x^+ = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} + w$$

with $\mathbb{W} := \{w \mid |w|_\infty \le 0.1\}$, $\mathbb{Z} := \{(x, u) \mid |x|_\infty \le 1, |u| \le 1\}$, and nominal control law $K := \begin{bmatrix} -0.4 & -1.2 \end{bmatrix}$. For increasing values of N, we calculate α such that $A_K^N \mathbb{W} \subset \alpha \mathbb{W}$ and $KA_K^N \mathbb{W} \subset \alpha K \mathbb{W}$.

Because $\mathbb{W}$ is a box, it is sufficient to check only its vertices, i.e., the four elements $w \in W := \{-0.1, 0.1\}^2$. Thus, we have

$$\alpha = \max\left(\frac{\max_{w \in W} \left| A_K^N w \right|_\infty}{\max_{w \in W} |w|_\infty}, \frac{\max_{w \in W} \left| KA_K^N w \right|_\infty}{\max_{w \in W} |Kw|_\infty} \right)$$

These values are shown in Figure 3.4. From here, we see that $N \ge 3$ is necessary for the approximation to hold. With the values of α, the

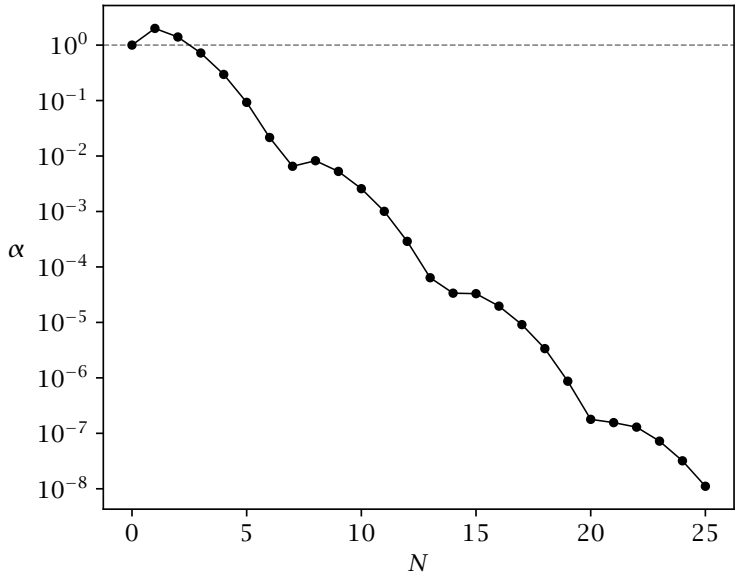


Figure 3.4: Minimum feasible α for varying N. Note that we require $\alpha \in [0, 1)$.

tightened constraint sets $\bar{\mathbb{Z}}$ can then be computed as above. Once again, because of the structure of $\mathbb{W}$, we need only check the vertices. Due to the symmetry of the system, each set is of the form

$$\bar{\mathbb{Z}} = \{(x, u) \mid |x_1| \leq \chi_1, |x_2| \leq \chi_2, |u| \leq \mu\}$$

The bounds χ_1, χ_2, and μ are shown in Figure 3.5. Note that while $N = 3$ gives a feasible value of α, we require at least $N = 4$ for $\bar{\mathbb{Z}}$ to be nonempty. □

Time-varying constraint set $\bar{\mathbb{Z}}(i)$. The tube-based model predictive controller is conservative in that the feasible set for $\bar{\mathbb{P}}_N(\bar{x})$ is unnecessarily small due to use of a constant constraint set $\bar{\mathbb{Z}} = \mathbb{Z} \ominus (S_K(\infty) \times KS_K(\infty))$. This reduces the region of attraction $\bar{\mathcal{X}}_N$, the set of states for which $\bar{\mathbb{P}}_N(\bar{x})$ is feasible. Tube-based model predictive control can be made less conservative by using time-varying constraint set $\bar{\mathbb{Z}}(i) = \mathbb{Z} \ominus (S_K(i) \times KS_k(i))$, $i \in \mathbb{I}_{0:N-1}$ for the initial optimal control problem that generates the control sequence $\mathbf{u}^0(\bar{x})$. The control applied

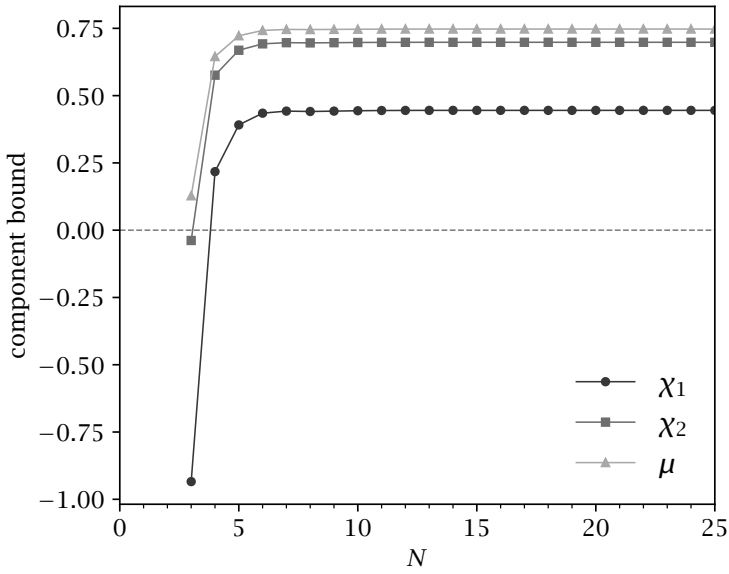


Figure 3.5: Bounds on tightened constraint set $\bar{\mathbb{Z}}$ for varying N. Bounds are $|x_1| \leq \chi_1$, $|x_2| \leq \chi_2$, and $|u| \leq \mu$.

to the uncertain system is $\bar{u}(i) + Ke(k)$; the infinite sequence $\bar{\mathbf{u}}$ is constructed as follows. The sequence $(\bar{u}(0), \bar{u}(1), \ldots, \bar{u}(N-1))$ is set equal to $\bar{\mathbf{u}}^0(\bar{x})$, the solution of the nominal optimal control problem at the initial state $\bar{x}$, with time-varying constraint sets $\bar{\mathbb{Z}}(i)$ and terminal constraint set $\bar{\mathbb{X}}(f)$. The associated state sequence is $(\bar{x}(0), \bar{x}(1), \ldots, \bar{x}(N))$ with $\bar{x}(N) \in \bar{\mathbb{X}}_f$. For $i \in \mathbb{I}_{\geq N}$, $\bar{u}(i)$ and $\bar{x}(i)$ are obtained as the solution at time i of

$$\bar{x}^+ = A\bar{x} + B\kappa_f(\bar{x}), \quad u = \kappa_f(\bar{x})$$

with initial state $\bar{x}(N)$ at time N. We now assume that $\bar{\mathbb{X}}_f$ satisfies $\bar{\mathbb{X}}_f \oplus S_K(\infty) \subset \mathbb{X}$. Since $\bar{x}(N) \in \bar{\mathbb{X}}_f$ it follows that $\bar{x}(i) \in \bar{\mathbb{X}}_f$ and $x(i) \in \mathbb{X}$ for all $i \in \mathbb{I}_{\geq N}$. Also, for all $i \in \mathbb{I}_{0:N-1}$, $\bar{x}(i) \in \bar{\mathbb{X}}(i) = \mathbb{X} \ominus S_K(i)$ and $e(i) \in S_k(i)$ so that $x(i) = \bar{x}(i) + e(i) \in \mathbb{X}$. Hence $x(i) \in \mathbb{X}$ for all $i \in \mathbb{I}_{\geq 0}$. Since $\bar{x}(i) \to 0$, the state $x(i)$ of the uncertain system tends to $S_K(\infty)$ as $i \to \infty$. Since $\bar{\mathbb{Z}}(i) \supset \bar{\mathbb{Z}}$, the region of attraction is larger than that for tube-based MPC using a constant constraint set.

3.5.4 Improved Tube-Based MPC of Linear Systems with Additive Disturbances

In this section we describe a version of the tube-based model predictive controller that has pleasing theoretical properties. We omitted, in the previous section, to make use of an additional degree of freedom available to the controller, namely the ability to change the state $\bar{x}$ of the nominal system. In Chisci, Rossiter, and Zappa (2001), $\bar{x}$ is set equal to x, the current state of the uncertain system, but there is no guarantee that an initial state x is superior to $\bar{x}$ in the sense of enhancing convergence to the origin of the nominal trajectory. To achieve more rapid convergence, we propose that an improved tube center $\bar{x}^*$ is chosen by minimizing the value function $\bar{V}_N^0(\cdot)$ of the nominal optimal control problem. It is necessary that the current state x remains in the tube with new center $\bar{x}^*$. To achieve this, at state $(x, \bar{x})$, a new optimal control problem $\bar{\mathbb{P}}_N^*(x)$, is solved online, to determine an improved center $\bar{x}^*$ and, simultaneously the subsequent center $\bar{x}^+$. We assume, for simplicity, that $\mathbb{Z} = \mathbb{X} \times \mathbb{U}$ and $\bar{\mathbb{Z}} = \bar{\mathbb{X}} \times \bar{\mathbb{U}}$. The new optimal control problem $\mathbb{P}_N^*(x)$ that replaces $\bar{\mathbb{P}}_N(\bar{x})$ is defined by

$$\begin{aligned}\mathbb{P}_N^*(x): \quad \bar{V}_N^*(x) &= \min_z \{\bar{V}_N^0(z) \mid x \in \{z\} \oplus S_K(\infty),\ z \in \bar{\mathbb{X}}\} \\ &= \min_{z,\bar{\mathbf{u}}} \{\bar{V}_N(z, \bar{\mathbf{u}}) \mid \bar{\mathbf{u}} \in \bar{\mathcal{U}}_N(z),\ x \in \{z\} \oplus S_K(\infty),\ z \in \bar{\mathbb{X}}\}\end{aligned}$$

The solution to problem $\mathbb{P}_N^*(x)$ is $(\bar{x}^*(x), \bar{\mathbf{u}}^*(x))$. The constraint $x \in \{z\} \oplus S_K(\infty)$ ensures that the current state x lies in $\{\bar{x}^*(x)\} \oplus S_K(\infty)$, the first element of the "new tube." The argument of $\mathbb{P}_N^*(x)$ is x because of the constraint $x \in \{z\} \oplus S_K(\infty)$; the solution to the problem generates both the improved current nominal state $\bar{x}^*(x)$ as well as its successor $\bar{x}^+$. If $(x, \bar{x})$ satisfies $\bar{x} \in \bar{\mathcal{X}}_N$ and $x \in \{\bar{x}\} \oplus S_K(\infty)$, then $(\bar{x}, \tilde{\mathbf{u}}(\bar{x}))$ is a warm start for $\mathbb{P}_N^*(x)$; here $\tilde{\mathbf{u}}(\bar{x})$ is a warm start for $\bar{\mathbb{P}}_N(\bar{x})$. The successor nominal state is

$$\bar{x}^+ = (\bar{x}^*(x))^+ = A\bar{x}^*(x) + B\bar{\kappa}_N(\bar{x}^*(x))$$

in which, as usual, $\bar{\kappa}_N(\bar{x}^*(x))$ is the first element in the control sequence $\bar{\mathbf{u}}^*(x)$. It follows that

$$\bar{V}_N^*(x) = \bar{V}_N^0(\bar{x}^*(x)) \le \bar{V}_N^0(\bar{x}), \quad \bar{\mathbf{u}}^*(x) = \bar{\mathbf{u}}^0(\bar{x}^*(x))$$

The control applied to the uncertain system at state x is

$$\bar{\kappa}_N^*(x) := \bar{\kappa}_N(\bar{x}^*(x)) + K(x - \bar{x}^*(x))$$

so the closed-loop uncertain system satisfies

$$x^+ = Ax + B\bar{\kappa}_N(\bar{x}^*(x)) + K(x - \bar{x}^*(x)) + w$$

and $e = x - \bar{x}^*(x)$ satisfies

$$e^+ = x^+ - (\bar{x}^*(x))^+ = Ae + BKe + w = A_K e + w$$

as before so that if $e \in S_K(\infty)$, then $e^+ \in S_K(\infty)$; hence $x \in \{\bar{x}^*(x)\} \oplus S_K(\infty)$ implies $x^+ \in \{(\bar{x}^*(x))^+\} \oplus S_K(\infty)$.

Suppose then that $\bar{x} \in \bar{\mathcal{X}}_N \subseteq \bar{\mathbb{X}}$ and $x \in \{\bar{x}\} \oplus S_K(\infty)$ so that $x \in \mathbb{X}$. If the usual assumptions for the nominal optimal control problem $\bar{\mathbb{P}}_N$ are satisfied and $\ell(\cdot)$ is quadratic and positive definite it follows that

$$\begin{aligned}
\bar{V}_N^*(x) &= \bar{V}_N^0(\bar{x}^*(x)) \geq c_1 \left|\bar{x}^*(x)\right|^2 \\
\bar{V}_N^*(x) &= \bar{V}_N^0(\bar{x}^*(x)) \leq c_2 \left|\bar{x}^*(x)\right|^2 \\
\bar{V}_N^*(x^+) &= \bar{V}_N^0(\bar{x}^*(x^+)) \leq \bar{V}_N^0((\bar{x}^*(x))^+) \leq \bar{V}_N^0(\bar{x}^*(x)) - c_1 \left|\bar{x}^*(x)\right|^2
\end{aligned}$$

The last inequality follows from the fact that $\bar{x}^+ = (\bar{x}^*(x))^+ = A\bar{x}^*(x) + B\bar{\kappa}_N^+(\bar{x}^*(x))$ and the descent property of the solution to $\bar{\mathbb{P}}_N^0(\bar{x}^*(x))$.

Proposition 3.14 (Recursive feasibility of tube-based MPC)**.** *Suppose that at time zero, $(x, \bar{x}) \in (\{\bar{x}\} \oplus S_K(\infty)) \times \bar{\mathcal{X}}_N$. Then, Problem $\bar{\mathbb{P}}_N^*$ is recursively feasible: $(x, \bar{x}) \in (\{\bar{x}\} \oplus S_K(\infty)) \times \bar{\mathcal{X}}_N$ implies $(x, \bar{x})^+ = (x^+, \bar{x}^+) \in (\{\bar{x}^+\} \oplus S_K(\infty)) \times \bar{\mathcal{X}}_N$.*

Proof. Suppose that $(x, \bar{x})$ satisfies $x \in \{\bar{x}\} \oplus S_K(\infty)$ and $\bar{x} \in \bar{\mathcal{X}}_N$. From the definition of $\bar{\mathbb{P}}_N^*$, any solution satisfies the tightened constraints so that $\bar{x}^*(x) \in \bar{\mathcal{X}}_N$. The terminal conditions ensure, by the usual argument, that the successor state $\bar{x}^*(x)^+$ also lies in $\bar{\mathcal{X}}_N$. The condition $x \in \{z\} \oplus S_K(\infty)$ in $\mathbb{P}_N^*(x)$ then implies that $x \in \{\bar{x}^*(x)\} \oplus S_K(\infty)$ so that $x^+ \in \{\bar{x}^+\} \oplus S_K(\infty)$ ($e^+ \in S_K(\infty)$). ■

Proposition 3.15 (Robust exponential stability of improved tube-based MPC)**.** *The set $S_K(\infty)$ is robustly exponentially stable in $\bar{\mathcal{X}}_N \oplus S_K(\infty)$ for the system $x^+ = Ax + B\big(\bar{\kappa}_N(\bar{x}^*(x)) + K(x - \bar{x}^*(x))\big) + w$.*

Proof. It follows from the upper and lower bounds on $\bar{V}_N^0(x^*(x))$, and the descent property listed above that

$$\bar{V}_N^0(x^*(x^+)) \leq \gamma \bar{V}_N^0(x^*(x))$$

with $\gamma = (1 - c_1/c_2) \in (0, 1)$. Hence, if $x(i)$ denotes the solution at time i of $x^+ = Ax + B(\bar{\kappa}_N(\bar{x}^*(x)) + K(x - x^*(x))) + w$, $\bar{V}_N^0(\bar{x}^*(x(i)))$

decays exponentially fast to zero. It then follows from the upper bound on $\bar{V}_N^0(\bar{x}^*(x))$ that $x^*(x(i))$ also decays exponentially to zero. Because $x(i) \in \{\bar{x}^*(x(i))\}$ for all $i \in \mathbb{I}_{\geq 0}$, it follows, similarly to the proof of Proposition 3.12, that the set $S_K(\infty)$ is robustly exponentially stable in $\bar{\mathcal{X}}_N \oplus S_K(\infty)$ for the system $x^+ = Ax + B\bar{\kappa}_N(\bar{x}^*(x)) + K(x - \bar{x}^*(x)) + w$. ■

3.6 Tube-Based MPC of Nonlinear Systems

Satisfactory control in the presence of uncertainty requires feedback. As shown in Section 3.5, MPC of uncertain systems ideally requires optimization over control policies rather than control sequences, resulting in an optimal control problem that is often impossibly complex. Practicality demands simplification. Hence, in tube-based MPC of constrained *linear* systems we replace the general control policy $\boldsymbol{\mu} = (\mu_0(\cdot), \mu_1(\cdot), \ldots, \mu_{N-1}(\cdot))$, in which each element $\mu_i(\cdot)$ is an arbitrary function, by the simpler policy $\boldsymbol{\mu}$ in which each element has the simple form $\mu_i(x) = \bar{u}(i) + K(x - \bar{x}(i))$; $\bar{u}(i)$ and $\bar{x}(i)$, the control and state of the nominal system at time i, are determined using conventional MPC.

The feedback gain K, which defines the local control law, is determined offline; it can be chosen so that all possible trajectories of the uncertain system lie in a tube centered on the nominal trajectory $(\bar{x}(0), \bar{x}(1), \ldots)$. The "cross section" of the tube is a constant set $S_K(\infty)$ so that every possible state of the uncertain system at time i lies in the set $\{\bar{x}(i)\} \oplus S_K(\infty)$. This enables the nominal trajectory to be determined using MPC, to ensure that all possible trajectories of the uncertain system satisfy the state and control constraints, and that all trajectories converge to an invariant set centered on the origin.

It would be desirable to extend this methodology to the control of constrained nonlinear systems, but we face some formidable challenges. It is possible to define a nominal system and, as shown in Chapter 2, to determine, using MPC with "tightened" constraints, a nominal trajectory that can serve as the center of a tube. But it seems to be prohibitively difficult to determine a local control law that steers all trajectories of the uncertain system toward the nominal trajectory, and of a set centered on the nominal trajectory in which these trajectories can be guaranteed to lie.

We can overcome these difficulties by first generating a nominal trajectory—either by MPC as in the linear case or by a single solution

of an optimal control problem—and then using MPC to steer the state of the uncertain system toward the nominal trajectory $\bar{x}(\cdot)$. The latter MPC controller replaces the linear controller $u = \bar{u} + K(x - \bar{x})$ employed in the linear case, and thereby avoids the difficulty of determining a local nonlinear version of this linear controller. The value function $(x, i) \mapsto V_N^0(x, i)$ of the optimal control problem that is used to determine the MPC controller is time varying and has the property that $V_N^0(\bar{x}(i), i) = 0$ for all i. The tube is now a sequence of sublevel sets $\left(\text{lev}_c\, V_N^0(\cdot, i)\right)_{i \in \mathbb{I}_{\geq 0}}$ and therefore, unlike the linear case, has a varying cross section. We show that if the initial state $x(0)$ lies in $\text{lev}_c\, V_N^0(\cdot, 0)$, then subsequent states $x(i)$ of the controlled system lie in $\text{lev}_c\, V_N^0(\cdot, i)$ for all $i \in \mathbb{I}_{\geq 0}$.

The system to be controlled is described by a nonlinear difference equation

$$x^+ = f(x, u, w) \tag{3.20}$$

in which the disturbance w is assumed to lie in the compact set $\mathbb{W}$ that contains the origin. The state x and the control u are required to satisfy the constraints

$$x \in \mathbb{X} \qquad u \in \mathbb{U}$$

Both $\mathbb{X}$ and $\mathbb{U}$ are assumed to be compact and to contain the origin in their interiors. The solution of (3.20) at time i, if the initial state at time zero is x_0 and the control is generated by policy $\boldsymbol{\mu}$, is $\phi(i; x_0, \boldsymbol{\mu}, \mathbf{w})$, in which $\mathbf{w}$ denotes, as usual, the disturbance sequence $(w(0), w(1), \ldots)$. Similarly, $\phi(i; x_0, \kappa, \mathbf{w})$ denotes the solution of (3.20) at time i, if the initial state at time zero is x_0 and the control is generated by a time-invariant control law $\kappa(\cdot)$.

The nominal system is obtained by neglecting the disturbance w and is therefore described by

$$\bar{x}^+ = \bar{f}(\bar{x}, \bar{u}) := f(\bar{x}, \bar{u}, 0)$$

Its solution at time i, if its initial state is $\bar{x}_0$, is denoted by $\bar{\phi}(i; \bar{x}_0, \bar{\mathbf{u}})$, in which $\bar{\mathbf{u}} := (\bar{u}(0), \bar{u}(1), \ldots)$ is the nominal control sequence. The deviation between the actual and nominal state is $e := x - \bar{x}$ and satisfies

$$e^+ = f(x, u, w) - f(\bar{x}, \bar{u}, 0) = f(x, u, w) - \bar{f}(\bar{x}, \bar{u})$$

Because $f(\cdot)$ is nonlinear, this difference equation cannot be simplified as in the linear case when e^+ is independent of x and $\bar{x}$, and depends only on their difference e and w.

3.6.1 The Nominal Trajectory

The nominal trajectory is a feasible trajectory for the nominal system that is sufficiently far from the boundaries of the original constraints to enable the model predictive controller for the uncertain system to satisfy these constraints. It is generated by the solution to a nominal optimal control problem $\bar{\mathbb{P}}_N(\bar{x})$ in which $\bar{x}$ is the state of the nominal system. The cost function $\bar{V}_N(\cdot)$ for the nominal optimal control problem is defined by

$$\bar{V}_N(\bar{x}, \bar{\mathbf{u}}) := \sum_{i=0}^{N-1} \ell(\bar{x}(i), \bar{u}(i)) \tag{3.21}$$

in which $\bar{x}(i) = \bar{\phi}(i; \bar{x}, \bar{\mathbf{u}})$ and $\bar{x}$ is the initial state. The function $\ell(\cdot)$ is defined by

$$\ell(\bar{x}, \bar{u}) := (1/2)(|\bar{x}|_Q^2 + |\bar{u}|_R^2)$$

in which Q and R are positive definite, $|\bar{x}|_Q^2 := \bar{x}^T Q \bar{x}$, and $|\bar{u}|_R^2 := \bar{u}^T R \bar{u}$. We impose the following state and control constraints on the nominal system

$$\bar{x} \in \bar{\mathbb{X}} \qquad \bar{u} \in \bar{\mathbb{U}}$$

in which $\bar{\mathbb{X}} \subset \mathbb{X}$ and $\bar{\mathbb{U}} \subset \mathbb{U}$. The choice of $\bar{\mathbb{X}}$ and $\bar{\mathbb{U}}$ is more difficult than in the linear case because it is difficult to bound the deviation $e = x - \bar{x}$ of the state x of the uncertain system from the state $\bar{x}$ of the nominal system; this is discussed below. The optimal nominal trajectories $\bar{\mathbf{u}}^0$ and $\bar{\mathbf{x}}^0$ are determined by minimizing $\bar{V}_N(\bar{x}_0, \bar{\mathbf{u}})$ subject to: $\bar{x}_0 = x_0$, the state and control constraints specified above, and the terminal constraint $\bar{x}(N) = 0$ (we omit the initial state $\bar{x}_0 = x_0$ in $\bar{\mathbf{u}}^0$ and $\bar{\mathbf{x}}^0$ to simplify notation). The state and control of the nominal system satisfy $\bar{x}(i) = 0$ and $\bar{u}(i) = 0$ for all $i \geq N$. This simplifies both analysis and implementation in that the control reverts to conventional MPC for all $i \geq N$.

3.6.2 Model Predictive Controller

The purpose of the model predictive controller is to maintain the state of the uncertain system $x^+ = f(x, u, w)$ close to the trajectory of the nominal system. This controller replaces the controller $u = v + K(x - \bar{x})$ employed in the linear case. Given the current state/time (x, t) of the uncertain system, we determine a control sequence that minimizes with respect to the control sequence $\mathbf{u}$, the cost over a horizon N of

the deviation between the state and control of the *nominal* system, with initial state x and control sequence $\mathbf{u}$, and the state and control of the *nominal* system, with initial state $\bar{x}^0(t)$ and control sequence $\bar{\mathbf{u}}_t^0 := (\bar{u}^0(t), \bar{u}^0(t+1), \ldots, \bar{u}^0(t+N-1))$. The cost $V_N(x,t,\mathbf{u})$ that measures the distance between these two trajectories is defined by

$$V_N(x,t,\mathbf{u}) := \sum_{i=0}^{N-1} \ell\big((x(i) - \bar{x}^0(t+i)), (u(i) - \bar{u}^0(t+i))\big) + V_f(x(N)) \tag{3.22}$$

in which $x(i) = \bar{\phi}(i;x,\mathbf{u})$. The optimal control problem solved online is defined by

$$\mathbb{P}_N(x,t): \qquad V_N^0(x,t) = \min_{\mathbf{u}} \{V_N(x,t,\mathbf{u}) \mid \mathbf{u} \in \mathbb{U}^N\}$$

The *only* constraint in $\mathbb{P}_N(x,t)$ is the control constraint. The control applied to the uncertain system is $\kappa_N(x,t)$, the first element of $\mathbf{u}^0(x,t) = (u^0(0;x,t), u^0(1;x,t), \ldots, u^0(N-1;x,t))$, the optimizing control sequence. The associated optimal state sequence is $\mathbf{x}^0(x,t) = (x^0(0;x,t), \ldots, x^0(1;x,t), \ldots, x^0(N-1;x,t))$. The terminal penalty $V_f(\cdot)$, and the functions $\bar{f}(\cdot)$ and $\ell(\cdot)$ are assumed to satisfy the usual assumptions 2.2, 2.3, and 2.14 for the nominal system $\bar{x}^+ = \bar{f}(\bar{x},\bar{u})$. In addition, $f : x \mapsto f(x,t,u)$ is assumed to be Lipschitz continuous for all $x \in \mathbb{R}^n$, uniformly in $(t,u) \in \mathbb{I}_{0:N} \times \mathbb{U}$, and $\ell(\cdot)$ is assumed to be quadratic and positive definite. Also, the linearization of $\bar{f}(\cdot)$ at $(0,0)$ is assumed to be stabilizable.

We first address the problem that $\mathbb{P}_N(\cdot)$ has no terminal constraint. The function $V_f'(\cdot)$ and associated controller $\kappa_f(\cdot)$ is chosen, as in Section 2.5.5, to be a local Lyapunov function for the nominal system $\bar{x}^+ = \bar{f}(\bar{x},\bar{u})$. The terminal cost $V_f(\cdot)$ is set equal to $\beta V_f'(\cdot)$ with β chosen as shown in the following proposition. The associated terminal constraint $\mathbb{X}_f := \{x \mid V_f'(x) \le \alpha\}$ for some $\alpha > 0$ is not employed in the optimal control problem, but is needed for analysis. For any state sequence $\bar{\mathbf{x}}$ let $\mathbf{X}_c(\bar{\mathbf{x}})$ denote the tube (sequence of sets) $(X_0^c(\bar{\mathbf{x}}), X_1^c(\bar{\mathbf{x}}), \ldots)$ in which the ith element of the sequence is $X_i^c(\bar{\mathbf{x}}) := \{x \mid V_N^0(x,i) \le c\}$. The tube $\mathbf{X}_d(\bar{\mathbf{x}})$ is similarly defined.

Proposition 3.16 (Implicit satisfaction of terminal constraint)**.** *For all $c > 0$ there exists a $\beta_c := c/\alpha$ such that, for any $i \in \mathbb{I}_{\ge 0}$ and any $x \in X_i^c(\bar{\mathbf{x}}^0)$, the terminal state $x^0(N;x_0,i)$ lies in $\mathbb{X}_f$ if $\beta \ge \beta_c$.*

Proof. Since $x \in X_i^c(\bar{\mathbf{x}}^0)$ implies $V_N^0(x,i) \le c$, we know $V_f(x^0(N;x,i)) = \beta V_f'(x^0(N;x,i)) \le c$ so that $x^0(N;x_0,i) \in \mathbb{X}_f$ if $\beta \ge \beta_c$. ∎

Proposition 3.16 shows that the constraint that the terminal state lies in $\mathbb{X}_f$ is implicitly satisfied if $\beta \geq \beta_c$ and the initial state lies in $X_i^c(\bar{\mathbf{x}}^0)$ for any $i \in \mathbb{I}_{\geq 0}$. The next Proposition establishes important properties of the value function $V_N^0(\cdot)$.

Proposition 3.17 (Properties of the value function)**.** *Suppose $\beta \geq \beta_c$. There exist constants $c_1 > 0$ and $c_2 > 0$ such that*

(a) $V_N^0(x,t) \geq c_1 \left| x - \bar{x}^0(t) \right|^2 \ \forall (x,t) \in \mathbb{R}^n \times \mathbb{I}_{\geq 0}$

(b) $V_N^0(x,t) \leq c_2 \left| x - \bar{x}^0(t) \right|^2 \ \forall (x,t) \in \mathbb{R}^n \times \mathbb{I}_{\geq 0}$

(c) $V_N^0((x,t)^+) \leq V_N^0(x,t) - c_1 \left| x - \bar{x}^0(t) \right|^2 \ \forall (x,t) \in X_i^c(\bar{x}^0) \times \mathbb{I}_{\geq 0}$

in which $(x,t)^+ = (x^+, t^+) = (\bar{f}(x, \kappa_N(x,t)), t+1)$.

It should be recalled that $\bar{x}^0(t) = 0$ and $\bar{u}^0(t) = 0$ for all $t \geq N$; the controller reverts to conventional MPC for $t \geq N$.

Proof.

(a) This follows from the fact that $V_N^0(x,t) \geq \ell(x - \bar{x}^0(t), u - \bar{u}^0(t))$ so that, by the assumptions on $\ell(\cdot)$, $V_N^0(x,t) \geq c_1 \left| x - \bar{x}^0(t) \right|^2$ for all $(x,t) \in \mathbb{R}^n \times \mathbb{I}_{\geq 0}$.

(b) We have that $V_N^0(x,t) = V_N(x, \mathbf{u}^0(x,t)) \leq V_N(x, \bar{\mathbf{u}}_t^0)$ with

$$V_N(x, \bar{\mathbf{u}}_t^0) = \sum_{i=0}^{N-1} \ell(x^0(i;x,t) - \bar{x}^0(t+i), 0) + V_f(x^0(N;x,t) - \bar{x}^0(t+N))$$

and $\bar{\mathbf{u}}_t^0 := (\bar{u}^0(t), \bar{u}^0(t+1), \bar{u}^0(t+2), \ldots)$. Lipschitz continuity of $f(\cdot)$ in x gives $\left| \bar{\phi}(i;x,\bar{\mathbf{u}}_t^0) - \bar{x}^0(i+t) \right| \leq L^i \left| x - \bar{x}^0(t) \right|$. Since $\ell(\cdot)$ and $V_f(\cdot)$ are quadratic, it follows that $V_N^0(x,t) \leq c_2 \left| x - \bar{x}^0(t) \right|^2$ for all $(x,t) \in \mathbb{R}^n \times \mathbb{I}_{\geq 0}$, for some $c_2 > 0$.

(c) It follows from Proposition 3.16 that the terminal state $x^0(N;x,t) \in \mathbb{X}_f$ so that the usual stabilizing condition is satisfied and

$$V_N^0((x,t)^+) \leq V_N^0(x,t) - \ell(x, \kappa_N(x,t))$$

The desired result follows from the lower bound on $\ell(\cdot)$. ■

It follows that the origin is asymptotically stable in the tube $\mathbf{X}_c(\bar{\mathbf{x}}^0)$ for the time-varying nominal system $(x,i)^+ = \bar{f}(x, \kappa_N(x,i))$. However, our main interest is the behavior of the uncertain system with the

controller $\kappa_N(\cdot)$. Before proceeding, we note that the tube $\mathbf{X}_c(\bar{\mathbf{x}}^0)$ is a "large" neighborhood of $\bar{\mathbf{x}}^0$ in the sense that any state/time (x, i) in this set can be controlled to $\mathbb{X}_f$ in $N - i$ steps by a control subject only to the control constraint. We wish to determine, if possible, a "small" neighborhood $\mathbf{X}_d(\bar{\mathbf{x}})$ of $\bar{\mathbf{x}}^0$, $d < c$, in which the trajectories of the uncertain system are contained by the controller $\kappa_N(\cdot)$. The size of these neighborhoods, however, are dictated by the size of the disturbance set $\mathbb{W}$ as we show next.

Proposition 3.18 (Neighborhoods of the uncertain system)**.** *Suppose* $\beta \geq \beta_c$.

(a) $V_N^0((x,t)^+) \leq \gamma V_N^0(x,t)$ *for all* $(x,t) \in X_t^d(\bar{\mathbf{x}}^0) \times \mathbb{I}_{\geq 0}$, *with* $(x,t)^+ = (x^+, t^+) = (f(x, \kappa_N(x,t), 0), t+1)$ *and* $\gamma := 1 - c_1/c_2 \in (0,1)$.

(b) $x \mapsto V_N^0(\,\cdot\,; t)$ *is Lipschitz continuous with Lipschitz constant* $c_3 > 0$ *in the compact set* $X_t^c(\bar{\mathbf{x}}^0) = \{x \mid V_N^0(x,t) \leq c\}$ *for all* $t \in \mathbb{I}_{0:N}$.

(c) $V_N^0(f(x, \kappa_N(x,t), w), t+1) \leq \gamma V_N^0(x,t) + c_3 |w|$ *for all* $(x,t) \in (X_t^c(\bar{\mathbf{x}}^0) \oplus \mathbb{W}) \times \mathbb{I}_{\geq 0}$.

Proof.

(a) This inequality follows directly from Proposition 3.17.

(b) This follows, as shown in Theorem C.29 in Appendix C, from the fact that $x \mapsto V_N^0(x,t)$ is Lipschitz continuous on bounded sets for each $t \in \mathbb{I}_{0:N}$, since $V_N(\cdot)$ is Lipschitz continuous on bounded sets and $\mathbf{u}$ lies in the compact set $\mathbb{U}^N$.

(c) The final inequality follows from (a), (b), and Proposition 3.17. ■

Proposition 3.19 (Robust positive invariance of tube-based MPC for nonlinear systems)**.**

(a) Suppose $\beta \geq \beta_c$ *and* $V_N^0(x,t) \leq d < c$ $(x \in X_t^d(\bar{\mathbf{x}}^0))$, *then* $V_N^0(x,t)^+ \leq d$ $(x \in X_{t+1}^d(\bar{\mathbf{x}}^0))$ *with* $(x,t)^+ = (x^+, t^+) = (f(x, \kappa_N(x,t), w), t+1)$ *if* $d \geq (c_3/(1-\gamma))\,|\mathbb{W}|$, $|\mathbb{W}| := \max_w \{|w| \mid w \in \mathbb{W}\}$.

(b) Suppose $\varepsilon > 0$. *Then* $V_N^0((x,t)^+) \leq V_N^0(x,t) - \varepsilon$ *if* $V_N^0(x) \geq d_\varepsilon := (c_3/(1-\gamma))\mathbb{W} + \varepsilon$.

Proof.

(a) It follows from Proposition 3.17 that

$$V_N^0(f(x, \kappa_N(x,t), w), t+1) \leq \gamma d + c_3 |w|$$

If $d \geq (c_3/(1-\gamma))\,|\mathbb{W}|$, then

$$V_N^0(f(x,\kappa_N(x,t),w),t+1) \leq [(\gamma c_3)/(1-\gamma)+c_3]\,|\mathbb{W}| \leq [c_3/(1-\gamma)]\,|\mathbb{W}|$$

(b) $V_N^0(f(x,\kappa_N(x,t),w) \leq V_N^0(x)-\varepsilon$ if $\gamma V_N^0(x)+c_3\mathbb{W} \leq V_N^0(x)-\varepsilon$, i.e., if $V_N^0(x) \geq [c_3/(1-\gamma)]\mathbb{W}+\varepsilon$. ■

These results show that—provided the inequalities $c \geq (c_3/(1-\gamma))\,|\mathbb{W}|$ and $d \geq (c_3/(1-\gamma))\,|\mathbb{W}|$ are satisfied—the tubes $\mathbf{X}_c(\bar{\mathbf{x}}^0)$ and $\mathbf{X}_d(\bar{\mathbf{x}}^0)$ are robustly positive invariant for $(x,t)^+ = (f(x,\kappa_N(x,t)), t+1)$, $w \in \mathbb{W}$ in the sense that if $x \in X_t^c(\bar{\mathbf{x}}^0)$ ($x \in X_t^d(\bar{\mathbf{x}}^0)$), then $x^+ \in X_{t+1}^c(\bar{\mathbf{x}}^0)$ ($x^+ \in X_{t+1}^d(\bar{\mathbf{x}}^0)$). The tubes $\mathbf{X}_c(\bar{\mathbf{x}}^0)$ and $\mathbf{X}_d(\bar{\mathbf{x}}^0)$ may be regarded as analogs of the sublevel sets $\mathrm{lev}_c\, V_N^0(\cdot)$ and $\mathrm{lev}_d\, V_N^0(\cdot)$ for time-invariant systems controlled by conventional MPC. If $d = d_\varepsilon$ and c is large (which implies $\beta = c/\alpha$ is large), are such that tube $\mathbf{X}_d(\bar{\mathbf{x}}^0) \subset \mathbf{X}_c(\bar{\mathbf{x}}^0)$, then any trajectory commencing at $x \in X_t^c(\bar{\mathbf{x}}^0)$ converges to the tube $\mathbf{X}_d(\bar{\mathbf{x}}^0)$ in finite time and thereafter remains in the tube $\mathbf{X}_d(\bar{\mathbf{x}}^0)$. It follows that $d_H(X_i^d(\bar{\mathbf{x}}^0), X_N^c(\bar{\mathbf{x}}^0)$ becomes zero when i exceeds some finite time not less than N.

3.6.3 Choosing the Nominal Constraint Sets $\bar{\mathbb{U}}$ and $\bar{\mathbb{X}}$

The first task is to choose d as small as possible given the constraint $d \geq d_\varepsilon$, and to choose c large. If the initial state x_0 lies in $X_0^c(\bar{\mathbf{x}}^0)$ (this can be ensured by setting $\bar{x}_0 = x_0$), then *all* state trajectories of the uncertain system lie in the tube $\mathbf{X}_c(\bar{\mathbf{x}}^0)$ and converge to the tube $\mathbf{X}_d(\bar{\mathbf{x}}^0)$. As $d \to 0$, the tube $\mathbf{X}_d(\bar{\mathbf{x}}^0)$ shrinks to the nominal trajectory $\bar{\mathbf{x}}^0$. If d is sufficiently small, and if $\bar{\mathbb{X}}$ is a sufficiently small subset of $\mathbb{X}$, all state trajectories of the uncertain system lie in the state constraint set $\mathbb{X}$. This is, of course, a consequence of the fact that the nominal trajectory $\bar{\mathbf{x}}^0$ lies in the tightened constraint set $\bar{\mathbb{X}}$.

The set $\bar{\mathbb{U}}$ is chosen next. Since $\mathbb{U}$ is often a box constraint, a simple choice would be $\bar{\mathbb{U}} = \theta\mathbb{U}$ with $\theta \in (0,1)$. This choice determines how much control is devoted to controlling $\bar{x}_0$ to 0, and how much to reduce the effect of the disturbance w. It is possible to change this choice online.

The main task is to choose $\bar{\mathbb{X}}$. This can be done as follows. Assume that $\mathbb{X}$ is defined by a set of inequalities of the form $g_i(x) \leq h_i$, $i \in \mathbb{I}_{1:J}$. Then $\bar{\mathbb{X}}$ may be defined by the set of "tightened" inequalities $g_i(x) \leq \alpha_i h_i$, $i \in \mathbb{I}_{1:J}$, in which each $\alpha_i \in (0,1)$. Let $\alpha := (\alpha_1, \alpha_2, \ldots, \alpha_J)$. Then the "design parameter" α is chosen to satisfy the constraint that the state trajectory of the controlled uncertain system lies in $\mathbb{X}$ for all $x_0 \in$

$\mathcal{X}_0$ (the set of potential initial states), and all disturbance sequences $\mathbf{w} \in \mathbb{W}^N$. This is a complex semi-infinite optimization problem, but can be solved offline using recent results in Monte Carlo optimization that show the constraints can be satisfied with "practical certainty," i.e., with probability exceeding $1 - \beta$, $\beta \ll 1$, using a manageable number of random samples of $\mathbf{w}$.

Example 3.20: Robust control of an exothermic reaction

Consider the control of a continuous-stirred-tank reactor. We use a model derived in Hicks and Ray (1971) and modified by Kameswaran and Biegler (2006). The reactor is described by the second-order differential equation

$$\dot{x}_1 = (1/\theta)(1 - x_1) - kx_1 \exp(-M/x_2)$$
$$\dot{x}_2 = (1/\theta)(x_f - x_2) + kx_1 \exp(-M/x_2) - \alpha u(x_2 - x_c) + w$$

in which x_1 is the product concentration, x_2 is the temperature, and u is the coolant flowrate. The model parameters are $\theta = 20$, $k = 300$, $M = 5$, $x_f = 0.3947$, $x_c = 0.3816$, and $\alpha = 0.117$. The state, control, and disturbance constraint sets are

$$\mathbb{X} = \{x \in \mathbb{R}^2 \mid x_1 \in [0, 2],\ x_2 \in [0, 2]\}$$
$$\mathbb{U} = \{u \in \mathbb{R} \mid u \in [0, 2]\}$$
$$\mathbb{W} = \{w \in \mathbb{R} \mid w \in [-0.001, 0.001]\}$$

The controller is required to steer the system from a locally stable steady state $x(0) = (0.9831, 0.3918)$ at time zero, to a locally unstable steady state $z_e = (0.2632, 0.6519)$. Because the desired terminal state is z_e rather than the origin, the stage cost $\ell(z, v)$ is replaced by $\ell(z - z_e, v - v_e)$ where $\ell(z, v) := (1/2)(|z|^2 + v^2)$ and (z_e, v_e) is an equilibrium pair satisfying $z_e = f(z_e, v_e)$; the terminal constraint set $\mathbb{Z}_f$ is chosen to be $\{z_e\}$. The constraint sets for the nominal control problem are $\mathbb{Z} = \mathbb{X}$ and $\mathbb{V} = [0.02, 2]$. Since the state constraints are not activated, there is no need to tighten $\mathbb{X}$. The disturbance is chosen to be $w(t) = A\sin(\omega t)$ where A and ω are independent uniformly distributed random variables, taking values in the sets $[0, 0.001]$ and $[0, 1]$, respectively. The horizon length is $N = 40$ and the sample time is $\Delta = 3$ giving a horizon time of 120. The model predictive controller uses $\ell_a(x, u) = (1/2)(|x|^2 + u^2)$, and the same horizon and sample time.

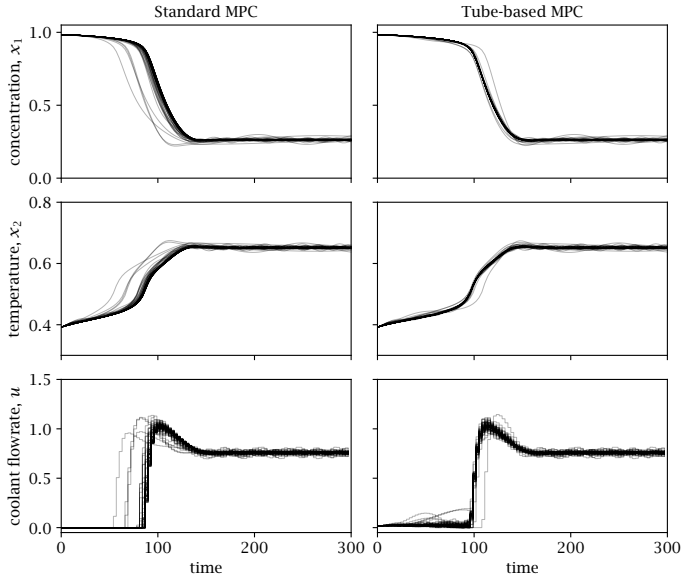


Figure 3.6: Comparison of 100 realizations of standard and tube-based MPC for the chemical reactor example.

For comparison, the performance of a standard MPC controller, using the same stage cost and the same terminal constraint set as that employed in the central-path controller, is simulated. Figure 3.6 (left) illustrates the performance of standard MPC, and Figure 3.6 (right) the performance of tube-based MPC for 100 realizations of the disturbance sequence. Tube-based MPC, as expected, has a smaller spread of state trajectories than is the case for standard MPC. Because each controller has the same stage cost and terminal constraint, the spread of trajectories in the steady-state phase is the same for the two controllers. Because the control constraint set for the central-path controller is tighter than that for the standard controller, the tube-based controller is somewhat slower than the standard controller.

The model predictive controller may be tuned to reduce more effectively the spread of trajectories due to the external disturbance. The main purpose of the central-path controller is to steer the system from one equilibrium state to another, while the purpose of the ancillary model predictive controller is to reduce the effect of the disturbance. These different objectives may require different stage costs.

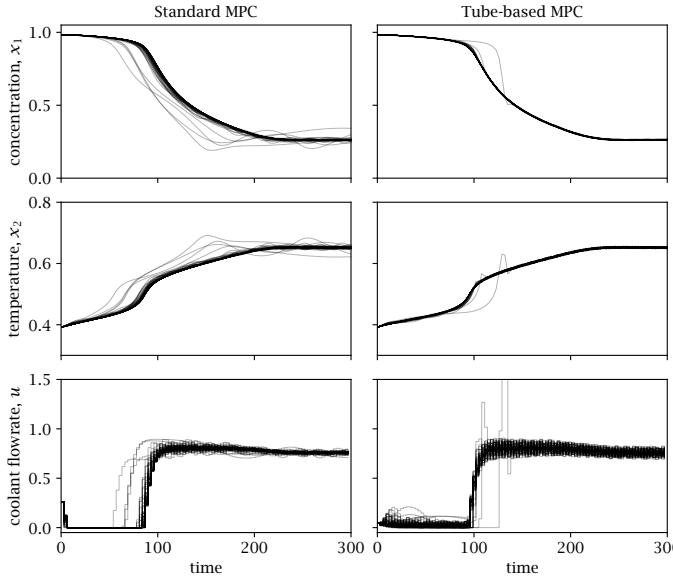


Figure 3.7: Comparison of standard and tube-based MPC with an aggressive model predictive controller.

The next simulation compares the performance of the standard and tube-based MPC when a more "aggressive" stage cost is employed for the model predictive controller. Figure 3.7 shows the performance of these two controllers when the central-path and standard MPC controller employ $\ell(z - z_e, v - v_e)$ with $\ell(z, v) := (1/2)|z|^2 + 5v^2$, and the ancillary model predictive controller employs $\ell_a(x, u) = 50|x|^2 + (1/20)u^2$. The tube-based MPC controller reduces the spread of the trajectories during both the transient *and* the steady-state phases.

It is also possible to tune the sample time of the ancillary model predictive controller. This feature may be useful when the disturbance frequency lies outside the pass band of the central-path controller. Figure 3.8 shows how concentration varies with time when the disturbance is $w(t) = 0.002\sin(0.4t)$, the sample time of the central-path controller is 12, whereas the sample time of the ancillary model predictive controller is 12 (left figure) and 8 (right figure). The central-path controller employs $\ell(z - z_e, v - v_e)$ where $\ell(z, v) := (1/2)(|z|^2 + v^2)$, and the model predictive controller employs the same stage cost $\ell_a(x,$

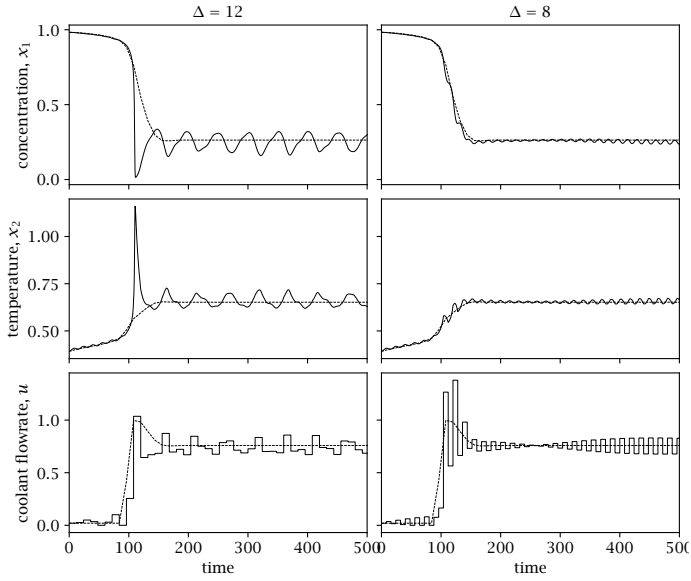


Figure 3.8: Concentration versus time for the ancillary model predictive controller with sample time $\Delta = 12$ (left) and $\Delta = 8$ (right).

$u) = \ell(x, u)$. The model predictive controller with the smaller sample time is more effective in rejecting the disturbance. □

3.7 Stochastic MPC

3.7.1 Introduction

In stochastic MPC, as in robust MPC, the system to be controlled is usually described by $x^+ = f(x, u, w)$, in which the disturbance w is a random variable that is assumed to take values in $\mathbb{W}$. The constraint set $\mathbb{W}$ is not necessarily assumed to be bounded as it is in robust MPC, although, to date, implementable versions appear to require boundedness of $\mathbb{W}$. The decision variable $\boldsymbol{\mu}$ is usually assumed, as in robust MPC, to be a policy $\boldsymbol{\mu} = (\mu_0(\cdot), \mu_1(\cdot), \ldots, \mu_{N-1}(\cdot))$ (a sequence of control laws) in order to contain the spread of trajectories that may result in a high cost and constraint violation. The functions

$\mu_i(\cdot)$ are usually parameterized to simplify optimization. A parameterization that is widely used when the system being controlled is linear is $\mu_i(x) = Kx + v_i$, in which case the decision variable is simply the sequence $\mathbf{v} = (v_0, v_1, \ldots, v_{N-1})$. Let $\phi(i; x, \boldsymbol{\mu}, \mathbf{w})$ denote the solution of $x^+ = f(x, u, w)$ at time i if the initial state at time zero is x, the control at (x, i) is $\mu_i(x)$, and the disturbance sequence is $\mathbf{w}$.

The cost that is minimized online is usually defined to be

$$V_N(x, \boldsymbol{\mu}) = \mathcal{E}_{|x}(J_N(x, \boldsymbol{\mu}, \mathbf{w}))$$
$$J_N(x, \mathbf{u}, \mathbf{w}) = \sum_{i=0}^{N-1} \ell(x(i), \mu_i(x(i))) + V_f(\phi(N; x, \boldsymbol{\mu}, \mathbf{w}))$$

in which $\mathcal{E}_{|x}(\cdot) = \mathcal{E}(\cdot \mid x(0) = x))$, $\mathcal{E}(\cdot)$ is the expectation under the probability measure of the underlying probability space, and $x(i) = \phi(i; x, \boldsymbol{\mu}, \mathbf{w})$. For simplicity, the nominal cost $V_N(x, \boldsymbol{\mu}) = \mathcal{E}_{|x}(J_N(x, \boldsymbol{\mu}, \mathbf{0}))$ is sometimes employed; here $\mathbf{0}$ is defined to be the sequence $(0, 0, \ldots, 0)$.

We consider briefly below three versions of MPC associated with three versions of the optimal control problem $\mathbb{P}_N(x)$ solved online. In the first version there are no constraints, permitting the disturbance to be unbounded. In the second version the hard constraints $x \in \mathbb{X}$, $u \in \mathbb{U}$ and the terminal constraint $x(N) \in \mathbb{X}_f$ are required to be satisfied. While satisfaction of the constraint $x \in \mathbb{X}$ almost surely is desirable, this constraint is often regarded as too conservative. The third version, therefore, replaces the hard constraint $x \in \mathbb{X}$ by the probabilistic (chance) constraint of the form

$$\Pr_{|x}(x(i) \in \mathbb{X}) \geq 1 - \varepsilon$$

for some suitably small $\varepsilon \in [0, 1]$. Some papers propose treating the hard control constraint $u \in \mathbb{U}$ similarly. This approach is not appropriate for process control since hard actuator constraints have to be satisfied; a valve cannot be more than fully open or less than fully closed. In a similar vein, softening of the terminal constraint may result in instability. Hence, the constraints in the third version on the system being controlled take the form

$$\Pr_{|x}(x(i) \in \mathbb{X}) \geq 1 - \varepsilon$$
$$u(i) \in \mathbb{U}$$

for all $i \in \mathbb{I}_{0:N}$. $\Pr(\cdot)$ denotes the probability measure of the underlying probability space and $\Pr_{|x}(\cdot)$ the probability measure conditional on $x(0) = x$. Also $x(i) := \phi(i; x, \boldsymbol{\mu}, \mathbf{w})$ and $u(i) = \mu_i(x(i))$.

Let $\Pi_N(x)$ denote the set of parameterized policies that satisfy the constraints appropriate to the version being considered and the initial state is x. The optimal control problem $\mathbb{P}_N(x)$ that is solved online can now be defined by

$$\mathbb{P}_N(x): \qquad V_N^0(x) = \min_{\boldsymbol{\mu}\in\Pi_N(x)} V_N(x,\boldsymbol{\mu})$$

subject to the constraints defined above as well as the hard terminal stability constraint $x(N) \in \mathbb{X}_f$. The solution to this problem, if it exists, is $\boldsymbol{\mu}^0(x) = (\mu_0^0(x), \mu_1^0(x), \ldots, \mu_{N-1}^0(x))$. The control applied to the uncertain system at state x is $\kappa_N(x) := \mu_0^0(x)$.

3.7.2 Stability of Stochastic MPC

Because the optimal control problem solved online has a finite horizon the resultant control law is not necessarily stabilizing. Stabilizing conditions involving the addition of a terminal cost and a terminal constraint set have been developed for deterministic and robust MPC but, as pointed out in Chatterjee and Lygeros (2015), no approaches to stochastic MPC prior to 2015 dealt "directly with stability under receding horizon control as a standalone and fundamental problem."

Version 1. A major contribution to stability and performance of stochastic MPC in the absence of hard constraints is given in the paper by Chatterjee and Lygeros that is the first paper proposing "standalone" stability conditions for unconstrained stochastic MPC. The problem considered in this paper is as stated above except that there are no constraints ($\mathbb{X} = \mathbb{X}_f = \mathbb{R}^n$, $\mathbb{U} = \mathbb{R}^m$) and the random disturbance w is merely assumed to take values in a measurable set $\mathbb{W}$ that is not necessarily bounded. The stabilizing assumption in Chatterjee and Lygeros (2015) is

Assumption 3.21 (Stabilizing conditions, stochastic MPC: Version 1)**.** There exists a measurable control law $\kappa_f(\cdot) : \mathbb{R}^n \to \mathbb{R}^m$ a number b and a bounded measurable set K such that

$$\mathcal{E}(V_f(f(x,\kappa_f(x),w)) \le V_f(x) - \ell(x,\kappa_f(x)) \quad \forall x \notin K$$

$$\sup_{x\in K}\{\mathcal{E}(V_f(f(x,\kappa_f(x),w))) - (V_f(x) - \ell(x,\kappa_f(x)))\} \le b$$

Under the basic assumptions that (i) the cost $V_N(x,\boldsymbol{\mu})$ is finite for all $x \in \mathbb{R}^n$ and all $\boldsymbol{\mu} \in \Pi_N(x)$; (ii) for all $x \in \mathbb{R}^n$, there exists a solution

$\boldsymbol{\mu}^0(x)$ that solves $\mathbb{P}_N(x)$; and (iii) the stage cost $\ell(\cdot)$ satisfies some modest conditions, it is shown in (Chatterjee and Lygeros, 2015) that, if Assumption 3.21 holds, then, for all $x \in \mathbb{R}^n$

$$\mathcal{E}_{|x}(V_N^0(x^+)) \leq V_N^0(x) - \ell(x, \kappa_N(x)) + b$$

Chatterjee and Lygeros then show that $V_N^0(\cdot)$ satisfies the geometric drift condition $\mathcal{E}_{|x}(V_N^0(x^+)) \leq V_N^0(x) - \ell(x, \kappa_N(x))$ outside of some compact subset of $\mathbb{R}^n$, and that the sequence $(\mathcal{E}_{|x}(V_N^0(x(t))))_{t \in \mathbb{I}_{\geq 0}}$ is bounded.

Version 2. While the results in Chatterjee and Lygeros (2015) hold for situations in which the disturbance is not restricted to lie in a compact set, they do require the absence of hard state constraints. In addition, determination of a function satisfying Assumption 3.21 is difficult. Stabilizing conditions suitable for version 2 of stochastic MPC (all constraints are hard and $\mathbb{W}$ is compact) are given in Mayne and Falugi (2019).

Assumption 3.22 (Stabilizing conditions, stochastic MPC: Version 2)**.** $V_f(\cdot)$, $\mathbb{X}_f$ and $\ell(\cdot)$ have the following properties.

(a) For all $x \in \mathbb{X}_f$ there exists a $u = \kappa_f(x) \in \mathbb{U}$ such that $V_f(f(x, \kappa_f(x), 0)) \leq V_f(x) - \ell(x, \kappa_f(x))$ and $f(x, \kappa_f(x), w) \in \mathbb{X}_f, \ \forall w \in \mathbb{W}$

(b) There exists a $\delta \in (0, \infty)$ such that for all $x \in \mathbb{X}_f$

$$\mathcal{E}_{|x}(V_f(f(x, \kappa_f(x), w))) \leq V_f(x) - \ell(x, \kappa_f(x)) + \delta$$

(c) $\mathbb{X}_f \subseteq \mathbb{X}$, $\mathbb{W}$ is compact.

(d) There exist constants $c_2 > c_1 > 0$ and $a > 0$ such that

$$\ell(x, u) \geq c_1 |x|^a, \ \forall x \in \mathbb{X}, \ \forall u \in \mathbb{U}$$
$$V_N^0(x) \leq c_2 |x|^a, \ \forall x \text{ such that } \Pi_N(x) \neq \varnothing$$

If this assumption is satisfied, it follows (Mayne and Falugi, 2019) that, for x such that $\Pi_N(x)$ is not empty, the optimal control problem $\mathbb{P}_N(x)$ is recursively feasible and

$$\mathcal{E}_{|x}(V_N^0(x^+)) \leq V_N^0(x) - \ell(x, \kappa_N(x)) + \delta \tag{3.23}$$

which is a modified descent property. Consider now an infinite random sequence $(x(i))_{i \in \mathbb{I}_{\geq 0}}$ generated by the control algorithm and stochastic system.

Proposition 3.23 (Expected cost bound)**.** *If Assumption 3.22 holds, then there exists* $\lambda \in (0,1)$ *such that the closed-loop trajectory* $x(k)$ *satisfies*

$$\mathcal{E}_{|x}(V_N^0(x(k))) \leq \lambda^k V_N^0(x) + \delta/(1-\lambda)$$

for all $k \in \mathbb{I}_{\geq 0}$ *and* $x \in \mathbb{X}$ *such that* $\Pi_N(x) \neq \emptyset$ *and* $x(0) = x$.

Proof. We proceed as in the deterministic case to obtain

$$\begin{aligned}\mathcal{E}_{|x}(V_N^0(x^+)) &\leq V_N(x) - c_1|x|^a + \delta \\ &\leq V_N(x) - (c_1/c_2)V_N(x) + \delta \\ &\leq \lambda V_N(x) + \delta\end{aligned}$$

with $\lambda = 1 - c_1/c_2$. Then $\mathcal{E}_{|x(0)}(V_N^0(x(1))) \leq \lambda V_N^0(x(0)) + \delta$ and, by law of iterated expectation, $\mathcal{E}_{|x(0)}(V_N^0(x(k))) = \mathcal{E}_{|x(0)}(\mathcal{E}_{|x(k-1)}(V_N^0(x(k))))$. By iterating we obtain our stability condition

$$\mathcal{E}_{|x(0)}(V_N^0(x(k))) \leq \lambda^k V_N(x) + \delta/(1-\lambda)$$

and the proof is complete. ∎

Remark. Version 1 is applicable to model predictive control of systems that have unbounded disturbances but do not have hard state and control constraints. Moreover it requires determination of a global Lyapunov-like function $V_f(\cdot)$ defined in Assumption 3.21. Version 2 requires compactness of $\mathbb{W}$ since it is applicable to model predictive control of systems that requires solution of a complex optimal control problem in which hard constraints have to be satisfied for all permitted disturbance sequences.

We turn next to an implementable version of stochastic MPC.

3.7.3 Tube-based stochastic MPC

To date, it appears that all implementable versions of stochastic MPC assume boundedness of the disturbance since, otherwise, it is difficult, if not impossible, to satisfy hard constraints. Even if the disturbance is bounded, however, satisfaction of hard constraints for *all disturbance sequences* is not simple. The tube-based approach, introduced in Chisci et al. (2001); Mayne and Langson (2001) appears to be the most practical method for handling hard constraints. The reason for this is the state and control of the uncertain system are forced to satisfy hard constraints merely by requiring the state and control of the nominal, deterministic, system to satisfy tighter versions of the same constraints—a

much simpler problem than forcing satisfaction of the constraints for all disturbance sequences using optimization. We present below a similar simple approach to stochastic MPC, the major difference from robust MPC being a minor modification required to tighten the constraints to permit a small probability of nonsatisfaction.

Control strategy. We present a control strategy that ensures stability in the sense that under reasonable assumptions the state converges to the optimal solution of the unconstrained linear system. The uncertain system to be controlled is described by

$$x^+ = Ax + Bu + w$$

and is subject to the constraints $x \in \mathbb{X}$ and $u \in \mathbb{U}$; $\mathbb{X}$ is closed, $\mathbb{U}$ is compact, and each set contains the origin in its interior. The disturbance w is a stationary random process and is assumed to lie in the compact set $\mathbb{W}$ that contains the origin in its interior. The nominal system and error are described by

$$\bar{x}^+ = A\bar{x} + B\bar{u} \qquad e := x - \bar{x}$$

Given state x at time t, we denote the solution of the nominal system by $\bar{x}(i)$ for given controls $\bar{u}(i)$, $i \in \mathbb{I}_{0:N-1}$ where the initial state of the nominal system is $\bar{x}(0) = x$. As in robust tube-based MPC, we employ the control policy $\boldsymbol{\mu} = (\mu_0(\cdot), \mu_1(\cdot), \ldots, \mu_{N-1}(\cdot))$ in which for each i, $\mu_i(\cdot)$ is defined for all x by

$$\mu_i(x) := \bar{u}(i) + K(x - \bar{x}(i))$$

so that $u(i) = \mu_i(x(i)) = \bar{u}(i) + K(x(i) - \bar{x}(i))$. The (x, e) pair then evolve as

$$x^+ = Ax + B\bar{u} + BKe + w \qquad e^+ = A_K e + w \qquad A_K := A + BK$$

The feedback matrix K is chosen so that A_K is Hurwitz. For practical reasons we assume that w and, hence, e are bounded. If $\mathbf{w}$ is an infinite sequence of independent, identically distributed, zero-mean random variables, then an optimal K may be obtained from the solution to the unconstrained problem

$$\min \lim_{N\to\infty} \mathcal{E}_{|x}(1/N) \sum_{i=0}^{N-1} \ell(x(i), u(i))$$

in which $\ell(x, u) = (1/2)(x'Qx + u'Ru)$ with both Q and R positive definite. Then $K = -(R + B'PB)^{-1}B'PA$, with P the solution of the

matrix Riccati equation $P = Q + A'P(R + B'PB)^{-1}PA$; $(1/2)x'Px$ is the minimum cost for the deterministic problem in which the disturbance w is identically zero.

Stochastic MPC differs from robust MPC in the definition of the control objective and in softening of the constraints that now take the form

$$\Pr_{|x}(x(i) \in \mathbb{X}) \geq 1 - \varepsilon, \ i \in \mathbb{I}_{\geq 0N}$$
$$u(i) \in \mathbb{U}, \ i \in \mathbb{I}_{\geq 0}$$

in which $x(i) = \phi(i; x, \boldsymbol{\mu}, \mathbf{w})$. The control $u(i)$ applied to the system at time i is

$$u(i) = \bar{u}(i) + K(x - \bar{x}(i)) \ \forall x$$

With this control policy, $e(i) := x(i) - \bar{x}(i)$ is the solution at time i of the difference equation

$$e^+ = A_K e + w, \ \ e(0) = 0$$

As shown earlier $e(i) \in S_K(i) := \sum_{j=0}^{i-1} A_K^j \mathbb{W}$ for all i. Because A_K is Hurwitz, $e(t)$ converges to a stationary process e_∞ as $t \to \infty$. To achieve robustness of stochastic MPC we adopt a policy similar to that employed in robust MPC. For each i, the control constraints are tightened by determining, for each i, a set $\bar{\mathbb{U}}(i)$ that ensures $\bar{u} + Ke(i) \in \mathbb{U}$ for all $\bar{u} \in \bar{\mathbb{U}}(i)$. The state constraints are tightened by determining, for each i, a set $\bar{\mathbb{X}}(i)$ that ensures $\Pr(bx + e(i) \in \mathbb{X}) \geq 1 - \varepsilon$ for all $\bar{x} \in \bar{\mathbb{X}}(i)$.

A model predictive controller is employed to steer the state and control of the nominal system, subject to the tightened constraints, to the origin. Since $x(t) = \bar{x}(t) + e(t)$ and $\bar{u}(t) = \bar{u}(t) + Ke(t)$ it follows that $x(t)$ converges to e_∞ and $u(t)$ converges to Ke_∞ as $t \to \infty$.

To implement this control it seems, at first sight, that we have to determine the tightened constraints for all $i \in \mathbb{I}_{\geq 0}$. We propose two practical alternatives. The first, which is similar to that employed for robust MPC, is determination of constant constraint sets $\bar{\mathbb{U}}_\infty$ and $\bar{\mathbb{X}}_\infty$ satisfying, respectively, $\bar{\mathbb{U}}_\infty \oplus KS_K(\infty) \subset \mathbb{U}$ and $\mathcal{P}\{\bar{x} + e_\infty \in \mathbb{X}\} \geq 1 - \varepsilon$ for all $\bar{x} \in \bar{\mathbb{X}}_\infty$. At each time i, when the composite state is $(x(i), \bar{x}(i))$, a standard nominal optimal control problem $\bar{\mathbb{P}}_N(\bar{x}(i))$ with constraints $\bar{x} \in \bar{\mathbb{X}}_\infty$, $\bar{u} \in \bar{\mathbb{U}}_\infty$ and the usual terminal constraint is solved. If standard stability conditions are satisfied, $\bar{x}(i)$ and $\bar{u}(i)$ converge to zero while satisfying the tightened constraints $\bar{u} \in \bar{\mathbb{U}}_\infty$ and $\bar{x} \in \bar{\mathbb{X}}_\infty$ as $i \to \infty$. The control applied to the system at time i is $u(i) = \bar{u}(i) + K(x(i) - \bar{x}(i))$. This procedure is conservative in that the constraints are tighter than necessary.

A better, albeit more complex, alternative is to solve, at time zero, a standard nominal optimal control problem $\bar{\mathbb{P}}_N(\bar{x})$ with a sequence of tightened control constraints $(\bar{\mathbb{U}}(0), \bar{\mathbb{U}}(1), \ldots, \bar{\mathbb{U}}(N-1))$ and state constraints $(\bar{\mathbb{X}}(1), \bar{\mathbb{X}}(2), \ldots, \bar{\mathbb{X}}(N-1), \mathbb{X}_f)$ specified below; the solution to this problem yields the nominal control and state sequences $(\bar{u}^0(0), \bar{u}^0(1), \bar{u}^0(N-1))$ and $(\bar{x}^0(1), \bar{x}^0(2), \ldots, \bar{x}^0(N))$ satisfying $\bar{u}^0(i) \in \bar{\mathbb{U}}(i)$, $\bar{x}^0(i) \in \bar{\mathbb{X}}(i)$, and $\bar{x}^0(N) \in \mathbb{X}_f$. At time $i = N$ and thereafter the control $\bar{u}(i)$ is set equal to $\kappa_f(\bar{x}(i))$ so that $\bar{x}(i+1) = A\bar{x}(i) + B\kappa_f(\bar{x}(i))$. If the usual stability conditions are satisfied, the nominal state $\bar{x}(i)$ remains in $\mathbb{X}_f$ for all $i \geq N$. The procedure therefore yields a control sequence consisting of the sequence $(\bar{u}^0(0), \bar{u}^0(1), \ldots, \bar{u}^0_{N-1})$ followed by the infinite control sequence $(\kappa_f(\bar{x}(N), \kappa_f(\bar{x}(N+1), \ldots)$. Moreover, the control law $\kappa_f(\cdot)$ ensures that $\bar{u}(i) \to 0$ and $\bar{x}(i) \to 0$ as $i \to \infty$. To implement this procedure we require an additional assumption.

Assumption 3.24 (Robust terminal set condition)**.** The terminal set satisfies $\mathbb{X}_f \oplus S_K(\infty) \subset \mathbb{X}$.

Both procedures ensure that $x(t)$ converges to the zero mean stationary process e_∞ to which $e(t)$ converges, and that $u(t)$ converges to Ke_∞ as $t \to \infty$.

Determination of tightened constraints. The tightened control constraints must satisfy $\bar{\mathbb{U}}(i) \oplus KS_K(i) \subset \mathbb{U}$ or, equivalently, $\bar{\mathbb{U}}(i) \subset \mathbb{U} \ominus KS_K(i)$ for all $i \in \mathbb{I}_{0:N-1}$. Provided we are able to tractably calculate $S_K(i)$ for any $i \in \mathbb{I}_{0:N-1}$, we may simply define $\bar{\mathbb{U}}(i) := \mathbb{U} \ominus KS_K(i)$. Alternatively, we may use any conservative estimate $\widetilde{S}_K(i) \supset S_K(i)$ and define $\bar{\mathbb{U}}(i) := \mathbb{U} \ominus K\widetilde{S}_K(i)$. For example, we may choose $\widetilde{S}_K(i) = S_K(\infty)$ and thereby define $\bar{\mathbb{U}}(i) = \bar{\mathbb{U}}_\infty$ for all $i \in \mathbb{I}_{\geq 0}$.

We now consider determination, for any $i \in \mathbb{I}_{0:N-1}$, of the state constraint set $\bar{\mathbb{X}}(i)$ that satisfies $\Pr(\bar{x} + e(i) \in \mathbb{X}) \geq 1 - \varepsilon$ for all $\bar{x} \in \bar{\mathbb{X}}(i)$. This is a stochastic optimization problem, a field in which, fortunately, there has been considerable recent progress. Tempo, Calafiore, and Dabbene (2013) give an excellent exposition of this subject.

Suppose $\mathbb{X}$ is defined by a single constraint of the form $\{x \mid c'x \leq d\}$. For each $i \in \mathbb{I}_{0:N-1}$, we wish to determine a tighter constraint $c'\bar{x} \leq \bar{d} := d - f$, $f \in [0, d]$, such that $c'\bar{x}(i) \leq \bar{d}$ implies $c'x(i) = c'\bar{x}(i) + c'e(i) \leq d$ with probability not less than $1 - \varepsilon$. To achieve this objective, we solve the stochastic problem $\mathbb{P}$ defined by

$$\min_{f \in [0,d]} \{f \mid \Pr(c'e(i) \leq f) \geq 1 - \varepsilon\}$$

with ε chosen to be suitably small; $c'e \le f$ and $c'\bar{x} \le d - f$ imply $c'x \le d$. In Calafiore and Campi (2006), the complex probability constrained problem $\mathbb{P}$ is replaced by a scenario convex optimization problem $\mathbb{P}^s$ defined by

$$\min_{f \in [0,d]} \{ f \mid c'e(i; \mathbf{w}^j) \le f, \ \forall j \in \mathbb{I}_{1:M} \} \tag{3.24}$$

Here $\mathbf{w}^j$ denotes the j^{th} sample of the finite sequence $\{w(0), w(1), w(2), \ldots, w(i-1)\}$ and $e(i)$ is replaced by $e(i; \mathbf{w}^j)$ to denote its dependence on the random sequence $\mathbf{w}^j$.

It is shown in Calafiore and Campi (2006) and Tempo et al. (2013) that given (ε, β), there exists a relatively modest number of samples $M^*(\varepsilon, \beta)$ such that if $M \ge M^*$, one of the following two conditions hold. For each $i \in \mathbb{I}_{0:N-1}$, either problem $\mathbb{P}^s$ is infeasible, in which case the robust control problem is infeasible; or its solution $f^0(i)$ satisfies

$$\Pr(c'e(i) \le f^0(i)) \ge 1 - \varepsilon$$

with probability $1 - \beta$ (i.e., with practical certainty if β is chosen sufficiently small). The tightened state constraint set is $\bar{\mathbb{X}}(i) = \{x \mid c'x \le d - f^0(i)\} \subset \mathbb{X}$. Note that $\bar{\mathbb{X}}_\infty \subset \bar{\mathbb{X}}(i)$, i.e., using $\bar{\mathbb{X}}(i)$ is less conservative than $\bar{\mathbb{X}}_\infty$. Tempo et al. (2013) give the value

$$M^*(\varepsilon, \beta) = \frac{2}{\varepsilon} \left(\log \left(\frac{1}{\beta} \right) + n_\theta \right) \tag{3.25}$$

If $\mathbb{X} := \{x \mid Cx \le d\}$ in which $d \in \mathbb{R}^p$, we apply the procedure above to each row $c_k'x \le d_k$ of the constraint yielding $f_K^0(i)$ satisfying

$$\Pr(c_k'e(i) \le f_k^0(i)) \ge 1 - \varepsilon_k$$

for each $k = 1, \ldots, p$ if the associated scenario problem is feasible. The probability that $x(i) \in \mathbb{X}(i)$ is not less than $1 - \varepsilon$ with $\varepsilon = \sum_{j=1}^p \varepsilon_k$.

Example 3.25: Constraint tightening via sampling

Consider the scalar system $x^+ = x + u + w$, with $\mathbb{X} = \mathbb{U} = [-1, 1]$ and w uniformly distributed in $\mathbb{W} = [-1/2, 1/2]$. Using costs $Q = 1/2$, $R = 1$, the LQR gain is $K = 1/2$, which gives $A_K = 1/2$, and thus

$$S_K(i) := \sum_{j=0}^{i-1} A_K^j \mathbb{W} = [-(1 - 2^{-i}), 1 - 2^{-i}]$$

for all $i \in \mathbb{I}_{\ge 0}$. Tightening the set $\mathbb{U}$, we have

$$\bar{\mathbb{U}}(i) := \mathbb{U} \ominus KS_K(i) = (1/2)[-(1 + 2^{-i}), 1 + 2^{-i}]$$

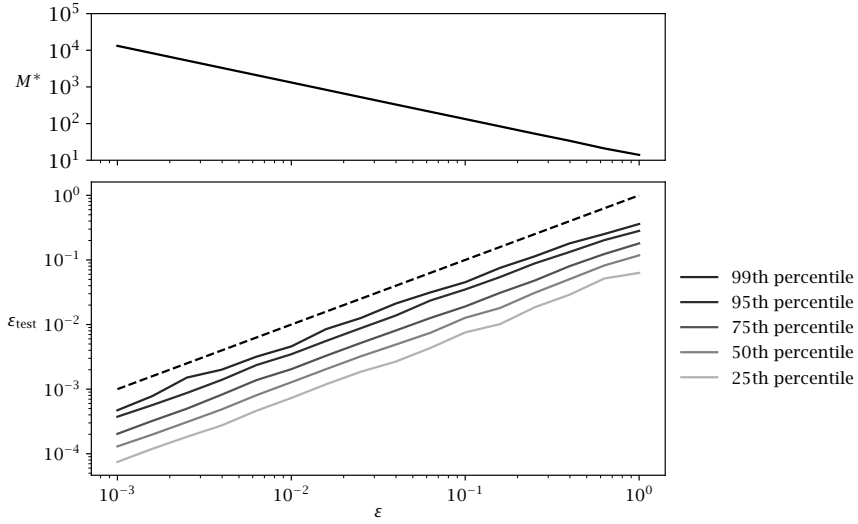


Figure 3.9: Observed probability $\varepsilon_{\text{test}}$ of constraint violation for $i = 10$. Distribution is based on 500 trials for each value of ε. Dashed line shows the outcome predicted by formula (3.25), i.e., $\varepsilon_{\text{test}} = \varepsilon$.

for all $i \in \mathbb{I}_{\geq 0}$. Note that the first control constraint does not need to be tightened at all, $\bar{\mathbb{U}}(0) = \mathbb{U}$, and all subsequent control constraints are less conservative than $\bar{\mathbb{U}}_\infty = [-1/2, 1/2]$.

To compute the tightened sets $\bar{\mathbb{X}}(i)$, we apply the sampling procedure for each $i \in \mathbb{I}_{0:N-1}$. For various values of ε, we compute the number of samples $M = M^*(\varepsilon, \beta)$ using (3.25) with $\beta = 0.01$. Then, we choose M samples of $\mathbf{w}$ and solve (3.24) for $f^0(i)$. To evaluate the actual probability of constraint violation, we then test the constraint violation using $M_{\text{test}} = 10^6$ different samples $\mathbf{w}_{\text{test}}$. That is, we compute

$$\varepsilon_{\text{test}} := \Pr\left(c'e(i; \mathbf{w}^j_{\text{test}}) > f^0(i),\ \forall j \in \mathbb{I}_{1:M_{\text{test}}}\right)$$

for each $i \in \mathbb{I}_{0:N-1}$. Note that since $\varepsilon_{\text{test}}$ is now a random variable depending on the particular M samples chosen, we repeat the process 500 times for each value of M. The distribution of $\varepsilon_{\text{test}}$ for $i = 10$ is shown in Figure 3.9. Notice that the formula (3.25) is slightly conservative, i.e., the observed probability $\varepsilon_{\text{test}}$ is half of the chosen probability ε for 99% of samples (with probability $1 - \beta$). This gap holds throughout the

entire range of the test. □

Evaluation. How does tube-based MPC compare with other methods? We discuss here two aspects: constraint handling and performance.

Traditional MPC handles constraints by solving online a finite horizon optimal control problem that involves minimization of a cost subject to control and state constraints for *every* realization of the disturbance process, a computationally expensive requirement. Tube-based MPC for linear systems appears to be the only method for avoiding this online expense but can only be employed for linear systems.

Tube-based MPC takes two different forms. One, introduced in Chisci et al. (2001), solves an online optimal control problem $\mathbb{P}_N(x)$ at the current state x and uses a tube to determine the set of all possible trajectories emanating from the current state x; the motive is to minimize the cost at each current state. The second form, introduced in Mayne and Langson (2001), solves a nominal optimal control problem $\mathbb{P}_N(x(0))$ at the *initial* state $x(0)$ and uses a single tube emanating from $x(0)$. The first form is closer to traditional MPC and attempts to minimize the cost of the particular trajectory generated by the controller as in deterministic MPC. The second form, on the other hand, minimizes the average cost over all trajectories emanating from the *initial* state $x(0)$ as in classical stochastic control in which controllers are developed offline.

While the first approach is closer to traditional MPC, its implementation is more difficult for the following reason: at each state x the successor state x^+ does not lie on the optimal trajectory emanating from x due to the disturbance w; recursive feasibility is therefore lost. Algorithmic modifications that are fairly complex have to be introduced; see, for example, Kouvaritakis and Cannon (2016); Lorenzen, Dabbene, Tempo, and Allgöwer (2016). These modifications increase computational expense and their effect on performance has not yet been studied.

The big advantage of the second approach is its simplicity; it is no more difficult to implement than traditional MPC, requiring only the determination of a nominal trajectory that converges to the origin. It is also possible to get an indication of its performance. If there are no constraints, if the horizon of the optimal control problem is infinite, and if $(w(i))_{i\in\mathbb{I}_{\geq 0}}$ is a sequence of independent, identically distributed random variables, then the optimal controller gain is $u = K(x)$ for both the stochastic and nominal systems. Thus, at composite state $(x, \bar{x})$,

the optimal control for the stochastic system at state x is $u^0 = Kx$ and the optimal control for the nominal system at state $\bar{x}$ is $\bar{u}^0 = K\bar{x}$. The control u determined by the control algorithm is $u = \bar{u}^0 + K(x^0 - \bar{x}^0) = K\bar{x}^0 + Kx^0 - K\bar{x}^0 = Kx^0$ and is therefore optimal. Next, if we accept that the control u is paramaterized by $u = \bar{u} + K(x - \bar{x})$ so that the decision variable is $\bar{\mathbf{u}}$, then, since $x = \bar{x} + e$ and $u = \bar{u} + Ke$, the performance index for the parameterized stochastic system is

$$\begin{aligned} V_N(x(0), \bar{\mathbf{u}}) &= \mathcal{E}_{|x(0)} \left[\sum_{i=0}^{N-1} \ell(x(i), u(i)) + V_f(x(N)) \right] \\ &= \sum_{i=0}^{N-1} \ell(\bar{x}(i), \bar{u}(i)) + V_f(\bar{x}(N)) + c \\ &= \bar{V}_N(\bar{x}(0), \bar{\mathbf{u}}) + c \end{aligned}$$

in which $\bar{V}_N$ is the performance index for the nominal system

$$c = \mathcal{E} \left[\sum_{i=0}^{N-1} \ell(e(i), Ke(i)) + V_f(e(N)) \right]$$

and $\ell(\cdot)$ and $V_f(\cdot)$ are quadratic functions. If, in addition, the system being controlled satisfies its control and probabilistic constraints if and only if the nominal system satisfies its tightened constraints, then the solution $\bar{\mathbf{u}}^0(x(0))$ of the nominal optimal control problem $\bar{\mathbb{P}}_N(x(0))$ is also the solution of the parameterized stochastic optimal control problem $\mathbb{P}_N(x(0))$.

3.8 Notes

Robust MPC. There is now a considerable volume of research on robust MPC; for a review of the literature up to 2000 see Mayne, Rawlings, Rao, and Scokaert (2000). Early literature examines robustness of nominal MPC under perturbations in Scokaert, Rawlings, and Meadows (1997); and robustness under model uncertainty in De Nicolao, Magni, and Scattolini (1996) and Magni and Sepulchre (1997). Sufficient conditions for robust stability of nominal MPC with modeling error are provided in Santos and Biegler (1999). Teel (2004) provides an excellent discussion of the interplay between nominal robustness and continuity of the Lyapunov function, and also presents some illuminating examples of nonrobust MPC. Robustness of the MPC controller described in

Chen and Allgöwer (1998), when employed to control a system without state constraints, is established in Yu, Reble, Chen, and Allgöwer (2011). The theory of inherent robustness is usefully extended in Pannocchia, Rawlings, and Wright (2011); Allan et al. (2017); and applied to optimal and suboptimal MPC.

Many papers propose solving online an optimal control problem in which the decision variable is a sequence of control actions that takes into account future disturbances. Thus, it is shown in Limon, Álamo, and Camacho (2002) that it is possible to determine a sequence of constraints sets that become tighter with time, and that ensure the state constraint is not transgressed if the control sequence satisfies these tightened constraints. This procedure was extended in Grimm, Messina, Tuna, and Teel (2007), who do not require the value function to be continuous and do not require the terminal cost to be a control Lyapunov function.

Predicted trajectories when the decision variable is a control sequence can diverge considerably with time, making satisfaction of state and terminal constraints difficult or even impossible. This has motivated the introduction of "feedback" MPC, in which the decision variable is a *policy* (sequence of control laws) rather than a sequence of control actions (Mayne, 1995; Kothare, Balakrishnan, and Morari, 1996; Mayne, 1997; Lee and Yu, 1997; Scokaert and Mayne, 1998). If arbitrary control laws are admissible, the implicit MPC control law is identical to that obtained by dynamic programming; see Section 3.1.3 and papers such as Magni, De Nicolao, Scattolini, and Allgöwer (2003), where a H_∞ MPC control law is obtained. But such results are *conceptual* because the decision variable is infinite dimensional. Hence practical controllers employ suboptimal policies that are finitely parameterized—an extreme example being nominal MPC. A widely used parameterization is $u = v + Kx$, particularly when the system being controlled is linear; this parameterization was first proposed in Rossiter, Kouvaritakis, and Rice (1998). The matrix K is chosen to stabilize the unconstrained linear system, and the decision variable is the sequence $(v(i))_{0:N-1}$.

The robust suboptimal controllers discussed in this chapter employ the concept of tubes introduced in the pioneering papers by Bertsekas and Rhodes (1971a,b), and developed for continuous time systems by Aubin (1991) and Khurzhanski and Valyi (1997). In robust MPC, local feedback is employed to confine all trajectories resulting from the random disturbance to lie in a tube that surrounds a nominal trajectory chosen to ensure the whole tube satisfies the state and control con-

straints. Robustly positive invariant sets are employed to construct the tubes as shown in (Chisci et al., 2001) and (Mayne and Langson, 2001). Useful references are the surveys by Blanchini (1999), and Kolmanovsky and Gilbert (1995), as well as the recent book by Blanchini and Miani (2008). Kolmanovsky and Gilbert (1995) provide extensive coverage of the theory and computation of minimal and maximal robust (disturbance) invariant sets.

The computation of approximations to robust invariant sets that are themselves invariant is discussed in a series of papers by Raković and colleagues (Raković, Kerrigan, Kouramas, and Mayne, 2003; Raković et al., 2005a; Raković, Mayne, Kerrigan, and Kouramas, 2005b; Kouramas, Raković, Kerrigan, Allwright, and Mayne, 2005). The tube-based controllers described above are based on the papers (Langson, Chryssochoos, Raković, and Mayne, 2004; Mayne, Serón, and Raković, 2005). Construction of robust invariant sets is restricted to systems of relatively low dimension, and is avoided in Section 3.6.3 by employing optimization directly to determine tightened constraints. A tube-based controller for nonlinear systems is presented in Mayne, Kerrigan, van Wyk, and Falugi (2011).

Because robust MPC is still an active area of research, other methods for achieving robustness have been proposed. Diehl, Bock, and Kostina (2006) simplify the robust nonlinear MPC problem by using linearization, also employed in (Nagy and Braatz, 2004), and present some efficient numerical procedures to determine an approximately optimal control sequence. Goulart, Kerrigan, and Maciejowski (2006) propose a control that is an affine function of current and past states; the decision variables are the associated parameters. This method subsumes the tube-based controllers described in this chapter, and has the advantage that a separate nominal trajectory is not required. A disadvantage is the increased complexity of the decision variable, although an efficient computational procedure that reduces computational time per iteration from $O(N^6)$ to $O(N^3)$ has been developed in Goulart, Kerrigan, and Ralph (2008). Interesting extensions to tube-based MPC are presented in Raković (2012), and Raković, Kouvaritakis, Cannon, Panos, and Findeisen (2012). The introduction of a novel parameterization by Raković (2012) enables him to establish that the solution obtained is equivalent to dynamic programming in at least three cases.

Considerable attention has recently been given to input-to-state stability of uncertain systems. Thus Limon, Alamo, Raimondo, de la Peña, Bravo, and Camacho (2008) present the theory of input-to-state sta-

bility as a unifying framework for robust MPC, generalizes the tube-based MPC described in (Langson et al., 2004), and extends existing results on min-max MPC. Another example of research in this vein is the paper by Lazar, de la Peña, Heemels, and Alamo (2008) that utilizes input-to-state practical stability to establish robust stability of feedback min-max MPC. A different approach is described by Angeli, Casavola, Franze, and Mosca (2008) where it is shown how to construct, for each time i, an ellipsoidal inner approximation $\mathcal{E}_i$ to the set $\mathcal{T}_i$ of states that can be robustly steered in i steps to a robust control invariant set $\mathcal{T}$. All that is required from the online controller is the determination of the minimum i such that the current state x lies in $\mathcal{E}_i$ and a control that steers $x \in \mathcal{E}_i$ into the set $\mathcal{E}_{i-1} \subset \mathcal{E}_i$.

Stochastic MPC. Interest in stochastic MPC has increased considerably. An excellent theoretical foundation is provided in Chatterjee and Lygeros (2015). Most papers address the stochastic constrained linear problem and propose that the online optimal control problem $\mathbb{P}_N(x)$ (x is the current state) minimizes a suitable objective function subject to satisfaction of state constraints with a specified probability as discussed above. If time-invariant probabilistic state constraints are employed, a major difficulty with this approach, as pointed out in Kouvaritakis, Cannon, Raković, and Cheng (2010) in the context of stochastic MPC for constrained linear systems, is that recursive feasibility is lost unless further measures are taken. It is assumed in this paper, as well as in a later paper Lorenzen et al. (2016), that the disturbance is bounded, enabling a combination of stochastic and hard constraints to be employed.

In contrast to these papers, which employ the control policy parameterization $u = Kx + v$, Chatterjee, Hokayem, and Lygeros (2011) employ the parameterization, first proposed in Goulart et al. (2006), in which the control law is an affine function of finite number of past disturbances. This parameterization, although not parsimonious, results in a convex optimal control problem, which is advantageous. Recursive feasibility is easily achieved in the tube-based controller proposed above, since it requires online solution of $\mathbb{P}_N(\bar{x})$ rather than $\mathbb{P}_N(x)$.

Tube-based MPC is well suited to handle hard constraints via constraint tightening Michalska and Mayne (1993); Chisci et al. (2001); Mayne and Langson (2001) and many subsequent papers. It has more recently been used for stochastic MPC in Lorenzen et al. (2016); Mayne (2016).

Another difficulty that arises in stochastic MPC, as pointed out above,

is determination of suitable terminal conditions. It is impossible, for example, to obtain a terminal cost $V_f(\cdot)$ and local controller $\kappa_f(\cdot)$ such that $V_f(x^+) < V_f(x)$ for all $x \in X_f$, $x \neq 0$, and all $x^+ = f(x, \kappa_f(x), w)$. For this reason, Chatterjee and Lygeros (2015) propose that it should be possible to decrease $V_f(x)$ outside of the terminal constraint set $\mathbb{X}_f$, but that $V_f(x)$ should be permitted to increase by a bounded amount inside $\mathbb{X}_f$. The terminal ingredients, $V_f(\cdot)$ and $\mathbb{X}_f$, that we propose for robust MPC in Assumption 3.8 have this property a difference being that Chatterjee and Lygeros (2015) require $V_f(\cdot)$ to be a global (stochastic) Lyapunov function.

In most proposals, $\mathbb{P}_N(x)$ is a stochastic optimization problem, an area of study in which there have been recent significant advances discussed briefly above. Despite this, the computational requirements for solving stochastic optimization problems online seems excessive for process control applications. It is therefore desirable that as much computation as possible is done offline as proposed in Kouvaritakis et al. (2010); Lorenzen et al. (2016); Mayne (2016); and above. In these papers, offline optimization is employed to choose tightened constraints that, if satisfied by the nominal system, ensure that the original constraints are satisfied by the uncertain system. It also is desirable, in process control applications, to avoid computation of polytopic sets, as in Section 3.6.3, since they cannot be reliably computed for complex systems.

Robustness against unstructured uncertainty has been considered in Løvaas, Serón, and Goodwin (2008); Falugi and Mayne (2011).

3.9 Exercises

Exercise 3.1: Removing the outer min in a min-max problem

Show that $V_i^0 : \mathcal{X}_i \to \mathbb{R}$ and $\kappa_i : \mathcal{X}_i \to \mathbb{U}$ defined by

$$V_i^0(x) = \min_{u\in\mathbb{U}} \max_{w\in\mathbb{W}} \{\ell(x,u,w) + V_{i-1}^0(f(x,u,w)) \mid f(x,u,\mathbb{W}) \subset \mathcal{X}_{i-1}\}$$
$$\kappa_i(x) = \arg\min_{u\in\mathbb{U}} \max_{w\in\mathbb{W}} \{\ell(x,u,w) + V_{i-1}^0(f(x,u,w)) \mid f(x,u,\mathbb{W}) \subset \mathcal{X}_{i-1}\}$$
$$\mathcal{X}_i = \{x \in \mathbb{X} \mid \exists u \in \mathbb{U} \text{ such that } f(x,u,\mathbb{W}) \subset \mathcal{X}_{i-1}\}$$

satisfy

$$V_i^0(x) = \max_{w\in\mathbb{W}} \{\ell(x,\kappa_i(x),w) + V_{i-1}^0(f(x,\kappa_i(x),w))\}$$

Exercise 3.2: Maximizing a difference

Prove the claim used in the proof of Theorem 3.9 that

$$\max_w\{a(w)\} - \max_w\{b(w)\} \le \max_w\{a(w) - b(w)\}$$

Also show the following minimization version

$$\min_w\{a(w)\} - \min_w\{b(w)\} \ge \min_w\{a(w) - b(w)\}$$

Exercise 3.3: Equivalent constraints

Assuming that S is a polytope and, therefore, defined by linear inequalities, show that the constraint $x \in \{z\} \oplus S$ (on z for given x) may be expressed as $Bz \le b + Bx$, i.e., z must lie in a polytope. If S is symmetric ($x \in S$ implies $-x \in S$), show that $x \in \{z\} \oplus S$ is equivalent to $z \in \{x\} \oplus S$.

Exercise 3.4: Hausdorff distance between translated sets

Prove that the Hausdorff distance between two sets $\{x\} \oplus S$ and $\{y\} \oplus S$, where S is a compact subset of $\mathbb{R}^n$ and x and y are points in $\mathbb{R}^n$, is $|x - y|$.

Exercise 3.5: Exponential convergence of $X(i)$

Complement the proof of Proposition 3.12 by proving the sequence of sets $(X(i))_{0:\infty}$, $X(i) := \{\tilde{x}(i)\} \oplus S_K(\infty)$, converges exponentially fast to the set $S_K(\infty)$ as $i \to \infty$ if $\tilde{x}(i)$ converges exponentially fast to 0 as $i \to \infty$.

Exercise 3.6: Simulating a robust MPC controller

This exercise explores robust MPC for linear systems with an additive bounded disturbance

$$x^+ = Ax + Bu + w$$

The first task, using the tube-based controller described in Section 3.5.3 is to determine state and control constraint sets $\mathbb{Z}$ and $\mathbb{V}$ such that if the nominal system $z^+ = Az + Bv$ satisfies $z \in \mathbb{Z}$ and $v \in \mathbb{V}$, then the actual system $x^+ = Ax + Bu + w$ with $u = v + K(x - z)$ where K is such that $A + BK$ is strictly stable, satisfies the constraints $x \in \mathbb{X}$ and $u \in \mathbb{U}$.

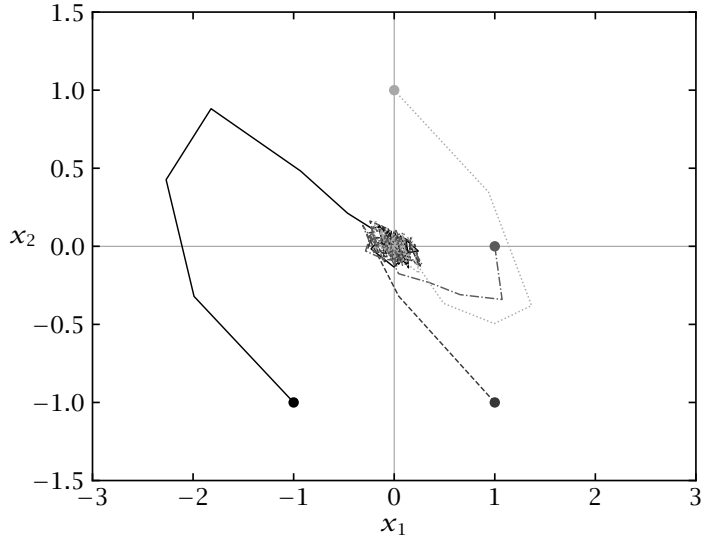


Figure 3.10: Closed-loop robust MPC state evolution with uniformly distributed $|w| \leq 0.1$ from four different x_0.

(a) To get started, consider the scalar system

$$x^+ = x + u + w$$

with constraint sets $\mathbb{X} = \{x \mid x \leq 2\}$, $\mathbb{U} = \{u \mid |u| \leq 1\}$, and $\mathbb{W} = \{w \mid |w| \leq 0.1\}$. Choose $K = -(1/2)$ so that $A_K = 1/2$. Determine $\mathbb{Z}$ and $\mathbb{V}$ so that if the nominal system $z^+ = z + v$ satisfies $z \in \mathbb{Z}$ and $v \in \mathbb{V}$, the uncertain system $x^+ = Ax + Bu + w$, $u = v + K(x - z)$ satisfies $x \in \mathbb{X}$, $u \in \mathbb{U}$.

(b) Repeat part (a) for the following uncertain system

$$x^+ = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u + w$$

with the constraint sets $\mathbb{X} = \{x \in \mathbb{R}^2 \mid x_1 \leq 2\}$, $\mathbb{U} = \{u \in \mathbb{R} \mid |u| \leq 1\}$ and $\mathbb{W} = [-0.1, 0.1]$. Choose $K = \begin{bmatrix} -0.4 & -1.2 \end{bmatrix}$.

(c) Determine a model predictive controller for the nominal system and constraint sets $\mathbb{Z}$ and $\mathbb{V}$ used in (b).

(d) Implement robust MPC for the uncertain system and simulate the closed-loop system for a few initial states and a few disturbance sequences for each initial state. The phase plot for initial states $[-1, -1]$, $[1, 1]$, $[1, 0]$, and $[0, 1]$ should resemble Figure 3.10.

Bibliography

D. A. Allan, C. N. Bates, M. J. Risbeck, and J. B. Rawlings. On the inherent robustness of optimal and suboptimal nonlinear MPC. *Sys. Cont. Let.*, 106: 68–78, August 2017.

D. Angeli, A. Casavola, G. Franze, and E. Mosca. An ellipsoidal off-line MPC scheme for uncertain polytopic discrete-time systems. *Automatica*, 44: 3113–3119, 2008.

J. P. Aubin. *Viability Theory*. Systems & Control: Foundations & Applications. Birkhauser, Boston, Basel, Berlin, 1991.

D. P. Bertsekas and I. B. Rhodes. Recursive state estimation for a set-membership description of uncertainty. *IEEE Trans. Auto. Cont.*, 16:117–128, 1971a.

D. P. Bertsekas and I. B. Rhodes. On the minimax reachability of target sets and target tubes. *Automatica*, 7(2):233–247, 1971b.

F. Blanchini. Set invariance in control. *Automatica*, 35:1747–1767, 1999.

F. Blanchini and S. Miani. *Set-Theoretic methods in Control*. Systems & Control: Foundations and Applications. Birkhäuser, 2008.

G. C. Calafiore and M. C. Campi. The scenario approach to robust control design. *IEEE Trans. Auto. Cont.*, 51(5):742–753, May 2006.

D. Chatterjee and J. Lygeros. On stability and performance of stochastic predictive control techniques. *IEEE Trans. Auto. Cont.*, 60(2):509–514, 2015.

D. Chatterjee, P. Hokayem, and J. Lygeros. Stochastic receding horizon control with bounded control inputs: a vector space approach. *IEEE Trans. Auto. Cont.*, 56(11):2704–2710, November 2011.

H. Chen and F. Allgöwer. A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability. *Automatica*, 34(10):1205–1217, 1998.

L. Chisci, J. A. Rossiter, and G. Zappa. Systems with persistent disturbances: predictive control with restricted constraints. *Automatica*, 37(7):1019–1028, 2001.

G. De Nicolao, L. Magni, and R. Scattolini. Robust predictive control of systems with uncertain impulse response. *Automatica*, 32(10):1475–1479, 1996.

M. Diehl, H. G. Bock, and E. Kostina. An approximation technique for robust nonlinear optimization. *Math. Prog.*, 107:213–230, 2006.

P. Falugi and D. Q. Mayne. Tube-based model predictive control for nonlinear systems with unstructured uncertainty. In *Proceedings of 50th IEEE Conference on Decision and Control*, pages 2656–2661, Orlando, Florida, USA, December 2011.

P. J. Goulart, E. C. Kerrigan, and J. M. Maciejowski. Optimization over state feedback policies for robust control with constraints. *Automatica*, 42:523–533, 2006.

P. J. Goulart, E. C. Kerrigan, and D. Ralph. Efficient robust optimization for robust control with constraints. *Math. Prog.*, 114(1):115–147, July 2008.

G. Grimm, M. J. Messina, S. E. Tuna, and A. R. Teel. Nominally robust model predictive control with state constraints. *IEEE Trans. Auto. Cont.*, 52(10): 1856–1870, October 2007.

G. A. Hicks and W. H. Ray. Approximation methods for optimal control synthesis. *Can. J. Chem. Eng.*, 49:522–528, August 1971.

S. Kameswaran and L. T. Biegler. Simultaneous dynamic optimization strategies: Recent advances and challenges. *Comput. Chem. Eng.*, 30:1560–1575, September 2006.

C. M. Kellett and A. R. Teel. Discrete-time asymptotic controllability implies smooth control-Lyapunov function. *Sys. Cont. Let.*, 52:349–359, 2004.

A. B. Khurzhanski and I. Valyi. *Ellipsoidal-valued dynamics for estimation and control*. Systems & Control: Foundations & Applications. Birkhauser, Boston, Basel, Berlin, 1997.

I. Kolmanovsky and E. G. Gilbert. Maximal output admissible sets for discrete-time systems with disturbance inputs. In *Proceedings of the American Control Conference*, Seattle, June 1995.

I. Kolmanovsky and E. G. Gilbert. Theory and computation of disturbance invariant sets for discrete-time linear systems. *Math. Probl. Eng.*, 4(4):317–367, 1998.

M. V. Kothare, V. Balakrishnan, and M. Morari. Robust constrained model predictive control using linear matrix inequalities. *Automatica*, 32(10):1361–1379, 1996.

K. I. Kouramas, S. V. Raković, E. C. Kerrigan, J. C. Allwright, and D. Q. Mayne. On the minimal robust positively invariant set for linear difference inclusions.

In *Proceedings of the 44th IEEE Conference on Decision and Control and European Control Conference ECC 2005*, pages 2296–2301, Sevilla, Spain, December 2005.

B. Kouvaritakis and M. Cannon. *Model predictive control*. Springer International Publishing, Switzerland, 2016.

B. Kouvaritakis, M. Cannon, S. V. Raković, and Q. Cheng. Explicit use of probabilistic distributions in linear predictive control. *Automatica*, 46:1719–1724, 2010.

W. Langson, I. Chryssochoos, S. V. Raković, and D. Q. Mayne. Robust model predictive control using tubes. *Automatica*, 40:125–133, January 2004.

M. Lazar, D. M. de la Peña, W. P. M. H. Heemels, and T. Alamo. On input-to-state stability of min-max nonlinear model predictive control. *Sys. Cont. Let.*, 57: 39–48, 2008.

J. H. Lee and Z. Yu. Worst-case formulations of model predictive control for systems with bounded parameters. *Automatica*, 33(5):763–781, 1997.

D. Limon, T. Álamo, and E. F. Camacho. Stability analysis of systems with bounded additive uncertainties based on invariant sets: stability and feasibility of MPC. In *Proceedings of the American Control Conference*, pages 364–369, Anchorage, Alaska, May 2002.

D. Limon, T. Alamo, D. M. Raimondo, D. M. de la Peña, J. M. Bravo, and E. F. Camacho. Input-to-state stability: an unifying framework for robust model predictive control. In L. Magni, D. M. Raimondo, and F. Allgöwer, editors, *International Workshop on Assessment and Future Directions of Nonlinear Model Predictive Control*, Pavia, Italy, September 2008.

M. Lorenzen, F. Dabbene, R. Tempo, and F. Allgöwer. Constraint-tightening and stability in stochastic model predictive control. *IEEE Trans. Auto. Cont.*, 62(7):3165–3177, 2016.

C. Løvaas, M. M. Serón, and G. C. Goodwin. Robust output feedback model predictive control for systems with unstructured uncertainty. *Automatica*, 44(8):1933–1943, August 2008.

L. Magni and R. Sepulchre. Stability margins of nonlinear receding-horizon control via inverse optimality. *Sys. Cont. Let.*, 32:241–245, 1997.

L. Magni, G. De Nicolao, R. Scattolini, and F. Allgöwer. Robust model predictive control for nonlinear discrete-time systems. *Int. J. Robust and Nonlinear Control*, 13:229–246, 2003.

D. Q. Mayne. Optimization in model based control. In *Proceedings of the IFAC Symposium Dynamics and Control of Chemical Reactors, Distillation Columns and Batch Processes*, pages 229–242, Helsingor, Denmark, June 1995.

D. Q. Mayne. Nonlinear model predictive control: An assessment. In J. C. Kantor, C. E. García, and B. Carnahan, editors, *Proceedings of Chemical Process Control – V*, pages 217–231. CACHE, AIChE, 1997.

D. Q. Mayne. Robust and stochastic model predictive control: Are we going in the right direction? *Annual Rev. Control*, 2016.

D. Q. Mayne and P. Falugi. Stabilizing conditions for model predictive control. *Int. J. Robust and Nonlinear Control*, 29(4):894–903, 2019.

D. Q. Mayne and W. Langson. Robustifying model predictive control of constrained linear systems. *Electron. Lett.*, 37(23):1422–1423, 2001.

D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert. Constrained model predictive control: Stability and optimality. *Automatica*, 36(6):789–814, 2000.

D. Q. Mayne, M. M. Serón, and S. V. Raković. Robust model predictive control of constrained linear systems with bounded disturbances. *Automatica*, 41 (2):219–224, February 2005.

D. Q. Mayne, E. C. Kerrigan, E. J. van Wyk, and P. Falugi. Tube based robust nonlinear model predictive control. *Int. J. Robust and Nonlinear Control*, 21 (11):1341–1353, 2011.

H. Michalska and D. Q. Mayne. Robust receding horizon control of constrained nonlinear systems. *IEEE Trans. Auto. Cont.*, 38(11):1623–1633, 1993.

Z. Nagy and R. Braatz. Open-loop and closed-loop robust optimal control of batch processes using distributional and worst-case analysis. *J. Proc. Cont.*, pages 411–422, 2004.

G. Pannocchia, J. B. Rawlings, and S. J. Wright. Conditions under which suboptimal nonlinear MPC is inherently robust. *Sys. Cont. Let.*, 60:747–755, 2011.

S. V. Raković. Invention of prediction structures and categorization of robust MPC syntheses. In *Proceedings of 4th IFAC Nonlinear Model Predictive Control Conference*, pages 245–273, Noordwijkerhout, NL, August 2012.

S. V. Raković, E. C. Kerrigan, K. I. Kouramas, and D. Q. Mayne. Approximation of the minimal robustly positively invariant set for discrete-time LTI systems with persistent state disturbances. In *Proceedings 42nd IEEE Conference on Decision and Control*, volume 4, pages 3917–3918, Maui, Hawaii, USA, December 2003.

S. V. Raković, E. C. Kerrigan, K. I. Kouramas, and D. Q. Mayne. Invariant approximations of the minimal robustly positively invariant sets. *IEEE Trans. Auto. Cont.*, 50(3):406–410, 2005a.

S. V. Raković, D. Q. Mayne, E. C. Kerrigan, and K. I. Kouramas. Optimized robust control invariant sets for constrained linear discrete-time systems. In *Proceedings of 16th IFAC World Congress on Automatic Control*, Prague, Czechoslavakia, 2005b.

S. V. Raković, B. Kouvaritakis, M. Cannon, C. Panos, and R. Findeisen. Parameterized tube model predictive control. *IEEE Trans. Auto. Cont.*, 57(11): 2746–2761, 2012.

R. T. Rockafellar and R. J.-B. Wets. *Variational Analysis*. Springer-Verlag, 1998.

J. A. Rossiter, B. Kouvaritakis, and M. J. Rice. A numerically robust state-space approach to stable-predictive control strategies. *Automatica*, 34(1):65–73, 1998.

L. O. Santos and L. T. Biegler. A tool to analyze robust stability for model predictive control. *J. Proc. Cont.*, 9:233–245, 1999.

P. O. M. Scokaert and D. Q. Mayne. Min–max feedback model predictive control for constrained linear systems. *IEEE Trans. Auto. Cont.*, 43(8):1136–1142, August 1998.

P. O. M. Scokaert, J. B. Rawlings, and E. S. Meadows. Discrete-time stability with perturbations: Application to model predictive control. *Automatica*, 33(3): 463–470, 1997.

A. R. Teel. Discrete time receding horizon control: is the stability robust. In Marcia S. de Queiroz, Michael Malisoff, and Peter Wolenski, editors, *Optimal control, stabilization and nonsmooth analysis*, volume 301 of *Lecture notes in control and information sciences*, pages 3–28. Springer, 2004.

R. Tempo, G. C. Calafiore, and F. Dabbene. *Randomized algorithms for analysis and control of uncertain systems: With applications*. Springer, second edition, 2013.

S. Yu, M. Reble, H. Chen, and F. Allgöwer. Inherent robustness properties of quasi-infinite horizon MPC. In *Proceedings of the 18th IFAC World Congress*, Milano, Italy, August, September 2011.

4
State Estimation

4.1 Introduction

We now turn to the general problem of estimating the state of a noisy dynamic system given noisy measurements. We assume that the system generating the measurements is given by

$$\begin{aligned} x^+ &= f(x, w) \\ y &= h(x) + v \end{aligned} \tag{4.1}$$

with the state $x \in \mathbb{X} \subseteq \mathbb{R}^n$, measurement $y \in \mathbb{Y} \subseteq \mathbb{R}^p$, process disturbance, $w \in \mathbb{W} \subseteq \mathbb{R}^g$, measurement disturbance, $v \in \mathbb{V} \subseteq \mathbb{R}^p$, and system initial state, $x(0) \in \mathbb{X}$. One of our main purposes is to provide a state estimate to the MPC regulator as part of a feedback control system, in which case the model changes to $x^+ = f(x, u, w)$ with both process disturbance w and control input u. But state estimation is a general technique that is often used in monitoring applications without any feedback control. So for simplicity of presentation, we start with state estimation as an independent subject and neglect the control input u as part of the system model as in (4.1).

Finally, in Section 4.5, we briefly treat the problem of combined MHE estimation and MPC regulation. In Chapter 5, we discuss the combined use of MHE and MPC in more detail.

4.2 Full Information Estimation

Of the estimators considered in this chapter, full information estimation will prove to have the best theoretical properties in terms of stability and optimality. Unfortunately, it will also prove to be computationally intractable except for the simplest cases, such as a linear system model. Its value therefore lies in clearly defining what is *desirable* in a

	System variable	Decision variable	Optimal decision
state	x	χ	$\hat{x}$
process disturbance	w	ω	$\hat{w}$
measured output	y	η	$\hat{y}$
measurement disturbance	v	ν	$\hat{v}$

Table 4.1: System and state estimator variables.

state estimator. One method for practical estimator design therefore is to come as close as possible to the properties of full information estimation (FIE) while maintaining a tractable online computation. This design philosophy leads directly to moving horizon estimation (MHE).

First we define some notation necessary to distinguish the system variables from the estimator variables. We have already introduced the system variables (x, w, y, v). In the estimator optimization problem, these have corresponding decision variables, which we denote $(\chi, \omega, \eta, \nu)$. The *optimal* decision variables are denoted $(\hat{x}, \hat{w}, \hat{y}, \hat{v})$, and these optimal decisions are the estimates provided by the state estimator. This notation is summarized in Table 4.1. Next we summarize the relationships between these variables

$$\begin{aligned} x^+ &= f(x, w) & y &= h(x) + v \\ \chi^+ &= f(\chi, \omega) & y &= h(\chi) + \nu \\ \hat{x}^+ &= f(\hat{x}, \hat{w}) & y &= h(\hat{x}) + \hat{v} \end{aligned}$$

Notice that it is always the system measurement y that appears in the second column of equations. We also can define the decision variable output, $\eta = h(\chi)$, but notice that ν measures the fitting error, $\nu = y - h(\chi)$, and we must use the system measurement y and not η in this relationship. Therefore, we do not satisfy a relationship like $\eta = h(\chi) + \nu$, but rather

$$\begin{aligned} y &= h(\chi) + \nu & \eta &= h(\chi) \\ y &= h(\hat{x}) + \hat{v} & \hat{y} &= h(\hat{x}) \end{aligned}$$

We begin with a reasonably general definition of the full information estimator that produces an estimator that is *stable*, which we also shall

define subsequently. The full information objective function is

$$V_T(\chi(0), \boldsymbol{\omega}) = \ell_x(\chi(0) - \overline{x}_0) + \sum_{i=0}^{T-1} \ell(\omega(i), \nu(i)) \tag{4.2}$$

subject to

$$\chi^+ = f(\chi, \omega) \qquad y = h(\chi) + \nu$$

in which T is the current time, $y(i)$ is the measurement at time i, and $\overline{x}_0$ is the prior estimate of the initial state.[1] Occasionally we shall consider input disturbances to an explicitly given *nominal* input. If we denote this nominal input trajectory as $\overline{\mathbf{w}}$, then we adjust the model constraint to $\chi^+ = f(\chi, \overline{w} + \omega)$, so that ω measures the difference from the nominal model's input. We recover the standard problem by setting $\overline{\mathbf{w}} = 0$. Because $\nu = y - h(\chi)$ is the error in fitting the measurement y, $\ell(\omega, \nu)$ penalizes the model disturbance and the fitting error. These are the two error sources we reconcile in all state estimation problems.

The full information estimator is then defined as the solution to

$$\mathbb{P}_T(\overline{x}_0, \overline{\mathbf{w}}_{0:k-1}, \mathbf{y}_{0:k-1}) \coloneqq \min_{\chi(0), \boldsymbol{\omega}} V_T(\chi(0), \boldsymbol{\omega}) \tag{4.3}$$

and we use the notation $\mathbb{P}_T(\overline{x}_0, \mathbf{y}_{0:k-1})$ for the usual case when the nominal input is $\overline{\mathbf{w}} = 0$. The solution to the optimization exists for all $T \in \mathbb{I}_{\geq 0}$ because $V_T(\cdot)$ is continuous, due to the continuity of $f(\cdot)$ and $h(\cdot)$, and because $V_T(\cdot)$ is an unbounded function of its arguments, as will be clear after stage costs $\ell_x(\cdot)$ and $\ell(\cdot)$ are defined. We denote the solution as $\hat{x}(0|T)$, $\hat{w}(i|T), 0 \leq i \leq T-1$, $T \geq 1$, and the optimal cost as V_T^0. We also use $\hat{x}(T) \coloneqq \hat{x}(T|T)$ to simplify the notation.

We require a definition of state estimation general enough to include this optimization approach. Attempting to express the state estimate as a finite dimensional dynamical system, as we do with the Kalman filter for linear systems, is not sufficient here. Instead we consider the state estimate at any time $k \in \mathbb{I}_{\geq 0}$ to be a *function* of the prior $\overline{x}_0$, nominal input (if nonzero), $\overline{\mathbf{w}}_{0:T-1}$, and the measurement $\mathbf{y}_{0:T-1}$.

Definition 4.1 (State Estimator)**.** A *state estimator* is a sequence of functions $(\Psi_T)_{T \geq 0}$ defined $\Psi_T : \mathbb{X} \times \mathbb{W}^T \times \mathbb{Y}^T \to \mathbb{X}$ for all $T \in \mathbb{I}_{\geq 0}$, and the

[1] Notice that we have dropped the final measurement $y(T)$ compared to the problem considered in Chapter 1 to formulate the prediction form rather than the filtering form of the state estimation problem. So what we denote here as $\hat{x}(T|T)$ would be $\hat{x}^-(T)$ in the notation of Chapter 1. This change is purely for notational convenience, and all results developed in this chapter also can be expressed in the filtering form of MHE.

state estimate at time T is denoted

$$\hat{x}(T) = \Psi_T(\overline{x}_0, \overline{\mathbf{w}}_{0:T-1}, \mathbf{y}_{0:T-1})$$

If the nominal input sequence is $\overline{\mathbf{w}} = 0$, as is usually the case, then we drop the second argument and write simply

$$\hat{x}(T) = \Psi_T(\overline{x}_0, \mathbf{y}_{0:T-1})$$

In the full information estimator, the function $\Psi(\cdot)$ denotes the final element of the state trajectory in the solution to (4.3). One important characteristic of optimization-based estimation worth bearing in mind as we progress is that $\hat{x}(T) = \Psi_T(\overline{x}_0, \mathbf{y}_{0:T-1})$ does *not* imply that $\hat{x}(T+1) = \Psi_1(\hat{x}(T), y_T)$, even though $\mathbf{y}_{0:T} := (\mathbf{y}_{0:T-1}, y_T)$. In (nonlinear) full information estimation, we have no convenient means to move from $\hat{x}(T)$ to $\hat{x}(T+1)$, and must instead recompute the entire optimal trajectory with $\Psi_{T+1}(\overline{x}_0, \mathbf{y}_{0:T})$. As we shall see subsequently, this confers some desirable properties on the estimator, but renders its online computation intractable since the size of the optimization problem increases with time.

Next we require a definition of robust stability suitable for state estimation in this general form. The standard attempt[2] would be to use the following type of bound in the definition of robust stability

$$|x(k) - \hat{x}(k)| \leq \alpha_x(|x(0) - \overline{x}_0|, k) + \gamma_w(\|\mathbf{w}\|_{0:k-1}) + \gamma_v(\|\mathbf{v}\|_{0:k-1}) \quad (4.4)$$

for all $k \in \mathbb{I}_{\geq 0}$ with $\alpha_x(\cdot) \in \mathcal{KL}$ and $\gamma_w(\cdot), \gamma_v(\cdot) \in \mathcal{K}$. But, for the general class of estimators under consideration here, an inequality of this type does not ensure that the estimate error converges to zero when the disturbances converge to zero. To ensure this desirable property we *strengthen* the definition of estimator stability to the following.

Definition 4.2 (Robustly globally asymptotically stable estimation)**.** A state estimator $(\Psi_k)_{k\geq 0}$ is robustly globally asymptotically stable (RGAS) if there exist $\mathcal{KL}$-functions $\alpha_x, \alpha_w, \alpha_v$ such that

$$|x(k) - \hat{x}(k)| \leq \alpha_x(|x(0) - \overline{x}_0|, k) \oplus \max_{j \in \mathbb{I}_{0:k-1}} \alpha_w(|w(j)|, k-j-1) \oplus \max_{j \in \mathbb{I}_{0:k-1}} \alpha_v(|v(j)|, k-j-1) \quad (4.5)$$

for all $k \in \mathbb{I}_{\geq 0}$, $x(0), \overline{x}_0 \in \mathbb{X}$, and $w \in \mathbb{W}$, $v \in \mathbb{V}$.

[2] See the previous printings of this chapter, for example.

We have chosen the convolution maximization form for the stability definition, where the notation $a \oplus b$ denotes $\max(a, b)$ for $a, b \in \mathbb{R}$. We choose to maximize on time index j rather than sum on j since we do not know a priori that the $\mathcal{KL}$-functions α_w, α_v decrease sufficiently quickly to ensure that the sums converge as $k \to \infty$.

We can then readily establish the following convergence result (Allan and Rawlings, 2020, Proposition 3.11)

Proposition 4.3 (RGAS plus convergent disturbances imply convergent estimates)**.** *If an estimator is RGAS and* $((w(k), v(k)))_{k\geq 0}$ *converges to zero, then the estimate error converges to zero.*

The proof of this proposition is discussed in Exercise 4.13.

Example 4.4: The Kalman filter of a linear system is RGAS

Show that the steady-state Kalman filter (predictor) of a detectable, stabilizable linear system

$$x^+ = Ax + Gw \qquad y = Cx + v$$

is RGAS and satisfies both (4.5) as well as (4.4).

Solution

For (A, C) detectable and (A, G) stabilizable, the steady-state Kalman predictor is nominally exponentially stable as discussed in Exercise 4.17. The steady-state estimator takes the form

$$\hat{x}^+ = A\hat{x} + L(y - C\hat{x}) \qquad \hat{x}(0) = \overline{x}_0$$

where L satisfies a steady-state Riccati equation and $A_L := (A - LC)$ is a stable matrix. Subtracting the estimator from the system gives

$$(x - \hat{x})^+ = A_L(x - \hat{x}) + Gw - Lv$$

Solving this linear system gives

$$x(k) - \hat{x}(k) = A_L^k(x(0) - \overline{x}_0) + \sum_{j=0}^{k-1} A_L^{k-j-1}(Gw(j) - Lv(j))$$

Since A_L is stable, we have the bound (Horn and Johnson, 1985, p.299) $|A_L^i| \leq c\lambda^i$ in which $\max |\operatorname{eig}(A_L)| < \lambda < 1$. Taking norms and using

this bound gives for all $k \geq 0$

$$|x(k) - \hat{x}(k)| \leq c\lambda^k |x(0) - \overline{x}_0| + c \sum_{j=0}^{k-1} \left(|G| \, |w(j)| + |L| \, |v(j)| \right) \lambda^{k-j-1} \quad (4.6)$$

Taking the largest disturbance terms outside and performing the sum then gives

$$|x(k) - \hat{x}(k)| \leq c\lambda^k |x(0) - \overline{x}_0| + \frac{c}{1-\lambda} \left[|G| \, \|\mathbf{w}\|_{0:k-1} + |L| \, \|\mathbf{v}\|_{0:k-1} \right]$$

So we have that (4.4) is satisfied after defining $\alpha_x(r,k) := cr\lambda^k$, which is an exponential $\mathcal{KL}$-function, and $\gamma_w(r) := (c|G|/(1-\lambda))\, r$ and $\gamma_v(r) := (c|L|/(1-\lambda))\, r$, which are linear $\mathcal{K}$-functions.

To obtain the stronger convolution maximization form, first note that for $0 \leq \lambda < 1$ and $z(j) > 0$

$$\sum_{j=0}^{k-1} z(j)\lambda^{k-j-1} = \sum_{j=0}^{k-1} \left(z(j)\lambda^{(k-j-1)/2} \right) \lambda^{(k-j-1)/2} \leq \frac{1}{1-\sqrt{\lambda}} \max_{j \in \mathbb{I}_{0:k-1}} z(j)\lambda^{(k-j-1)/2}$$

Using this result in (4.6) and letting $\eta := \sqrt{\lambda} > \lambda$ so that $0 \leq \eta < 1$, we have that

$$|x(k) - \hat{x}(k)| \leq c\eta^k |x(0) - \overline{x}_0| + c|G|/(1-\eta) \max_{j \in \mathbb{I}_{0:k-1}} |w(j)| \, \eta^{k-j-1} + c|L|/(1-\eta) \max_{j \in \mathbb{I}_{0:k-1}} |v(j)| \, \eta^{k-j-1}$$

Finally, using Exercise 4.6(d), we convert the sum to maximization and satisfy (4.5) with

$$\alpha_x(r,k) := 3cr\,\eta^k \qquad \alpha_w(r,k) := 3c|G|/(1-\eta) r\,\eta^k$$
$$\alpha_v(r,k) := 3c|L|/(1-\eta) r\,\eta^k$$

and the steady-state Kalman predictor is RGAS, and the $\mathcal{KL}$-functions $\alpha_x, \alpha_w, \alpha_v$ are of exponential form. □

The next order of business is to decide what class of systems to consider if the goal is to obtain a stable state estimator. A standard choice

in most nonlinear estimation literature is to assume system observability. The drawback with this choice is that it is overly restrictive, even for linear systems. As discussed in Chapter 1, for linear systems we require only detectability for stable estimation (Exercise 1.33). We therefore start instead with an assumption of detectability that is appropriate for nonlinear systems, called incremental input/output-to-state stability (i-IOSS) Sontag and Wang (1997). This definition is an incremental property in which we compare two trajectories starting at different initial conditions $x_1, x_2 \in \mathbb{X}$ and experiencing different disturbance sequences, $\mathbf{w}_1, \mathbf{w}_2 \in \mathbb{W}^\infty$. We use $x(k; x, \mathbf{w})$ to denote the solution to (4.1) for initial condition x and disturbance sequence $\mathbf{w}$. To compress the notation, we define the incremental differences in state $\Delta x(k) := x(k, x_1, \mathbf{w}_1) - x(k, x_2, \mathbf{w}_2)$, input difference $\Delta \mathbf{w}(k) := w_1(k) - w_2(k)$, and output $\Delta y(k) := h(x(k, x_1, \mathbf{w}_1)) - h(x(k, x_2, \mathbf{w}_2))$. For convenience, we choose a detectability assumption that is similar in structure to our choice of stability definition.

Definition 4.5 (i-IOSS)**.** The system $x^+ = f(x, w), y = h(x)$ is *incrementally input/output-to-state stable* (i-IOSS) if there exist functions $\beta_x(\cdot), \beta_w(\cdot), \beta_v(\cdot) \in \mathcal{KL}$ such that

$$|\Delta x(k)| \le \beta_x(|\Delta x(0)|, k) \oplus \max_{j \in \mathbb{I}_{0:k-1}} \beta_w(|\Delta w(j)|, k-j-1) \oplus \max_{j \in \mathbb{I}_{0:k-1}} \beta_v(|\Delta y(j)|, k-j-1) \quad (4.7)$$

for all $k \in \mathbb{I}_{\ge 0}$, all initial states $x_1, x_2 \in \mathbb{X}$, and all disturbance sequences $\mathbf{w}_1, \mathbf{w}_2 \in \mathbb{W}^\infty$.

In previous versions of the text we used the more traditional definition of i-IOSS that has a single $\mathcal{KL}$-function β_x and two asymptotic gain $\mathcal{K}$-functions γ_w, γ_v and the following bound in place of (4.7)

$$|\Delta x(k)| \le \beta_x(|\Delta x(0)|, k) + \gamma_w(|\Delta \mathbf{w}|) + \gamma_v(|\Delta \mathbf{y}|) \quad (4.8)$$

It is straightforward to show that the bound in (4.7) implies the bound in (4.8). Although it is not straightforward, Allan, Rawlings, and Teel (2020, Proposition 4) show that the bound in (4.8) also implies the one in (4.7). Therefore the choice of the form of the bound in Definition 4.5 is indeed one of convenience, as we shall see in the proof of the next proposition.

System properties such as i-IOSS are generically difficult to check for a given nonlinear application of interest. It is therefore important

to ask whether the assumption is overly restrictive. We show that it is not overly restrictive if the goal is to build an RGAS estimator for the system (Allan et al., 2020, Proposition 5).

Proposition 4.6 (RGAS estimator implies i-IOSS)**.** *If a system admits an RGAS estimator* $(\Psi_k)_{k\geq 0}$*, then the system is i-IOSS.*

Proof. Consider two initial conditions denoted $x_{1,0}$ and $x_{2,0}$, two input sequences $\mathbf{w}_1$ and $\mathbf{w}_2$ generating from (4.1) two corresponding state trajectories $\mathbf{x}_1$ and $\mathbf{x}_2$. Now consider input and output disturbance sequences $\widetilde{w}_1(j) = w_1(j) - w_2(j)$, and $v_1(j) = h(x_2(j)) - h(x_1(j))$ for $j \in \mathbb{I}_{\geq 0}$. Let the system generating the measurements for state estimation be $x(k) = x(k; x_{1,0}, \mathbf{w}_2 + \widetilde{\mathbf{w}}_1), y = h(x) + v_1$. Note that the system generating the measurements has initial condition $x_{1,0}$, *nominal* input, $\overline{\mathbf{w}} = \mathbf{w}_2$, but *disturbed* or actual input $\mathbf{w}_1$ since $\mathbf{w}_2 + \widetilde{\mathbf{w}}_1 = \mathbf{w}_1$; so we have that $x(k) = x_1(k)$ for $k \in \mathbb{I}_{\geq 0}$. The output measurements are exactly $h(\mathbf{x}_2)$ because of the output disturbance. The state estimator is therefore based on nominal input $\overline{\mathbf{w}} = \mathbf{w}_2$ and output measurement $h(\mathbf{x}_2)$. Let the state estimator then have $x_{2,0}$ as its prior. The information given to the estimator is then consistent, and it produces $\hat{x}(k) = \Psi_k(x_{2,0}, \mathbf{w}_2, h(\mathbf{x}_2)) = x_2(k)$ for $k \in \mathbb{I}_{\geq 0}$. If the estimator is RGAS, then (4.5) gives for this system and estimator

$$
\begin{aligned}
|x_1(k) - x_2(k)| \leq\ & \alpha_x(|x_{1,0} - x_{2,0}|, k) \\
& \oplus \max_{j \in \mathbb{I}_{0:k-1}} \alpha_w(|\widetilde{w}_1(j)|, k-j-1) \oplus \max_{j \in \mathbb{I}_{0:k-1}} \alpha_v(|v_1(j)|, k-j-1)
\end{aligned}
$$

and substituting the defined disturbances

$$
\begin{aligned}
|x_1(k) - x_2(k)| \leq\ & \alpha_x(|x_{1,0} - x_{2,0}|, k) \\
& \oplus \max_{j \in \mathbb{I}_{0:k-1}} \alpha_w(|w_1(j) - w_2(j)|, k-j-1) \\
& \oplus \max_{j \in \mathbb{I}_{0:k-1}} \alpha_v(|h(x_1(j)) - h(x_2(j))|, k-j-1)
\end{aligned}
$$

for all $k \in \mathbb{I}_{\geq 0}$. Note that since $x_{1,0}, x_{2,0}, \mathbf{w}_1, \mathbf{w}_2$ are arbitrary, the system is i-IOSS. ∎

Sontag and Wang (1997, Proposition 23) derived an earlier result of this style but restricted to estimators in the class of observers evolving in the same state space as x with output injection.

We shall find an i-IOSS Lyapunov function useful to establish the estimator's stability. We have the following definition.

Definition 4.7 (i-IOSS Lyapunov function)**.** A function $\Lambda : \mathbb{X} \times \mathbb{X} \to \mathbb{R}_{\geq 0}$ is an *i-IOSS Lyapunov function* for the system (4.1) if there exist $\mathcal{K}_\infty$-functions $\alpha_1, \alpha_2, \alpha_3$ and $\mathcal{K}$-functions σ_w, σ_v such that

$$\alpha_1(|x_1 - x_2|) \leq \Lambda(x_1, x_2) \leq \alpha_2(|x_1 - x_2|) \tag{4.9}$$

$$\begin{aligned}\Lambda(f(x_1, w_1), f(x_2, w_2)) \leq{}& \Lambda(x_1, x_2) - \alpha_3(|x_1 - x_2|) \\ &+ \sigma_w(|w_1 - w_2|) \\ &+ \sigma_v(|h(x_1) - h(x_2)|)\end{aligned} \tag{4.10}$$

for all $x_1, x_2 \in \mathbb{X}$ and $w_1, w_2 \in \mathbb{W}$.

The following converse theorem establishes that a system is i-IOSS if and only if the system admits an i-IOSS Lyapunov function.

Theorem 4.8 (i-IOSS and Lyapunov function equivalence)**.** *A system* (4.1) *is i-IOSS if and only if it admits an i-IOSS Lyapunov function.*

The proof that an i-IOSS Lyapunov function implies i-IOSS is provided in (Allan and Rawlings, 2019, Proposition 5, Remark 6). The converse implication is more involved and is provided in (Allan et al., 2020, Theorem 8).

The last element that we require is system stabilizability. Most of the literature on FIE and MHE has not stressed this requirement and sometimes tacitly assumes an unnecessarily strong form of it by expressing the system as $x^+ = f(x) + w$, but we obtain sharper conclusions by addressing it. We take the following definition of stabilizability.

Definition 4.9 (Incremental Stabilizability with respect to stage cost $L(\cdot)$)**.** A nonlinear system $x^+ = f(x, u)$ is said to be *incrementally stabilizable with respect to stage cost* $L(\cdot)$ if there exists $\mathcal{K}$-function $\overline{\alpha}$ such that for every two initial conditions $x_1, x_2 \in \mathbb{X}$ and control sequence $\mathbf{w}_1 \in \mathbb{W}^\infty$, another control sequence $\mathbf{w}_2 \in \mathbb{W}^\infty$ exists such that

$$\sum_{k=0}^{\infty} L(x_1(k), x_2(k), w_1(k), w_2(k)) \leq \overline{\alpha}(|x_1 - x_2|)$$

With all of the basic concepts introduced, we can state our working assumptions for the full-information state estimation problem.

Assumption 4.10 (Continuity)**.** The functions $f(\cdot)$, $h(\cdot)$, $\ell_x(\cdot)$, and $\ell(\cdot)$ are continuous, $\ell_x(0) = 0$, and $\ell(0, 0) = 0$. The sets $\mathbb{X}$ and $\mathbb{W}$ are closed.

Assumption 4.11 (Positive-definite stage cost)**.** The stage cost $\ell(\cdot)$ satisfies

$$\sigma_w(|\omega|) + \sigma_v(|\nu|) \le \ell(\omega, \nu) \le \overline{\sigma}_w(|\omega|) + \overline{\sigma}_v(|\nu|)$$

for all $\omega \in \mathbb{W}, \nu \in \mathbb{V}$ for some $\mathcal{K}_\infty$-functions $\overline{\sigma}_w$ and $\overline{\sigma}_v$, and the $\mathcal{K}_\infty$-functions σ_w and σ_v come from (4.10) of the i-IOSS Lyapunov function. Furthermore, we have that

$$\sigma_x(|\chi - \overline{x}_0|) \le \ell_x(\chi - \overline{x}_0) \le \overline{\sigma}_x(|\chi - \overline{x}_0|)$$

for all $\chi, \overline{x}_0 \in \mathbb{X}$ for some $\mathcal{K}_\infty$-functions σ_x and $\overline{\sigma}_x$.

Assumption 4.12 (Stabilizability)**.** The system (4.1) is stabilizable with respect to the stage cost $L(x_1, x_2, w_1, w_2) := \ell(w_2 - w_1, h(x_1) - h(x_2))$.

Assumption 4.13 (Detectability)**.** The system (4.1) is i-IOSS.

Remark.

(a) Assumptions 4.10 and 4.11 guarantee that a solution to (4.3) exists for all finite $T \ge 0$ (Rawlings and Ji, 2012).

(b) From Theorem 4.8, Assumption 4.13 implies the existence of an i-IOSS Lyapunov function satisfying (4.9)–(4.10).

(c) Notice that the stage cost is chosen to be compatible with the system's detectability properties in Assumption 4.11.

(d) A similar case can be made in regulation that one must choose the regulator's stage cost to be compatible with the system's stabilizability properties. We did not emphasize this issue in Chapter 2, and instead allowed the stage cost to affect the MPC regulator's feasibility set $\mathcal{X}_N$. The consequence of choosing the stage cost inappropriately in the zero-state MPC regulator would therefore be a catastrophic reduction in the size of the feasibility set, with the worst case being $\mathcal{X}_N = \{0\}$.

(e) If we strengthen the detectability property to exponential detectability, then the stage cost restriction is relaxed. For example, any positive definite quadratic stage cost is compatible with exponential detectability as discussed in Exercise 4.12.

(f) The stage cost also is chosen to be compatible with the system's stabilizability properties in Assumption 4.12.

(g) It is not strictly necessary to assume the upper bounds in Assumption 4.11. From Assumption 4.10, $\ell(\cdot)$ and $\ell_x(\cdot)$ are continuous and

therefore have upper-bounding $\mathcal{K}_\infty$-functions (Rawlings and Risbeck, 2015, Proposition 19). But it is helpful to name these upper-bounding functions here.

4.2.1 Nominal Estimator Stability

In this section we set $w = 0$, $v = 0$, and estimator stability in Definition 4.2 reduces to existence of a $\mathcal{KL}$-function α_x such that for all $k \in \mathbb{I}_{\geq 0}$, $x(0), \overline{x}_0 \in \mathbb{X}$

$$|x(k) - \hat{x}(k)| \leq \alpha_x(|x(0) - \overline{x}_0|, k) \tag{4.11}$$

We refer to this property as "nominal" stability. Since the main purpose of state estimation is to deal with nonzero disturbances w, v, one may wonder why we should bother analyzing nominal stability in the first place. The motivation is to illustrate in this simple setting a new analysis tool, termed a Q-function.[3] This function takes the place of a Lyapunov function in our estimator stability analysis. It has the characteristics that we expect of a Lyapunov function, but it has some additional features: two time arguments instead of one, and an extra inequality involving the estimator's prior and the system's initial condition. The extra complexity seems to be required by the fact that, unlike the zero-state regulator, the evolution of the estimate error cannot be expressed as a simple dynamical system. We introduce the Q-function in this setting where the stability arguments can be presented in their entirety. We closely follow the development in Allan and Rawlings (2019) in this section. Then the same tools introduced in this section can be used in the next section to treat bounded disturbances w, v, which is the case of most interest. The arguments for that case become significantly longer and more detailed, so we will have to be content to state the main results and point to the appropriate references for the proofs. The results in that section mainly follow Allan and Rawlings (2020); Allan (2020).

First we consider the estimation problem (4.3) on the infinite horizon, i.e., in the limit $T \to \infty$. For the zero disturbance case, the choice $\chi(0) = x(0)$, and $\omega(j) = 0$ for all $j \in \mathbb{I}_{0:T-1}$ is feasible and gives cost $\ell_x(x(0) - \overline{x}_0)$, which is independent of T. So we have the following upper bound for the optimal FIE cost for all $T \in \mathbb{I}_{\geq 0}$

$$V_T^0(x(0)) \leq \ell_x(x(0) - \overline{x}_0)$$

[3] Allan and Rawlings (2019, 2020) introduce the name Q-function to commemorate the seminal contributions of David Q. Mayne to control and estimation theory.

and from (Keerthi and Gilbert, 1985, Theorem 2), a solution to the infinite horizon problem exists. If we consider the solution of a k-stage problem, optimality of the infinite horizon problem gives

$$V_\infty^0 \le V_k^0 + \min_{\boldsymbol{\omega}_{k:\infty}} \sum_{i=k}^{\infty} \ell(\omega(i), \nu(i)) \tag{4.12}$$

subject to

$$\chi^+ = f(\chi, \omega) \qquad \mathcal{Y} = h(\chi) + \nu \qquad \chi(k) = \hat{x}(k|k)$$

The stabilizability assumption then provides an upper bound for the minimization in (4.12) as follows. In the sum, the system generating the data starts at $x(k)$ and experiences zero input disturbance. The estimator starts at $\hat{x}(k|k)$ and optimizes the input sequence to fit the data. The definition of stabilizability and Assumption 4.12 gives

$$\sum_{i=k}^{\infty} L(x(i), \chi(i), 0, \omega(i)) \le \overline{\alpha}(|x(k) - \hat{x}(k|k)|)$$

for $L(x(i), \chi(i), 0, \omega(i)) = \ell(\omega(i), \mathcal{Y}(i) - h(\chi(i))) = \ell(\omega(i), \nu(i))$. Using this bound in (4.12) then gives

$$V_\infty^0 \le V_k^0 + \overline{\alpha}(|x(k) - \hat{x}(k|k)|)$$

In previous versions of FIE analysis, we made use of the fact that the optimal solution of the estimation problem at time $k+1$ gives feasible, but possibly suboptimal decision variables at time k. That argument leads to the inequality

$$V_k^0 \le V_{k+1}^0 - \ell(\hat{w}(k|k+1), \hat{v}(k|k+1)) \tag{4.13}$$

which shows that the sequence $\left(V_k^0\right)_{k\ge 0}$ is nondecreasing. Since it is bounded above by $\ell_x(x(0) - \overline{x}_0)$, it converges, and that implies that $\ell(\hat{w}(k|k+1), \hat{v}(k|k+1)) \to 0$ as $k \to \infty$. The problem with this approach is that it compares two different trajectories, and does not generalize well to the bounded disturbance case where the infinite horizon problem is not bounded above. So we change course from previous analysis and consider instead a single trajectory, but different times within the trajectory by introducing partial sums

$$V^0(j|k) = \ell_x(\hat{x}(0|k) - \overline{x}_0) + \sum_{i=0}^{j-1} \ell(\hat{x}(i|k), \hat{v}(i|k))$$

with $j \le k \in \mathbb{I}_{\ge 0}$. Changing j rather than k is then straightforward

$$V^0(j|k) = V^0(j+1|k) - \ell(\hat{w}(j|k), \hat{v}(j|k)) \tag{4.14}$$

Note that we have an equality here, not even an inequality as arises in (4.13) when comparing optimal costs at k and $k+1$.

The Q-function. We now modify the optimal cost of the estimation problem to create something that operates similarly to a Lyapunov function in this context. First we flip the function so that it decreases rather than increases with k. We define $Y(j|k)$

$$Y(j|k) := V_\infty^0 - V^0(j \mid k)$$

for $j \le k \in \mathbb{I}_{\ge 0}$. We know that $V^0(j|k) \le V^0(k|k) \le V_\infty^0$ for all $j \le k$ because the objective function is a sum of positive stage costs. We can also deduce that

$$V_\infty^0 \le V^0(j|k) + \overline{\alpha}(\,|x(j) - \hat{x}(j|k)|\,)$$

for all $j \le k$ using the same argument as we used above with $j = k$. These give the corresponding bounds for the flipped function $Y(\cdot)$

$$0 \le Y(j|k) \le \overline{\alpha}(\,|x(j) - \hat{x}(j|k)|\,)$$

for all $j \le k$. Substituting (4.14) into the definition of $Y(\cdot)$ then gives a cost decrease equality

$$Y(j+1|k) = Y(j|k) - \ell(\hat{w}(j|k), \hat{v}(j|k))$$

for $j \le k-1$.

The last step is to use the i-IOSS Lyapunov function implied by the detectability Assumption 4.13. Applying (4.9)–(4.10) to the values $x(j)$ and $\hat{x}(j|k)$ gives

$$\begin{aligned}\alpha_1(\,|x(j) - \hat{x}(j|k)|\,) \le \Lambda(x(j), \hat{x}(j|k)) &\le \alpha_2(\,|x(j) - \hat{x}(j|k)|\,)\\ \Lambda(x(j+1), \hat{x}(j+1|k)) &\le \Lambda(x(j), \hat{x}(j|k)) - \alpha_3(\,|x(j) - \hat{x}(j|k)|\,)\\ &\quad + \sigma_w(\,|\hat{w}(j|k)|\,) + \sigma_v(\,|\hat{v}(j|k)|\,)\end{aligned}$$

for all $j \le k$. We define the Q-function as the sum of $\Lambda(\cdot)$ and $Y(\cdot)$

$$Q(j|k) := Y(j|k) + \Lambda(x(j), \hat{x}(j|k))$$

Substituting the bounds on $Y(\cdot)$ and $\Lambda(\cdot)$ into this definition gives positive upper and lower bounds on $Q(\cdot)$

$$\alpha_1(\,|x(j) - \hat{x}(j|k)|\,) \le Q(j|k) \le \overline{\alpha}_2(\,|x(j) - \hat{x}(j|k)|\,)$$

with $\mathcal{K}_\infty$-function $\overline{\alpha}_2 := \overline{\alpha} + \alpha_2$, and the following descent condition

$$\begin{aligned} Q(j+1|k) &\leq Q(j|k) - \alpha_3(\left|x(j) - \hat{x}(j|k)\right|) \\ &\quad + \sigma_w(\left|\hat{w}(j|k)\right|) + \sigma_v(\left|\hat{v}(j|k)\right|) - \ell(\hat{w}(j|k), \hat{v}(j|k)) \\ &\leq Q(j|k) - \alpha_3(\left|x(j) - \hat{x}(j|k)\right|) \end{aligned}$$

where we used Assumption 4.11 to achieve the final inequality. Note that choosing an appropriate stage cost in estimation is what allows the decrease in cost due to optimization to overcome the effect of the positive supply rate in the i-IOSS Lyapunov function.

The inequalities established for $Q(j|k)$ make it well suited for stability analysis except for one remaining issue, which we resolve next. The upper bound at $j = 0$ gives

$$Q(0|k) \leq \overline{\alpha}_2(|x(0) - \hat{x}(0|k)|) \tag{4.15}$$

But to achieve nominal stability in (4.11) we require a bound that depends on the distance of the initial state from the *prior* $\overline{x}_0$, not the *estimated* initial state at time k, $\hat{x}(0|k)$. So we create that bound next. Note from Assumption 4.11 and the previous discussion of the infinite horizon problem

$$\sigma_x(|\hat{x}(0|k) - \overline{x}_0|) \leq \ell_x(\hat{x}(0|k) - \overline{x}_0) \leq V_k^0 \leq V_\infty^0 \leq \ell_x(x(0) - \overline{x}_0)$$

for all $k \in \mathbb{I}_{\geq 0}$, and $x(0), \overline{x}_0 \in \mathbb{X}$. From Assumption 4.11, we also have the upper bound

$$\ell_x(x(0) - \overline{x}_0) \leq \overline{\sigma}_x(|x(0) - \overline{x}_0|)$$

Combining these gives

$$|\hat{x}(0|k) - \overline{x}_0| \leq \sigma_x^{-1}\left(\overline{\sigma}_x(|x(0) - \overline{x}_0|)\right)$$

Using the triangle inequality and this result gives

$$|\hat{x}(0|k) - x(0)| \leq |\hat{x}(0|k) - \overline{x}_0| + |x(0) - \overline{x}_0| \leq \overline{\sigma}_0(|x(0) - \overline{x}_0|)$$

with $\overline{\sigma}_0(\cdot) := (\cdot) + \sigma_x^{-1}(\overline{\sigma}_x(\cdot))$. Substituting this result in (4.15) then gives the desired bound

$$Q(0|k) \leq \alpha_0(|x(0) - \overline{x}_0|)$$

with $\mathcal{K}_\infty$-function $\alpha_0 := \overline{\alpha}_2 \circ \overline{\sigma}_0$.

Summarizing, we have established that FIE provides a Q-function that meets the following definition.

Definition 4.14 (Q-function for estimation)**.** A function $Q(j|k)$ is a Q-function for some state estimator if there exist $\mathcal{K}_\infty$-functions $\mu_0, \mu_1, \mu_2, \mu_3$ such that

$$Q(0|k) \leq \mu_0(|x(0) - \overline{x}_0|) \tag{4.16}$$

$$\mu_1(\left|x(j) - \hat{x}(j|k)\right|) \leq Q(j|k) \leq \mu_2(\left|x(j) - \hat{x}(j|k)\right|) \tag{4.17}$$

$$Q(j+1|k) \leq Q(j|k) - \mu_3(\left|x(j) - \hat{x}(j|k)\right|) \tag{4.18}$$

for all $j \leq k \in \mathbb{I}_{\geq 0}$ for (4.16) and (4.17) and $j \leq k-1 \in \mathbb{I}_{\geq 0}$ for (4.18).

Next we establish a Q-function theorem for nominal stability (Allan and Rawlings, 2019, Theorem 14).

Theorem 4.15 (Q-function theorem for global asymptotic stability)**.** *If a state estimator admits a Q-function, then it is globally asymptotically stable (GAS).*

Proof. First combine (4.17) and (4.18) to obtain

$$Q(j+1|k) \leq Q(j|k) - \mu_3(\mu_2^{-1}(Q(j|k)))$$

Next use the same standard construction shown in Appendix B, Theorem B.15, to obtain a $\mathcal{K}_\infty$-function σ satisfying $\sigma(s) < s$ for $s > 0$ and $\sigma(s) \geq s - \mu_3(\mu_2^{-1}(s))$, which gives

$$Q(j+1|k) \leq \sigma(Q(j|k))$$

Applying this result recursively starting at $j = 0$ gives

$$Q(j|k) \leq \sigma^j(Q(0|k))$$

Combining this with (4.16) and (4.17) then gives for all $j \leq k$

$$\begin{aligned}\left|x(j) - \hat{x}(j|k)\right| &\leq \mu_1^{-1}(\sigma^j(\mu_0(|x(0) - \overline{x}_0|))) \\ &:= \alpha_x(|x(0) - \overline{x}_0|, j)\end{aligned}$$

Note that $\alpha_x(\cdot) \in \mathcal{KL}$, and on choosing $j = k$, we have that

$$|x(k) - \hat{x}(k|k)| \leq \alpha_x(|x(0) - \overline{x}_0|, k)$$

for all $k \in \mathbb{I}_{\geq 0}$, and the state estimator is GAS. ■

So applying this theorem establishes that FIE is globally asymptotically stable for the case of zero input and output disturbances. We summarize the result in the following theorem.

Theorem 4.16 (Stability of full information estimation)**.** *Let Assumptions 4.10–4.13 hold. Then full information estimation is GAS.*

4.2.2 Robust Estimator Stability

The reason for increasing the abstraction level in the current presentation is not to handle nominal stability. That simple problem can be addressed with simple tools. The point is to address finally FIE with *bounded disturbances.* We are now in a good position to accomplish that. Let's first recall what we concluded about the steady-state Kalman filter (predictor) with bounded disturbances. We showed in Example 4.4 that the Kalman predictor is RGAS and that the estimate error satisfies (4.5). So that result represents the gold standard of FIE for a nonlinear system with bounded disturbances. We'll see next how close we can come to the same conclusion for nonlinear systems.

The system continuity and detectability conditions from the nominal case are unchanged when treating the bounded disturbance case. But the stage cost and stabilizability assumptions require modification. We state the new conditions next.

Assumption 4.17 (Stage cost under disturbances)**.** The stage cost $\ell(\cdot)$ satisfies

$$\sigma_w(2\,|\omega|) + \sigma_v(2\,|\nu|) \leq \ell(\omega,\nu) \leq \overline{\sigma}_w(|\omega|) + \overline{\sigma}_v(|\nu|)$$

for all $\omega \in \mathbb{W}, \nu \in \mathbb{V}$, for some $\mathcal{K}_\infty$-functions $\overline{\sigma}_w$ and $\overline{\sigma}_v$, and the $\mathcal{K}_\infty$-functions σ_w and σ_v come from (4.10) of the i-IOSS Lyapunov function. Furthermore, we have that

$$\alpha_2(2\,|\chi - \overline{x}_0|) \leq \ell_x(\chi - \overline{x}_0) \leq \overline{\sigma}_x(|\chi - \overline{x}_0|)$$

for all $\chi, \overline{x}_0 \in \mathbb{X}$, for some $\mathcal{K}_\infty$-function $\overline{\sigma}_x$, and the $\mathcal{K}_\infty$-function α_2 comes from (4.9) of the i-IOSS Lyapunov function.

Assumption 4.18 (Stabilizability under disturbances)**.** There exists $\mathcal{K}$-function $\overline{\gamma}$ such that for every finite sequences $\mathbf{w} \in \mathbb{W}^k$ and $\mathbf{v} \in \mathbb{V}^k$ and any $\chi, x \in \mathbb{X}$, there exists $\boldsymbol{\omega} \in \mathbb{W}^\infty$ such that the following holds for all $k \geq 0$

$$\sum_{i=0}^{\infty} \ell(\omega(j), \nu(j)) \leq \overline{\alpha}(|\chi - x|) + \sum_{i=0}^{k} \overline{\gamma}(|w(i), v(i)|)$$

in which

$$\chi^+ = f(\chi,\omega) \qquad y = h(\chi) + \nu$$

$$x^+ = \begin{cases} f(x,w) & \text{for } i \in \mathbb{I}_{0:k-1} \\ f(x,0) & \text{for } i \in \mathbb{I}_{k:\infty} \end{cases} \qquad y = \begin{cases} h(x) + v & \text{for } i \in \mathbb{I}_{0:k-1} \\ h(x) & \text{for } i \in \mathbb{I}_{k:\infty} \end{cases}$$

Remark.

(a) Note the introduction of the factor of two in the lower bound of $\ell(\cdot)$ in Assumption 4.17 compared to the nominal case, Assumption 4.11.

(b) Note the new compatibility restriction on the lower bound for $\ell_x(\cdot)$ in Assumption 4.17 compared to the nominal case, Assumption 4.11.

(c) In the stabilizability assumption note that the upper bound on the infinite horizon cost grows linearly with time for the case of bounded disturbances. It is anticipated that the full-information optimal cost also increases without bound for this bounded disturbance case. The divergence of the optimal cost presents one of the primary challenges in the estimator stability analysis.

It also will prove insightful to break out a stronger case of detectability, termed exponential detectability, defined as follows.

Definition 4.19 (Exponentially i-IOSS)**.** The system $x^+ = f(x,w), y = h(x)$ is *exponentially incrementally input/output-to-state stable* (exponentially i-IOSS) if there exist $0 \leq \lambda < 1$ and positive constants b_x, b_w, b_v such that for all $k \in \mathbb{I}_{\geq 0}$, all initial states $x_1, x_2 \in \mathbb{X}$, and all disturbance sequences $\mathbf{w}_1, \mathbf{w}_2 \in \mathbb{W}^\infty$

$$|x_1(k) - x_2(k)| \leq b_x |x_1 - x_2| \lambda^k \oplus \max_{j \in \mathbb{I}_{0:k-1}} b_w \left|\Delta w(j)\right| \lambda^{k-j-1} \oplus \max_{j \in \mathbb{I}_{0:k-1}} b_v \left|\Delta y(j)\right| \lambda^{k-j-1} \quad (4.19)$$

where $x_1(k) = x(k; x_1, \mathbf{w}_1), x_2(k) = x(k; x_2, \mathbf{w}_2)$, $\Delta w(k) = w_1(k) - w_2(k)$, and $\Delta y(k) = h(x_1(k)) - h(x_2(k))$.

Note that we have restricted the $\mathcal{KL}$-functions of (asymptotic) detectability to an exponential form. We shall see subsequently that this stronger form of detectability makes the analysis of moving horizon estimation particularly straightforward. Note also that detectable linear systems satisfy this property.

When we assume exponential detectability, we also achieve a stronger form of stability, termed robust global exponential stability, defined in convolution maximization form as follows.

Definition 4.20 (Robustly globally exponentially stable estimation)**.** A state estimator $(\Psi_k)_{k \geq 0}$ is robustly globally exponentially stable (RGES)

if there exist $0 \le \lambda < 1$ and positive constants a_x, a_w, a_v such that

$$|x(k) - \hat{x}(k)| \le a_x |x(0) - \overline{x}_0| \lambda^k \oplus \max_{j \in \mathbb{I}_{0:k-1}} a_w |w(j)| \lambda^{k-j-1} \oplus \max_{j \in \mathbb{I}_{0:k-1}} a_v |v(j)| \lambda^{k-j-1} \quad (4.20)$$

for all $k \in \mathbb{I}_{\ge 0}$, $x(0), \overline{x}_0 \in \mathbb{X}$, and $w \in \mathbb{W}$, $v \in \mathbb{V}$.

It is often convenient to compress the notation and combine the disturbances as $d(j) := (w(j), v(j)), j \in \mathbb{I}_{\ge 0}$ with $\mathbb{D} := \mathbb{W} \times \mathbb{V}$, and use the following equivalent definition of RGES.

Proposition 4.21 (Equivalent definition of RGES)**.** *A state estimator $(\Psi_k)_{k \ge 0}$ is robustly globally exponentially stable (RGES) if there exist $0 \le \lambda < 1$ and positive constant a_d such that*

$$|x(k) - \hat{x}(k)| \le a_x |x(0) - \overline{x}_0| \lambda^k \oplus \max_{j \in \mathbb{I}_{0:k-1}} a_d |d(j)| \lambda^{k-j-1} \quad (4.21)$$

for all $k \in \mathbb{I}_{\ge 0}$, $x(0), \overline{x}_0 \in \mathbb{X}$, and $d \in \mathbb{D}$.

Proof of this proposition is discussed in Exercise 4.20.

Next we strengthen the asymptotic Assumptions 4.11–4.13 to their exponential versions.

Assumption 4.22 (Power-law bounds for stage costs)**.** There exist positive constants $\underline{c}_\ell, \underline{c}_x, \overline{c}_\ell, \overline{c}_x$ and $\sigma \ge 1$ such that

$$\begin{aligned} \underline{c}_\ell |(\omega, \nu)|^\sigma \le \quad & \ell(\omega, \nu) \quad \le \overline{c}_\ell |(\omega, \nu)|^\sigma \\ \underline{c}_x |\chi - \overline{x}_0|^\sigma \le \ell_x(\chi - \overline{x}_0) & \le \overline{c}_x |\chi - \overline{x}_0|^\sigma \end{aligned}$$

for all $\omega \in \mathbb{W}, \nu \in \mathbb{V}$, and $\chi, \overline{x}_0 \in \mathbb{X}$.

Assumption 4.23 (Exponential stabilizability)**.** The system (4.1) is exponentially incrementally stabilizable, i.e., there exists positive constant $\overline{c} > 0$ such that for every two initial conditions $x_1, x_2 \in \mathbb{X}$ and input sequence $\boldsymbol{\omega}_1 \in \mathbb{W}^\infty$, there exists $\boldsymbol{\omega}_2 \in \mathbb{W}^\infty$ such that

$$\sum_{k=0}^{\infty} \ell(\omega_2(k) - \omega_1(k), h(x_1(k)) - h(x_2(k))) \le \overline{c} |x_1 - x_2|^\sigma$$

where $\sigma \ge 1$ comes from Assumption 4.22.

Assumption 4.24 (Exponential detectability)**.** The system (4.1) is exponentially i-IOSS.

We then have the following result for robust stability of FIE under disturbances.

Theorem 4.25 (Robust stability of full information estimation)**.**

(a) Let Assumptions 4.10, 4.13, 4.17, and 4.18 hold. Then full information estimation is RGAS.

(b) Let Assumptions 4.10 and 4.22–4.24 hold. Then full information estimation is RGES.

The proof for RGES is given in (Allan and Rawlings, 2020, Theorem 3.16). The considerably more involved proof for RGAS is given in (Allan, 2020, Theorem 5.18).

Theorem 4.25 is a reasonable resting place for the theory of full information estimation. We can finally handle bounded disturbances in a fairly clean theoretical development with reasonable assumptions on the system's detectability and stabilizability. If one is willing to strengthen the detectability assumption to *exponential* detectability as in Theorem 4.25(b), the theoretical development is reasonably compact, and can be easily extended to MHE as we show subsequently. Moreover, by strengthening the definitions of RGAS and RGES using the convolution maximization form, we have the desirable and anticipated consequence that stability implies convergence of estimate error given convergence of disturbances.

4.2.3 Interlude—Linear System Review

Given the many structural similarities between estimation and regulation, the reader may wonder why the stability analysis of the full information estimator presented in the previous sections looks rather different than the zero-state regulator stability analysis presented in Chapter 2.

State Estimation as Optimal Control of Estimate Error

To provide some insight into essential *differences*, as well as similarities, between estimation and regulation, consider again the estimation problem in the simplest possible setting with a linear time-invariant model and Gaussian noise

$$\begin{aligned} x^+ &= Ax + Gw \qquad & w &\sim N(0, Q) \\ y &= Cx + v \qquad & v &\sim N(0, R) \end{aligned} \tag{4.22}$$

and random initial state $x(0) \sim N(\overline{x}_0, P^-(0))$. In FIE, we define the objective function

$$V_T(\chi(0), \boldsymbol{\omega}) = \frac{1}{2}\Big(|\chi(0) - \overline{x}_0|^2_{(P^-(0))^{-1}} + \sum_{i=0}^{T-1} |\omega(i)|^2_{Q^{-1}} + |\nu(i)|^2_{R^{-1}} \Big)$$

subject to $\chi^+ = A\chi + G\omega$, $y = C\chi + \nu$. Denote the solution to this optimization as

$$(\hat{x}(0|T), \hat{\mathbf{w}}_T) = \arg\min_{\chi(0),\boldsymbol{\omega}} V_T(\chi(0), \boldsymbol{\omega})$$

and the trajectory of state estimates comes from the model $\hat{x}(i+1|T) = A\hat{x}(i|T) + G\hat{w}(i|T)$. We define estimate error as $\tilde{x}(i|T) = x(i) - \hat{x}(i|T)$ for $0 \le i \le T-1$, $T \ge 1$.

The simplest stability question is nominal stability, i.e., if noise-free data are provided to the estimator, $(w(i), v(i)) = 0$ for all $i \ge 0$ in (4.22), is the estimate error asymptotically stable as $T \to \infty$ for all x_0? We next make this statement precise. First we note that the noise-free measurement satisfies $y(i) - C\hat{x}(i|T) = C\tilde{x}(i|T), 0 \le i \le T$ and the initial condition term can be written in estimate error as $\hat{x}(0) - \overline{x}(0) = -(\tilde{x}(0) - a)$ in which $a = x(0) - \overline{x}_0$. For the noise-free measurement we can therefore rewrite the cost function as

$$V_T(a, \tilde{x}(0), \mathbf{w}) = \frac{1}{2}\Big(|\tilde{x}(0) - a|^2_{(P^-(0))^{-1}} + \sum_{i=0}^{T-1} |C\tilde{x}(i)|^2_{R^{-1}} + |w(i)|^2_{Q^{-1}} \Big) \tag{4.23}$$

in which we list explicitly the dependence of the cost function on parameter a. For estimation we solve

$$\min_{\tilde{x}(0),\mathbf{w}} V_T(a, \tilde{x}(0), \mathbf{w}) \tag{4.24}$$

subject to $\tilde{x}^+ = A\tilde{x} + Gw$. Now consider problem (4.24) as an optimal control problem (OCP) using w as the manipulated variable and minimizing an objective that measures size of estimate error $\tilde{x}$ and control w. We denote the optimal solution as $\tilde{x}^0(0; a)$ and $\mathbf{w}^0(a)$. Substituting these into the model equation gives optimal estimate error $\tilde{x}^0(j|T; a), 0 \le j \le T, 0 \le T$. Parameter a denotes how far $x(0)$, the system's initial state generating the measurement, is from $\overline{x}_0$, the prior. If we are lucky and $a = 0$, the optimal solution is $(\tilde{x}^0, \mathbf{w}^0) = 0$, and we achieve zero cost in V_T^0 and zero estimate error $\tilde{x}(j|T)$ at all time in

the trajectory $0 \leq j \leq T$ for all time $T \geq 1$. The stability analysis in estimation is to show that the origin for $\widetilde{x}$ is asymptotically stable. In other words, we wish to show there exists a KL function β such that $\left|\widetilde{x}^0(T;a)\right| \leq \beta(|a|,T)$ for all $T \in \mathbb{I}_{\geq 0}$.

We note the following differences between standard regulation and the estimation problem (4.24). First we see that (4.24) is slightly nonstandard because it contains an extra decision variable, the initial state, and an extra term in the cost function, (4.23). Indeed, without this extra term, the regulator could choose $\widetilde{x}(0) = 0$ to zero the estimate error immediately, choose $\mathbf{w} = 0$, and achieve zero cost in $V_T^0(a)$ for all a. The nonstandard regulator allows $\widetilde{x}(0)$ to be manipulated as a decision variable, but penalizes its distance from a. Next we look at the stability question.

The stability analysis is to show there exists KL function β such that $\left|\widetilde{x}^0(T;a)\right| \leq \beta(|a|,T)$ for all $T \in \mathbb{I}_{\geq 0}$. Here convergence is a question about the terminal state in a sequence of *different* OCPs with increasing horizon length T. That is also not the standard regulator convergence question, which asks how the state trajectory evolves using the optimal control law. In standard regulation, we inject the optimal first input and ask whether we are successfully moving the system to the origin as time increases. In estimation, we do not inject anything into the system; we are provided more information as time increases and ask whether our explanation of the data is improving (terminal estimate error is decreasing) as time increases.

Because stability is framed around the behavior of the terminal state, we would not choose *backward* dynamic programming (DP) to solve (4.24), as in standard regulation. We do not seek the optimal first control move as a function of a known initial state. Rather we seek the optimal terminal state $\widetilde{x}^0(T;a)$ as a function of the parameter a appearing in the cost function. This problem is better handled by *forward* DP as discussed in Sections 1.3.2 and 1.4.3 of Chapter 1 when solving the full information state estimation problem. Exercise 4.16 discusses how to solve (4.24); we obtain the following recursion for the optimal terminal state

$$\widetilde{x}^0(k+1;a) = (A - \widetilde{L}(k)C)\,\widetilde{x}^0(k;a) \tag{4.25}$$

for $k \geq 0$. The initial condition for the recursion is $\widetilde{x}^0(0;a) = a$. The time-varying gains $\widetilde{L}(k)$ and associated cost matrices $P^-(k)$ required

are

$$P^-(k+1) = GQG' + AP^-(k)A' - AP^-(k)C'(CP^-(k)C' + R)^{-1}CP^-(k)A' \tag{4.26}$$

$$\widetilde{L}(k) = AP^-(k)C'(CP^-(k)C' + R)^{-1} \tag{4.27}$$

in which $P^-(0)$ is specified in the problem. As expected, these are the standard estimator recursions developed in Chapter 1. Jazwinski (1970, Theorem 7.4) follows an argument introduced by Deyst and Price (1968), assumes controllability and observability, and tries to establish stability for this more restrictive case by showing that $V(k, \widetilde{x}) := (1/2)\widetilde{x}'P(k)^{-1}\widetilde{x}$ is a Lyapunov function for (4.25). Notice that this Lyapunov function candidate is *not* the optimal cost of (4.24) as in a standard regulation problem. The optimal cost of (4.24), $V_T^0(a)$, is an *increasing* function of T rather than a decreasing function of T as required for a Lyapunov function. Although one can find Lyapunov functions valid for estimation, they do not have the same simple connection to optimal cost functions as in standard regulation problems, even in the linear, unconstrained case. Stability arguments based instead on properties of $V_T^0(a)$ are simpler and more easily adapted to cover new situations arising in research problems.

Duality of Linear Estimation and Regulation

For linear systems, the estimate error $\widetilde{x}$ in FIE and state x in regulation to the origin display an interesting duality that we summarize briefly here. Consider the following steady-state estimation and infinite horizon regulation problems.

Estimator problem.

$$x(k+1) = Ax(k) + Gw(k)$$
$$y(k) = Cx(k) + v(k)$$

$$R > 0 \qquad Q > 0 \qquad (A, C) \text{ detectable} \qquad (A, G) \text{ stabilizable}$$

$$\widetilde{x}(k+1) = \left(A - \widetilde{L}C\right)\widetilde{x}(k)$$

Regulator problem.

$$x(k+1) = Ax(k) + Bu(k)$$
$$y(k) = Cx(k)$$

Regulator	Estimator
A	A'
B	C'
C	G'
k	$l = N - k$
$\Pi(k)$	$P^-(l)$
$\Pi(k-1)$	$P^-(l+1)$
Π	P^-
Q	Q
R	R
P_f	$P^-(0)$
K	$-\widetilde{L}'$
$A + BK$	$(A - \widetilde{L}C)'$
x	$\widetilde{x}'$

Regulator	Estimator
$R > 0, \quad Q > 0$	$R > 0, \quad Q > 0$
(A, B) stabilizable	(A, C) detectable
(A, C) detectable	(A, G) stabilizable

Table 4.2: Duality variables and stability conditions for linear quadratic regulation and least squares estimation.

$$R > 0 \qquad Q > 0 \qquad (A, B) \text{ stabilizable} \qquad (A, C) \text{ detectable}$$

$$x(k+1) = (A + BK)\, x(k)$$

In Appendix A, we derive the dual dynamic system following the approach in Callier and Desoer (1991), and obtain the duality variables in regulation and estimation listed in Table 4.2.

We also have the following result connecting controllability of the original system and observability of the dual system.

Lemma 4.26 (Duality of controllability and observability)**.** *(A, B) is controllable (stabilizable) if and only if (A', B') is observable (detectable).*

This result can be established directly using the Hautus lemma and is left as an exercise. This lemma and the duality variables allow us to translate stability conditions for infinite horizon regulation problems into stability conditions for FIE problems, and vice versa. For example, the following is a basic theorem covering convergence of Riccati equations in the form that is useful in establishing exponential stability of regulation as discussed in Chapter 1.

Theorem 4.27 (Riccati iteration and regulator stability)**.** *Given (A, B) stabilizable, (A, C) detectable, $Q > 0$, $R > 0$, $P_f \geq 0$, and the discrete*

Riccati equation

$$\begin{aligned}\Pi(k-1) &= C'QC + A'\Pi(k)A - \\ &\qquad A'\Pi(k)B(B'\Pi(k)B+R)^{-1}B'\Pi(k)A, \quad k = N,\dots,1 \\ \Pi(N) &= P_f\end{aligned}$$

Then

(a) There exists $\Pi \geq 0$ *such that for every* $P_f \geq 0$

$$\lim_{k\to-\infty} \Pi(k) = \Pi$$

and Π *is the unique solution of the steady-state Riccati equation*

$$\Pi = C'QC + A'\Pi A - A'\Pi B(B'\Pi B + R)^{-1}B'\Pi A$$

among the class of positive semidefinite matrices.

(b) The matrix $A + BK$*, in which*

$$K = -(B'\Pi B + R)^{-1}B'\Pi A$$

is a stable matrix.

Bertsekas (1987, pp.59–64) provides a proof for a slightly different version of this theorem. Exercise 4.17 explores translating this theorem into the form that is useful for establishing exponential convergence of FIE.

4.3 Moving Horizon Estimation

As displayed in Figure 1.5 of Chapter 1, in MHE we consider only the N most recent measurements, $\mathbf{y}_N(T) = \big(y(T-N), y(T-N+1),\dots, y(T-1)\big)$. For $T > N$, the MHE objective function is given by

$$\hat{V}_T(\chi(T-N), \boldsymbol{\omega}) = \Gamma_{T-N}(\chi(T-N)) + \sum_{i=T-N}^{T-1} \ell(\omega(i), v(i))$$

subject to $\chi^+ = f(\chi,\omega)$, $y = h(\chi) + v$. The MHE problem is defined to be

$$\hat{\mathbb{P}}_T(\overline{x}_{T-N}, \mathbf{y}_N(T)) := \min_{\chi(T-N),\boldsymbol{\omega}} \hat{V}_T(\chi(T-N), \boldsymbol{\omega}) \tag{4.28}$$

in which $\boldsymbol{\omega} = (\omega(T-N),\dots,\omega(T-1))$. The designer chooses the prior weighting $\Gamma_k(\cdot)$ for $k > 0$. Until the data horizon is full, i.e., for times $T \leq N$, we generally *define* the MHE problem to be the full information problem.

4.3.1 Zero Prior Weighting

Here we discount the early data completely and choose $\Gamma_i(\cdot) = 0$ for all $i \geq 0$. Because it discounts the past data completely, this form of MHE must be able to asymptotically reconstruct the state using only the most recent N measurements. The first issue is establishing existence of the solution. Unlike the full information problem, in which the positive definite initial penalty guarantees that the optimization takes place over a bounded (compact) set, here there is zero initial penalty. So we must restrict the system further than i-IOSS to ensure solution existence. We show next that observability is sufficient for this purpose.

Definition 4.28 (Observability)**.** The system $x^+ = f(x,w), y = h(x)$ is *observable* if there exist finite $N_o \in \mathbb{I}_{\geq 1}$, $\gamma_w(\cdot)$, $\gamma_v(\cdot) \in \mathcal{K}$ such that for every two initial states z_1 and z_2, and any two disturbance sequences $\mathbf{w}_1, \mathbf{w}_2$, and all $k \geq N_o$

$$|z_1 - z_2| \leq \gamma_w(\|\mathbf{w}_1 - \mathbf{w}_2\|_{0:k-1}) + \gamma_v(\|\mathbf{y}_{z_1,\mathbf{w}_1} - \mathbf{y}_{z_2,\mathbf{w}_2}\|_{0:k-1})$$

Let Assumption 4.10 hold. Then the MHE objective function $\hat{V}_T(\chi(T-N), \boldsymbol{\omega})$ is a continuous function of its arguments because $f(\cdot)$ and $h(\cdot)$ are continuous. We next show that $\hat{V}_T(\cdot)$ is an unbounded function of its arguments, which establishes existence of the solution of the MHE optimization problem. Let Assumption 4.11 hold. Then we have that

$$\hat{V}_T(\chi(T-N), \boldsymbol{\omega}) = \sum_{i=T-N}^{T-1} \ell(\omega(i), \nu(i)) \geq \sum_{i=T-N}^{T-1} \sigma_w(|\omega(i)|) + \sigma_v(|\nu(i)|) \quad (4.29)$$

From observability we have that for $N \geq N_o$

$$|x(T-N) - \chi(T-N)| \leq \gamma_w(\|\mathbf{w} - \boldsymbol{\omega}\|_{T-N:T-1}) + \gamma_v(\|\mathbf{v} - \boldsymbol{\nu}\|_{T-N:T-1}) \quad (4.30)$$

Consider arbitrary but fixed values of time T, horizon length $N \geq N_o$, and the system state and measurement sequence. Let the decision variables $|(\chi(T-N), \boldsymbol{\omega})| \to \infty$. Then we have that either $|\chi(T-N)| \to \infty$ or $|\boldsymbol{\omega}| \to \infty$. If $|\boldsymbol{\omega}| \to \infty$, we have directly from (4.29) that $\hat{V}_T \to \infty$. On the other hand, if $|\chi(T-N)| \to \infty$, then from (4.30), since

$x(T-N)$, $\mathbf{w}$ and $\mathbf{v}$ are fixed, we have that either $\|\boldsymbol{\omega}\|_{T-N:T-1} \to \infty$ or $\|\boldsymbol{\nu}\|_{T-N:T-1} \to \infty$, which implies from (4.29) that $\hat{V}_T \to \infty$. We conclude that $\hat{V}_T(\chi(T-N), \boldsymbol{\omega}) \to \infty$ if $|(\chi(T-N), \boldsymbol{\omega})| \to \infty$. Therefore the objective function is a continuous and unbounded function of its arguments, and existence of the solution of the MHE problem can be established from the Weierstrass theorem (Proposition A.7). The solution does not have to be unique.

We show next that final-state observability is a less restrictive and more natural system requirement for MHE with zero prior weighting to provide stability and convergence.

Definition 4.29 (Final-state observability)**.** The system $x^+ = f(x,w)$, $y = h(x)$ is *final-state observable* (FSO) if there exist finite $N_o \in \mathbb{I}_{\geq 1}$, $\overline{\gamma}_w(\cdot)$, $\overline{\gamma}_v(\cdot) \in \mathcal{K}$ such that for every two initial states z_1 and z_2, and any two disturbance sequences $\mathbf{w}_1, \mathbf{w}_2$, and all $k \geq N_o$

$$|x(k;z_1,\mathbf{w}_1) - x(k;z_2,\mathbf{w}_2)| \leq \overline{\gamma}_w(\,\|\mathbf{w}_1 - \mathbf{w}_2\|_{0:k-1}\,) + \overline{\gamma}_v(\,\|\mathbf{y}_{z_1,\mathbf{w}_1} - \mathbf{y}_{z_2,\mathbf{w}_2}\|_{0:k-1}\,)$$

Notice that FSO is not the same as observable. For sufficiently restricted $f(\cdot)$, FSO is weaker than observable and stronger than i-IOSS (detectable) as discussed in Exercise 4.14.

To ensure FSO, we restrict the system as follows.

Definition 4.30 (Globally $\mathcal{K}$-continuous)**.** A function $f : \mathcal{X} \to \mathcal{Y}$ is globally $\mathcal{K}$-continuous if there exists function $\sigma(\cdot) \in \mathcal{K}$ such that for all $x_1, x_2 \in \mathcal{X}$

$$|\,f(x_1) - f(x_2)\,| \leq \sigma(|x_1 - x_2|) \tag{4.31}$$

We then have the following result.

Proposition 4.31 (Observable and global $\mathcal{K}$-continuous imply FSO)**.** *An observable system $x^+ = f(x,w), y = h(x)$ with globally $\mathcal{K}$-continuous $f(\cdot)$ is final-state observable.*

The proof of this proposition is discussed in Exercise 4.14. Consider two equal disturbance sequences, $\mathbf{w}_1 = \mathbf{w}_2$, and two equal measurement sequences $\mathbf{y}_1 = \mathbf{y}_2$. FSO implies that for every pair z_1 and z_2, $x(N_o;z_1,\mathbf{w}_1) = x(N_o;z_2,\mathbf{w}_1)$; we know the *final* states at time $k = N_o$ are equal. FSO does not imply that the *initial* states are equal as required when the system is observable. We can of course add the nonnegative term $\beta(|z_1 - z_2|\,, k)$ to the right-hand side of the FSO inequality and obtain the i-IOSS inequality, so FSO implies i-IOSS. Exercise 4.11

treats observable, FSO, and detectable for the linear time-invariant system, which can be summarized compactly in terms of the eigenvalues of the partitioned state transition matrix corresponding to the unobservable modes.

Definition 4.32 (RGAS estimation (observable case))**.** The estimate is based on the *noisy* measurement $\mathbf{y} = h(\mathbf{x}(x_0, \mathbf{w})) + \mathbf{v}$. The estimator is RGAS (observable case) if there exist $N_o \in \mathbb{I}_{\geq 1}$ and function $\delta(\cdot) \in \mathcal{K}$ such that the following holds for all $x_0, \overline{x}_0 \in \mathbb{X}$, $w \in \mathbb{W}$, $v \in \mathbb{V}$, and $k \geq N_o$

$$\left| x(k; x_0, \mathbf{w}) - x(k; \hat{x}(0|k), \hat{\mathbf{w}}_k) \right| \leq \delta(\|(\mathbf{w}, \mathbf{v})\|_{k-N_o:k-1})$$

Remark. Notice that the definition of RGAS estimation in the observable case is silent about what happens to estimate error at early times, $k < N_o$, while the estimator is collecting enough measurements to obtain its first valid state estimate.

We have the following theorem for this estimator.

Theorem 4.33 (MHE is RGAS (observable case))**.** *Consider an observable system with globally $\mathcal{K}$-continuous $f(\cdot)$, and measurement sequence generated by* (4.1) *with bounded disturbances. Let Assumptions 4.10 and 4.11 hold. Then the MHE estimator using zero prior weighting and $N \geq N_o$ is RGAS (observability case).*

Proof. Consider the system to be at state $x(k-N)$ at time $k-N$ and subject to disturbance sequence $(\mathbf{w}_k, \mathbf{v}_k)$. Due to system observability and Assumption 4.10, the MHE problem has a solution for all $k \geq N \geq N_o$. Denote the estimator solution at such time k as initial state $\hat{x}(k-N|k)$ and disturbance sequence $(\hat{w}_k, \hat{v}_k)$. We start by noting that the optimal MHE cost satisfies the bounds

$$\sum_{i=k-N}^{k-1} \ell(\hat{w}_k(i), \hat{v}_k(i)) = \hat{V}_k^0 \leq \sum_{i=k-N}^{k-1} \ell(w(i), v(i))$$

Using the upper and lower bounds in Assumption 4.11, $\sigma_w, \sigma_v, \overline{\sigma}_w$, $\overline{\sigma}_v$, (B.1), and noting that $\max(|a|, |b|) \leq |(a,b)| \leq |a| + |b|$, we can convert these bounds into

$$\underline{\sigma}(\left|\left|(\hat{\mathbf{w}}_k, \hat{\mathbf{v}}_k)\right|\right|_{k-N:k-1}) \leq \hat{V}_k^0 \leq \overline{\sigma}(\|(\mathbf{w}_k, \mathbf{v}_k)\|_{k-N:k-1})$$

where $\underline{\sigma}(\cdot) := \min(\sigma_w, \sigma_v)((\cdot)/2)$ and $\overline{\sigma} := 2N \max(\overline{\sigma}_w, \overline{\sigma}_v)$. Note that $\underline{\sigma}(\cdot), \overline{\sigma}(\cdot) \in \mathcal{K}$. The system is FSO by Proposition 4.31 since the

system is observable and $f(\cdot)$ is globally $\mathcal{K}$-continuous. Considering $x(k-N)$ and $\hat{x}(k-N|k)$ as two initial conditions and applying the FSO bound gives

$$|x(k) - \hat{x}(k)| \leq \overline{\gamma}_w\big(\big\|\mathbf{w}_k - \hat{\mathbf{w}}_k\big\|_{k-N:k-1}\big) + \overline{\gamma}_v\big(\big\|(\mathbf{v}_k - \hat{\mathbf{v}}_k)\big\|_{k-N:k-1}\big)$$

for some $\mathcal{K}$-functions $\overline{\gamma}_w, \overline{\gamma}_v$. Again using $|(a,b)| \geq \max(|a|, |b|)$ and the triangle inequality, this bound can be rearranged into

$$|x(k) - \hat{x}(k)| \leq \overline{\gamma}_x\big(\|(\mathbf{w}_k, \mathbf{v}_k)\|_{k-N:k-1}\big) + \overline{\gamma}_x\big(\big\|(\hat{\mathbf{w}}_k, \hat{\mathbf{v}}_k)\big\|_{k-N:k-1}\big)$$

where $\overline{\gamma}_x(\cdot) := 2\max(\overline{\gamma}_w, \overline{\gamma}_v)(2(\cdot))$. Note that $\overline{\gamma}_x(\cdot) \in \mathcal{K}$. Next apply $\underline{\sigma}^{-1}$ to the $\hat{V}_k^0$ inequality above to obtain

$$\big\|(\hat{\mathbf{w}}_k, \hat{\mathbf{v}}_k)\big\|_{k-N:k-1} \leq \underline{\sigma}^{-1} \circ \overline{\sigma}(\|(\mathbf{w}_k, \mathbf{v}_k)\|_{k-N:k-1})$$

and substitute this result into the previous inequality to obtain for all $k \geq N \geq N_0$

$$|x(k) - \hat{x}(k)| \leq \delta\big(\|(\mathbf{w}_k, \mathbf{v}_k)\|_{k-N:k-1}\big)$$

with $\delta := \overline{\gamma}_x + \overline{\gamma}_x \circ \underline{\sigma}^{-1} \circ \overline{\sigma}$, which is also a $\mathcal{K}$-function. We have therefore established that MHE with zero prior weighting is RGAS (observable case). ■

Notice that unlike in FIE, the estimate error bound does not require the initial error $x(0) - \overline{x}_0$ since we have zero prior weighting and as a result have assumed observability rather than detectability. Notice also that RGAS implies estimate error converges to zero for convergent disturbances. Finally, the $\mathcal{K}$-functions $\overline{\sigma}$ and hence δ increase with N, which shows that this analysis can likely be tightened to remove this N dependence. See also the Notes discussion on this point.

4.3.2 Nonzero Prior Weighting

The two drawbacks of zero prior weighting are: the system had to be assumed *observable* rather than detectable to ensure existence of the solution to the MHE problem; and a large horizon N may be required to obtain performance comparable to full information estimation. We address these two disadvantages by using nonzero prior weighting. To get started, we use forward DP, as we did in Chapter 1 for the unconstrained linear case, to decompose the FIE problem exactly into the MHE problem (4.28) in which $\Gamma_k(\cdot)$ is chosen as arrival cost.

Definition 4.34 (Full information arrival cost)**.** The full information arrival cost is defined as

$$Z_T(p) = \min_{\chi(0),\boldsymbol{\omega}} V_T(\chi(0),\boldsymbol{\omega}) \tag{4.32}$$

subject to

$$\chi^+ = f(\chi,\omega) \qquad y = h(\chi) + v \qquad \chi(T;\chi(0),\boldsymbol{\omega}) = p$$

We have the following equivalence.

Lemma 4.35 (MHE and FIE equivalence)**.** *The MHE problem* (4.28) *is equivalent to the full information problem* (4.3) *for the choice* $\Gamma_k(\cdot) = Z_k(\cdot)$ *for all* $k > N$ *and* $N \geq 1$*.*

The proof is left as an exercise. This lemma is the essential insight provided by the DP recursion. But notice that evaluating arrival cost in (4.32) has the same computational complexity as solving a full information problem. So next we generate an MHE problem that has simpler computational requirements, but retains the excellent stability properties of full information estimation.

4.3.3 RGES of MHE under exponential assumptions

We consider the simplest case of MHE in which we penalize deviation from $\hat{x}(k|k)$ with prior weighting that has power-law upper and lower bounds with time-invariant parameters described by the following assumption.

Assumption 4.36 (MHE prior weighting bounds)**.** For all $k \in \mathbb{I}_{\geq 0}$, $\Gamma_k : \mathbb{X} \times \mathbb{X} \to \mathbb{R}_{\geq 0}$ is continuous and there exist constants $\underline{c}_\Gamma, \overline{c}_\Gamma \geq 0$ such that the following bounds hold uniformly in k for all $\chi, \hat{x}(k|k) \in \mathbb{X}$

$$\underline{c}_\Gamma \, |\chi - \hat{x}(k|k)|^\sigma \leq \Gamma_k(\chi) \leq \overline{c}_\Gamma \, |\chi - \hat{x}(k|k)|^\sigma \tag{4.33}$$

in which $\sigma \geq 1$ comes from Assumption 4.22.

So, when solving the MHE problem at time T, we bound the prior weighting on the initial state at time $T-N$ using the deviation from the estimate $\hat{x}(T-N|T-N)$. Choosing a constant c_Γ satisfying $\underline{c}_\Gamma \leq c_\Gamma \leq \overline{c}_\Gamma$ and corresponding prior weighting $\Gamma_k(\chi) = c_\Gamma \, |\chi - \hat{x}(k|k)|^\sigma$ would be the simplest choice meeting this assumption.

We next establish that MHE is RGES under the exponential case assumptions with this so-called filtering prior and constant prior weighting bounds (Allan and Rawlings, 2020, Theorem 4.2).

Theorem 4.37 (MHE is RGES)**.** *Let Assumptions 4.10, 4.22–4.24, and 4.36 hold. Then there exists a horizon length $\underline{N}$ such that MHE is RGES for all $N \geq \underline{N}$.*

Proof. Let $e(k) \coloneqq x(k) - \hat{x}(k|k)$ and $\overline{e}_0 \coloneqq x(0) - \overline{x}_0$ to compress the notation. Let any time $k \in \mathbb{I}_{\geq 0}$ be expressed as $k = k_0 + pN$ for $k_0 \in \mathbb{I}_{0:N-1}$ and $p \geq 0$. Since $k_0 \leq N-1$, the horizon at time $k = k_0$ is not yet filled, and the MHE problem reduces to the FIE problem; we have from Theorem 4.25(b) and (4.21) that

$$|e(k_0)| \leq a_x \, |\overline{e}_0| \, \lambda^{k_0} \oplus \max_{j \in \mathbb{I}_{0:k_0-1}} a_d \, |d(k_0 - j - 1)| \, \lambda^j$$

with $0 \leq \lambda < 1$. Now consider the time to be one horizon length later. The MHE problem at this time has identical structure to the FIE problem, but with different data: the initial prior $\overline{x}_0$ is replaced by $\hat{x}(k_0|k_0)$, the bounds on $\ell_x(\cdot)$ are replaced by the bounds on $\Gamma_k(\cdot)$, and the initial and final times $(0, k_0)$ are replaced by $(k_0, k_0 + N)$. We therefore have that

$$|e(k_0 + N)| \leq a_\Gamma \, |e(k_0)| \, \lambda^N \oplus \max_{j \in \mathbb{I}_{0:N-1}} a_d \, |d(k_0 + N - j - 1)| \, \lambda^j$$

where the RGES constant a_x is altered by the new data to a new constant denoted $a_\Gamma > 0$.[4] Using the previous bound for $e(k_0)$ then gives

$$\begin{aligned} |e(k_0 + N)| \leq |\overline{e}_0| \, a_x (a_\Gamma \lambda^{k_0+N}) \oplus a_\Gamma \lambda^N \max_{j \in \mathbb{I}_{0:k_0-1}} a_d \, |d(k_0 - j - 1)| \, \lambda^j \\ \oplus \max_{j \in \mathbb{I}_{0:N-1}} a_d \, |d(k_0 + N - j - 1)| \, \lambda^j \end{aligned}$$

We next choose N large enough so that $a_\Gamma \lambda^N < 1$. Choose $\underline{N} \in \mathbb{I}_{\geq 1}$ as the smallest value such that $a_\Gamma \lambda^{\underline{N}} < 1$, and we restrict the horizon to $N \geq \underline{N}$. Repeating this bounding argument gives for $p \geq 0$

$$\begin{aligned} &|e(k_0 + pN)| \leq \\ &\quad |\overline{e}_0| \, a_x (a_\Gamma^p \lambda^{k_0+pN}) \oplus (a_\Gamma \lambda^N)^p \max_{j \in \mathbb{I}_{0:k_0-1}} a_d \, |d(k_0 - j - 1)| \, \lambda^j \\ &\qquad \bigoplus_{i=0}^{p-1} (a_\Gamma \lambda^N)^i \max_{j \in \mathbb{I}_{0:N-1}} a_d \, |d(k_0 + (p-i)N - j - 1)| \, \lambda^j \end{aligned}$$

[4] The constant a_x is derived in the proof of Theorem 3.16 in Allan and Rawlings (2020) and shown to be $a_x \coloneqq \left[(\overline{c}_x + c_2 2^{\sigma-1}(1 + \overline{c}_x / \underline{c}_x))/c_1 \right]^{1/\sigma}$ where $c_1 \leq c_2$ are the constants in the power-law bounds for the exponential i-IOSS Lyapunov function corresponding to Assumption 4.24, and $\underline{c}_x, \overline{c}_x$ are from Assumption 4.22. The value of a_Γ is therefore given by replacing $\underline{c}_x$ and $\overline{c}_x$ in this expression with $\underline{c}_\Gamma$ and $\overline{c}_\Gamma$, respectively. Note that $a_x, a_\Gamma \geq 1$ since $c_1 \leq c_2$.

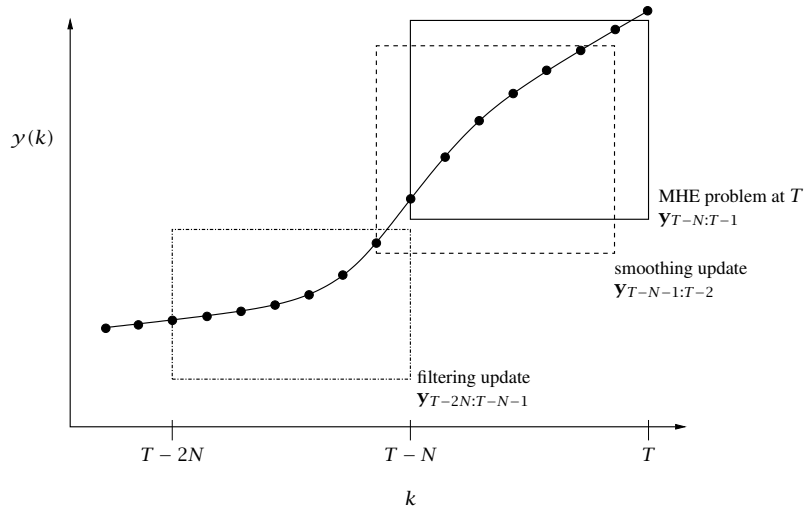


Figure 4.1: Smoothing update.

Now let $\eta := a_\Gamma^{1/N}\lambda$, and note that $\lambda \leq \eta < 1$ by the choice of $\underline{N}$. Substituting η into the previous equation and noting that $a_\Gamma \geq 1$ gives the bound

$$|e(k_0+pN)| \leq |\overline{e}_0|\, a_x \eta^{k_0+pN} \oplus \max_{j\in\mathbb{I}_{0:k_0-1}} a_d\, |d(k_0-j-1)|\, \eta^{pN+j}$$
$$\bigoplus_{i=0}^{p-1} \max_{j\in\mathbb{I}_{0:N-1}} a_d\, |d(k_0+(p-i)N-j-1)|\, \eta^{iN+j}$$

Now substitute $k = k_0 + pN$ and note that the maximizations simplify giving

$$|e(k)| \leq a_x\, |\overline{e}_0|\, \eta^k \oplus \max_{j\in\mathbb{I}_{0:k-1}} a_d\, |d(k-j-1)|\, \eta^j$$

for all $k \geq 0$ with $N \geq \underline{N}$, and MHE is RGES from Proposition 4.21. ■

Filtering versus smoothing update. The MHE approach discussed to this point uses, at all time $T > N$, the MHE estimate $\hat{x}(T-N)$ and prior weighting function $\Gamma_{T-N}(\cdot)$, which may be regarded as our best approximation of the arrival cost. We call this approach a "filtering update" because the prior weight at time T is derived from the solution of the MHE "filtering problem" at time $T-N$, i.e., the estimate of $\hat{x}(T-N) := \hat{x}(T-N|T-N)$ given measurements up to time $T-N-1$. For

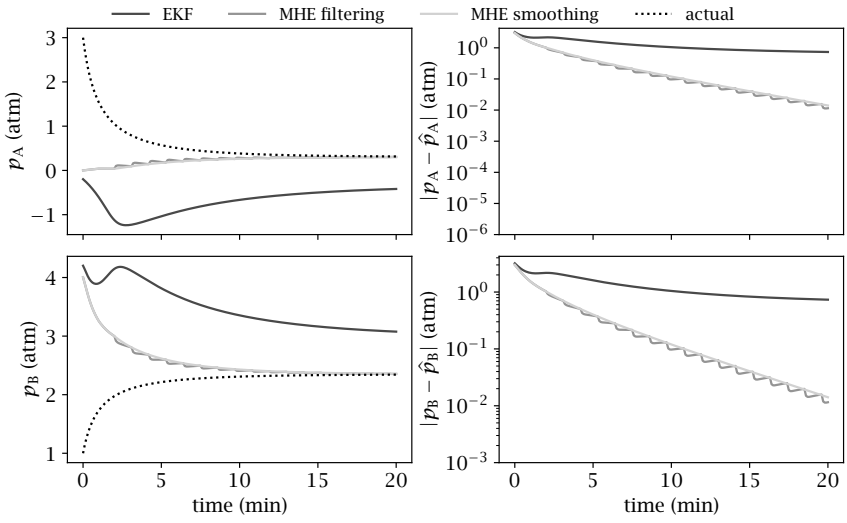


Figure 4.2: Comparison of filtering and smoothing updates for the batch reactor system. Second column shows absolute estimate error.

implementation, this choice requires storage of a window of N prior filtering estimates to be used in the prior weighting functions as time progresses.

Next we describe a "smoothing update" that can be used instead. As depicted in Figure 4.1, in the smoothing update we wish to use $\hat{x}(T - N|T - 1)$ (instead of $\hat{x}(T - N|T - N)$) for the prior and wish to find an appropriate prior weighting based on this choice. For the linear *unconstrained* problem we can find an exact prior weighting that gives an equivalence to the full information problem. See Rao, Rawlings, and Lee (2001) and Rao (2000, pp.80–93) for a derivation of this equivalence, with minor error corrections provided in the first edition of this text (Rawlings and Mayne, 2009, p.292).

We illustrate with the following example why the smoothing update may be useful in nonlinear models.

Example 4.38: Filtering and smoothing updates

Consider a constant-volume batch reactor in which the reaction $2A \rightleftharpoons B$ takes place (Tenny and Rawlings, 2002). The system state x consists

of the partial pressures (p_A, p_b) that evolve according to

$$\frac{dp_A}{dt} = -2k_1 p_A^2 + 2k_2 p_B$$
$$\frac{dp_B}{dt} = k_1 p_A^2 - k_2 p_B$$

with $k_1 = 0.16$ min^{-1} atm^{-1} and $k_2 = 0.0064$ min^{-1}. The only measurement is total pressure, $y = p_A + p_B$.

Starting from initial condition $x = (3, 1)$, the system is measured with sample time $\Delta = 0.1$ min. The model is exact and there are no disturbances. Using a poor initial estimate $\overline{x}_0 = (0.1, 4.5)$, parameters

$$Q = \begin{bmatrix} 10^{-4} & 0 \\ 0 & 0.01 \end{bmatrix} \qquad R = \begin{bmatrix} 0.01 \end{bmatrix} \qquad P = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

and horizon $N = 10$, MHE is performed on the system using the filtering and smoothing updates for the prior weighting. For comparison, the EKF is also used. The resulting estimates are plotted in Figure 4.2.

In this simulation, MHE performs well with either update formula. Due to the structure of the filtering update, every $N = 10$ time steps, a poor state estimate is used as the prior, which leads to undesirable periodic behavior in the estimated state. Due to the poor initial state estimate, the EKF produces negative pressure estimates, leading to large estimate errors throughout the simulation. □

Summary remarks. The results presented in this section are representative of what is currently known about MHE for bounded disturbances, but we expect that this analysis remains far from finished. Several questions remain open.

- Is MHE RGAS if the system is asymptotically (rather than exponentially) i-IOSS? What are the required compatibility conditions between the allowable stage costs and the i-IOSS condition to achieve an RGAS MHE estimator?
- What are the best methods to update the MHE initial penalty, $\Gamma_k(\cdot)$ to obtain an accurate estimator with a small horizon N for computational efficiency?
- Is MHE with a smoothing update instead of a filtering update RGAS? What stage costs are allowable to achieve RGAS of an MHE estimator with a smoothing update?

4.4 Other Nonlinear State Estimators

State estimation for nonlinear systems has a long history, and moving horizon estimation is a rather new approach to the problem. As with model predictive control, the optimal estimation problem on which moving horizon is based has a long history, but only the rather recent advances in computing technology have enabled moving horizon estimation to be considered as a viable option in online applications. It is therefore worthwhile to compare moving horizon estimation to other less computationally demanding nonlinear state estimators.

4.4.1 Particle Filtering

An extensive discussion and complete derivation of particle filtering appeared in the first edition of the text (Rawlings and Mayne, 2009, pp.301–355). This material is available electronically on the text's website. As with many sample-based procedures, however, it seems that all of the available sampling strategies in particle filtering do run into the "curse of dimensionality." The low density of samples in a reasonably large-dimensional space (say $n \geq 5$) lead to inaccurate state estimates. For this reason we omit further discussion of particle filtering in this edition.

Feedback particle filtering has recently been suggested as an alternative to overcome many of the drawbacks of the particle filter (Yang, Mehta, and Meyn, 2013). In feedback particle filtering, one uses the measurements to influence the particle locations by solving an optimal control problem for repositioning the particles to obtain an accurate posterior distribution after measurement. Application examples and a burgeoning literature on the theoretical properties of different algorithms indicate that this technique may provide a valuable addition to nonlinear estimation (Berntorp and Grover, 2018).

4.4.2 Extended Kalman Filtering

The extended Kalman filter (EKF) generates estimates for *nonlinear* systems by first linearizing the nonlinear system, and then applying the linear Kalman filter equations to the linearized system. The approach can be summarized in a recursion similar in structure to the Kalman

filter (Stengel, 1994, pp.387–388)

$$\begin{aligned}\hat{x}^-(k+1) &= f(\hat{x}(k),0)\\ P^-(k+1) &= \overline{A}(k)P(k)\overline{A}(k)' + \overline{G}(k)Q\overline{G}(k)'\\ \hat{x}^-(0) &= \overline{x}_0 \qquad P^-(0) = Q_0\end{aligned}$$

The mean and covariance after measurement are given by

$$\begin{aligned}\hat{x}(k) &= \hat{x}^-(k) + L(k)\big(y(k) - h(\hat{x}^-(k))\big)\\ L(k) &= P^-(k)\overline{C}(k)'(R + \overline{C}(k)P^-(k)\overline{C}(k)')^{-1}\\ P(k) &= P^-(k) - L(k)\overline{C}(k)P^-(k)\end{aligned}$$

with the following linearizations

$$\overline{A}(k) = \left.\frac{\partial f(x,w)}{\partial x}\right|_{(\hat{x}(k),0)} \quad \overline{G}(k) = \left.\frac{\partial f(x,w)}{\partial w}\right|_{(\hat{x}(k),0)} \quad \overline{C}(k) = \left.\frac{\partial h(x)}{\partial x}\right|_{\hat{x}^-(k)}$$

The densities of w, v, and x_0 are assumed to be normal. Many variations on this theme have been proposed, such as the iterated EKF and the second-order EKF (Gelb, 1974, 190–192). Of the nonlinear filtering methods, the EKF method has received the most attention due to its relative simplicity and demonstrated effectiveness in handling some nonlinear systems. Examples of implementations include estimation for the production of silicon/germanium alloy films (Middlebrooks and Rawlings, 2006), polymerization reactions (Prasad, Schley, Russo, and Bequette, 2002), and fermentation processes (Gudi, Shah, and Gray, 1994). The EKF is at best an *ad hoc* solution to a difficult problem, however, and hence there exist many pitfalls to the practical implementation of EKFs (see, for example, (Wilson, Agarwal, and Rippin, 1998)). These problems include the inability to accurately incorporate physical state constraints, and the naive use of linearization of the nonlinear model.

Until recently, few properties regarding the stability and convergence of the EKF have been established. Recent research shows bounded estimation error and exponential convergence for the continuous and discrete EKF forms given observability, small initial estimation error, small noise terms, and no model error (Reif, Günther, Yaz, and Unbehauen, 1999; Reif and Unbehauen, 1999; Reif, Günther, Yaz, and Unbehauen, 2000). Depending on the system, however, the bounds on initial estimation error and noise terms may be unrealistic. Also, initial estimation error may result in bounded estimate error but not exponential convergence, as illustrated by Chaves and Sontag (2002).

Julier and Uhlmann (2004a) summarize the status of the EKF as follows.

> The extended Kalman filter is probably the most widely used estimation algorithm for nonlinear systems. However, more than 35 years of experience in the estimation community has shown that it is difficult to implement, difficult to tune, and only reliable for systems that are almost linear on the time scale of the updates.

We seem to be making a transition from a previous era in which new approaches to nonlinear filtering were criticized as overly complex because "the EKF works," to a new era in which researchers are demonstrating ever simpler examples in which the EKF fails completely. The unscented Kalman filter is one of the methods developed specifically to overcome the problems caused by the naive linearization used in the EKF.

4.4.3 Unscented Kalman Filtering

The linearization of the nonlinear model at the current state estimate may not accurately represent the dynamics of the nonlinear system behavior even for one sample time. In the EKF prediction step, the mean propagates through the full nonlinear model, but the covariance propagates through the linearization. The resulting error is sufficient to throw off the correction step and the filter can diverge even with a perfect model. The unscented Kalman filter (UKF) avoids this linearization at a single point by sampling the nonlinear response at several points. The points are called sigma points, and their locations and weights are chosen to satisfy the given starting mean and covariance (Julier and Uhlmann, 2004a,b).[5] Given $\hat{x}$ and P, choose sample points, z^i, and weights, w^i, such that

$$\hat{x} = \sum_i w^i z^i \qquad P = \sum_i w^i (z^i - \hat{x})(z^i - \hat{x})'$$

Similarly, given $w \sim N(0, Q)$ and $v \sim N(0, R)$, choose sample points n^i for w and m^i for v. Each of the sigma points is propagated forward at each sample time using the nonlinear system model. The locations

[5] Note that this idea is fundamentally different than the idea of particle filtering. The sigma points are chosen deterministically, for example, as points on a selected covariance contour ellipse or a simplex. The particle filtering points are chosen by random sampling.

and weights of the transformed points then update the mean and covariance

$$z^i(k+1) = f(z^i(k), n^i(k))$$
$$\eta^i = h(z^i) + m^i \quad \text{all } i$$

From these we compute the forecast step

$$\hat{x}^- = \sum_i w^i z^i \qquad \hat{y}^- = \sum_i w^i \eta^i$$
$$P^- = \sum_i w^i (z^i - \hat{x}^-)(z^i - \hat{x}^-)'$$

After measurement, the EKF correction step is applied after first expressing this step in terms of the covariances of the innovation and state prediction. The output error is given as $\tilde{y} := y - \hat{y}^-$. We next rewrite the Kalman filter update as

$$\hat{x} = \hat{x}^- + L(y - \hat{y}^-)$$
$$L = \underbrace{\mathcal{E}((x - \hat{x}^-)\tilde{y}')}_{P^-C'} \underbrace{\mathcal{E}(\tilde{y}\tilde{y}')^{-1}}_{(R+CP^-C')^{-1}}$$
$$P = P^- - L \underbrace{\mathcal{E}((x - \hat{x}^-)\tilde{y}')'}_{CP^-}$$

in which we approximate the two expectations with the sigma-point samples

$$\mathcal{E}((x - \hat{x}^-)\tilde{y}') \approx \sum_i w^i (z^i - \hat{x}^-)(\eta^i - \hat{y}^-)'$$
$$\mathcal{E}(\tilde{y}\tilde{y}') \approx \sum_i w^i (\eta^i - \hat{y}^-)(\eta^i - \hat{y}^-)'$$

See Julier, Uhlmann, and Durrant-Whyte (2000); Julier and Uhlmann (2004a); van der Merwe, Doucet, de Freitas, and Wan (2000) for more details on the algorithm. An added benefit of the UKF approach is that the partial derivatives $\partial f(x,w)/\partial x, \partial h(x)/\partial x$ are not required. See also Nørgaard, Poulsen, and Ravn (2000) for other derivative-free nonlinear filters of comparable accuracy to the UKF. See Lefebvre, Bruyninckx, and De Schutter (2002); Julier and Uhlmann (2002) for an interpretation of the UKF as a use of statistical linear regression.

The UKF has been tested in a variety of simulation examples taken from different application fields including aircraft attitude estimation,

tracking and ballistics, and communication systems. In the chemical process control field, Romanenko and Castro (2004); Romanenko, Santos, and Afonso (2004) have compared the EKF and UKF on a strongly nonlinear exothermic chemical reactor and a pH system. The reactor has nonlinear dynamics and a linear measurement model, i.e., a subset of states is measured. In this case, the UKF performs significantly better than the EKF when the process noise is large. The pH system has linear dynamics but a strongly nonlinear measurement, i.e., the pH measurement. In this case, the authors show a modest improvement in the UKF over the EKF.

4.4.4 EKF, UKF, and MHE Comparison

One nice feature enjoyed by the EKF and UKF formulations is the recursive update equations. One-step recursions are computationally efficient, which may be critical in online applications with short sample times. The MHE computational burden may be reduced by shortening the length of the moving horizon, N. But use of short horizons may produce inaccurate estimates, especially after an unmodeled disturbance. This unfortunate behavior is the result of the system's nonlinearity. As we saw in Sections 1.4.3–1.4.4, for *linear systems*, the full information problem and the MHE problem are identical to a one-step recursion using the appropriate state penalty coming from the filtering Riccati equation. Losing the equivalence of a one-step recursion to full information or a finite moving horizon problem brings into question whether the one-step recursion can provide equivalent estimator performance. We show in the following example that the EKF and the UKF do not provide estimator performance comparable to MHE.

Example 4.39: EKF, UKF, and MHE performance comparison

Consider the following set of reversible reactions taking place in a well-stirred, isothermal, gas-phase batch reactor

$$\mathrm{A} \underset{k_{-1}}{\overset{k_1}{\rightleftharpoons}} \mathrm{B} + \mathrm{C} \qquad 2\mathrm{B} \underset{k_{-2}}{\overset{k_2}{\rightleftharpoons}} \mathrm{C}$$

The material balance for the reactor is

$$\frac{d}{dt}\begin{bmatrix} c_A \\ c_B \\ c_C \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 1 & -2 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} k_1 c_A - k_{-1} c_B c_C \\ k_2 c_B^2 - k_{-2} c_C \end{bmatrix}$$

$$\frac{dx}{dt} = f_c(x)$$

with states and measurement

$$x = \begin{bmatrix} c_A & c_B & c_C \end{bmatrix}' \qquad y = RT \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} x$$

in which c_j denotes the concentration of species j in mol/L, R is the gas constant, and T is the reactor temperature in K. The measurement is the reactor pressure in atm, and we use the ideal gas law to model the pressure. The model is nonlinear because of the two second-order reactions. We model the system plus disturbances with the following discrete time model

$$\begin{aligned} x^+ &= f(x) + w \\ y &= Cx + v \end{aligned}$$

in which f is the solution of the ordinary differential equations (ODEs) over the sample time, Δ, i.e., if $s(t, x_0)$ is the solution of $dx/dt = f_c(x)$ with initial condition $x(0) = x_0$ at $t = 0$, then $f(x) = s(\Delta, x)$. The state and measurement disturbances, w and v, are assumed to be zero-mean independent normals with constant covariances Q and R. The following parameter values are used in the simulations

$$RT = 32.84 \,\text{mol} \cdot \text{atm/L}$$

$$\Delta = 0.25 \quad k_1 = 0.5 \quad k_{-1} = 0.05 \quad k_2 = 0.2 \quad k_{-2} = 0.01$$

$$C = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} RT \qquad P(0) = (0.5)^2 I \qquad Q = (0.001)^2 I \qquad R = (0.25)^2$$

$$\overline{x}_0 = \begin{bmatrix} 1 \\ 0 \\ 4 \end{bmatrix} \qquad x(0) = \begin{bmatrix} 0.5 \\ 0.05 \\ 0 \end{bmatrix}$$

The prior density for the initial state, $N(\overline{x}_0, P(0))$, is deliberately chosen to poorly represent the actual initial state to model a large initial disturbance to the system. We wish to examine how the different estimators recover from this large unmodeled disturbance.

Solution

Figure 4.3 (top) shows a typical EKF performance for these conditions. Note that the EKF cannot reconstruct the state for this system and that the estimates converge to incorrect steady states displaying negative concentrations of A and B. For some realizations of the noise sequences, the EKF may converge to the correct steady state. Even for these cases,

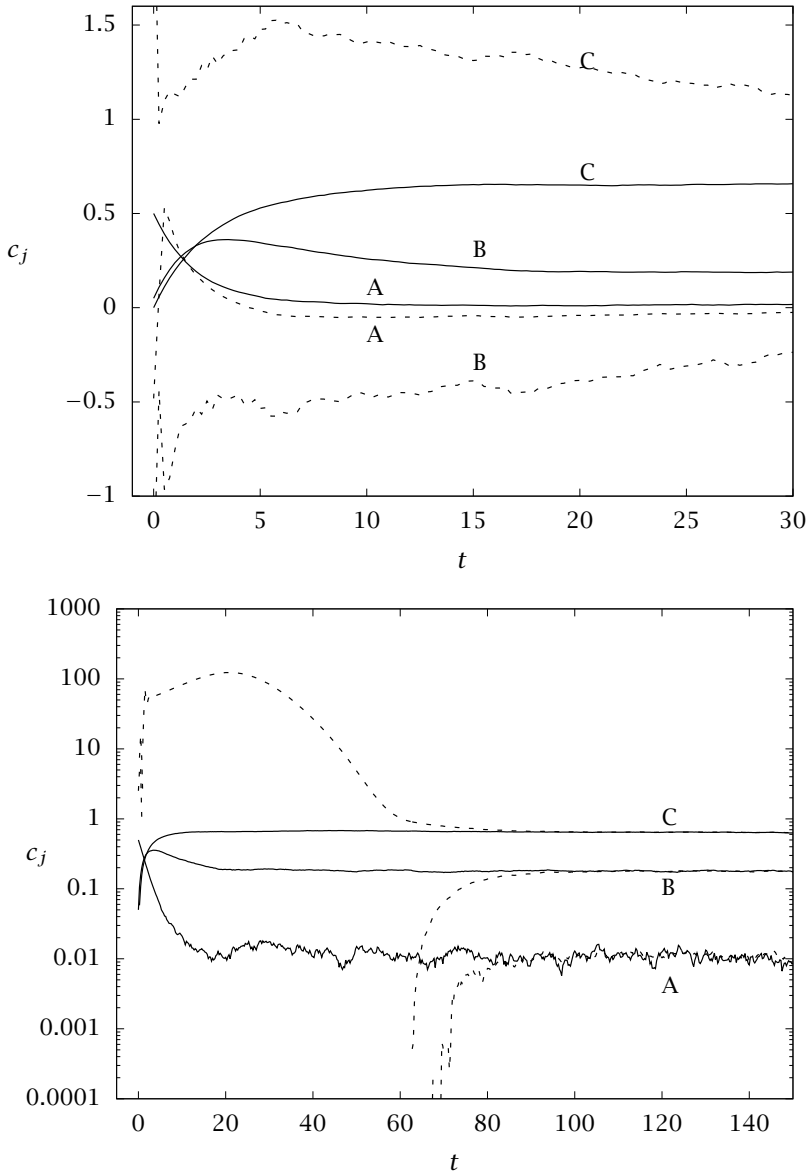


Figure 4.3: Evolution of the state (solid line) and EKF state estimate (dashed line). Top plot shows negative concentration estimates with the standard EKF. Bottom plot shows large estimate errors and slow convergence with the clipped EKF.

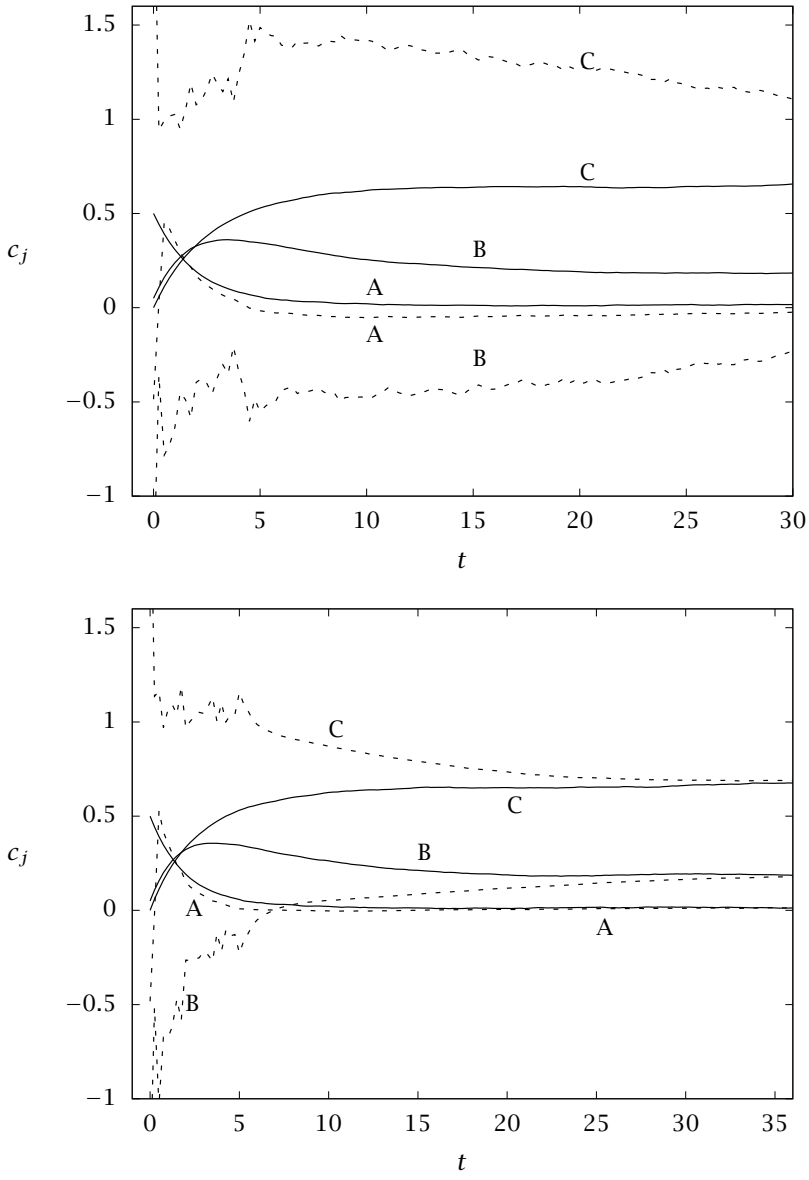


Figure 4.4: Evolution of the state (solid line) and UKF state estimate (dashed line). Top plot shows negative concentration estimates with the standard UKF. Bottom plot shows similar problems even if constraint scaling is applied.

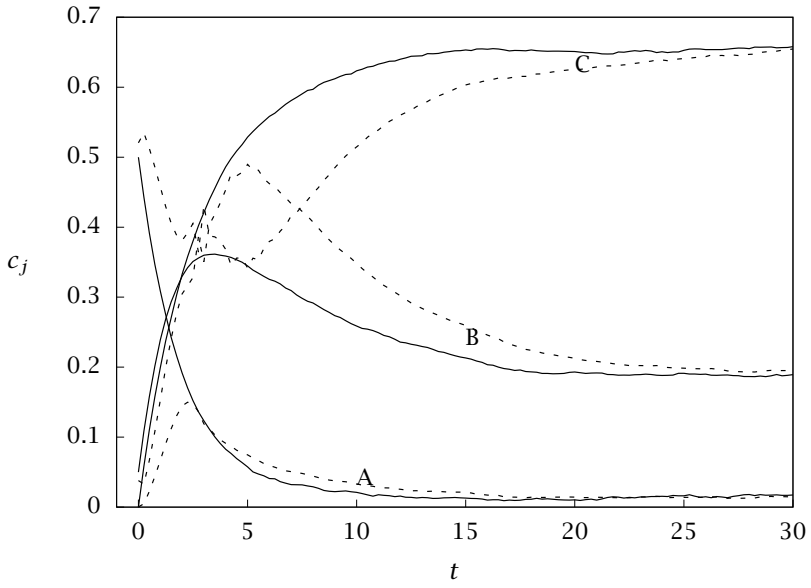


Figure 4.5: Evolution of the state (solid line) and MHE state estimate (dashed line).

however, negative concentration estimates still occur during the transient, which correspond to physically impossible states. Figure 4.3 (bottom) presents typical results for the clipped EKF, in which negative values of the filtered estimates are set to zero. Note that although the estimates converge to the system states, this estimator gives pressure estimates that are two orders of magnitude larger than the measured pressure before convergence is achieved.

The standard UKF achieves results similar to the EKF as shown in Figure 4.4 (top). Vachhani, Narasimhan, and Rengaswamy (2006) have proposed a modification to the UKF to handle constrained systems. In this approach, the sigma points that violate the constraints are scaled back to the feasible region boundaries and the sigma-point weights are modified accordingly. If this constrained version of the UKF is applied to this case study, the estimates do not significantly improve as shown in Figure 4.4 (bottom). The UKF formulations used here are based on the algorithm presented by Vachhani et al. (2006, Sections 3 and 4) with the tuning parameter κ set to $\kappa = 1$. Adjusting this parameter

using other suggestions from the literature (Julier and Uhlmann, 1997; Qu and Hahn, 2009; Kandepu, Imsland, and Foss, 2008) and trial and error, does not substantially improve the UKF estimator performance.

Better performance is obtained in this example if the sigma points that violate the constraints are simply saturated rather than rescaled to the feasible region boundaries. But, this form of clipping still does not prevent the occurrence of negative concentrations in this example. Negative concentration estimates are not avoided by either scaling or clipping of the sigma points. As a solution to this problem, the use of constrained optimization for the sigma points is proposed (Vachhani et al., 2006; Teixeira, Tôrres, Aguirre, and Bernstein, 2008). If one is willing to perform online optimization, however, MHE with a short horizon is likely to provide more accurate estimates at similar computational cost compared to approaches based on optimizing the locations of the sigma points.

The authors have only recently become aware of yet another approach to handling constraints in the UKF that does work well on this example (Kolås, Foss, and Schei, 2009). It remains to be seen whether further examples can be constructed that this approach cannot address.

Finally, Figure 4.5 presents typical results of applying constrained MHE to this example. For this simulation we choose $N = 10$ and the smoothing update for the arrival cost approximation. Note that MHE recovers well from the poor initial prior. Comparable performance is obtained if the filtering update is used instead of the smoothing update to approximate the arrival cost. The MHE estimates are also insensitive to the choice of horizon length N for this example. □

The EKF, UKF, and all one-step recursive estimation methods, suffer from the "short horizon syndrome" by *design*. One can try to reduce the harmful effects of a short horizon through tuning various other parameters in the estimator, but the basic problem remains. Large initial state errors lead to inaccurate estimation and potential estimator divergence. The one-step recursions such as the EKF and UKF can be viewed as one extreme in the choice between speed and accuracy in that only a single measurement is considered at each sample. That is similar to an MHE problem in which the user chooses $N = 1$. Situations in which $N = 1$ lead to poor MHE performance often lead to unreliable EKF and UKF performance as well.

4.5 On combining MHE and MPC

Estimating the state of a system is an interesting problem in its own right, with many important applications having no connection to feedback control. But in some applications the goal of state estimation is indeed to provide a state feedback controller with a good estimate of the system state based on the available measurements. We close this chapter with a look at the properties of such control systems consisting of a moving horizon estimator that provides the state estimate to a model predictive controller.

What's desirable. Consider the evolution of the system $x^+ = f(x, u, w)$ and its measurement $y = h(x) + v$ when taking control using MPC based on the state estimate

$$x^+ = f(x, \kappa_N(\hat{x}), w) \qquad y = h(x) + v$$

with $f : \mathbb{Z} \times \mathbb{W} \to \mathbb{R}^n$, $h : \mathbb{R}^n \to \mathbb{R}^p$, in which $w \in \mathbb{W}$ is the process disturbance and $v \in \mathbb{V}$ is the measurement disturbance, $u = \kappa_N(\hat{x})$ is the control from the MPC regulator, and $\hat{x}$ is generated by the MHE estimator. We assume, as we have through the text, that $f(\cdot)$ and $h(\cdot)$ are continuous functions. Again we denote estimate error by $e := x - \hat{x}$, which gives for the state evolution

$$x^+ = f(x, \kappa_N(x - e), w) \qquad y = h(x) + v \tag{4.34}$$

The obvious difficulty with analyzing the effect of estimate error is the coupling of estimation and control. Unlike the problem studied earlier in the chapter, where $x^+ = f(x, w)$, we now have estimate error also influencing state evolution. This coupling precludes obtaining the simple bounds on $|e(k)|$ in terms of $(e(0), \mathbf{w}, \mathbf{v})$ as we did in the previous sections.

What's possible. Here we lower our sights from the analysis of the fully coupled problem and consider only the effect of *bounded estimate error* on the combined estimation/regulation problem. To make this precise, consider the following definition of an incrementally, uniformly input/output-to-state stable (i-UIOSS) system.

Definition 4.40 (i-UIOSS)**.** The system

$$x^+ = f(x, u, w) \qquad y = h(x)$$

is *incrementally uniformly input/output-to-state stable* (i-UIOSS) if there exist functions $\alpha(\cdot) \in \mathcal{KL}$ and $\gamma_w(\cdot)$, $\gamma_v(\cdot) \in \mathcal{K}$ such that for any

two initial states z_1 and z_2, any input sequence $\mathbf{u}$, and any two disturbance sequences $\mathbf{w}_1$ and $\mathbf{w}_2$ generating state sequences $\mathbf{x}_1(z_1, \mathbf{u}, \mathbf{w}_1)$ and $\mathbf{x}_2(z_2, \mathbf{u}, \mathbf{w}_2)$, the following holds for all $k \in \mathbb{I}_{\geq 0}$

$$|x(k; z_1, \mathbf{u}, \mathbf{w}_1) - x(k; z_2, \mathbf{u}, \mathbf{w}_2)| \leq \alpha(|z_1 - z_2|, k) \oplus \gamma_w(\|\mathbf{w}_1 - \mathbf{w}_2\|_{0:k-1}) \oplus \gamma_v(\|h(\mathbf{x}_1) - h(\mathbf{x}_2)\|_{0:k-1}) \quad (4.35)$$

Notice that the bound is uniform in the sense that it is independent of the input sequence $\mathbf{u}$ generated by a controller. See Cai and Teel (2008, Definition 3.4) for similar definitions. Exercise 4.15 discusses how to establish that a detectable linear system $x^+ = Ax + Bu + Gw$, $y = Cx$ is i-UIOSS.

Given this strong form of detectability, we assume that we can derive an error bound of the form

Assumption 4.41 (Bounded estimate error)**.** There exists $\delta > 0$ and $\beta(\cdot) \in \mathcal{KL}$ and $\sigma(\cdot) \in \mathcal{K}$ such that for all $\|(\mathbf{w}, \mathbf{v})\| \leq \delta$ and for all $k \geq 0$ the following holds

$$|e(k)| \leq \beta(|e(0)|, k) + \sigma(\|(\mathbf{w}, \mathbf{v})\|)$$

Next we note that the evolution of the state in the form of (4.34) is not a compelling starting point for analysis because the estimate error perturbation appears inside a possibly discontinuous function, $\kappa_N(\cdot)$ (recall Example 2.8). Therefore, as in (Roset, Heemels, Lazar, and Nijmeijer, 2008), we instead express the equivalent evolution, but in terms of the state estimate as

$$\hat{x}^+ = f(\hat{x} + e, \kappa_N(\hat{x}), w) - e^+ \qquad y = h(\hat{x} + e) + v$$

which is more convenient because the estimate error appears inside continuous functions $f(\cdot)$ and $h(\cdot)$.

We require that the system not leave an invariant set due to the disturbance.

Definition 4.42 (Robust positive invariance)**.** A set $\mathcal{X} \subseteq \mathbb{R}^n$ is robustly positive invariant with respect to a difference inclusion $x^+ \in f(x, d)$ if there exists some $\delta > 0$ such that $f(x, d) \subseteq \mathcal{X}$ for all $x \in \mathcal{X}$ and all disturbance sequences $\mathbf{d}$ satisfying $\|\mathbf{d}\| \leq \delta$.

So, we define robust asymptotic stability as input-to-state stability on a robust positive invariant set.

Definition 4.43 (Robust asymptotic stability)**.** The origin of a perturbed difference inclusion $x^+ \in f(x,d)$ is RAS in $\mathcal{X}$ if there exists some $\delta > 0$ such that for all disturbance sequences $\mathbf{d}$ satisfying $\|\mathbf{d}\| \leq \delta$ we have both that $\mathcal{X}$ is robustly positive invariant and that there exist $\beta(\cdot) \in \mathcal{KL}$ and $\gamma(\cdot) \in \mathcal{K}$ such that for each $x \in \mathcal{X}$, we have for all $k \in \mathbb{I}_{\geq 0}$ that all solutions $\phi(k;x,\mathbf{d})$ satisfy

$$\left|\phi(k;x,\mathbf{d})\right| \leq \beta(|x|,k) + \gamma(\|\mathbf{d}\|) \tag{4.36}$$

To establish input-to-state stability, we define an ISS Lyapunov function for a difference inclusion, similar to an ISS Lyapunov function defined in Jiang and Wang (2001); Lazar, Heemels, and Teel (2013). See also Definition B.45 in Appendix B.

Definition 4.44 (ISS Lyapunov function)**.** $V(\cdot)$ is an ISS Lyapunov function in the robust positive invariant set $\mathcal{X}$ for the difference inclusion $x^+ \in f(x,d)$ if there exists some $\delta > 0$, functions $\alpha_1(\cdot), \alpha_2(\cdot)$, $\alpha_3(\cdot) \in \mathcal{K}_\infty$, and function $\sigma(\cdot) \in \mathcal{K}$ such that for all $x \in \mathcal{X}$ and $\|\mathbf{d}\| \leq \delta$

$$\alpha_1(|x|) \leq V(x) \leq \alpha_2(|x|) \tag{4.37}$$

$$\sup_{x^+ \in f(x,d)} V(x^+) \leq V(x) - \alpha_3(|x|) + \sigma(|d|) \tag{4.38}$$

The value of an ISS Lyapunov function is analogous to having a Lyapunov function in standard stability analysis: it allows us to conclude input-to-state stability and therefore *robust* asymptotic stability. The following result is therefore highly useful in robustness analysis.

Proposition 4.45 (ISS Lyapunov stability theorem)**.** *If a difference inclusion $x^+ \in f(x,d)$ admits an ISS Lyapunov function in a robust positive invariant set $\mathcal{X}$ for all $\|\mathbf{d}\| \leq \delta$ for some $\delta > 0$, then the origin is RAS in $\mathcal{X}$ for all $\|\mathbf{d}\| \leq \delta$.*

The proof of this proposition follows Jiang and Wang (2001) as modified for a difference inclusion on a robust positive invariant set in Allan, Bates, Risbeck, and Rawlings (2017, Proposition 19).

Combined MHE/MPC is RAS. Our strategy now is to establish that $V_N^0(x)$ is an ISS Lyapunov function for the combined MHE/MPC system subject to process and measurement disturbances on a robust positive invariant set. We have already established the upper and lower bounding inequalities

$$\alpha_1(|x|) \leq V_N^0(x) \leq \alpha_2(|x|)$$

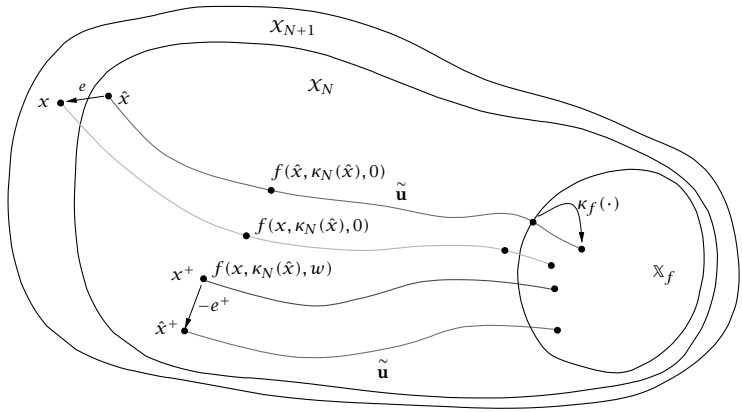


Figure 4.6: Although the nominal trajectory from $\hat{x}$ may terminate on the boundary of $\mathbb{X}_f$, the three perturbed trajectories, including the one from $\hat{x}^+$, terminate in $\mathbb{X}_f$. After Allan et al. (2017).

So we require only

$$\sup_{x^+ \in f(x,d)} V_N^0(x^+) \leq V_N^0(x) - \alpha_3(|x|) + \sigma(|d|)$$

with disturbance d defined here as $d := (e, w, e^+)$. That plus robust positive invariance establishes that the controlled system is RAS.

Figure 4.6 gives the picture of the argument we are going to make. We have that $\hat{x}^+ = f(\hat{x} + e, \kappa_N(\hat{x}), w) - e^+$ and $x^+ = f(x, \kappa_N(\hat{x}), w)$. We create the standard candidate input sequence by dropping the first input and applying the terminal control law to the terminal state, i.e., $\tilde{\mathbf{u}} = \left(u^0(1; \hat{x}), \ldots, u^0(N-1; \hat{x}), \kappa_f(x^0(N; \hat{x}))\right)$. We then compute difference in cost of trajectories starting at $f(\hat{x}, \kappa_N(\hat{x}), 0)$ and $\hat{x}^+$ using the same input sequence $\tilde{\mathbf{u}}$. We choose the terminal region to be a sublevel set of the terminal cost, $\mathbb{X}_f = \text{lev}_\tau V_f$, $\tau > 0$. Note that $\tilde{\mathbf{u}}$ is feasible for both initial states, i.e., both trajectories terminate in $\mathbb{X}_f$, if $|(e, w, e^+)|$ is small enough.

As in Chapter 3, we make use of Proposition 3.4 to bound the size of the change to a continuous function (Allan et al., 2017, Proposition 20). Since $V_N(x, \mathbf{u})$ is continuous, Proposition 3.4 gives

$$|V_N(\hat{x}^+, \tilde{\mathbf{u}}) - V_N(f(\hat{x}, \kappa_N(\hat{x}), 0), \tilde{\mathbf{u}})| \leq \sigma_V(|\hat{x}^+ - f(\hat{x}, \kappa_N(\hat{x}), 0)|)$$

with $\sigma_V(\cdot) \in \mathcal{K}$. Note that we are *not* using the possibly discontinuous $V_N^0(x)$ here. Since $f(x,u,w)$ is also continuous

$$\begin{aligned}|\hat{x}^+ - f(\hat{x},\kappa_N(\hat{x}),0)| &= |f(\hat{x}+e,\kappa_N(\hat{x}),w) - e^+ - f(\hat{x},\kappa_N(\hat{x}),0)| \\ &\le |f(\hat{x}+e,\kappa_N(\hat{x}),w) - f(\hat{x},\kappa_N(\hat{x}),0)| + |e^+| \\ &\le \sigma_f(|(e,w)|) + |e^+| \\ &\le \widetilde{\sigma}_f(|d|)\end{aligned}$$

with $d := (e,w,e^+)$ and $\widetilde{\sigma}_f(\cdot) \in \mathcal{K}$. Therefore

$$\begin{aligned}|V_N(\hat{x}^+,\tilde{\mathbf{u}}) - V_N(f(\hat{x},\kappa_N(\hat{x}),0),\tilde{\mathbf{u}})| &\le \sigma_V \circ \widetilde{\sigma}_f(|d|) := \sigma(|d|) \\ V_N(\hat{x}^+,\tilde{\mathbf{u}}) &\le V_N(f(\hat{x},\kappa_N(\hat{x}),0),\tilde{\mathbf{u}}) + \sigma(|d|)\end{aligned}$$

with $\sigma(\cdot) \in \mathcal{K}$. Note that for the candidate sequence, $V_N(f(\hat{x},\kappa_N(\hat{x}),0),\tilde{\mathbf{u}}) \le V_N^0(\hat{x}) - \ell(\hat{x},\kappa_N(\hat{x}))$, so we have that

$$V_N(f(\hat{x},\kappa_N(\hat{x}),0),\tilde{\mathbf{u}}) \le V_N^0(\hat{x}) - \alpha_1(|\hat{x}|)$$

since $\alpha_1(|x|) \le \ell(x,\kappa_N(x))$ for all x. Therefore, we finally have

$$\begin{aligned}V_N(\hat{x}^+,\tilde{\mathbf{u}}) &\le V_N^0(\hat{x}) - \alpha_1(|\hat{x}|) + \sigma(|d|) \\ V_N^0(\hat{x}^+) &\le V_N^0(\hat{x}) - \alpha_1(|\hat{x}|) + \sigma(|d|)\end{aligned} \tag{4.39}$$

and we have established that $V_N^0(\cdot)$ satisfies the inequality of an ISS-Lyapunov function. This analysis leads to the following main result.

Theorem 4.46 (Combined MHE/MPC is RAS)**.** *For the MPC regulator, let the standard Assumptions 2.2, 2.3, and 2.14 hold, and choose* $\mathbb{X}_f = \operatorname{lev}_\tau V_f$ *for some* $\tau > 0$*. For the moving horizon estimator, let Assumption 4.41 hold. Then for every* $\rho > 0$ *there exists* $\delta > 0$ *such that if* $\|\mathbf{d}\| \le \delta$*, the origin is RAS for the system* $\hat{x}^+ = f(\hat{x}+e,\kappa_N(\hat{x}),w) - e^+$*,* $y = h(\hat{x}+e) + v$*, in the set* $\mathcal{X}_\rho = \operatorname{lev}_\rho V_f$*.*

A complete proof of this theorem, for the more general case of *suboptimal* MPC, is given in Allan et al. (2017, Theorem 21). The proof proceeds by first showing that $\mathcal{X}_\rho$ is robustly positive invariant for all $\rho > 0$. That argument is similar to the one presented in Chapter 3 before Proposition 3.5. The proof then establishes that inequality (4.39) holds for all $\hat{x} \in \mathcal{X}_\rho$. Proposition 4.45 is then invoked to establish that the origin is RAS.

Notice that neither $V_N^0(\cdot)$ nor $\kappa_N(\cdot)$ need be continuous for this combination of MHE and MPC to be inherently robust. Since $x = \hat{x} + e$, Theorem 4.46 also gives robust asymptotic stability of the evolution of x in addition to $\hat{x}$ for the closed-loop system with bounded disturbances.

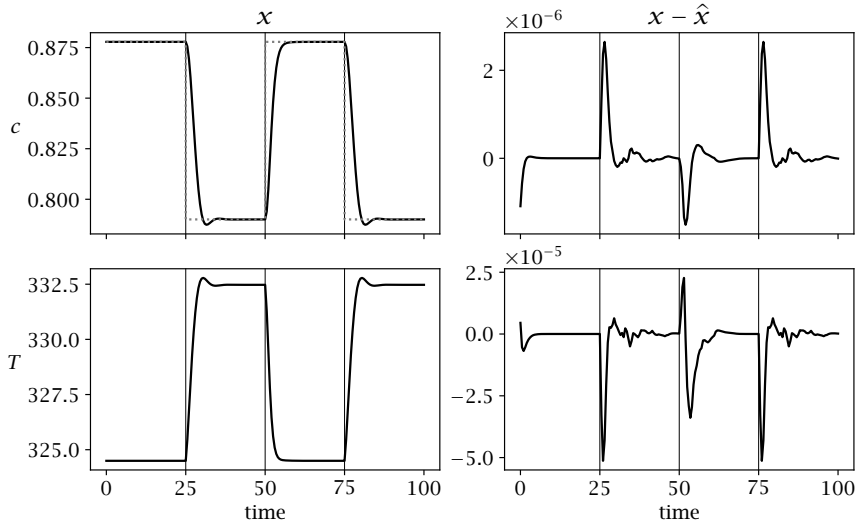


Figure 4.7: Closed-loop performance of combined nonlinear MHE/MPC with no disturbances. First column shows system states, and second column shows estimation error. Dashed line shows concentration setpoint. Vertical lines indicate times of setpoint changes.

Example 4.47: Combined MHE/MPC

Consider the nonlinear reactor system from Example 1.11 with sample time $\Delta = 0.5\,\text{min}$ and height h and inlet flow F fixed to their steady-state values. The resulting system has two states (concentration c and temperature T) and one input (cooling temperature T_c). The only measured output is temperature, which means the reactor concentration must be estimated via MHE.

To illustrate the performance of combined MHE/MPC, closed-loop control to a changing setpoint is simulated. Figure 4.7 shows the changing states x and estimate errors $x - \hat{x}$. Note that each setpoint change leads to a temporary increase in estimate error, which eventually decays back to zero. Note that zero prior weighting is used in the MHE formulation.

To illustrate the response to disturbances, the simulation is repeated for varying disturbance sizes. The system itself is subject to a disturbance w, which adds to the evolution of concentration, while the tem-

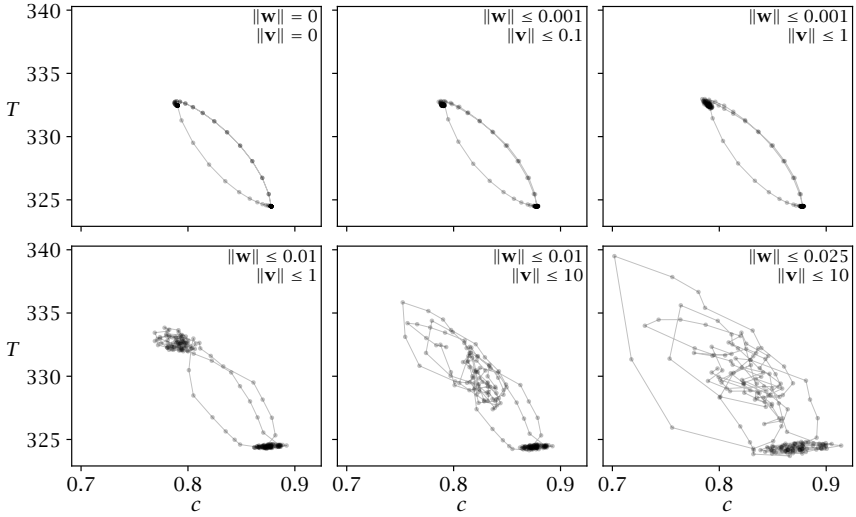


Figure 4.8: Closed-loop performance of combined nonlinear MHE/MPC for varying disturbance size. The system is controlled between two steady states.

perature measurement is subject to noise v. Figure 4.8 shows a phase plot of system evolution subject to the same setpoint changes as before. As the disturbances become larger, the system deviates further from its setpoint. Note that the same MHE objective function (with zero prior weight) is used for all cases. □

4.6 Notes

State estimation is a fundamental topic appearing in many branches of science and engineering, and has a large literature. A nice and brief annotated bibliography describing the early contributions to optimal state estimation of the *linear Gaussian* system is provided by Åström (1970, pp. 252-255). Kailath (1974) provides a comprehensive and historical review of *linear* filtering theory including the historical development of Wiener-Kolmogorov theory for filtering and prediction that preceded Kalman filtering (Wiener, 1949; Kolmogorov, 1941).

Jazwinski (1970) provides an early and comprehensive treatment of the optimal stochastic state estimation problem for linear and *nonlin-*

ear systems. As mentioned in Section 4.2.3, Jazwinski (1970) follows Deyst and Price (1968) and proposes $V(k,x) = (1/2)x'P(k)^{-1}x$ as a Lyapunov function candidate for the linear controllable and observable time-varying system. Note that the estimate error dynamic system is time varying even if the model is time invariant because the optimal estimator gains are time varying. This choice of Lyapunov function has been used to establish estimator stability in many subsequent textbooks (Stengel, 1994, pp.474-475). The most complete treatment of the linear problem in the literature seems to be (Anderson and Moore, 1981), which assumes uniform stabilizability and detectability for the time-varying system and establishes exponential stability. Kailath (1974, p.152) remarks that the known proofs that the optimal filter is stable "are somewhat difficult, and it is significant that only a small fraction of the vast literature on the Kalman filter deals with this problem."

For establishing stability of the *steady-state* optimal linear estimator, simpler arguments suffice because the estimate error equation is time invariant. Establishing duality with the optimal regulator is a favorite technique for establishing estimator stability in this case. See, for example, Kwakernaak and Sivan (1972, Theorem 4.11) for a general steady-state stability theorem for the linear Gaussian case.

Many of the full information and MHE results in this chapter are motivated by early results in Rao (2000) and Rao, Rawlings, and Mayne (2003). The full information analysis given here is more general because (i) we assume nonlinear detectability rather than nonlinear observability, and (ii) we establish asymptotic stability under process and measurement disturbances, which were neglected in previous analysis.

Muske, Rawlings, and Lee (1993) and Meadows, Muske, and Rawlings (1993) apparently were the first to use the increasing property of the optimal cost to establish classical (not KL) asymptotic stability for full information estimation for linear models with constraints. Robertson and Lee (2002) present the interesting statistical interpretation of MHE for the constrained linear system. Michalska and Mayne (1995) establish stability of moving horizon estimation with zero prior weighting for the continuous time nonlinear system. Alessandri, Baglietto, and Battistelli (2008) also provide a stability analysis of MHE with an observability assumption and quadratic stage cost.

Rawlings and Ji (2012) streamlined the presentation of the full information problem for the case of convergent disturbances, and pointed to MHE of bounded disturbances, and suboptimal MHE as two signifi-

cant open research problems. Next Ji, Rawlings, Hu, Wynn, and Diehl (2016); Hu, Xie, and You (2015) provided the first analysis of full information estimation for bounded disturbances by introducing a max term in the estimation objective function, and assuming stronger forms of the i-IOSS detectability condition. This reformulation did provide RAS of full information estimation with bounded disturbances, but had the unfortunate side effect of removing convergent estimate error for convergent disturbances.

In a major step forward, Müller (2017) examined MHE with bounded disturbances for similarly restrictive i-IOSS conditions, and established bounds on arrival cost penalty and horizon length that provide both RAS for bounded disturbances and convergence of estimate error for convergent disturbances. Hu (2017) generalized the detectability conditions in Ji et al. (2016) and treated both full information with the max term and MHE estimation. At this stage of development, all the bounds for robust stability became *worse* with increasing horizon length, which seems problematic since the use of more measurements should *improve* estimation. In another significant step, Knüfer and Müller (2018) next introduced a fading memory formulation of FIE and MHE for exponentially i-IOSS systems whose bounds improved with horizon length. But this formulation required that the stage cost satisfy the triangle inequality, which excludes the quadratic penalty commonly used in estimation, especially for linear systems.

As described in detail throughout the chapter, Allan (2020) introduced explicit stabilizability assumptions into the analysis and established a converse theorem for i-IOSS. He then showed for general stage costs that FIE is RGAS for (asymptotic) i-IOSS systems, thus removing the exponential part of the assumption, and that MHE is RGES for exponentially i-IOSS systems. As mentioned in the chapter, whether MHE is RGAS for (asymptotic) i-IOSS systems remains an open question. Finally, numerous application papers using MHE have appeared in the last several years indicating a growing interest in this approach to state estimation.

For the case of output feedback, there are of course alternatives to simply combining independently designed MHE estimators and MPC regulators as briefly analyzed in Section 4.5. Recently Copp and Hespanha (2017) propose solving instead a single min-max optimization for simultaneous estimation and control. Because of the excellent resultant closed-loop properties, this class of approaches certainly warrants further attention and development.

4.7 Exercises

Exercise 4.1: Input-to-state stability and convergence

Assume the nonlinear system

$$x^+ = f(x,u)$$

is input-to-state stable (ISS) so that for all $x_0 \in \mathbb{R}^n$, input sequences $\mathbf{u}$, and $k \geq 0$

$$|x(k;x_0,\mathbf{u})| \leq \beta(|x_0|,k) + \gamma(\|\mathbf{u}\|)$$

in which $x(k;x_0,\mathbf{u})$ is the solution to the system equation at time k starting at state x_0 using input sequence $\mathbf{u}$, and $\gamma \in \mathcal{K}$ and $\beta \in \mathcal{KL}$.

(a) Show that the ISS property also implies

$$|x(k;x_0,\mathbf{u})| \leq \beta(|x_0|,k) + \gamma(\|\mathbf{u}\|_{0:k-1})$$

in which $\|\mathbf{u}\|_{a:b} = \max_{a\leq j\leq b} |u(j)|$.

(b) Show that the ISS property implies the "converging-input converging-state" property (Jiang and Wang, 2001), (Sontag, 1998, p. 330), i.e., show that if the system is ISS, then $u(k) \to 0$ implies $x(k) \to 0$.

Exercise 4.2: Output-to-state stability and convergence

Assume the nonlinear system

$$x^+ = f(x) \qquad y = h(x)$$

is output-to-state stable (OSS) so that for all $x_0 \in \mathbb{R}^n$ and $k \geq 0$

$$|x(k;x_0)| \leq \beta(|x_0|,k) + \gamma(\|\mathbf{y}\|_{0:k})$$

in which $x(k;x_0)$ is the solution to the system equation at time k starting at state x_0, and $\gamma \in \mathcal{K}$ and $\beta \in \mathcal{KL}$.

Show that the OSS property implies the "converging-output converging-state" property (Sontag and Wang, 1997, p. 281) i.e., show that if the system is OSS, then $y(k) \to 0$ implies $x(k) \to 0$.

Exercise 4.3: i-IOSS and convergence

Establish that if system

$$x^+ = f(x,w) \qquad y = g(x)$$

is i-IOSS, and $w_1(k) \to w_2(k)$ and $y_1(k) \to y_2(k)$ as $k \to \infty$, then

$$x(k;z_1,\mathbf{w}_1) \to x(k;z_2,\mathbf{w}_2) \quad \text{as } k \to \infty \quad \text{for all } z_1, z_2$$

Exercise 4.4: Observability and detectability of linear time-invariant systems and OSS

Consider the linear time-invariant system

$$x^+ = Ax \qquad y = Cx$$

(a) Show that if the system is observable, then the system is OSS.

(b) Show that the system is detectable if and only if the system is OSS.

Exercise 4.5: Observability and detectability of linear time-invariant system and IOSS

Consider the linear time-invariant system with input

$$x^+ = Ax + Gw \qquad y = Cx$$

(a) Show that if the system is observable, then the system is IOSS.

(b) Show that the system is detectable if and only if the system is IOSS.

Exercise 4.6: Max or sum?

To facilitate complicated arguments involving $\mathcal{K}$ and $\mathcal{KL}$ functions, it is often convenient to interchange sum and max operations. First some suggestive notation: let the max operator over scalars be denoted with the $\oplus$ symbol so that

$$a \oplus b := \max(a, b)$$

(a) Show that the $\oplus$ operator is commutative and associative, i.e., $a \oplus b = b \oplus a$ and $(a \oplus b) \oplus c = a \oplus (b \oplus c)$ for all a, b, c, so that the following operation is well defined and the order of operation is inconsequential

$$a_1 \oplus a_2 \oplus a_3 \cdots \oplus a_n := \bigoplus_{i=1}^{n} a_i$$

(b) Find scalars d and e such that for all $a, b \geq 0$, the following holds

$$d(a + b) \leq a \oplus b \leq e(a + b)$$

(c) Find scalars $\overline{d}$ and $\overline{e}$ such that for all $a, b \geq 0$, the following holds

$$\overline{d}(a \oplus b) \leq a + b \leq \overline{e}(a \oplus b)$$

(d) Generalize the previous result to the n term sum; find $d_n, e_n, \overline{d}_n, \overline{e}_n$ such that the following holds for all $a_i \geq 0$, $i = 1, 2, \ldots, n$

$$d_n \sum_{i=1}^{n} a_i \leq \bigoplus_{i=1}^{n} a_i \leq e_n \sum_{i=1}^{n} a_i$$
$$\overline{d}_n \bigoplus_{i=1}^{n} a_i \leq \sum_{i=1}^{n} a_i \leq \overline{e}_n \bigoplus_{i=1}^{n} a_i$$

(e) Show that establishing convergence (divergence) in sum or max is equivalent, i.e., consider a time sequence $(s(k))_{k\geq 0}$

$$s(k) = \sum_{i=1}^{n} a_i(k) \qquad \overline{s}(k) = \bigoplus_{i=1}^{n} a_i(k)$$

Show that

$$\lim_{k\to\infty} s(k) = 0 \ (\infty) \text{ if and only if } \lim_{k\to\infty} \overline{s}(k) = 0 \ (\infty)$$

Exercise 4.7: Where did my constants go?

Once $\mathcal{K}$ and $\mathcal{KL}$ functions appear, we may save some algebra by switching from the sum to the max.

In the following, let $\gamma(\cdot)$ be any $\mathcal{K}$ function and $a_i \in \mathbb{R}_{\geq 0}, i \in \mathbb{I}_{1:n}$.

(a) If you choose to work with sum, derive the following bounding inequalities (Rawlings and Ji, 2012)

$$\gamma(a_1 + a_2 + \cdots + a_n) \leq \gamma(na_1) + \gamma(na_2) + \cdots + \gamma(na_n)$$
$$\gamma(a_1 + a_2 + \cdots + a_n) \geq \frac{1}{n}\Big(\gamma(a_1) + \gamma(a_2) + \cdots + \gamma(a_n)\Big)$$

(b) If you choose to work with max instead, derive instead the following simpler result

$$\gamma(a_1 \oplus a_2 \oplus \cdots \oplus a_n) = \gamma(a_1) \oplus \gamma(a_2) \oplus \cdots \oplus \gamma(a_n)$$

Notice that you have an equality rather than an inequality, which leads to tighter bounds.

Exercise 4.8: Linear systems and incremental stability

Show that for a linear time-invariant system, i-ISS (i-OSS, i-IOSS) is equivalent to ISS (OSS, IOSS).

Exercise 4.9: Nonlinear observability and Lipschitz continuity implies i-OSS

Consider the following definition of observability for nonlinear systems in which f and h are Lipschitz continuous. A system

$$x^+ = f(x) \qquad y = h(x)$$

is observable if there exists $N_o \in \mathbb{I}_{\geq 1}$ and $\mathcal{K}$ function γ such that

$$\sum_{k=0}^{N_o-1} \big|\gamma(k; x_1) - \gamma(k; x_2)\big| \geq \gamma(|x_1 - x_2|) \tag{4.40}$$

holds for all $x_1, x_2 \in \mathbb{R}^n$. This definition was used by Rao et al. (2003) in showing stability of nonlinear MHE to initial condition error under zero state and measurement disturbances.

(a) Show that this form of nonlinear observability implies i-OSS.

(b) Show that i-OSS does not imply this form of nonlinear observability and, therefore, i-OSS is a weaker assumption.

The i-OSS concept generalizes the linear system concept of detectability to nonlinear systems.

Exercise 4.10: Equivalance of detectability and IOSS for continuous time, linear, time-invariant system

Consider the continuous time, linear, time-invariant system with input

$$\dot{x} = Ax + Bu \qquad y = Cx$$

Show that the system is detectable if and only if the system is IOSS.

Exercise 4.11: Observable, FSO, and detectable for linear systems

Consider the linear time-invariant system

$$x^+ = Ax \qquad y = Cx$$

and its observability canonical form. What conditions must the system satisfy to be

(a) observable?

(b) final-state observable (FSO)?

(c) detectable?

Exercise 4.12: Exponential detectability and compatibility of stage cost

We commented in the text that working with exponential detectability lessens the requirement for stage-cost compatibility in Assumption 4.11 that is necessary with (asymptotic) detectability. To see why, consider the noise-free case and assume system 4.1 is exponentially i-IOSS. Without loss of generality, the exponential i-IOSS Lyapunov function can then be chosen quadratic (Allan, 2020, Corollary 2.15). The descent condition is then given by

$$\Lambda(f(x_1,w_1),f(x_2,w_2))) \leq \Lambda(x_1,x_2) - a_3\,|x_1-x_2|^2 + a_w\,|w_1-w_2|^2 + a_v\,|h(x_1)-h(x_2)|^2$$

which holds for all $x_1, x_2 \in \mathbb{X}$ and $w_1, w_2 \in \mathbb{W}$. Assume we have chosen the usual least-squares stage cost

$$\ell(w,v) = |w|^2_{Q_w^{-1}} + |v|^2_{R_v^{-1}}$$

where $Q_w, R_v > 0$ are estimates of the variances of process and measurement disturbances w, v, respectively. The standard descent condition in the noise-free case is

$$\begin{aligned} Y(j+1|k) &= Y(j|k) - \ell(\hat{w}(j|k),\hat{v}(j|k) \\ &\leq Y(j|k) - \underline{\sigma}(Q_w^{-1})\,|\,\hat{w}(j|k)\,|^2 - \underline{\sigma}(R_v^{-1})\,|\,\hat{v}(j|k)\,|^2 \end{aligned}$$

where $\underline{\sigma}(A)$ is the smallest singular value of matrix A. Define $Q(j|k) \coloneqq Y(j|k) + \Lambda(x(j),\hat{x}(j|k))$, and to establish estimator stability we need to show that the Q-function has a descent condition

$$Q(j+1|k) \leq Q(j|k) - c_3\,\big|\,x(j)-\hat{x}(j|k)\,\big|^2$$

for some $c_3 > 0$.

But how can we have a descent condition when we have not assumed any relationship between matrices Q_w, R_v in the stage cost and constants a_w and a_v in the system's detectability condition?

Hint: consider what you are asked to show in Exercise B.3(b) about the converse theorem for exponential stability. Use a similar idea here.

Exercise 4.13: Convergent disturbances

Prove Proposition 4.3, i.e., show that if an estimator is RGAS and $(w(k),v(k)) \to 0$ as $k \to \infty$, then $\hat{x}(k) \to x(k)$ as $k \to \infty$. Hint: in the definition of RGAS, break the maximization over interval $[0:k-1]$ into maximization over two intervals $[0:M-1] \cup [M:k-1]$ and choose M to control the size of each maximization.

Exercise 4.14: Observabilty plus $\mathcal{K}$-continuity imply FSO

Prove Proposition 4.31. Hint: first try replacing global $\mathcal{K}$-continuity with the stronger assumption of global Lipschitz continuity to get a feel for the argument.

Exercise 4.15: Detectable linear time-invariant system and i-UIOSS

Show that the detectable linear time-invariant system $x^+ = Ax + Bu + Gw, y = Cx$ is i-UIOSS from Definition 4.40.

Exercise 4.16: Dynamic programming recursion for Kalman predictor

In the Kalman predictor, we use forward DP to solve at stage k

$$\min_{x,w} \ell(x,w) + V_k^-(x) \qquad \text{s.t. } z = Ax + w$$

in which x is the state at the current stage and z is the state at the next stage. The stage cost and arrival cost are given by

$$\ell(x,w) = (1/2)(\,|y(k) - Cx|^2_{R^{-1}} + w'Q^{-1}w) \qquad V_k^-(x) = (1/2)\,|x - \hat{x}^-(k)|^2_{(P^-(k))^{-1}}$$

and we wish to find the value function $V^0(z)$, which we denote $V_{k+1}^-(z)$ in the Kalman predictor estimation problem.

(a) Combine the two x terms to obtain

$$\min_{x,w} \frac{1}{2}\Big(w'Q^{-1}w + (x - \hat{x}(k))'P(k)^{-1}(x - \hat{x}(k))\Big) \qquad \text{s.t. } z = Ax + w$$

and, using the third part of Example 1.1, show

$$\begin{aligned}
P(k) &= P^-(k) - P^-(k)C'(CP^-(k)C' + R)^{-1}CP^-(k)\\
L(k) &= P^-(k)C'(CP^-(k)C' + R)^{-1}\\
\hat{x}(k) &= \hat{x}^-(k) + L(k)(y(k) - C\hat{x}^-(k))
\end{aligned}$$

(b) Add the w term and use the inverse form in Exercise 1.18 to show the optimal cost is given by

$$\begin{aligned}
V^0(z) &= (1/2)(z - A\hat{x}^-(k+1))'(P^-(k+1))^{-1}(z - A\hat{x}^-(k+1))\\
\hat{x}^-(k+1) &= A\hat{x}(k)\\
P^-(k+1) &= AP(k)A' + Q
\end{aligned}$$

Substitute the results for $\hat{x}(k)$ and $P(k)$ above and show

$$\begin{aligned}
V_{k+1}^-(z) &= (1/2)(z - \hat{x}^-(k+1))'(P^-(k+1))^{-1}(z - \hat{x}(k+1))\\
P^-(k+1) &= Q + AP^-(k)A' - AP^-(k)C'(CP^-(k)C' + R)^{-1}CP^-(k)A\\
\hat{x}^-(k+1) &= A\hat{x}^-(k) + \widetilde{L}(k)(y(k) - C\hat{x}^-(k))\\
\widetilde{L}(k) &= AP^-(k)C'(CP^-(k)C' + R)^{-1}
\end{aligned}$$

(c) Compare and contrast this form of the estimation problem to the one given in Exercise 1.29 that describes the Kalman filter.

Exercise 4.17: Duality, cost to go, and covariance

Using the duality variables of Table 4.2, translate Theorem 4.27 into the version that is relevant to the state estimation problem.

Exercise 4.18: Estimator convergence for (A, G) not stabilizable

What happens to the stability of the optimal estimator if we violate the condition

$$(A, G) \text{ stabilizable}$$

(a) Is the steady-state Kalman filter a stable estimator? Is the full information estimator a stable estimator? Are these two answers contradictory? Work out the results for the case $A = 1, G = 0, C = 1, P^-(0) = 1, Q = 1, R = 1$.
Hint: you may want to consult de Souza, Gevers, and Goodwin (1986).

(b) Can this phenomenon happen in the LQ regulator? Provide the interpretation of the time-varying regulator that corresponds to the time-varying filter given above. Does this make sense as a regulation problem?

Exercise 4.19: Exponential stability of the Kalman predictor

Establish that the Kalman predictor defined in Section 4.2.3 is a globally exponentially stable estimator. What is the corresponding linear quadratic regulator?

Exercise 4.20: Equivalent definition of RGES

Prove Proposition 4.21.
Hint: Consider arbitrary $w \in \mathbb{R}^g, v \in \mathbb{R}^p$. Show that

1. For every $a_w, a_v > 0$, there exists $a_d > 0$ such that $a_w |w| \oplus a_v |v| \leq a_d |(w, v)|$;
2. For every $a_d > 0$, there exist $a_w, a_v > 0$ such that $a_d |(w, v)| \leq a_w |w| \oplus a_v |v|$.

Bibliography

A. Alessandri, M. Baglietto, and G. Battistelli. Moving-horizon state estimation for nonlinear discrete-time systems: New stability results and approximation schemes. *Automatica*, 44(7):1753–1765, 2008.

D. A. Allan. *A Lyapunov-like Function for Analysis of Model Predictive Control and Moving Horizon Estimation*. PhD thesis, University of Wisconsin–Madison, August 2020.

D. A. Allan and J. B. Rawlings. A Lyapunov-like function for full information estimation. In *American Control Conference*, pages 4497–4502, Philadelphia, PA, July 10–12, 2019.

D. A. Allan and J. B. Rawlings. Robust stability of full information estimation. SIAM J. Cont. Opt., 2020. Submitted 4/16/2020.

D. A. Allan, C. N. Bates, M. J. Risbeck, and J. B. Rawlings. On the inherent robustness of optimal and suboptimal nonlinear MPC. *Sys. Cont. Let.*, 106: 68–78, August 2017.

D. A. Allan, J. B. Rawlings, and A. R. Teel. Nonlinear detectability and incremental input/output-to-state stability. Technical Report 2020–01, TWCCC Technical Report, July 2020.

B. D. O. Anderson and J. B. Moore. Detectability and stabilizability of time-varying discrete-time linear systems. *SIAM J. Cont. Opt.*, 19(1):20–32, 1981.

K. J. Åström. *Introduction to Stochastic Control Theory*. Academic Press, San Diego, California, 1970.

K. Berntorp and P. Grover. Feedback particle filter with data-driven gain-function approximation. *IEEE Trans. Aero. Elec. Sys.*, 54(5):2118–2130, 2018.

D. P. Bertsekas. *Dynamic Programming*. Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1987.

C. Cai and A. R. Teel. Input–output-to-state stability for discrete-time systems. *Automatica*, 44(2):326 – 336, 2008.

F. M. Callier and C. A. Desoer. *Linear System Theory*. Springer-Verlag, New York, 1991.

M. Chaves and E. D. Sontag. State-estimators for chemical reaction networks of Feinberg-Horn-Jackson zero deficiency type. *Eur. J. Control*, 8(4):343–359, 2002.

D. A. Copp and J. P. Hespanha. Simultaneous nonlinear model predictive control and state estimation. *Automatica*, 77:143–154, 2017.

C. E. de Souza, M. R. Gevers, and G. C. Goodwin. Riccati equation in optimal filtering of nonstabilizable systems having singular state transition matrices. *IEEE Trans. Auto. Cont.*, 31(9):831–838, September 1986.

J. Deyst and C. Price. Conditions for asymptotic stability of the discrete minimum-variance linear estimator. *IEEE Trans. Auto. Cont.*, 13(6):702–705, Dec 1968.

A. Gelb, editor. *Applied Optimal Estimation*. The M.I.T. Press, Cambridge, Massachusetts, 1974.

R. Gudi, S. Shah, and M. Gray. Multirate state and parameter estimation in an antibiotic fermentation with delayed measurements. *Biotech. Bioeng.*, 44: 1271–1278, 1994.

R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press, 1985.

W. Hu. Robust Stability of Optimization-based State Estimation Under Bounded Disturbances. *ArXiv e-prints*, Feb 2017.

W. Hu, L. Xie, and K. You. Optimization-based state estimation under bounded disturbances. In *2015 54th IEEE Conference on Decision and Control CDC*, pages 6597–6602, Dec 2015.

A. H. Jazwinski. *Stochastic Processes and Filtering Theory*. Academic Press, New York, 1970.

L. Ji, J. B. Rawlings, W. Hu, A. Wynn, and M. Diehl. Robust stability of moving horizon estimation under bounded disturbances. *IEEE Trans. Auto. Cont.*, 61(11):3509–3514, November 2016.

Z.-P. Jiang and Y. Wang. Input-to-state stability for discrete-time nonlinear systems. *Automatica*, 37:857–869, 2001.

S. J. Julier and J. K. Uhlmann. A new extension of the Kalman filter to nonlinear systems. In *International Symposium Aerospace/Defense Sensing, Simulation and Controls*, pages 182–193, 1997.

S. J. Julier and J. K. Uhlmann. Author's reply. *IEEE Trans. Auto. Cont.*, 47(8): 1408–1409, August 2002.

S. J. Julier and J. K. Uhlmann. Unscented filtering and nonlinear estimation. *Proc. IEEE*, 92(3):401–422, March 2004a.

S. J. Julier and J. K. Uhlmann. Corrections to unscented filtering and nonlinear estimation. *Proc. IEEE*, 92(12):1958, December 2004b.

S. J. Julier, J. K. Uhlmann, and H. F. Durrant-Whyte. A new method for the nonlinear transformation of means and covariances in filters and estimators. *IEEE Trans. Auto. Cont.*, 45(3):477–482, March 2000.

T. Kailath. A view of three decades of linear filtering theory. *IEEE Trans. Inform. Theory*, IT-20(2):146–181, March 1974.

R. Kandepu, L. Imsland, and B. A. Foss. Constrained state estimation using the unscented kalman filter. In *Procedings of the 16th Mediterranean Conference on Control and Automation*, pages 1453–1458, Ajaccio, France, June 2008.

S. S. Keerthi and E. G. Gilbert. An existence theorem for discrete-time infinite-horizon optimal control problems. *IEEE Trans. Auto. Cont.*, 30(9):907–909, September 1985.

S. Knüfer and M. A. Müller. Robust global exponential stability for moving horizon estimation. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 3477–3482, Dec 2018.

S. Kolås, B. A. Foss, and T. S. Schei. Constrained nonlinear state estimation based on the UKF approach. *Comput. Chem. Eng.*, 33(8):1386–1401, 2009.

A. N. Kolmogorov. Interpolation and extrapolation of stationary random sequences. *Bull. Moscow Univ., USSR, Ser. Math. 5*, 1941.

H. Kwakernaak and R. Sivan. *Linear Optimal Control Systems*. John Wiley and Sons, New York, 1972.

M. Lazar, W. P. M. H. Heemels, and A. R. Teel. Further input-to-state stability subtleties for discrete-time systems. *Automatic Control, IEEE Transactions on*, 58(6):1609–1613, June 2013.

T. Lefebvre, H. Bruyninckx, and J. De Schutter. Comment on "A new method for the nonlinear transformation of means and covariances in filters and estimators". *IEEE Trans. Auto. Cont.*, 47(8):1406–1408, August 2002.

E. S. Meadows, K. R. Muske, and J. B. Rawlings. Constrained state estimation and discontinuous feedback in model predictive control. In *Proceedings of the 1993 European Control Conference*, pages 2308–2312, 1993.

H. Michalska and D. Q. Mayne. Moving horizon observers and observer-based control. *IEEE Trans. Auto. Cont.*, 40(6):995–1006, 1995.

S. A. Middlebrooks and J. B. Rawlings. State estimation approach for determining composition and growth rate of $Si_{1-x}Ge_x$ chemical vapor deposition utilizing real-time ellipsometric measurements. *Applied Opt.*, 45:7043–7055, 2006.

M. A. Müller. Nonlinear moving horizon estimation in the presence of bounded disturbances. *Automatica*, 79:306–314, 2017.

K. R. Muske, J. B. Rawlings, and J. H. Lee. Receding horizon recursive state estimation. In *Proceedings of the 1993 American Control Conference*, pages 900–904, June 1993.

M. Nørgaard, N. K. Poulsen, and O. Ravn. New developments in state estimation for nonlinear systems. *Automatica*, 36:1627–1638, 2000.

V. Prasad, M. Schley, L. P. Russo, and B. W. Bequette. Product property and production rate control of styrene polymerization. *J. Proc. Cont.*, 12(3):353–372, 2002.

C. C. Qu and J. Hahn. Computation of arrival cost for moving horizon estimation via unscented Kalman filtering. *J. Proc. Cont.*, 19(2):358–363, 2009.

C. V. Rao. *Moving Horizon Strategies for the Constrained Monitoring and Control of Nonlinear Discrete-Time Systems*. PhD thesis, University of Wisconsin–Madison, 2000.

C. V. Rao, J. B. Rawlings, and J. H. Lee. Constrained linear state estimation – a moving horizon approach. *Automatica*, 37(10):1619–1628, 2001.

C. V. Rao, J. B. Rawlings, and D. Q. Mayne. Constrained state estimation for nonlinear discrete-time systems: stability and moving horizon approximations. *IEEE Trans. Auto. Cont.*, 48(2):246–258, February 2003.

J. B. Rawlings and L. Ji. Optimization-based state estimation: Current status and some new results. *J. Proc. Cont.*, 22:1439–1444, 2012.

J. B. Rawlings and D. Q. Mayne. *Model Predictive Control: Theory and Design*. Nob Hill Publishing, Madison, WI, 2009. 668 pages, ISBN 978-0-9759377-0-9.

J. B. Rawlings and M. J. Risbeck. On the equivalence between statements with epsilon-delta and K-functions. Technical Report 2015–01, TWCCC Technical Report, December 2015.

K. Reif and R. Unbehauen. The extended Kalman filter as an exponential observer for nonlinear systems. *IEEE Trans. Signal Process.*, 47(8):2324–2328, August 1999.

K. Reif, S. Günther, E. Yaz, and R. Unbehauen. Stochastic stability of the discrete-time extended Kalman filter. *IEEE Trans. Auto. Cont.*, 44(4):714–728, April 1999.

K. Reif, S. Günther, E. Yaz, and R. Unbehauen. Stochastic stability of the continuous-time extended Kalman filter. *IEE Proceedings-Control Theory and Applications*, 147(1):45–52, January 2000.

D. G. Robertson and J. H. Lee. On the use of constraints in least squares estimation and control. *Automatica*, 38(7):1113–1124, 2002.

A. Romanenko and J. A. A. M. Castro. The unscented filter as an alternative to the EKF for nonlinear state estimation: a simulation case study. *Comput. Chem. Eng.*, 28(3):347–355, March 15 2004.

A. Romanenko, L. O. Santos, and P. A. F. N. A. Afonso. Unscented Kalman filtering of a simulated pH system. *Ind. Eng. Chem. Res.*, 43:7531–7538, 2004.

B. J. P. Roset, W. P. M. H. Heemels, M. Lazar, and H. Nijmeijer. On robustness of constrained discrete-time systems to state measurement errors. *Automatica*, 44(4):1161 – 1165, 2008.

E. D. Sontag. *Mathematical Control Theory*. Springer-Verlag, New York, second edition, 1998.

E. D. Sontag and Y. Wang. Output-to-state stability and detectability of nonlinear systems. *Sys. Cont. Let.*, 29:279–290, 1997.

R. F. Stengel. *Optimal Control and Estimation*. Dover Publications, Inc., 1994.

B. O. S. Teixeira, L. A. B. Tôrres, L. A. Aguirre, and D. S. Bernstein. Unscented filtering for interval-constrained nonlinear systems. In *Proceedings of the 47th IEEE Conference on Decision and Control*, pages 5116–5121, Cancun, Mexico, December 9-11 2008.

M. J. Tenny and J. B. Rawlings. Efficient moving horizon estimation and nonlinear model predictive control. In *Proceedings of the American Control Conference*, pages 4475–4480, Anchorage, Alaska, May 2002.

P. Vachhani, S. Narasimhan, and R. Rengaswamy. Robust and reliable estimation via unscented recursive nonlinear dynamic data reconciliation. *J. Proc. Cont.*, 16(10):1075–1086, December 2006.

R. van der Merwe, A. Doucet, N. de Freitas, and E. Wan. The unscented particle filter. Technical Report CUED/F-INFENG/TR 380, Cambridge University Engineering Department, August 2000.

N. Wiener. *The Extrapolation, Interpolation, and Smoothing of Stationary Time Series with Engineering Applications.* Wiley, New York, 1949. Originally issued as a classified MIT Rad. Lab. Report in February 1942.

D. I. Wilson, M. Agarwal, and D. W. T. Rippin. Experiences implementing the extended Kalman filter on an industrial batch reactor. *Comput. Chem. Eng.*, 22(11):1653–1672, 1998.

T. Yang, P. G. Mehta, and S. P. Meyn. Feedback particle filter. *IEEE Trans. Auto. Cont.*, 58(10):2465–2480, 2013.

5
Output Model Predictive Control

5.1 Introduction

In Chapter 2 we show how model predictive control (MPC) may be employed to control a *deterministic* system, that is, a system in which there are no uncertainties and the state is known. In Chapter 3 we show how to control an *uncertain* system in which uncertainties are present but the state is known. Here we address the problem of MPC of an uncertain system in which the state is *not* fully known. We assume that there are outputs available that may be used to estimate the state as shown in Chapter 4. These outputs are used by the model predictive controller to generate control actions; hence the name *output MPC*.

The state is not known, but a noisy measurement $y(t)$ of the state is available at each time t. Since the state x is not known, it is replaced by a hyperstate p that summarizes all prior information (previous inputs and outputs and the prior distribution of the initial state) and that has the "state" property: future values of p can be determined from the current value of p, and current and future inputs and outputs. Usually $p(t)$ is the conditional density of $x(t)$ given the prior density $p(0)$ of $x(0)$, and the current available "information" $I(t) := (y(0), y(1), \ldots, y(t-1), u(0), u(1), \ldots, u(t-1))$.

For the purpose of control, future hyperstates have to be predicted since future noisy measurements of the state are not known. So the hyperstate satisfies an uncertain difference equation of the form

$$p^+ = \phi(p, u, \psi) \tag{5.1}$$

where $(\psi(t))_{t \in \mathbb{I}_{\geq 0}}$ is a sequence of random variables. The problem of controlling a system with unknown state x is transformed into the problem of controlling an uncertain system with known state p. For

example, if the underlying system is described by

$$x^+ = Ax + Bu + w$$
$$y = Cx + v$$

where $(w(t))_{t\in\mathbb{I}_{\geq 0}}$ and $(v(t))_{t\in\mathbb{I}_{\geq 0}}$ are sequences of zero-mean, normal, independent random variables with variances Σ_w and Σ_v, respectively, and if the prior density $p(0)$ of $x(0)$ is normal with density $n(\bar{x}_0, \Sigma_0)$, then, as is well known, $p(t)$ is the normal density $n(\hat{x}(t), \Sigma(t))$ so that the hyperstate $p(t)$ is finitely parameterized by $(\hat{x}(t), \Sigma(t))$. Hence the evolution equation for $p(t)$ may be replaced by the simpler evolution equation for $(\hat{x}, \Sigma)$, that is by

$$\hat{x}(t+1) = A\hat{x}(t) + Bu + L(t)\psi(t) \tag{5.2}$$
$$\Sigma(t+1) = \Phi(\Sigma(t)) \tag{5.3}$$

in which

$$\Phi(\Sigma) := A\Sigma A' - A\Sigma C'(C'\Sigma C + \Sigma_v)^{-1}C\Sigma A' + \Sigma_w$$
$$\psi(t) := y(t) - C\hat{x}(t) = C\tilde{x}(t) + v(t)$$
$$\tilde{x}(t) := x(t) - \hat{x}(t)$$

The initial conditions for (5.2) and (5.3) are

$$\hat{x}(0) = \bar{x}_0 \qquad \Sigma(0) = \Sigma_0$$

These are, of course, the celebrated Kalman filter equations derived in Chapter 1. The random variables $\tilde{x}$ and ψ have the following densities: $\tilde{x}(t) \sim n(0, \Sigma(t))$ and $\psi(t) \sim n(0, \Sigma_v + C'\Sigma(t)C)$. The finite dimensional equations (5.2) and (5.3) replace the difference equation (5.1) for the hyperstate p that is a conditional density and, therefore, infinite dimensional in general. The sequence $(\psi(t))_{t\in\mathbb{I}_{\geq 0}}$ is known as the *innovations* sequence; $\psi(t)$ is the "new" information contained in $y(t)$.

Output control, in general, requires control of the hyperstate p that may be computed with difficulty, since p satisfies a complex evolution equation $p^+ = \phi(p, u, \psi)$ where ψ is a random disturbance. Controlling p is a problem of the same type as that considered in Chapter 3, but considerably more complex since the function $p(\cdot)$ is infinite dimensional. Because of the complexity of the evolution equation for p, a simpler procedure is often adopted. Assuming that the state x is known, a stabilizing controller $u = \kappa(x)$ is designed. An observer or

filter yielding an estimate $\hat{x}$ of the state is then separately designed and the control $u = \kappa(\hat{x})$ is applied to the plant. Indeed, this form of control is actually optimal for the linear quadratic Gaussian (LQG) optimal control problem considered in Chapter 1, but is not necessarily optimal and stabilizing when the system is nonlinear and constrained. We propose a variant of this procedure, modified to cope with state and control constraints.

The state estimate $\hat{x}$ satisfies an uncertain difference equation with an additive disturbance of the same type as that considered in Chapter 3. Hence we employ tube MPC, similar to that employed in Chapter 3, to obtain a nominal trajectory satisfying tightened constraints. We then construct a tube that has as its center the nominal trajectory, and which includes every possible realization of $\hat{\mathbf{x}} = (\hat{x}(t))_{t \in \mathbb{I}_{\geq 0}}$. We then construct a second tube that includes the first tube in its interior, and is such that every possible realization of the sequence $\mathbf{x} = (x(t))_{t \in \mathbb{I}_{\geq 0}}$ lies in its interior. The tightened constraints are chosen to ensure every possible realization of $\mathbf{x} = (x(t))_{t \in \mathbb{I}_{\geq 0}}$ does not transgress the original constraints. An advantage of the method presented here is that its online complexity is comparable to that of conventional MPC.

As in Chapter 3, a caveat is necessary. Because of the inherent complexity of output MPC, different compromises between simplicity and efficiency are possible. For this reason, output MPC remains an active research area and alternative methods, available or yet to be developed, may be preferred.

5.2 A Method for Output MPC

Suppose the system to be controlled is described by

$$\begin{aligned} x^+ &= Ax + Bu + w \\ y &= Cx + v \end{aligned}$$

The state and control are required to satisfy the constraints $x(t) \in \mathbb{X}$ and $u(t) \in \mathbb{U}$ for all t, and the disturbance is assumed to lie in the compact set $\mathbb{W}$. It is assumed that the origin lies in the interior of the sets $\mathbb{X}$, $\mathbb{U}$, and $\mathbb{W}$. The state estimator $(\hat{x}, \mathbb{Z})$ evolves, as shown in the sequel, according to

$$\hat{x}^+ = \phi(\hat{x}, u, y) \tag{5.4}$$

$$\mathbb{Z}^+ = \Phi(\mathbb{Z}) \tag{5.5}$$

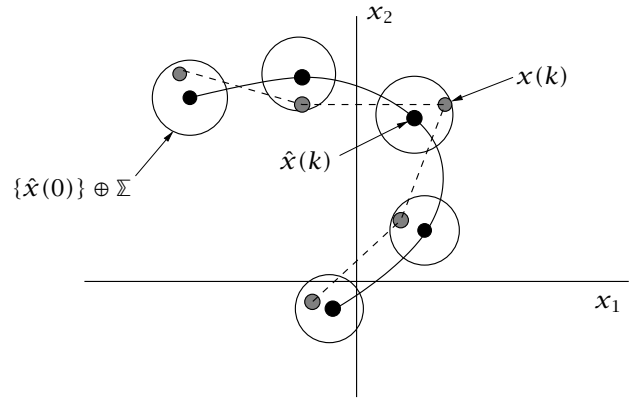


Figure 5.1: State estimator tube. The solid line $\hat{x}(t)$ is the center of the tube, and the dashed line is a sample trajectory of $x(t)$.

in which ψ is a random variable in the stochastic case, and a bounded disturbance taking values in $\mathbb{W}$ when w and v are bounded. In the latter case, $x \in \{\hat{x}\} \oplus \mathbb{Z}$ implies $x^+ \in \{\hat{x}^+\} \oplus \mathbb{Z}^+$ for all $\psi \in \mathbb{W}$.

As illustrated in Figure 5.1, the evolution equations generate a *tube*, which is the set sequence $(\{\hat{x}(t)\} \oplus \mathbb{Z}(t))_{t \in \mathbb{I}_{\geq 0}}$; at time t the center of the tube is $\hat{x}(t)$ and the "cross section" is $\mathbb{Z}(t)$. When the disturbances are bounded, which is the only case we consider in the sequel, all possible realizations of the state trajectory $(x(t))$ lie in the set $\{\hat{x}(t)\} \oplus \mathbb{Z}(t)$ for all t; the dashed line is a sample trajectory of $x(t)$.

From (5.4), the estimator trajectory $(\hat{x}(t))_{t \in \mathbb{I}_{\geq 0}}$ is influenced both by the control that is applied and by the disturbance sequence $(\psi(t))_{t \in \mathbb{I}_{\geq 0}}$. If the trajectory were influenced only by the control, we could choose the control to satisfy the control constraints, and to cause the estimator tube to lie in a region such that the state constraints are satisfied by all possible realizations of the state trajectory. Hence the output MPC problem would reduce to a conventional MPC problem with modified constraints in which the state is $\hat{x}$, rather than x. The new state constraint is $\hat{x} \in \hat{\mathbb{X}}$ where $\hat{\mathbb{X}}$ is chosen to ensure that $\hat{x} \in \hat{\mathbb{X}}$ implies $x \in \mathbb{X}$ and, therefore, satisfies $\hat{\mathbb{X}} \subseteq \mathbb{X} \ominus \mathbb{Z}$ if $\mathbb{Z}$ does not vary with time t.

But the estimator state $\hat{x}(t)$ is influenced by the disturbance ψ (see (5.4)), so it cannot be precisely controlled. The problem of controlling the system described by (5.4) is the same type of problem studied in

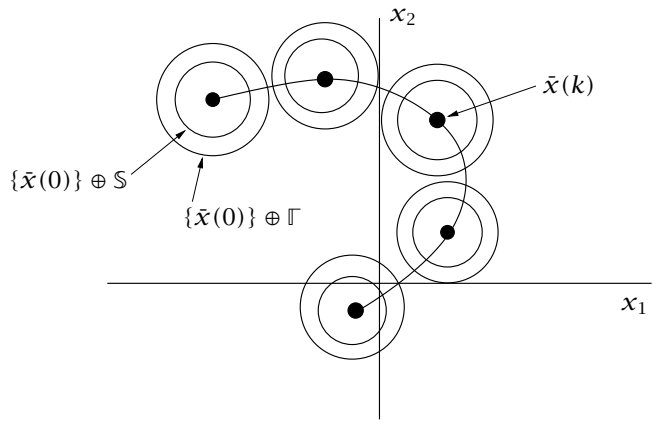


Figure 5.2: The system with disturbance. The state estimate lies in the inner tube, and the state lies in the outer tube.

Chapter 3, where the system was described by $x^+ = f(x, u, w)$ with the estimator state $\hat{x}$, which is accessible, replacing the state x. Hence we may use the techniques presented in Chapter 3 to choose a control that forces $\hat{x}$ to lie in another tube $(\{\bar{x}(t)\} \oplus \mathbb{S}(t))_{t \in \mathbb{I}_{\geq 0}}$ where the set sequence $(\mathbb{S}(t))_{t \in \mathbb{I}_{\geq 0}}$ that defines the cross section of the tube is precomputed. The sequence $(\bar{x}(t))_{t \in \mathbb{I}_{\geq 0}}$ that defines the center of the tube is the state trajectory of the nominal (deterministic) system defined by

$$\bar{x}^+ = \phi(\bar{x}, \bar{u}, 0) \tag{5.6}$$

the nominal version of (5.4). Thus we get two tubes, one embedded in the other. At time t the estimator state $\hat{x}(t)$ lies in the set $\{\bar{x}(t)\} \oplus \mathbb{S}(t)$, and $x(t)$ lies in the set $\{\hat{x}(t)\} \oplus \mathbb{\Sigma}(t)$, so that for all t

$$x(t) \in \{\bar{x}(t)\} \oplus \mathbb{\Gamma}(t) \qquad \mathbb{\Gamma}(t) := \mathbb{\Sigma}(t) \oplus \mathbb{S}(t)$$

Figure 5.2 shows the tube $(\{\bar{x}(t)\} \oplus \mathbb{S}(t))$, in which the trajectory $(\hat{x}(t))$ lies, and the tube $(\{\bar{x}(t)\} \oplus \mathbb{\Gamma}(t))$, in which the state trajectory $(x(t))$ lies.

5.3 Linear Constrained Systems: Time-Invariant Case

5.3.1 Introduction

We consider the following uncertain linear time-invariant system

$$\begin{aligned} x^+ &= Ax + Bu + w \\ y &= Cx + v \end{aligned} \tag{5.7}$$

in which $x \in \mathbb{R}^n$ is the current state, $u \in \mathbb{R}^m$ is the current control action, x^+ is the successor state, $w \in \mathbb{R}^n$ is an unknown state disturbance, $y \in \mathbb{R}^p$ is the current measured output, $v \in \mathbb{R}^p$ is an unknown output disturbance, the pair (A, B) is assumed to be controllable, and the pair (A, C) observable. The state and additive disturbances w and v are known only to the extent that they lie, respectively, in the C-sets[1] $\mathbb{W} \subseteq \mathbb{R}^n$ and $\mathbb{N} \subseteq \mathbb{R}^p$. Let $\phi(i; x(0), \mathbf{u}, \mathbf{w})$ denote the solution of (5.7) at time i if the initial state at time 0 is $x(0)$, and the control and disturbance sequences are, respectively, $\mathbf{u} := (u(0), u(1), \ldots)$ and $\mathbf{w} := (w(0), w(1), \ldots)$. The system (5.7) is subject to the following set of hard state and control constraints

$$x \in \mathbb{X} \qquad u \in \mathbb{U} \tag{5.8}$$

in which $\mathbb{X} \subseteq \mathbb{R}^n$ and $\mathbb{U} \subseteq \mathbb{R}^m$ are polyhedral and polytopic sets respectively; both sets contain the origin as an interior point.

5.3.2 State Estimator

To estimate the state a Kalman filter or Luenberger observer is employed

$$\begin{aligned} \hat{x}^+ &= A\hat{x} + Bu + L(y - \hat{y}) \\ \hat{y} &= C\hat{x} \end{aligned} \tag{5.9}$$

in which $\hat{x} \in \mathbb{R}^n$ is the current observer state (state estimate), $u \in \mathbb{R}^m$ is the current control action, $\hat{x}^+$ is the successor state of the observer system, $\hat{y} \in \mathbb{R}^p$ is the current observer output, and $L \in \mathbb{R}^{n \times p}$. The output injection matrix L is chosen to satisfy $\rho(A_L) < 1$ where $A_L := A - LC$.

The estimated state $\hat{x}$ therefore satisfies the following uncertain difference equation

$$\hat{x}^+ = A\hat{x} + Bu + L(C\tilde{x} + v)$$

[1] Recall, a C-set is a convex, compact set containing the origin.

The state estimation error $\tilde{x}$ is defined by $\tilde{x} := x - \hat{x}$ so that $x = \hat{x} + \tilde{x}$. Since $x^+ = Ax + Bu + w$, the state estimation error $\tilde{x}$ satisfies

$$\tilde{x}^+ = A_L\tilde{x} + \tilde{w} \qquad \tilde{w} := w - Lv \tag{5.10}$$

Because w and v are bounded, so is $\tilde{w}$; in fact, $\tilde{w}$ takes values in the C-set $\bar{\mathbb{W}}$ defined by

$$\bar{\mathbb{W}} := \mathbb{W} \oplus (-L\mathbb{N})$$

We recall the following standard definitions (Blanchini, 1999).

Definition 5.1 (Positive invariance; robust positive invariance)**.** A set $\Omega \subseteq \mathbb{R}^n$ is *positive invariant* for the system $x^+ = f(x)$ and the constraint set $\mathbb{X}$ if $\Omega \subseteq \mathbb{X}$ and $f(x) \in \Omega,\ \forall x \in \Omega$.
A set $\Omega \subseteq \mathbb{R}^n$ is *robust positive invariant* for the system $x^+ = f(x, w)$ and the constraint set $(\mathbb{X}, \mathbb{W})$ if $\Omega \subseteq \mathbb{X}$ and $f(x, w) \in \Omega,\ \forall w \in \mathbb{W}$, $\forall x \in \Omega$.

Since $\rho(A_L) < 1$ and $\tilde{\mathbb{W}}$ is compact, there exists, as shown in Kolmanovsky and Gilbert (1998), Theorem 4.1, a robust positive invariant set $\mathbb{\Sigma} \subseteq \mathbb{R}^n$, satisfying

$$A_L\mathbb{\Sigma} \oplus \bar{\mathbb{W}} = \mathbb{\Sigma} \tag{5.11}$$

Hence, for all $\tilde{x} \in \mathbb{\Sigma}$, $\tilde{x}^+ = A_L\tilde{x} + \tilde{w} \in \mathbb{\Sigma}$ for all $\tilde{w} \in \tilde{\mathbb{W}}$; the term *robust* in the description of $\mathbb{\Sigma}$ refers to this property. In fact, $\mathbb{\Sigma}$ is the *minimal* robust, positive invariant set for $\tilde{x}^+ = A_L\tilde{x} + \tilde{w}$, $\tilde{w} \in \tilde{\mathbb{W}}$, i.e., a set that is a subset of all robust positive invariant sets. There exist techniques (Raković, Kerrigan, Kouramas, and Mayne, 2005) for obtaining, for every $\epsilon > 0$, a polytopic, nonminimal, robust, positive invariant set $\mathbb{\Sigma}^0$ that satisfies $d_H(\mathbb{\Sigma}, \mathbb{\Sigma}^0) \leq \epsilon$ where $d_H(\cdot, \cdot)$ is the Hausdorff metric. However, it is not necessary to compute the set Σ or Σ^0 as shown in Chapter 3. An immediate consequence of (5.11) is the following.

Proposition 5.2 (Proximity of state and state estimate)**.** *If the initial system and observer states, $x(0)$ and $\hat{x}(0)$ respectively, satisfy $\{x(0)\} \in \{\hat{x}(0)\} \oplus \mathbb{\Sigma}$, then $x(i) \in \{\hat{x}(i)\} \oplus \mathbb{\Sigma}$ for all $i \in \mathbb{I}_{\geq 0}$, and all admissible disturbance sequences $\mathbf{w}$ and $\boldsymbol{v}$.*

The assumption that $\tilde{x}(i) \in \mathbb{\Sigma}$ for all i is a *steady-state* assumption; if $\tilde{x}(0) \in \mathbb{\Sigma}$, then $\tilde{x}(i) \in \mathbb{\Sigma}$ for all i. If, on the other hand, $\tilde{x}(0) \in \mathbb{\Sigma}(0)$ where $\mathbb{\Sigma}(0) \supseteq \mathbb{\Sigma}$, then it is possible to show that $\tilde{x}(i) \in \mathbb{\Sigma}(i)$ for all $i \in \mathbb{I}_{\geq 0}$ where $\mathbb{\Sigma}(i) \to \mathbb{\Sigma}$ in the Hausdorff metric as $i \to \infty$; the sequence $(\mathbb{\Sigma}(i))$ satisfies $\mathbb{\Sigma}(0) \supseteq \mathbb{\Sigma}(1) \supseteq \mathbb{\Sigma}(2) \supseteq \cdots \supseteq \mathbb{\Sigma}$. Hence, it is reasonable

to assume that if the estimator has been running for a "long" time, it is in steady state.

Hence we have obtained a state estimator, with "state" $(\hat{x}, \mathbb{\Sigma})$ satisfying

$$\begin{aligned} \hat{x}^+ &= A\hat{x} + Bu + L(y - \hat{y}) \\ \mathbb{\Sigma}^+ &= \mathbb{\Sigma} \end{aligned} \tag{5.12}$$

and $x(i) \in \hat{x}(i) \oplus \mathbb{\Sigma}$ for all $i \in \mathbb{I}_{\geq 0}$, thus meeting the requirements specified in Section 5.2. Knowing this, our remaining task is to control $\hat{x}(i)$ so that the resultant closed-loop system is stable and satisfies all constraints.

5.3.3 Controlling $\hat{x}$

Since $\tilde{x}(i) \in \mathbb{\Sigma}$ for all i, we seek a method for controlling the observer state $\hat{x}(i)$ in such a way that $x(i) = \hat{x}(i) + \tilde{x}(i)$ satisfies the state constraint $x(i) \in \mathbb{X}$ for all i. The state constraint $x(i) \in \mathbb{X}$ will be satisfied if we control the estimator state to satisfy $\hat{x}(i) \in \mathbb{X} \ominus \mathbb{\Sigma}$ for all i. The estimator state satisfies (5.12) which can be written in the form

$$\hat{x}^+ = A\hat{x} + Bu + \delta \tag{5.13}$$

where the disturbance δ is defined by

$$\delta := L(y - \hat{y}) = L(C\tilde{x} + v)$$

and, therefore, always lies in the C-set $\mathbb{\Delta}$ defined by

$$\mathbb{\Delta} := L(C\mathbb{\Sigma} \oplus \mathbb{N})$$

The problem of controlling $\hat{x}$ is, therefore, the same as that of controlling an uncertain system with known state. This problem was extensively discussed in Chapter 3. We can therefore use the approach of Chapter 3 here with $\hat{x}$ replacing x, δ replacing w, $\mathbb{X} \ominus \mathbb{\Sigma}$ replacing $\mathbb{X}$ and $\mathbb{\Delta}$ replacing $\mathbb{W}$.

To control (5.13) we use, as in Chapter 3, a combination of open-loop and feedback control, i.e., we choose the control u as follows

$$u = \bar{u} + Ke \qquad e := \hat{x} - \bar{x} \tag{5.14}$$

where $\bar{x}$ is the state of a nominal (deterministic) system that we shall shortly specify; $\bar{u}$ is the feedforward component of the control u, and

Ke is the feedback component. The matrix K is chosen to satisfy $\rho(A_K) < 1$ where $A_K := A + BK$. The feedforward component v of the control u generates, as we show subsequently, a trajectory $(\bar{x}(i))$, which is the center of the tube in which the state estimator trajectory $(\hat{x}(i))$ lies. The feedback component Ke attempts to steer the trajectory $(\hat{x}(i))$ of the state estimate toward the center of the tube, and thereby controls the cross section of the tube. The controller is *dynamic* since it incorporates the nominal dynamic system.

With this control, $\hat{x}$ satisfies the following difference equation

$$\hat{x}^+ = A\hat{x} + B\bar{u} + BKe + \delta \qquad \delta \in \mathbb{\Delta} \tag{5.15}$$

The nominal (deterministic) system describing the evolution of $\bar{x}$ is obtained by neglecting the disturbances BKe and δ in (5.15) yielding

$$\bar{x}^+ = A\bar{x} + B\bar{u}$$

The deviation $e = \hat{x} - \bar{x}$ between the state $\hat{x}$ of the estimator and the state $\bar{x}$ of the nominal system satisfies

$$e^+ = A_K e + \delta \qquad A_K := A + BK \tag{5.16}$$

The feedforward component $\bar{u}$ of the control u generates the trajectory $(\bar{x}(i))$, which is the center of the tube in which the state estimator trajectory $(\hat{x}(i))$ lies. Because $\mathbb{\Delta}$ is a C-set and $\rho(A_K) < 1$, there exists a robust positive invariant C-set $\mathbb{S}$ satisfying

$$A_K\mathbb{S} \oplus \mathbb{\Delta} = \mathbb{S}$$

An immediate consequence is the following.

Proposition 5.3 (Proximity of state estimate and nominal state)**.** *If the initial states of the estimator and nominal system, $\hat{x}(0)$ and $\bar{x}(0)$ respectively, satisfy $\hat{x}(0) \in \{\bar{x}(0)\} \oplus \mathbb{S}$, then $\hat{x}(i) \in \{\bar{x}(i)\} \oplus \mathbb{S}$ and $u(i) \in \{\bar{u}(i)\} \oplus K\mathbb{S}$ for all $i \in \mathbb{I}_{\geq 0}$, and all admissible disturbance sequences $\mathbf{w}$ and $\boldsymbol{\nu}$.*

It follows from Proposition 5.3 that the state estimator trajectory $\hat{\mathbf{x}}$ remains in the tube $(\{\bar{x}(i)\} \oplus \mathbb{S})_{i \in \mathbb{I}_{\geq 0}}$ and the control trajectory $\bar{\mathbf{u}}$ remains in the tube $(\{\bar{u}(i)\} \oplus K\mathbb{S})_{i \in \mathbb{I}_{\geq 0}}$ provided that $e(0) \in \mathbb{S}$. Hence, from Propositions 5.2 and 5.3, the state trajectory $\mathbf{x}$ lies in the tube $(\{\bar{x}(i)\} \oplus \mathbb{\Gamma})_{i \in \mathbb{I}_{\geq 0}}$ where $\mathbb{\Gamma} := \mathbb{S} \oplus \mathbb{Z}$ provided that $\tilde{x}(0) = x(0) - \hat{x}(0) \in \mathbb{Z}$ *and* $e(0) \in \mathbb{S}$. This information may be used to construct a robust output feedback model predictive controller using the procedures outlined

in Chapter 3 for robust state feedback MPC of systems; the major difference is that we now control the estimator state $\hat{x}$ and use the fact that the actual state x lies in $\{\hat{x}\} \oplus \mathbb{Z}$.

5.3.4 Output MPC

Model predictive controllers now can be constructed as described in Chapter 3, which dealt with robust control when the state was known. There is an obvious difference in that we now are concerned with controlling $\hat{x}$ whereas, in Chapter 3, our concern was control of x. We describe here the appropriate modification of the simple model predictive controller presented in Section 3.5.2. We adopt the same procedure of defining a nominal optimal control problem with tighter constraints than in the original problem. The solution to this problem defines the center of a tube in which solutions to the original system lie, and the tighter constraints in the nominal problem ensure that the original constraints are satisfied by the actual system.

The nominal system is described by

$$\bar{x}^+ = A\bar{x} + B\bar{u} \tag{5.17}$$

The nominal optimal control problem is the minimization of the cost function $\bar{V}_N(\bar{x}, \bar{\mathbf{u}})$ with

$$\bar{V}_N(\bar{x}, \bar{\mathbf{u}}) := \sum_{k=0}^{N-1} \ell(\bar{x}(k), \bar{u}(k)) + V_f(\bar{x}(N)) \tag{5.18}$$

subject to satisfaction by the state and control sequences of (5.17) and the *tighter* constraints

$$\bar{x}(i) \in \bar{\mathbb{X}} \subseteq \mathbb{X} \ominus \mathbb{\Gamma} \qquad \mathbb{\Gamma} := \mathbb{S} \oplus \mathbb{Z} \tag{5.19}$$

$$\bar{u}(i) \in \bar{\mathbb{U}} \subseteq \mathbb{U} \ominus K\mathbb{S} \tag{5.20}$$

as well as a terminal constraint $\bar{x}(N) \in \bar{\mathbb{X}}_f \subseteq \bar{\mathbb{X}}$. Notice that $\mathbb{\Gamma}$ appears in (5.19) whereas $\mathbb{S}$, the set in which $e = \hat{x} - \bar{x}$ lies, appears in (5.20); this differs from the case studied in Chapter 3 where the same set appears in both equations. The sets $\mathbb{W}$ and $\mathbb{N}$ are assumed to be sufficiently small to ensure satisfaction of the following condition.

Assumption 5.4 (Constraint bounds)**.** $\mathbb{\Gamma} = \mathbb{S} \oplus \mathbb{Z} \subseteq \mathbb{X}$ and $K\mathbb{S} \subseteq \mathbb{U}$.

If Assumption 5.4 holds, the sets on the right-hand side of (5.19) and (5.20) are not empty; it can be seen from their definitions that the

sets $\mathbb{Z}$ and $\mathbb{S}$ tend to the set $\{0\}$ as $\mathbb{W}$ and $\mathbb{N}$ tend to the set $\{0\}$ in the sense that $d_H(\mathbb{W}, \{0\}) \to 0$ and $d_H(\mathbb{N}, \{0\}) \to 0$.

It follows from Propositions 5.2 and 5.3, if Assumption 5.4 holds, that satisfaction of the constraints (5.19) and (5.20) by the nominal system ensures satisfaction of the constraints (5.8) by the original system. The nominal optimal control problem is, therefore

$$\mathbb{P}_N(\bar{x}): \quad \bar{V}_N^0(\bar{x}) = \min_{\bar{\mathbf{u}}} \{\bar{V}_N(\bar{x}, \bar{\mathbf{u}}) \mid \bar{\mathbf{u}} \in \bar{\mathcal{U}}_N(\bar{x})\}$$

in which the constraint set $\bar{\mathcal{U}}_N(\bar{x})$ is defined by

$$\bar{\mathcal{U}}_N(\bar{x}) := \{\bar{\mathbf{u}} \mid \bar{u}(k) \in \bar{\mathbb{U}} \text{ and } \bar{\phi}(k; \bar{x}, \bar{\mathbf{u}}) \in \bar{\mathbb{X}} \ \forall k \in \{0, 1, \ldots, N-1\}, \ \bar{\phi}(N; \bar{x}, \bar{\mathbf{u}}) \in \bar{\mathbb{X}}_f\} \quad (5.21)$$

In (5.21), $\bar{\mathbb{X}}_f \subseteq \bar{\mathbb{X}}$ is the terminal constraint set, and $\bar{\phi}(k; \bar{x}, \bar{\mathbf{u}})$ denotes the solution of $\bar{x}^+ = A\bar{x} + B\bar{u}$ at time k if the initial state at time 0 is $\bar{x}$ and the control sequence is $\bar{\mathbf{u}} = (\bar{u}(0), \bar{u}(1), \ldots, \bar{u}(N-1))$. The terminal constraint, which is not desirable in process control applications, may be omitted, as shown in Chapter 2, if the set of admissible initial states is suitably restricted. Let $\bar{\mathbf{u}}^0(\bar{x})$ denote the minimizing control sequence; the stage cost $\ell(\cdot)$ is chosen to ensure uniqueness of $\bar{\mathbf{u}}^0(\bar{x})$. The implicit model predictive control law for the nominal system is $\bar{\kappa}_N(\cdot)$ defined by

$$\bar{\kappa}_N(\bar{x}) := \bar{u}^0(0; \bar{x})$$

where $\bar{u}^0(0; \bar{x})$ is the first element in the sequence $\bar{\mathbf{u}}^0(\bar{x})$. The domain of $\bar{V}_N^0(\cdot)$ and $\bar{\mathbf{u}}^0(\cdot)$, and, hence, of $\bar{\kappa}_N(\cdot)$, is $\bar{\mathcal{X}}_N$ defined by

$$\bar{\mathcal{X}}_N := \{\bar{x} \in \bar{\mathbb{X}} \mid \bar{\mathcal{U}}_N(\bar{x}) \neq \emptyset\} \quad (5.22)$$

$\bar{\mathcal{X}}_N$ is the set of initial states $\bar{x}$ that can be steered to $\bar{\mathbb{X}}_f$ by an admissible control $\bar{\mathbf{u}}$ that satisfies the state and control constraints, (5.19) and (5.20), and the terminal constraint. From (5.14), the implicit control law for the state estimator $\hat{x}^+ = A\hat{x} + Bu + \delta$ is $\kappa_N(\cdot)$ defined by

$$\kappa_N(\hat{x}, \bar{x}) := \bar{\kappa}_N(\bar{x}) + K(\hat{x} - \bar{x})$$

The controlled composite system with state $(\hat{x}, \bar{x})$ satisfies

$$\hat{x}^+ = A\hat{x} + B\kappa_N(\hat{x}, \bar{x}) + \delta \quad (5.23)$$

$$\bar{x}^+ = A\bar{x} + B\bar{\kappa}_N(\bar{x}) \quad (5.24)$$

with initial state $(\hat{x}(0), \bar{x}(0))$ satisfying $\hat{x}(0) \in \{\bar{x}(0)\} \oplus \mathbb{S}$, $\bar{x}(0) \in \bar{\mathcal{X}}_N$. These constraints are satisfied if $\bar{x}(0) = \hat{x}(0) \in \bar{\mathcal{X}}_N$. The control algorithm may be formally stated as follows.

Algorithm 5.5 (Robust control algorithm (linear constrained systems))**.** First set $i = 0$, set $\hat{x} = \hat{x}(0)$, and set $\bar{x} = \hat{x}$. Then repeat

1. At time i, solve the nominal optimal control problem $\bar{\mathbb{P}}_N(\bar{x})$ to obtain the current nominal control action $\bar{u} = \bar{\kappa}_N(\bar{x})$ and the control $u = \bar{x} + K(\hat{x} - \bar{x})$.

2. Apply the control u to the system being controlled.

3. Compute the successor state estimate $\hat{x}^+$ and the successor state of the nominal system $\bar{x}^+$

$$\hat{x}^+ = A\hat{x} + Bu + L(y - C\hat{x}) \qquad \bar{x}^+ = A\bar{x} + B\bar{u}$$

4. Set $(\hat{x}, \bar{x}) = (\hat{x}^+, \bar{x}^+)$, set $i = i + 1$.

If the terminal cost $V_f(\cdot)$ and terminal constraint set $\bar{\mathbb{X}}_f$ satisfy the stability Assumption 2.14, and if Assumption 5.4 is satisfied, the value function $\bar{V}_N^0(\cdot)$ satisfies

$$\begin{aligned} \bar{V}_N^0(\bar{x}) &\geq \ell(\bar{x}, \bar{\kappa}_N(\bar{x})) && \forall \bar{x} \in \bar{\mathcal{X}}_N \\ \bar{V}_N^0(\bar{x}) &\leq V_f(\bar{x}) && \forall \bar{x} \in \bar{\mathcal{X}}_N \\ \bar{V}_N^0(f(\bar{x}, \bar{\kappa}_N(\bar{x}))) &\leq \bar{V}_N^0(\bar{x}) - \ell(\bar{x}, \bar{\kappa}_N(\bar{x})) && \forall \bar{x} \in \bar{\mathcal{X}}_N \end{aligned}$$

in which $\Delta\bar{V}_N^0(\bar{x}) := \bar{V}_N^0(f(\bar{x}, \bar{\kappa}_N(\bar{x}))) - \bar{V}_N^0(\bar{x})$.

As shown in Section 3.5.3, if, in addition to Assumption 5.4

1. the stability Assumption 2.14 is satisfied,

2. $\ell(\bar{x}, \bar{u}) = (1/2)(|\bar{x}|_Q^2 + |\bar{u}|_R^2)$ where Q and R are positive definite,

3. $V_f(\bar{x}) = (1/2)\,|\bar{x}|_{P_f}^2$ where P_f is positive definite, and

4. $\bar{\mathcal{X}}_N$ is a C-set,

then there exist positive constants c_1 and c_2 such that

$$\begin{aligned} \bar{V}_N^0(\bar{x}) &\geq c_1\,|\bar{x}|^2 && \forall \bar{x} \in \bar{\mathcal{X}}_N \\ \bar{V}_N^0(\bar{x}) &\leq c_2\,|\bar{x}|^2 && \forall \bar{x} \in \bar{\mathcal{X}}_N \\ \bar{V}_N^0(f(\bar{x}, \bar{\kappa}_N(\bar{x}))) &\leq \bar{V}_N^0(\bar{x}) - c_1\,|\bar{x}|^2 && \forall \bar{x} \in \bar{\mathcal{X}}_N \end{aligned}$$

It follows from Chapter 2 that the origin is exponentially stable for the nominal system $\bar{x}^+ = A\bar{x} + B\bar{\kappa}_N(\bar{x})$ with a region of attraction $\bar{\mathcal{X}}_N$ so that there exists a $c > 0$ and a $\gamma \in (0,1)$ such that

$$|\bar{x}(i)| \leq c\,|\bar{x}(0)|\,\gamma^i$$

for all $\bar{x}(0) \in \bar{\mathcal{X}}_N$, all $i \in \mathbb{I}_{\geq 0}$. Also $\bar{x}(i) \in \bar{\mathcal{X}}_N$ for all $i \in \mathbb{I}_{\geq 0}$ if $\bar{x}(0) \in \bar{\mathcal{X}}_N$ so that problem $\mathbb{P}_N(\bar{x}(i))$ is always feasible. Because the state $\hat{x}(i)$ of the state estimator always lies in $\{\bar{x}(i)\} \oplus \mathbb{S}$, and the state $x(i)$ of the system being controlled always lies in $\{\bar{x}(i)\} \oplus \mathbb{\Gamma}$, it follows that $\hat{x}(i)$ converges robustly and exponentially fast to $\mathbb{S}$, and $x(i)$ converges robustly and exponentially fast to $\mathbb{\Gamma}$. We are now in a position to establish exponential stability of $\mathcal{A} := \mathbb{S} \times \{0\}$ with a region of attraction $(\bar{\mathcal{X}}_N \oplus \mathbb{S}) \times \bar{\mathcal{X}}_N$ for the composite system (5.23) and (5.24).

Proposition 5.6 (Exponential stability of output MPC)**.** *The set* $\mathcal{A} := \mathbb{S} \times \{0\}$ *is exponentially stable with a region of attraction* $(\bar{\mathcal{X}}_N \oplus \mathbb{S}) \times \bar{\mathcal{X}}_N$ *for the composite system* (5.23) *and* (5.24)*.*

Proof. Let $\phi := (\hat{x}, \bar{x})$ denote the state of the composite system. Then $|\phi|_{\mathcal{A}}$ is defined by

$$|\phi|_{\mathcal{A}} = |\hat{x}|_{\mathbb{S}} + |\bar{x}|$$

where $|\hat{x}|_{\mathbb{S}} := d(\hat{x}, \mathbb{S})$. But $\hat{x} \in \{\bar{x}\} \oplus \mathbb{S}$ implies $\hat{x} = \bar{x} + e$ for some $e \in S$ so that

$$|\hat{x}|_{\mathbb{S}} = d(\hat{x}, \mathbb{S}) = d(\bar{x} + e, \mathbb{S}) \leq d(\bar{x} + e, e) = |\bar{x}|$$

since $e \in \mathbb{S}$. Hence $|\phi|_{\mathcal{A}} \leq 2\,|\bar{x}|$ so that

$$|\phi(i)|_{\mathcal{A}} \leq 2\,|\bar{x}(i)| \leq 2c\,|\bar{x}(0)|\,\gamma^i \leq 2c\,|\phi(0)|\,\gamma^i$$

for all $\phi(0) \in (\bar{\mathcal{X}}_N \oplus \mathbb{S}) \times \bar{\mathcal{X}}_N$. Since for all $\bar{x}(0) \in \bar{\mathcal{X}}_N$, $\bar{x}(i) \in \bar{\mathbb{X}}$ and $\bar{u}(i) \in \bar{\mathbb{U}}$, it follows that $\hat{x}(i) \in \{\bar{x}(i)\} \oplus \mathbb{S}$, $x(i) \in \mathbb{X}$, and $u(i) \in \mathbb{U}$ for all $i \in \mathbb{I}_{\geq 0}$. Thus $\mathcal{A} := \mathbb{S} \times \{0\}$ is exponentially stable with a region of attraction $(\bar{\mathcal{X}}_N \oplus \mathbb{S}) \times \bar{\mathcal{X}}_N$ for the composite system (5.23) and (5.24). ■

It follows from Proposition 5.6 that $x(i)$, which lies in the set $\{\bar{x}(i)\} \oplus \mathbb{\Gamma}$, $\mathbb{\Gamma} := \mathbb{S} \oplus \mathbb{\Sigma}$, converges to the set $\mathbb{\Gamma}$. In fact $x(i)$ converges to a set that is, in general, smaller than $\mathbb{\Gamma}$ since $\mathbb{\Gamma}$ is a conservative bound on $\tilde{x}(i) + e(i)$. We determine this smaller set as follows. Let $\phi := (\tilde{x}, e)$ and let $\psi := (w, v)$; ϕ is the state of the two error systems and ψ is a bounded disturbance lying in a C-set $\mathbb{\Psi} := \mathbb{W} \times \mathbb{N}$. Then, from (5.10) and (5.16), the state ϕ evolves according to

$$\phi^+ = \tilde{A}\phi + \tilde{B}\psi \tag{5.25}$$

where

$$\widetilde{A} := \begin{bmatrix} A_L & 0 \\ LC & A_K \end{bmatrix} \qquad \widetilde{B} := \begin{bmatrix} I & -L \\ 0 & L \end{bmatrix}$$

Because $\rho(A_L) < 1$ and $\rho(A_K) < 1$, it follows that $\rho(\widetilde{A}) < 1$. Since $\rho(\widetilde{A}) < 1$ and $\mathbb{\Psi}$ is compact, there exists a robust positive invariant set $\mathbb{\Phi} \subseteq \mathbb{R}^n \times \mathbb{R}^n$ for (5.25) satisfying

$$\widetilde{A}\mathbb{\Phi} \oplus \widetilde{B}\mathbb{\Psi} = \mathbb{\Phi}$$

Hence $\phi(i) \in \mathbb{\Phi}$ for all $i \in \mathbb{I}_{\geq 0}$ if $\phi(0) \in \mathbb{\Phi}$. Since $x(i) = \bar{x}(i) + e(i) + \widetilde{x}(i)$, it follows that $x(i) \in \{\bar{x}(i)\} \oplus H\mathbb{\Phi}$, $H := \begin{bmatrix} I_n & I_n \end{bmatrix}$, for all $i \in \mathbb{I}_{\geq 0}$ provided that $x(0)$, $\hat{x}(0)$, and $\bar{x}(0)$ satisfy $(\widetilde{x}(0), e(0)) \in \mathbb{\Phi}$ where $\widetilde{x}(0) = x(0) - \hat{x}(0)$ and $e(0) = \hat{x}(0) - \bar{x}(0)$. If these initial conditions are satisfied, $x(i)$ converges robustly and exponentially fast to the set $H\mathbb{\Phi}$.

The remaining robust controllers presented in Section 3.5 may be similarly modified to obtain a robust output model predictive controller.

5.3.5 Computing the Tightened Constraints

The analysis above shows the tightened state and control constraint sets $\bar{\mathbb{X}}$ and $\bar{\mathbb{U}}$ for the nominal optimal control problem can, in principle, be computed using set algebra. Polyhedral set computations are not robust, however, and usually are limited to sets in $\mathbb{R}^n$ with $n \leq 15$. So we present here an alternative method for computing tightened constraints, similar to that described in 3.5.3.

We next show how to obtain a conservative approximation to $\bar{\mathbb{X}} \subseteq \mathbb{X} \ominus \mathbb{\Gamma}$, $\mathbb{\Gamma} = \mathbb{S} \oplus \mathbb{Z}$. Suppose $c'x \leq d$ is one of the constraints defining $\mathbb{X}$. Since $e = \hat{x} - \bar{x}$, which lies in $\mathbb{S}$, and $\widetilde{x} = x - \hat{x}$, which lies in $\mathbb{Z}$, satisfy $e^+ = A_K e + LC\widetilde{x} + L\nu$ and $\widetilde{x}^+ = A_L\widetilde{x} + w - L\nu$, the constraint $c'x \leq d$ (one of the constraints defining $\mathbb{X}$), the corresponding constraint in $\bar{\mathbb{X}}$ should be $c'x \leq d - \phi_\infty^{\bar{\mathbb{X}}}$ in which

$$\begin{aligned}\phi_\infty^{\bar{\mathbb{X}}} &= \max\{c'e \mid e \in \mathbb{S}\} + \max\{c'\widetilde{x} \mid \widetilde{x} \in \mathbb{Z}\} \\ &= \max_{(w(i),\nu(i))} \sum_{j=0}^{\infty} A_K^j(LC\widetilde{x}(j) + L\nu(j)) + \max_{(w(i),\nu(i))} \sum_{j=0}^{\infty} A_L^j(w(j) - L\nu(j))\end{aligned}$$

in which $\widetilde{x}(j) = \sum_{i=0}^{j-1} A_L^i(w(i) - L\nu(i))$. The maximizations are subject to the constraints $w(i) \in \mathbb{W}$ and $\nu(i) \in \mathbb{N}$ for all $i \in \mathbb{I}_{\geq 0}$. Because

maximization over an infinite horizon is impractical, we determine, as in 3.5.3, a horizon $N \in \mathbb{I}_{\geq 0}$ and an $\alpha \in (0,1)$ such that $A_K^N \mathbb{W} \subset \alpha \mathbb{W}$ and $A_L^N \mathbb{N} \subset \alpha \mathbb{N}$, and define the constraint in $\bar{\mathbb{X}}$ corresponding to the constraint $c'x \leq d$ in $\mathbb{X}$ to be $c'x \leq d - (1-\alpha^{-1})\phi_N^{\bar{\mathbb{X}}}$ with

$$\phi_N^{\bar{\mathbb{X}}} = \max_{(w(i),v(i))} \sum_{j=0}^{N-1} A_K^j (LC\tilde{x}(j) + Lv(j)) + \max_{(w(i),v(i))} \sum_{j=0}^{N-1} A_L^j (w(j) - Lv(j))$$

The tightened constraints yielding a conservative approximation to $\bar{\mathbb{U}} := \mathbb{U} \ominus KS$ may be similarly computed. The constraint $c'u \leq d$, one of the constraints defining $\mathbb{U}$, should be replaced by $c'u \leq d - (1-\alpha)^{-1}\phi_N^{\bar{\mathbb{U}}}$ with

$$\phi_N^{\bar{\mathbb{U}}} = \max\{c'e \mid e \in KS\} = \max_{(w(i),v(i))} \sum_{j=0}^{N-1} KA_K^j (LC\tilde{x}(j) + Lv(j))$$

The maximizations for computing $\phi_N^{\bar{\mathbb{X}}}$ and $\phi_N^{\bar{\mathbb{U}}}$ are subject to the constraints $w(i) \in \mathbb{W}$ and $v(i) \in \mathbb{N}$ for all $i \in \mathbb{I}_{\geq 0}$.

5.4 Linear Constrained Systems: Time-Varying Case

The time-invariant case corresponds to the "steady-state" situation in which the sets $\mathbb{S}(t)$ and $\mathbb{Z}(t)$ have settled down to their steady-state values $\mathbb{S}$ and $\mathbb{Z}$, respectively. As a result the constraint sets $\bar{\mathbb{X}}$ and $\bar{\mathbb{U}}$ are also time invariant. When the state is accessible, the constraint $x \in \bar{\mathbb{X}}(i) := \mathbb{X} \ominus S(i)$ is less conservative than $x \in \bar{\mathbb{X}} = \mathbb{X} \ominus S$, in which $S = S(\infty)$. This relaxation of the constraint may be useful in some applications. The version of tube-based MPC employed here is such that $\mathbb{S}(t+1) \supset \mathbb{S}(t)$ for all t so that $\mathbb{S}(t)$ converges to $\mathbb{S}(\infty)$ as $t \to \infty$. In other versions of tube-based MPC, in which $\mathbb{P}_N(x)$ rather $\mathbb{P}_N(\bar{x})$ is solved online, $\mathbb{S}(t)$ is reset to the empty set so that advantage in using $\mathbb{S}(t)$ rather than $\mathbb{S}(\infty)$ is larger. On the other hand, the state estimation set $\mathbb{Z}(t)$ may increase or decrease with t depending on prior information. The time-varying version of tube-based MPC is fully discussed in Mayne, Raković, Findeisen, and Allgöwer (2009).

5.5 Offset-Free MPC

Offset-free MPC was introduced in Chapters 1 and 2 in a deterministic context; see also Pannocchia and Rawlings (2003). Suppose the system

Set	Definition	Membership
$\mathbb{X}$	state constraint set	$x \in \mathbb{X}$
$\mathbb{U}$	input constraint set	$u \in \mathbb{U}$
$\mathbb{W}_x$	state disturbance set	$w_x \in \mathbb{W}_x$
$\mathbb{W}_d$	integrating disturbance set	$w_d \in \mathbb{W}_d$
$\mathbb{W}$	total state disturbance set, $\mathbb{W}_x \times \mathbb{W}_d$	$w \in \mathbb{W}$
$\mathbb{N}$	measurement error set	$v \in \mathbb{N}$
$\widetilde{\mathbb{W}}$	estimate error disturbance set, $\mathbb{W} \oplus (-L\mathbb{N})$	$\tilde{w} \in \widetilde{\mathbb{W}}$
Φ	total estimate error disturbance set, $\Phi = \widetilde{A}_L \Phi \oplus \widetilde{\mathbb{W}}$	$\phi \in \Phi$
$\mathbb{E}_x$	state estimate error disturbance set, $\begin{bmatrix} I_n & 0 \end{bmatrix} \Phi$	$\tilde{x} \in \mathbb{E}_x$
$\mathbb{E}_d$	integrating disturbance estimate error set, $\begin{bmatrix} 0 & I_p \end{bmatrix} \Phi$	$\tilde{d} \in \mathbb{E}_d$
$\mathbb{A}$	innovation set, $L(\widetilde{C}\Phi \oplus \mathbb{N})$	$L\tilde{y} \in \mathbb{A}$
$\mathbb{A}_x$	set containing state component of innovation, $L_x(\widetilde{C}\Phi \oplus \mathbb{N})$	$L_x\tilde{y} \in \mathbb{A}_x$
$\mathbb{A}_d$	set containing integrating disturbance component of innovation, $L_d(\widetilde{C}\Phi \oplus \mathbb{N})$	$L_d\tilde{y} \in \mathbb{A}_d$
$\mathbb{S}$	nominal state tracking error invariance set, $A_K\mathbb{S} \oplus \mathbb{A}_x = \mathbb{S}$	$e \in \mathbb{S}$ $\hat{x} \in \{\bar{x}\} + \mathbb{S}$
$\mathbb{\Gamma}$	state tracking error invariance set, $\mathbb{S} + \mathbb{E}_x$	$x \in \{\bar{x}\} + \mathbb{\Gamma}$
$\bar{\mathbb{U}}$	nominal input constraint set, $\bar{\mathbb{U}} = \mathbb{U} \ominus K\mathbb{S}$	$\bar{u} \in \bar{\mathbb{U}}$
$\bar{\mathbb{X}}$	nominal state constraint set, $\bar{\mathbb{X}} = \mathbb{X} \ominus \mathbb{\Gamma}$	$: \bar{x} \in \bar{\mathbb{X}}$

Table 5.1: Summary of the sets and variables used in output MPC.

to be controlled is described by

$$\begin{aligned} x^+ &= Ax + B_d d + Bu + w_x \\ y &= Cx + C_d d + v \\ r &= Hy \qquad \tilde{r} = r - \bar{r} \end{aligned}$$

in which w_x and v are unknown bounded disturbances taking values, respectively, in the compact sets $\mathbb{W}_x$ and $\mathbb{N}$ containing the origin in their interiors. In the following discussion, $y = Cx + C_d d$ is the output of the system being controlled, r is the controlled variable, and $\bar{r}$ is its setpoint. The variable $\tilde{r}$ is the tracking error that we wish to minimize. We assume d is nearly constant but drifts slowly, and model its

behavior by

$$d^+ = d + w_d$$

where w_d is a bounded disturbance taking values in the compact set $\mathbb{W}_d$; in practice d is bounded, although this is not implied by our model. We assume that $x \in \mathbb{R}^n$, $d \in \mathbb{R}^p$, $u \in \mathbb{R}^m$, $y \in \mathbb{R}^r$, and $e \in \mathbb{R}^q$, $q \leq r$, and that the system to be controlled is subject to the usual state and control constraints

$$x \in \mathbb{X} \qquad u \in \mathbb{U}$$

We assume $\mathbb{X}$ is polyhedral and $\mathbb{U}$ is polytopic.

Given the many sets that are required to specify the output feedback case we are about to develop, Table 5.1 may serve as a reference for the sets defined in the chapter and the variables that are members of these sets.

5.5.1 Estimation

Since both x and d are unknown, it is necessary to estimate them. For estimation purposes, it is convenient to work with the composite system whose state is $\phi := (x, d)$. This system may be described more compactly by

$$\phi^+ = \widetilde{A}\phi + \widetilde{B}u + w$$
$$y = \widetilde{C}\phi + v$$

in which $w = (w_x, w_d)$ and

$$\widetilde{A} := \begin{bmatrix} A & B_d \\ 0 & I \end{bmatrix} \qquad \widetilde{B} := \begin{bmatrix} B \\ 0 \end{bmatrix} \qquad \widetilde{C} := \begin{bmatrix} C & C_d \end{bmatrix}$$

and $w := (w_x, w_d)$ takes values in $\mathbb{W} = \mathbb{W}_x \times \mathbb{W}_d$. A necessary and sufficient condition for the detectability of $(\widetilde{A}, \widetilde{C})$ is given in Lemma 1.8. A sufficient condition is detectability of (A, C), coupled with invertibility of C_d. If $(\widetilde{A}, \widetilde{C})$ is detectable, the state may be estimated using the time-invariant observer or filter described by

$$\hat{\phi}^+ = \widetilde{A}\hat{\phi} + \widetilde{B}u + \delta \qquad \delta := L(y - \widetilde{C}\hat{\phi})$$

in which L is such that $\rho(\widetilde{A}_L) < 1$ where $\widetilde{A}_L := \widetilde{A} - L\widetilde{C}$. Clearly $\delta = L\widetilde{y}$ where $\widetilde{y} = \widetilde{C}\widetilde{\phi} + v$. The estimation error $\widetilde{\phi} := \phi - \hat{\phi}$ satisfies

$$\widetilde{\phi}^+ = \widetilde{A}\widetilde{\phi} + w - L(\widetilde{C}\widetilde{\phi} + v)$$

or, in simpler form

$$\tilde{\phi}^+ = \tilde{A}_L\tilde{\phi} + \tilde{w} \qquad \tilde{w} := w - L\nu$$

Clearly $\tilde{w} = w - L\nu$ takes values in the compact set $\tilde{\mathbb{W}}$ defined by

$$\tilde{\mathbb{W}} := \mathbb{W} \oplus (-L\mathbb{N})$$

If w and ν are zero, $\tilde{\phi}$ decays to zero exponentially fast so that $\hat{x} \to \tilde{x}$ and $\hat{d} \to d$ exponentially fast. Since $\rho(\tilde{A}_L) < 1$ and $\tilde{\mathbb{W}}$ is compact, there exists a robust positive invariant set Φ for $\tilde{\phi}^+ = \tilde{A}_L\tilde{\phi} + \tilde{w}$, $\tilde{w} \in \tilde{\mathbb{W}}$ satisfying

$$\Phi = \tilde{A}_L\Phi \oplus \tilde{\mathbb{W}}$$

Hence $\tilde{\phi}(i) \in \Phi$ for all $i \in \mathbb{I}_{\geq 0}$ if $\tilde{\phi}(0) \in \Phi$. Since $\tilde{\phi} = (\tilde{x}, \tilde{d}) \in \mathbb{R}^n \times \mathbb{R}^p$ where $\tilde{x} := x - \hat{x}$ and $\tilde{d} := d - \hat{d}$, we define the sets $\mathbb{Z}_x$ and $\mathbb{Z}_d$ as follows

$$\mathbb{Z}_x := \begin{bmatrix} I_n & 0 \end{bmatrix} \Phi \qquad \mathbb{Z}_d := \begin{bmatrix} 0 & I_p \end{bmatrix} \Phi$$

It follows that $\tilde{x}(i) \in \mathbb{Z}_x$ and $\tilde{d}(i) \in \mathbb{Z}_d$ so that $x(i) \in \{\hat{x}\} \oplus \Sigma_x$ and $d(i) \in \{\hat{d}(i)\} \oplus \Sigma_d$ for all $i \in \mathbb{I}_{\geq 0}$ if $\tilde{\phi}(0) = (\tilde{x}(0), \tilde{d}(0)) \in \Phi$. That $\tilde{\phi}(0) \in \Phi$ is a steady-state assumption.

5.5.2 Control

The estimation problem has a solution similar to previous solutions. The control problem is more difficult. As before, we control the estimator state, making allowance for state estimation error. The estimator state $\hat{\phi}$ satisfies the difference equation

$$\hat{\phi}^+ = \tilde{A}\hat{\phi} + \tilde{B}u + \delta$$

where the disturbance δ is defined by

$$\delta := L\tilde{y} = L(\tilde{C}\tilde{\phi} + \nu)$$

The disturbance $\delta = (\delta_x, \delta_d)$ lies in the $C-$set $\mathbb{\Delta}$ defined by

$$\mathbb{\Delta} := L(\tilde{C}\Phi \oplus \mathbb{N})$$

where the set Φ is defined in Section 5.5.1. The system $\hat{\phi}^+ = \tilde{A}\hat{\phi} + \tilde{B}u + \delta$ is not stabilizable, however, so we examine the subsystems with states $\hat{x}$ and $\hat{d}$

$$\hat{x}^+ = A\hat{x} + B_d\hat{d} + Bu + \delta_x$$
$$\hat{d}^+ = \hat{d} + \delta_d$$

where the disturbances δ_x and δ_d are components of $\delta = (\delta_x, \delta_d)$ and are defined by

$$\delta_x := L_x\tilde{y} = L_x(\tilde{C}\tilde{\phi} + \nu) \qquad \delta_d := L_d\tilde{y} = L_d(\tilde{C}\tilde{\phi} + \nu)$$

The matrices L_x and L_d are the corresponding components of L. The disturbance δ_x and δ_d lie in the $C-$sets $\mathbb{\Delta}_x$ and $\mathbb{\Delta}_d$ defined by

$$\mathbb{\Delta}_x := \begin{bmatrix} I_n & 0 \end{bmatrix} \mathbb{\Delta} = L_x[\tilde{C}\Phi \oplus \mathbb{N}] \qquad \mathbb{\Delta}_d := \begin{bmatrix} 0 & I_p \end{bmatrix} \mathbb{\Delta} = L_d[\tilde{C}\Phi \oplus \mathbb{N}]$$

We assume that (A, B) is a stabilizable pair so the tube methodology may be employed to control $\hat{x}$. The system $\hat{d}^+ = \hat{d} + \delta_d$ is uncontrollable. The central trajectory is therefore chosen to be the nominal version of the difference equation for $(\hat{x}, \hat{d})$ and is described by

$$\begin{aligned} \bar{x}^+ &= A\bar{x} + B_d\hat{d} + B\bar{u} \\ \bar{d}^+ &= \bar{d} \end{aligned}$$

in which the initial state is $(\hat{x}, \hat{d})$. We obtain $\bar{u} = \bar{\kappa}_N(\bar{x}, \bar{d}, \bar{r})$ by solving a nominal optimal control problem defined later and set $u = \bar{u} + Ke$, $e := \hat{x} - \bar{x}$ where K is chosen so that $\rho(A_K) < 1$, $A_K := A + BK$; this is possible since (A, B) is assumed to be stabilizable. It follows that $e := \hat{x} - \bar{x}$ satisfies the difference equation

$$e^+ = A_K e + \delta_x \qquad \delta_x \in \mathbb{\Delta}_x$$

Because $\mathbb{\Delta}_x$ is compact and $\rho(A_K) < 1$, there exists a robust positive invariant set $\mathbb{S}$ for $e^+ = A_K e + \delta_x$, $\delta_x \in \mathbb{\Delta}_x$ satisfying

$$A_K\mathbb{S} \oplus \mathbb{\Delta}_x = \mathbb{S}$$

Hence $e(i) \in \mathbb{S}$ for all $i \in \mathbb{I}_{\geq 0}$ if $e(0) \in \mathbb{S}$. So, as in Proposition 5.3, the states and controls of the estimator and nominal system satisfy $\hat{x}(i) \in \{\bar{x}(i)\} \oplus \mathbb{S}$ and $u(i) \in \{\bar{u}(i)\} \oplus K\mathbb{S}$ for all $i \in \mathbb{I}_{\geq 0}$ if the initial states $\hat{x}(0)$ and $\bar{x}(0)$ satisfy $\hat{x}(0) \in \{\bar{x}(0)\} \oplus \mathbb{S}$. Using the fact established previously that $\tilde{x}(i) \in \mathbb{Z}_x$ for all i, we can also conclude that $x(i) = \bar{x}(i) + e(i) + \tilde{x}(i) \in \{\bar{x}(i)\} \oplus \mathbb{\Gamma}$ and that $u(i) = \bar{u}(i) + Ke(i) \in \{\bar{u}(i)\} + K\mathbb{S}$ for all i where $\mathbb{\Gamma} := \mathbb{S} \oplus \mathbb{Z}_x$ provided, of course, that $\phi(0) \in \{\hat{\phi}(0)\} \oplus \Phi$ and $x(0) \in \{\hat{x}(0)\} \oplus \mathbb{S}$. These conditions are equivalent to $\tilde{\phi}(0) \in \Phi$ and $e(0) \in \mathbb{S}$ where, for all i, $e(i) := \hat{x}(i) - \bar{x}(i)$. Hence $x(i)$ lies in $\mathbb{X}$ and $u(i)$ lies in $\mathbb{U}$ if $\bar{x}(i) \in \bar{\mathbb{X}} := \mathbb{X} \ominus \mathbb{\Gamma}$ and $\bar{u}(i) \in \bar{\mathbb{U}} := \mathbb{U} \ominus K\mathbb{S}$.

Thus $\hat{x}(i)$ and $x(i)$ evolve in known neighborhoods of the central state $\bar{x}(i)$ that we can control. Although we know that the uncontrollable state $d(i)$ lies in the set $\{\hat{d}(i)\} \oplus i\mathbb{Z}_d$ for all i, the evolution of $\hat{d}(i)$

is an uncontrollable random walk and is, therefore, unbounded. If the initial value of $\hat{d}$ at time 0 is $\hat{d}_0$, then $\hat{d}(i)$ lies in the set $\{\hat{d}_0\} \oplus i\mathbb{Z}_d$ that increases without bound as i increases. This behavior is a defect in our model for the disturbance d; the model is useful for estimation purposes, but is unrealistic in permitting unbounded values for d. Hence we assume in the sequel that d evolves in a compact $C-$set X_d. We can modify the observer to ensure that $\hat{d}$ lies in X_d, but find it simpler to observe that if d lies in X_d, $\hat{d}$ must lie in $X_d \oplus \mathbb{Z}_d$.

Target Calculation. Our first task is to determine the target state $\bar{x}_s$ and associated control $\bar{u}_s$; we require our estimate of the tracking error $\tilde{r} = r - \bar{r}$ to be zero in the absence of any disturbances. We follow the procedure outlined in Pannocchia and Rawlings (2003). Since our estimate of the measurement noise v is 0 and since our best estimate of d when the target state is reached is $\hat{d}$, we require

$$\hat{r} - \bar{r} = H(C\bar{x}_s + C_d\hat{d}) - \bar{r} = 0$$

We also require the target state to be an equilibrium state satisfying, therefore, $\bar{x}_s = A\bar{x}_s + B_d\hat{d} + B\bar{u}_s$ for some control $\bar{u}_s$. Given $(\hat{d}, \bar{r})$, the target equilibrium pair $(\bar{x}_s, \bar{u}_s)(\hat{d}, \bar{r})$ is computed as follows

$$(\bar{x}_s, \bar{u}_s)(\hat{d}, \bar{r}) = \arg\min_{\bar{x},\bar{u}} \{L(\bar{x}, \bar{u}) \mid \bar{x} = A\bar{x} + B_d\hat{d} + B\bar{u}, \\ H(C\bar{x} + C_d\hat{d}) = \bar{r}, \bar{x} \in \mathbb{X},\ \bar{u} \in \mathbb{U}\}$$

where $L(\cdot)$ is an appropriate cost function; e.g., $L(\bar{x}, \bar{u}) = (1/2)\,|\bar{u}|_R^2$. The equality constraints in this optimization problem can be satisfied if the matrix $\left[\begin{smallmatrix} I-A & -B \\ HC & 0 \end{smallmatrix}\right]$ has full rank. As the notation indicates, the target equilibrium pair $(\bar{x}_s, \bar{u}_s)(\hat{d}, \bar{r})$ is not constant, but varies with the estimate of the disturbance state d.

MPC algorithm. The control objective is to steer the central state $\bar{x}$ to a small neighborhood of the target state $\bar{x}_s(\hat{d}, \bar{r})$ while satisfying the state and control constraints $x \in \mathbb{X}$ and $u \in \mathbb{U}$. It is desirable that $\bar{x}(i)$ converges to $\bar{x}_s(\hat{d}, \bar{r})$ if $\hat{d}$ remains constant, in which case $x(i)$ converges to the set $\{\bar{x}_s(\hat{d}, \bar{r})\} \oplus \mathbb{\Gamma}$. We are now in a position to specify the optimal control problem whose solution yields $\bar{u} = \bar{\kappa}_N(\bar{x}, \hat{d}, \bar{r})$ and, hence, $u = \bar{u} + K(\hat{x} - \bar{x})$.To achieve this objective, we define the deterministic optimal control problem

$$\bar{\mathbb{P}}_N(\bar{x}, \hat{d}, \bar{r}): \quad V_N^0(\bar{x}, \hat{d}, \bar{r}) := \min_{\bar{\mathbf{u}}} \{V_N(\bar{x}, \hat{d}, \bar{r}, \bar{\mathbf{u}}) \mid \bar{\mathbf{u}} \in \bar{\mathcal{U}}_N(\bar{x}, \hat{d}, \bar{r})\}$$

in which the cost $V_N(\cdot)$ and the constraint set $\bar{\mathcal{U}}_N(\bar{x}, \hat{d}, \bar{r})$ are defined by

$$V_N(\bar{x}, \hat{d}, \bar{r}, \bar{\mathbf{u}}) := \sum_{i=0}^{N-1} \ell(\bar{x}(i) - \bar{x}_s(\hat{d}, \bar{r}), \bar{u}(i) - \bar{u}_s(\hat{d}, \bar{r})) + V_f(\bar{x}(N), \bar{x}_s(\hat{d}, \bar{r}))$$

$$\bar{\mathcal{U}}_N(\bar{x}, \hat{d}, \bar{r}) := \{\bar{\mathbf{u}} \mid \bar{x}(i) \in \bar{\mathbb{X}}, \bar{u}(i) \in \bar{\mathbb{U}} \ \forall i \in \mathbb{I}_{0:N-1}, \ \bar{x}(N) \in \bar{\mathbb{X}}_f(\bar{x}_s(\hat{d}, \bar{r}))\}$$

and, for each i, $\bar{x}(i) = \bar{\phi}(i; \bar{x}, \hat{d}, \bar{\mathbf{u}})$, the solution of $\bar{x}^+ = A\bar{x} + B_d\hat{d} + B\bar{u}$ when the initial state is $\bar{x}$, the control sequence is $\bar{\mathbf{u}}$, and the disturbance $\hat{d}$ is constant, i.e., satisfies the nominal difference equation $\hat{d}^+ = \hat{d}$. The set of feasible $(\bar{x}, \hat{d}, \bar{r})$ and the set of feasible states $\bar{x}$ for $\bar{\mathbb{P}}_N(\bar{x}, \hat{d}, \bar{r})$ are defined by

$$\bar{\mathcal{F}}_N := \{(\bar{x}, \hat{d}, \bar{r}) \mid \mathcal{U}_N(\bar{x}, \hat{d}, \bar{r}) \neq \emptyset\} \qquad \bar{\mathcal{X}}_N(\hat{d}, \bar{r}) := \{\bar{x} \mid (\bar{x}, \hat{d}, \bar{r}) \in \bar{\mathcal{F}}_N\}$$

The terminal cost is zero when the terminal state is equal to the target state. The solution to $\bar{\mathbb{P}}_N(\bar{x}, \hat{d}, \bar{r})$ is

$$\bar{\mathbf{u}}^0(\bar{x}, \hat{d}, \bar{r}) = \{\bar{u}^0(0; \bar{x}, \hat{d}, \bar{r}), \bar{u}^0(1; \bar{x}, \hat{d}, \bar{r}), \ldots, \bar{u}^0(N-1; \bar{x}, \hat{d}, \bar{r})\}$$

and the implicit model control law $\bar{\kappa}_N(\cdot)$ is defined by

$$\bar{\kappa}_N(\bar{x}, \hat{d}, \bar{r}) := \bar{u}^0(0; \bar{x}, \hat{d}, \bar{r})$$

where $\bar{u}^0(0; \bar{x}, \hat{d}, \bar{r})$ is the first element in the sequence $\bar{\mathbf{u}}^0(\bar{x}, \hat{d}, \bar{r})$. The control u applied to the plant and the observer is $u = \kappa_N(\hat{x}, \bar{x}, \hat{d}, \bar{r})$ where $\kappa_N(\cdot)$ is defined by

$$\kappa_N(\hat{x}, \bar{x}, \hat{d}, \bar{r}) := \bar{\kappa}_N(\bar{x}, \hat{d}, \bar{r}) + K(\hat{x} - \bar{x})$$

Although the optimal control problem $\bar{\mathbb{P}}_N(\bar{x}, \hat{d}, \bar{r})$ is deterministic, $\hat{d}$ is random, so that the sequence $(\bar{x}(i))$, which satisfies $\bar{x}^+ = A\bar{x} + B_d\hat{d} + B\bar{\kappa}_N(\bar{x}, \hat{d}, \bar{r})$, is random, unlike the case discussed in Chapter 3. The control algorithm may now be formally stated.

Algorithm 5.7 (Robust control algorithm (offset-free MPC))**.**

1. At time 0, set $i = 0$, set $\hat{\phi} = \hat{\phi}(0)$ ($\hat{\phi} = (\hat{x}, \hat{d})$, and set $\bar{x} = \hat{x}$.

2. At time i, solve the "nominal" optimal control problem $\bar{\mathbb{P}}_N(\bar{x}, \hat{d}, \bar{r})$ to obtain the current "nominal" control action $\bar{u} = \bar{\kappa}_N(\bar{x}, \hat{d}, \bar{r})$ and the control action $u = \bar{u} + K(\hat{x} - \bar{x})$.

3. If $\bar{\mathbb{P}}_N(\bar{x}, \hat{d}, \bar{r})$ is infeasible, adopt safety/recovery procedure.

4. Apply the control u to the system being controlled.

5. Compute the successor state estimate $\hat{\phi}^+ = \widetilde{A}\hat{x} + \widetilde{B}u + L(y - \widetilde{C}\hat{\phi})$.

6. Compute the successor state $\bar{x}^+ = A\bar{x} + B_d\hat{d} + B\bar{u}$ of the nominal system.

7. Set $(\hat{\phi}, \bar{x}) = (\hat{\phi}^+, \bar{x}^+)$, set $i = i + 1$.

In normal operation, Step 2 is not activated; Propositions 5.2 and 5.3 ensure that the constraints $\hat{x} \in \{\bar{x}\} \oplus \mathbb{S}$ and $u \in \{\bar{u}\} \oplus K\mathbb{S}$ are satisfied. If an unanticipated event occurs and Step 2 is activated, the controller can be reinitialized by setting $\bar{u} = \bar{\kappa}_N(\hat{x}, \hat{d}, \bar{r})$, setting $u = \bar{u}$, and relaxing constraints if necessary.

5.5.3 Convergence Analysis

We give here an informal discussion of the stability properties of the controller. The controller described above is motivated by the following consideration: nominal MPC is able to handle "slow" uncertainties such as the drift of a target point."Fast" uncertainties, however, are better handled by the tube controller that generates, using MPC, a suitable central trajectory and a "fast" ancillary controller to steer trajectories of the uncertain system toward the central trajectory. As shown above, the controller ensures that $x(i) \in \{\bar{x}(i)\} \oplus \mathbb{\Gamma}$ for all i; its success therefore depends on the ability of the controlled nominal system $\bar{x}^+ = A\bar{x} + B_d\hat{d} + B\bar{\kappa}_N(\bar{x}, \hat{d}, \bar{r})$, $\bar{u} = \bar{\kappa}_N(\bar{x}, \hat{d}, \bar{r})$, to track the target $\bar{x}_s(\hat{d}, \bar{r})$ that varies as $\hat{d}$ evolves.

Assuming that the standard stability assumptions are satisfied for the nominal optimal control problem $\bar{\mathbb{P}}_N(\bar{x}, \hat{d}, \bar{r})$ defined above, we have

$$
\begin{aligned}
V_N^0(\bar{x}, \hat{d}, \bar{r}) &\geq c_1 \left| \bar{x} - \bar{x}_s(\hat{d}, \bar{r}) \right|^2 \\
V_N^0(\bar{x}, \hat{d}, \bar{r}) &\leq c_2 \left| \bar{x} - \bar{x}_s(\hat{d}, \bar{r}) \right|^2 \\
V_N^0(\bar{x}^+, \hat{d}, \bar{r}) &\leq V_N^0(\bar{x}, \hat{d}, \bar{r}) - c_1 \left| \bar{x} - \bar{x}_s(\hat{d}, \bar{r}) \right|^2
\end{aligned}
$$

with $\bar{x}^+ = A\bar{x} + B_d\hat{d} + B\bar{\kappa}_N(\bar{x}, \hat{d}, \bar{r})$, for all $(\bar{x}, \hat{d}, \bar{r}) \in \bar{\mathcal{F}}_N$. The first and last inequalities follow from our assumptions; we assume the existence of the upper bound in the second inequality. The inequalities hold for all $(\bar{x}, \hat{d}, \bar{r}) \in \bar{\mathcal{F}}_N$. Note that the last inequality does NOT ensure $V_N^0(\bar{x}^+, \hat{d}^+, \bar{r}) \le V_N^0(\bar{x}, \hat{d}, \bar{r}) - c_1 \left| \bar{x} - \bar{x}_s(\hat{d}, \bar{r}) \right|^2$ with $\bar{x}^+ = A\bar{x} + B_d\hat{d} + B\bar{\kappa}_N(\bar{x}, \hat{d}, \bar{r})$ and $\hat{d}^+ := \hat{d} + \delta_d$. The perturbation due to δ_d has to be taken into account when analyzing stability.

Constant $\hat{d}$. If $\hat{d}$ remains constant, $\bar{x}_s(\hat{d}, \bar{r})$ is exponentially stable for $\bar{x}^+ = A\bar{x} + B_d\hat{d} + B\bar{\kappa}_N(\bar{x}, \hat{d}, \bar{r})$ with a region of attraction $\bar{\mathcal{X}}_N(\hat{d}, \bar{r})$. It can be shown, as in the proof of Proposition 5.6, that the set $\mathcal{A}(\hat{d}, \bar{r}) := (\{\bar{x}_s(\hat{d}, \bar{r})\} \oplus \mathbb{S}) \times \{\bar{x}_s(\hat{d}, \bar{r})\}$ is exponentially stable for the composite system $\hat{x}^+ = A\hat{x} + B_d\hat{d} + B\kappa_N(\hat{x}, \bar{x}, \hat{d}, \bar{r}) + \delta_x$, $\bar{x}^+ = A\bar{x} + B_d\hat{d} + B\bar{\kappa}_N(\bar{x}, \hat{d})$, $\delta_x \in \mathbb{\Delta}_x$, with a region of attraction $(\bar{\mathcal{X}}_N(\hat{d}, \bar{r}) \oplus \mathbb{S}) \times \bar{\mathcal{X}}_N(\hat{d}, \bar{r})$. Hence $x(i) \in \{\bar{x}(i)\} \oplus \mathbb{\Gamma}$ tends to the set $\{\bar{x}_s(\hat{d}, \bar{r})\} \oplus \mathbb{\Gamma}$ as $i \to \infty$. If, in addition, $\mathbb{W} = \{0\}$ and $\mathbb{N} = \{0\}$, then $\mathbb{\Delta} = \{0\}$ and $\mathbb{\Gamma} = \mathbb{\Sigma} = \mathbb{S} = \{0\}$ so that $x(i) \to \bar{x}_s(d, \bar{r})$ and $\widetilde{r}(i) \to 0$ as $i \to \infty$.

Slowly varying $\hat{d}$. If $\hat{d}$ is varying, the descent property of $V_N^0(\cdot)$ is modified and it is necessary to obtain an upper bound for $V_N^0(A\bar{x} + B_d(\hat{d} + \delta_d) + B\bar{\kappa}_N(\bar{x}, \hat{d}, \bar{r}), \hat{d} + \delta_d, \bar{r})$. We make use of Proposition 3.4 in Chapter 3. If $\bar{\mathcal{X}}_N$ is compact and if $(\hat{d}, \bar{r}) \mapsto \bar{x}_s(\hat{d}, \bar{r})$ and $(\hat{d}, \bar{r}) \mapsto \bar{u}_s(\hat{d}, \bar{r})$ are both continuous in some compact domain, then, since $V_N(\cdot)$ is then continuous in a compact domain $\mathcal{A}$, it follows from the properties of $V_N^0(\cdot)$ and Proposition 3.4 that there exists a $\mathcal{K}_\infty$ function $\alpha(\cdot)$ such that

$$
\begin{aligned}
V_N^0(\bar{x}, \hat{d}, \bar{r}) &\ge c_1 \left| \bar{x} - \bar{x}_s(\hat{d}, \bar{r}) \right|^2 \\
V_N^0(\bar{x}, \hat{d}, \bar{r}) &\le c_2 \left| \bar{x} - \bar{x}_s(\hat{d}, \bar{r}) \right|^2 \\
V_N^0(\bar{x}^+, \hat{d}^+, \bar{r}) &\le V_N^0(\bar{x}, \hat{d}, \bar{r}) - c_1 \left| \bar{x} - \bar{x}_s(\hat{d}, \bar{r}) \right|^2 + \alpha(\delta_d)
\end{aligned}
$$

for all $(\bar{x}, \hat{d}, \delta_d, \bar{r}) \in \mathcal{V}$; here $(\bar{x}, \hat{d})^+ := (\bar{x}^+, \hat{d}^+)$, $\bar{x}^+ = A\bar{x} + B_d(\hat{d} + \delta_d) + B\bar{\kappa}_N(\bar{x}, \hat{d}, \bar{r})$ and $\hat{d}^+ = \hat{d} + \delta_d$. A suitable choice for $\mathcal{A}$ is $\mathcal{V} \times \mathcal{D} \times \{\bar{r}\} \times \mathbb{U}^N$ with $\mathcal{V}$ the closure of $\text{lev}_a V_N^0(\cdot)$ for some $a > 0$, and $\mathcal{D}$ a compact set containing d and $\hat{d}$. It follows that there exists a $\gamma \in (0, 1)$ such that

$$
V_N^0((\bar{x}, \hat{d})^+, \bar{r}) \le \gamma V_N^0(\bar{x}, \hat{d}, \bar{r}) + \alpha(\delta_d)
$$

with $\gamma = 1 - c_1/c_2 \in (0, 1)$. Assuming that $\mathbb{P}_N(\bar{x}, \hat{d}, \bar{r})$ is recursively feasible

$$
V_N^0(\bar{x}(i), \hat{d}(i), \bar{r}) \le \gamma^i V_N^0(\bar{x}(0), \hat{d}(0), \bar{r}) + \alpha(\delta_d)(1 - \gamma^i)/(1 - \gamma)
$$

in which $\bar{x}(0) = x(0)$ and $\hat{d}(0) = d(0)$. It then follows from the last inequality and the bounds on $V_N^0(\cdot)$ that

$$\left|\bar{x}(i) - \bar{x}_s(\hat{d}(i), \bar{r})\right| \leq \gamma^{i/2}(c_2/c_1)^{1/2}\left|\bar{x}(0) - \bar{x}_s(\hat{d}(0), \bar{r})\right| + c(i)$$

with $c(i) := [\alpha(\delta_d)(1-\gamma^i)/(1-\gamma)]^{1/2}$ so that $c(i) \to c := [\alpha(\delta_d)/(1-\gamma)]^{1/2}$ and $\left|\bar{x}(i) - \bar{x}_s(\hat{d}(i), \bar{r})\right| \to c$ as $i \to \infty$. Here we have made use of the fact that $(a+b)^{1/2} \leq a^{1/2} + b^{1/2}$.

Let $C \subset \mathbb{R}^n$ denote the set $\{x \mid |x| \leq c\}$. Then $\bar{x}(i) \to \{\bar{x}_s(\hat{d}(i), \bar{r})\} \oplus C$, $\hat{x}(i) \to \{\bar{x}_s(\hat{d}(i), \bar{r})\} \oplus C \oplus S$ and $x(i) \to \{\bar{x}_s(\hat{d}(i), \bar{r})\} \oplus C \oplus S \oplus \Sigma$ as $i \to \infty$. Since $c(i) = [\alpha(\delta_d)(1-\gamma^i)/(1-\gamma)]^{1/2} \to 0$ as $\delta_d \to 0$, it follows that $\bar{x}(i) \to \bar{x}_s(\hat{d}(i), \bar{r})$ as $i \to \infty$. The sizes of S and Σ are dictated by the process and measurement disturbances, w and v respectively.

Recursive feasibility. The result that $x(i) \to \{\bar{x}_s(\hat{d}(i), \bar{r})\} \oplus C \oplus \Gamma$, $\Gamma := S \oplus \Sigma$, is useful because it gives an asymptotic bound on the tracking error. But it does depend on the recursive feasibility of the optimal control problem $\mathbb{P}_N(\cdot)$, which does not necessarily hold because of the variation of $\hat{d}$ with time. Tracking of a random reference signal has been considered in the literature, but not in the context of output MPC. We show next that $\mathbb{P}_N(\cdot)$ *is* recursively feasible and that the tracking error remains bounded if the estimate $\hat{d}$ of the disturbance d varies sufficiently slowly—that is if δ_d in the difference equation $\hat{d}^+ = \hat{d} + \delta_d$ is sufficiently small. This can be ensured by design of the state estimator.

To establish recursive feasibility, assume that the current "state" is $(\bar{x}, \hat{d}, \bar{r})$ and $\bar{x} \in \bar{\mathcal{X}}(\hat{d}, \bar{r})$. In other words, we assume $\bar{\mathbb{P}}_N(\bar{x}, \hat{d}, \bar{r})$ is feasible and $\bar{x}_N := \bar{\phi}(N; \bar{x}, \bar{\kappa}_N(\bar{x}, \hat{d}, \bar{r})) \in \mathbb{X}_f(\bar{x}_s(\hat{d}, \bar{r}))$. If the usual stability conditions are satisfied, problem $\mathbb{P}_N(\bar{x}^+, \hat{d}, \bar{r})$ is also feasible so that $\bar{x}^+ = A\bar{x} + B_d\hat{d} + B\bar{\kappa}_N(\bar{x}, \hat{d}, \bar{r}) \in \bar{\mathcal{X}}_N(\hat{d}, \bar{r})$. But $\hat{d}^+ = \hat{d} + \delta_d$ so that $\mathbb{P}_N(\bar{x}^+, \hat{d}^+, \bar{r})$ is *not* necessarily feasible since $\bar{x}_N$, which lies in $\mathbb{X}_f(\bar{x}_s(\hat{d}, \bar{r}))$, does not necessarily lie in $\mathbb{X}_f(\bar{x}_s(\hat{d}^+, \bar{r}))$. Let the terminal set $\mathbb{X}_f(\bar{x}_s(\hat{d}, \bar{r})) := \{x \mid V_f(x - \bar{x}_s(\hat{d}, \bar{r})) \leq c\}$. If the usual stability conditions are satisfied, for each $\bar{x}_N \in \mathbb{X}_f(\bar{x}_s(\hat{d}, \bar{r}))$, there exists a $u = \kappa_f(\bar{x}_N)$ that steers $\bar{x}_N$ to a state $\bar{x}_N^+$ in $\{x \mid V_f(x - \bar{x}_s(\hat{d}, \bar{r})) \leq e\}$, $e < c$. Consequently, there exists a feasible control sequence $\tilde{\mathbf{u}}(\bar{x}) \in \bar{\mathcal{U}}_N(\bar{x}, \hat{d}, \bar{r})$ that steers $\bar{x}^+$ to a state $\bar{x}_N^+ \in \{x \mid V_f(x - \bar{x}_s(\hat{d}, \bar{r})) \leq e\}$. If the map $\hat{d} \mapsto \bar{x}_s(\hat{d}, \bar{r})$ is uniformly continuous, there exists a constant $a > 0$ such that $|\delta_d| \leq a$ implies that $\bar{x}_N^+$ lies also in $\mathbb{X}_f(\bar{x}_s(\hat{d}^+, \bar{r})) = \{x \mid V_f(x - \bar{x}_s(\hat{d}^+, \bar{r})) \leq c\}$. Thus the control sequence $\tilde{\mathbf{u}}(\bar{x})$ also steers $\bar{x}^+$ to the set $\mathbb{X}_f(\bar{x}_s(\hat{d}^+, \bar{r}))$ and hence lies in $\bar{\mathcal{U}}_N(\bar{x}, \hat{d}^+, \bar{r})$. Hence

problem $\bar{\mathbb{P}}_N(\bar{x}^+, \hat{d}^+, \bar{r})$ is feasible so that $\bar{\mathbb{P}}_N$ is recursively feasible if $\sup_{i \in \mathbb{I}_{0:\infty}} |\delta_d(i)| \le e$.

Computing the tightened constraints. The first step in the control algorithm requires solution of the problem $\mathbb{P}_N(\bar{x}, \hat{d}, \bar{r})$, in which the state and control constraints are, respectively, $\bar{x} \in \bar{\mathbb{X}}$ and $\bar{u} \in \bar{\mathbb{U}}$. Since the sets $\bar{\mathbb{X}}$ and $\bar{\mathbb{U}}$ are difficult to compute, we replace them by tightened versions of the original constraints as described in Section 5.3.5.

> Summarizing, if the usual stability assumptions are satisfied, if $\hat{d}(i)$ remains in a compact set $X_{\hat{d}}$ for all i, if the map $\hat{d} \mapsto \bar{x}_s(\hat{d}, \bar{r})$ is continuous in $X_{\hat{d}}$, if $\ell(\cdot)$ and $V_f(\cdot)$ are quadratic and positive definite, and $|\delta_d(i)| \le a$ for all i, then the asymptotic error $x(i) - \bar{x}_s(\hat{d}(i), \bar{r})$ lies in the compact set $C \oplus \Gamma$ ($\Gamma = S + \Sigma$) that converges to the set $\{0\}$ as the sets $\mathbb{W}$ and $\mathbb{N}$ that bound the disturbances converge to the zero set $\{0\}$. Similarly, the tracking error $r - \bar{r}$ is also bounded and converges to 0 as $\mathbb{W}$ and $\mathbb{N}$ converge to the zero set $\{0\}$.

5.6 Nonlinear Constrained Systems

When the system being controlled is nonlinear, the state can be estimated using moving horizon estimation (MHE), as described in Chapter 4. But establishing stability of nonlinear output MPC that employs MHE does not appear to have received much attention, with one important exception mentioned in Section 5.7.

5.7 Notes

The problem of output feedback control has been extensively discussed in the general control literature. For linear systems, it is well known that a stabilizing state feedback controller and an observer may be separately designed and combined to give a stabilizing output feedback controller (the separation principle). For nonlinear systems, Teel and Praly (1994) show that global stabilizability and complete uniform observability are sufficient to guarantee semiglobal stabilizability when a dynamic observer is used, and provide useful references to related work on this topic.

Although output MPC, in which nominal MPC is combined with a separately designed observer, is widely used in industry since the state

is seldom available, it has received relatively little attention in the literature because of the inherent difficulty in establishing asymptotic stability. An extra complexity in MPC is the presence of hard constraints. A useful survey, more comprehensive than these notes, is provided in Findeisen, Imsland, Allgöwer, and Foss (2003). Earlier Michalska and Mayne (1995) show for deterministic systems that for any subset of the region of attraction of the full state feedback system, there exists a sampling time and convergence rate for the observer such that the subset also lies in the region of attraction of the output feedback system. A more sophisticated analysis in Imsland, Findeisen, Allgöwer, and Foss (2003) using continuous time MPC shows that the region of attraction and rate of convergence of the output feedback system can approach that of the state feedback system as observer gain increases.

We consider systems with input disturbances and noisy state measurement, and employ the "tube" methodology that has its roots in the work of Bertsekas and Rhodes (1971), and Glover and Schweppe (1971) on constrained discrete time systems subject to bounded disturbances. Reachability of a "target set" and a "target tube" are considered in these papers. These concepts were substantially developed in the context of continuous time systems in Khurzhanski and Valyi (1997); Aubin (1991); Kurzhanski and Filippova (1993).

The theory for discrete time systems is considerably simpler; a modern tube-based theory for optimal control of discrete time uncertain systems with imperfect state measurement appears in Moitié, Quincampoix, and Veliov (2002). As in this chapter, they regard a set X of states x that are consistent with past measurements as the "state" of the optimal control problem. The set X satisfies an uncertain "full information" difference equation of the form $X^+ = f^*(X, u, \mathbb{W}, v)$ so the output feedback optimal control problem reduces to robust control of an uncertain system with known state X.

The optimal control problem remains difficult because the state X, a subset of $\mathbb{R}^n$, is difficult to obtain numerically and determination of a control law as a function of (X, t) prohibitive. In Mayne, Raković, Findeisen, and Allgöwer (2006); Mayne et al. (2009) the output feedback problem is simplified considerably by replacing $X(t)$ by a simple outer approximation $\{\hat{x}(t)\} \oplus \mathbb{Z}_x$ in the time-invariant case, and by $\{\hat{x}(t)\} \oplus \mathbb{Z}_x(t)$ in the time-varying case. The set $\mathbb{Z}_x$, or the sequence $(\mathbb{Z}_x(t))$, may be precomputed so the difficult evolution equation for X is replaced by a simple evolution equation for $\hat{x}$; in the linear case, the Luenberger observer or Kalman filter describes the evolution of $\hat{x}$. The output

feedback control problem reduces to control of an uncertain system with known state $\hat{x}$.

Artstein and Raković (2008) provide an interesting extension of the invariant sets given in (5.11) to the nonlinear case $x^+ \in F(x) + V$ when $F(\cdot)$ is a contraction mapping and V is compact.

While the tube approach may be successfully employed for output MPC when the system being controlled is linear, there seems to be no literature on combining moving horizon estimation (MHE) with MPC when the system being controlled is nonlinear, except for the paper Copp and Hespanha (2014). The novel proposal in this paper is to replace separate solutions of the control and estimation problems by a single min-max problem in which the cost is, unusually, over the interval $(-\infty, \infty)$ or $[-T, T]$, and combines the cost of both estimation and control. The authors also propose an efficient interior point algorithm for solving the complex min-max problem.

The output MPC problem involves tracking of a possibly random reference, a problem that has extra difficulty when zero offset is required. There is a growing literature dealing with tracking random references not necessarily in the context of output MPC. Examples of papers dealing with this topic are Limon, Alvarado, Alamo, and Camacho (2008); Ferramosca, Limon, Alvarado, Alamo, and Camacho (2009); Falugi and Mayne (2013).

5.8 Exercises

Exercise 5.1: Hausdorff distance between a set and a subset

Show that $d_H(\mathbb{A}, \mathbb{B}) = \max_{a \in \mathbb{A}} d(a, \mathbb{B})$ if $\mathbb{A}$ and $\mathbb{B}$ are two compact subsets of $\mathbb{R}^n$ satisfying $\mathbb{B} \subseteq \mathbb{A}$.

Exercise 5.2: Hausdorff distance between sets $\mathbb{A} \oplus \mathbb{B}$ and $\mathbb{B}$

Show that $d_H(\mathbb{A} \oplus \mathbb{B}, \mathbb{A}) \leq |\mathbb{B}|$ if $\mathbb{A}$ and $\mathbb{B}$ are two compact subsets of $\mathbb{R}^n$ satisfying $0 \in \mathbb{B}$ in which $|\mathbb{B}| := \max_b \{|b| \mid b \in \mathbb{B}\}$.

Exercise 5.3: Hausdorff distance between sets $\{z\} \oplus \mathbb{B}$ and $\mathbb{A}$

Show that $d_H(\{z\} \oplus \mathbb{B}, \mathbb{A}) \leq |z| + d_H(\mathbb{A}, \mathbb{B})$ if $\mathbb{A}$ and $\mathbb{B}$ are two compact sets in $\mathbb{R}^n$.

Exercise 5.4: Hausdorff distance between sets $\{z\} \oplus \mathbb{A}$ and $\mathbb{A}$

Show that $d_H(\{z\} \oplus \mathbb{A}, \mathbb{A}) = |z|$ if z is a point and $\mathbb{A}$ is a compact set in $\mathbb{R}^n$.

Exercise 5.5: Hausdorff distance between sets $\mathbb{A} \oplus \mathbb{C}$ and $\mathbb{B} \oplus \mathbb{C}$

Show that $d_H(\mathbb{A} \oplus \mathbb{C}, \mathbb{B} \oplus \mathbb{C}) \leq d_H(\mathbb{A}, \mathbb{B})$ if $\mathbb{A}$, $\mathbb{B}$, and $\mathbb{C}$ are compact subsets of $\mathbb{R}^n$.

Exercise 5.6: Hausdorff distance between sets $F\mathbb{A}$ and $F\mathbb{B}$

Let $\mathbb{A}$ and $\mathbb{B}$ be two compact sets in $\mathbb{R}^n$, and let $F \in \mathbb{R}^{n \times n}$. Show that

$$d_H(F\mathbb{A}, F\mathbb{B}) \leq |F| \, d_H(\mathbb{A}, \mathbb{B})$$

in which $|F|$ is the induced norm of F satisfying $|Fx| \leq |F| \, |x|$ and $|x| := d(x, 0)$.

Exercise 5.7: Linear combination of sets; $\lambda_1 \mathbb{W} \oplus \lambda_2 \mathbb{W} = (\lambda_1 + \lambda_2)\mathbb{W}$

If $\mathbb{W}$ is a convex set, show that $\lambda_1 \mathbb{W} \oplus \lambda_2 \mathbb{W} = (\lambda_1 + \lambda_2)\mathbb{W}$ for any $\lambda_1, \ \lambda_2 \in \mathbb{R}_{\geq 0}$. Hence show $\mathbb{W} \oplus \lambda \mathbb{W} \oplus \lambda^2 \mathbb{W} \oplus \ \cdots = (1 - \lambda)^{-1} \mathbb{W}$ if $\lambda \in [0, 1)$.

Exercise 5.8: Hausdorff distance between the sets $\Phi(i)$ and Φ

Show that there exist $c > 0$ and $\gamma \in (0, 1)$ such that

$$d_H(\Phi(i), \Phi) \leq c d_H(\Phi(0), \Phi) \gamma^i$$

in which

$$\Phi(i) = \widetilde{A}\Phi(i-1) \oplus \widetilde{B}\Psi$$
$$\Phi = \widetilde{A}\Phi \oplus \widetilde{B}\Psi$$

and $\widetilde{A}$ is a stable matrix ($\rho(\widetilde{A}) < 1$).

Bibliography

Z. Artstein and S. V. Raković. Feedback and invariance under uncertainty via set-iterates. *Automatica*, 44(2):520–525, February 2008.

J. P. Aubin. *Viability Theory*. Systems & Control: Foundations & Applications. Birkhauser, Boston, Basel, Berlin, 1991.

D. P. Bertsekas and I. B. Rhodes. Recursive state estimation for a set-membership description of uncertainty. *IEEE Trans. Auto. Cont.*, 16:117–128, 1971.

F. Blanchini. Set invariance in control. *Automatica*, 35:1747–1767, 1999.

D. A. Copp and J. P. Hespanha. Nonlinear output-feedback model predictive control with moving horizon estimation. In *Proceedings of the 53rd Conference on Decision and Control*, December 2014.

P. Falugi and D. Q. Mayne. Model predictive control for tracking random references. In *Proceedings of the 2013 European Control Conference*, pages 518–523, 2013.

A. Ferramosca, D. Limon, I. Alvarado, T. Alamo, and E. F. Camacho. MPC for tracking of constrained nonlinear systems. In *Proceedings of the 48th Conference on Decision and Control, and the 28th Chinese Control Conference*, pages 7978–7983, 2009.

R. Findeisen, L. Imsland, F. Allgöwer, and B. A. Foss. State and output feedback nonlinear model predictive control: An overview. *Eur. J. Control*, 9(2–3):190–206, 2003.

J. D. Glover and F. C. Schweppe. Control of linear dynamic systems with set constrained disturbances. *IEEE Trans. Auto. Cont.*, 16:411–423, 1971.

L. Imsland, R. Findeisen, F. Allgöwer, and B. A. Foss. A note on stability, robustness and performance of output feedback nonlinear model predictive control. *J. Proc. Cont.*, 13:633–644, 2003.

A. B. Khurzhanski and I. Valyi. *Ellipsoidal-valued dynamics for estimation and control*. Systems & Control: Foundations & Applications. Birkhauser, Boston, Basel, Berlin, 1997.

I. Kolmanovsky and E. G. Gilbert. Theory and computation of disturbance invariant sets for discrete-time linear systems. *Math. Probl. Eng.*, 4(4):317–367, 1998.

A. B. Kurzhanski and T. F. Filippova. On the theory of trajectory tubes: A mathematical formalism for uncertain dynamics, viability and control. In A. B. Kurzhanski, editor, *Advances in Nonlinear Dynamics and Control: A Report from Russia*, volume 17 of *PSCT*, pages 122–188. Birkhauser, Boston, Basel, Berlin, 1993.

D. Limon, I. Alvarado, T. Alamo, and E. F. Camacho. MPC for tracking piecewise constant references for constrained linear systems. *Automatica*, pages 2382–2387, 2008.

D. Q. Mayne, S. V. Raković, R. Findeisen, and F. Allgöwer. Robust output feedback model predictive control of constrained linear systems. *Automatica*, 42(7):1217–1222, July 2006.

D. Q. Mayne, S. V. Raković, R. Findeisen, and F. Allgöwer. Robust output feedback model predictive control of constrained linear systems: Time varying case. *Automatica*, 45(9):2082–2087, September 2009.

H. Michalska and D. Q. Mayne. Moving horizon observers and observer-based control. *IEEE Trans. Auto. Cont.*, 40(6):995–1006, 1995.

R. Moitié, M. Quincampoix, and V. M. Veliov. Optimal control of discrete-time uncertain systems with imperfect measurement. *IEEE Trans. Auto. Cont.*, 47 (11):1909–1914, November 2002.

G. Pannocchia and J. B. Rawlings. Disturbance models for offset-free MPC control. *AIChE J.*, 49(2):426–437, 2003.

S. V. Raković, E. C. Kerrigan, K. I. Kouramas, and D. Q. Mayne. Invariant approximations of the minimal robustly positively invariant sets. *IEEE Trans. Auto. Cont.*, 50(3):406–410, 2005.

A. R. Teel and L. Praly. Global stabilizability and observability implies semiglobal stabilizability by output feedback. *Sys. Cont. Let.*, 22:313–325, 1994.

6
Distributed Model Predictive Control

6.1 Introduction and Preliminary Results

In many large-scale control applications, it becomes convenient to break the large plantwide problem into a set of smaller and simpler subproblems in which the local inputs are used to regulate the local outputs. The overall plantwide control is then accomplished by the composite behavior of the interacting, local controllers. There are many ways to design the local controllers, some of which produce guaranteed properties of the overall plantwide system. We consider four control approaches in this chapter: decentralized, noncooperative, cooperative, and centralized control. The first three methods require the local controllers to optimize over only their local inputs. Their computational requirements are identical. The communication overhead is different, however. Decentralized control requires no communication between subsystems. Noncooperative and cooperative control require the input sequences and the current states or state estimates for all the other local subsystems. Centralized control solves the large, complex plantwide optimization over all the inputs. Communication is not a relevant property for centralized control because all information is available in the single plantwide controller. We use centralized control in this chapter to provide a benchmark of comparison for the distributed controllers.

We have established the basic properties of centralized MPC, both with and without state estimation, in Chapters 2, 3, and 5. In this chapter, we analyze some basic properties of the three distributed approaches: decentralized, noncooperative, and cooperative MPC. We show that the conditions required for closed-loop stability of decentralized control and noncooperative control are often violated for coupled multivariable systems under reasonable decompositions into subsystems. For ensuring closed-loop stability of a wide class of plantwide

models and decomposition choices, cooperative control emerges as the most attractive option for distributed MPC. We then establish the closed-loop properties of cooperative MPC for unconstrained and constrained linear systems with and without state estimation. We also discuss current challenges facing this method, such as input constraints that are coupled between subsystems.

In our development of distributed MPC, we require some basic results on two topics: how to organize and solve the linear algebra of linear MPC, and how to ensure stability when using suboptimal MPC. We cover these two topics in the next sections, and then turn to the distributed MPC approaches.

6.1.1 Least Squares Solution

In comparing various forms of linear distributed MPC it proves convenient to see the MPC quadratic program for the sequence of states and inputs as a single large linear algebra problem. To develop this linear algebra problem, we consider first the *unconstrained* linear quadratic (LQ) problem of Chapter 1, which we solved efficiently with dynamic programming (DP) in Section 1.3.3

$$V(x(0),\mathbf{u}) = \frac{1}{2}\sum_{k=0}^{N-1}\left(x(k)'Qx(k)+u(k)'Ru(k)\right)+(1/2)x(N)'P_f x(N)$$

subject to

$$x^+ = Ax+Bu$$

In this section, we first take the direct but brute-force approach to finding the optimal control law. We write the model solution as

$$\begin{bmatrix} x(1)\\ x(2)\\ \vdots\\ x(N)\end{bmatrix} = \underbrace{\begin{bmatrix} A\\ A^2\\ \vdots\\ A^N\end{bmatrix}}_{\mathcal{A}} x(0) + \underbrace{\begin{bmatrix} B & 0 & \cdots & 0\\ AB & B & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ A^{N-1}B & A^{N-2}B & \cdots & B\end{bmatrix}}_{\mathcal{B}} \begin{bmatrix} u(0)\\ u(1)\\ \vdots\\ u(N-1)\end{bmatrix} \tag{6.1}$$

or using the input and state sequences

$$\mathbf{x} = \mathcal{A}x(0)+\mathcal{B}\mathbf{u}$$

The objective function can be expressed as

$$V(x(0),\mathbf{u}) = (1/2)\left(x'(0)Qx(0)+\mathbf{x}'\mathcal{Q}\mathbf{x}+\mathbf{u}'\mathcal{R}\mathbf{u}\right)$$

in which

$$\begin{aligned}\mathcal{Q} &= \text{diag}\left(\begin{bmatrix} Q & Q & \dots & P_f \end{bmatrix}\right) \in \mathbb{R}^{Nn\times Nn} \\ \mathcal{R} &= \text{diag}\left(\begin{bmatrix} R & R & \dots & R \end{bmatrix}\right) \in \mathbb{R}^{Nm\times Nm}\end{aligned} \tag{6.2}$$

Eliminating the state sequence. Substituting the model into the objective function and *eliminating* the state sequence gives a quadratic function of $\mathbf{u}$

$$V(x(0),\mathbf{u}) = (1/2)x'(0)(Q + \mathcal{A}'\mathcal{Q}\mathcal{A})x(0) + \mathbf{u}'(\mathcal{B}'\mathcal{Q}\mathcal{A})x(0) + (1/2)\mathbf{u}'(\mathcal{B}'\mathcal{Q}\mathcal{B} + \mathcal{R})\mathbf{u} \tag{6.3}$$

and the optimal solution for the entire set of inputs is obtained in one shot

$$\mathbf{u}^0(x(0)) = -(\mathcal{B}'\mathcal{Q}\mathcal{B} + \mathcal{R})^{-1}\mathcal{B}'\mathcal{Q}\mathcal{A}\, x(0)$$

and the optimal cost is

$$V^0(x(0)) = \left(\frac{1}{2}\right)x'(0)\left(Q + \mathcal{A}'\mathcal{Q}\mathcal{A} - \mathcal{A}'\mathcal{Q}\mathcal{B}(\mathcal{B}'\mathcal{Q}\mathcal{B} + \mathcal{R})^{-1}\mathcal{B}'\mathcal{Q}\mathcal{A}\right)x(0)$$

If used explicitly, this procedure for computing $\mathbf{u}^0$ would be inefficient because $\mathcal{B}'\mathcal{Q}\mathcal{B} + \mathcal{R}$ is an $(mN \times mN)$ matrix. Notice that in the DP formulation one has to invert instead an $(m\times m)$ matrix N times, which is computationally less expensive.[1] Notice also that unlike DP, the least squares approach provides *all* input moves as a function of the *initial* state, $x(0)$. The gain for the control law comes from the first input move in the sequence

$$K(0) = -\begin{bmatrix} I_m & 0 & \cdots & 0 \end{bmatrix}(\mathcal{B}'\mathcal{Q}\mathcal{B} + \mathcal{R})^{-1}\mathcal{B}'\mathcal{Q}\mathcal{A}$$

It is not immediately clear that the $K(0)$ and V^0 given above from the least squares approach are equivalent to the result from the Riccati iteration, (1.10)–(1.14) of Chapter 1, but since we have solved the same optimization problem, the two results are the same.[2]

Retaining the state sequence. In this section we set up the least squares problem again, but with an eye toward improving its efficiency. Retaining the state sequence and adjoining the model equations as

[1] Would you prefer to invert by hand 100 (1×1) matrices or a single (100×100) dense matrix?

[2] Establishing this result directly is an exercise in using the partitioned matrix inversion formula. The next section provides another way to show they are equivalent.

equality constraints is a central idea in optimal control and is described in standard texts (Bryson and Ho, 1975, p. 44). We apply this standard approach here. Wright (1997) provides a discussion of this problem in the linear model MPC context and the extensions required for the quadratic programming problem when there are inequality constraints on the states and inputs. Including the state with the input in the sequence of unknowns, we define the enlarged vector $\mathbf{z}$ to be

$$\mathbf{z} = \begin{bmatrix} u(0) \\ x(1) \\ u(1) \\ x(2) \\ \vdots \\ u(N-1) \\ x(N) \end{bmatrix}$$

The objective function is

$$\min_{\mathbf{u}} (1/2)(x'(0)Qx(0) + \mathbf{z}'H\mathbf{z})$$

in which

$$H = \operatorname{diag}\left(\begin{bmatrix} R & Q & R & Q & \cdots & R & P_f \end{bmatrix}\right)$$

The constraints are

$$D\mathbf{z} = d$$

in which

$$D = -\begin{bmatrix} B & -I & & & & & \\ & A & B & -I & & & \\ & & & \ddots & & & \\ & & & & A & B & -I \end{bmatrix} \qquad d = \begin{bmatrix} A \\ 0 \\ \vdots \\ 0 \end{bmatrix} x(0)$$

We now substitute these results into (1.57) and obtain the linear algebra problem

$$\begin{bmatrix} R & & & & & & & B' & & & \\ & Q & & & & & & -I & A' & & \\ & & R & & & & & & B' & & \\ & & & Q & & & & & -I & & \\ & & & & \ddots & & & & & \ddots & \\ & & & & & R & & & & & B' \\ & & & & & & P_f & & & & -I \\ B & -I & & & & & & & & & \\ & A & B & -I & & & & & & & \\ & & & & \ddots & & & & & & \\ & & & & & B & -I & & & & \end{bmatrix} \begin{bmatrix} u(0) \\ x(1) \\ u(1) \\ x(2) \\ \vdots \\ u(N-1) \\ x(N) \\ \lambda(1) \\ \lambda(2) \\ \vdots \\ \lambda(N) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ -A \\ 0 \\ \vdots \\ 0 \end{bmatrix} x(0)$$

Method	FLOPs
dynamic programming (DP)	Nm^3
dense least squares	N^3m^3
banded least squares	$N(2n+m)(3n+m)^2$

Table 6.1: Computational cost of solving finite horizon LQR problem.

This equation is rather cumbersome, but if we reorder the unknown vector to put the Lagrange multiplier together with the state and input from the same time index, and reorder the equations, we obtain the following banded matrix problem

$$\begin{bmatrix} R & B' & & & & & & & & & & \\ B & & -I & & & & & & & & & \\ & -I & Q & & & & & & & & & \\ & & & R & & & & & & & & \\ & & & \ddots & \ddots & & & & & & & \\ & & & & & R & B' & & & & & \\ & & & & A & B & & -I & & & & \\ & & & & & & -I & Q & & A' & & \\ & & & & & & & & R & B' & & \\ & & & & & & & A & B & & -I & \\ & & & & & & & & & -I & P_f \end{bmatrix} \begin{bmatrix} u(0) \\ \lambda(1) \\ x(1) \\ u(1) \\ \vdots \\ u(N-2) \\ \lambda(N-1) \\ x(N-1) \\ u(N-1) \\ \lambda(N) \\ x(N) \end{bmatrix} = \begin{bmatrix} 0 \\ -A \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} x(0) \quad (6.4)$$

The banded structure allows a more efficient solution procedure. The floating operation (FLOP) count for the factorization of a banded matrix is $O(LM^2)$ in which L is the dimension of the matrix and M is the bandwidth. This compares to the regular FLOP count of $O(L^3)$ for the factorization of a regular dense matrix. The bandwidth of the matrix in (6.4) is $3n+m$ and the dimension of the matrix is $N(2n+m)$. Therefore the FLOP count for solving this equation is $O(N(2n+m)(3n+m)^2)$. Notice that this approach reduces the N^3 dependence of the previous MPC solution method. That is the computational advantage provided by these adjoint methods for treating the model constraints. Table 6.1 summarizes the computational cost of the three approaches for the linear quadratic regulator (LQR) problem. As shown in the table, DP is highly efficient. When we add input and state inequality constraints to the control problem and the state dimension is large, however, we cannot conveniently apply DP. The dense least squares computational cost is high if we wish to compute a large number of moves in the horizon. Note the cost of dense least squares scales with the third

power of horizon length N. As we have discussed in Chapter 2, considerations of control theory favor large N. Another factor increasing the computational cost is the trend in industrial MPC implementations to larger multivariable control problems with more states and inputs, i.e., larger m and n. Therefore, the adjoint approach using banded least squares method becomes important for industrial applications in which the problems are large and a solid theoretical foundation for the control method is desirable.

We might obtain more efficiency than the banded structure if we view (6.4) as a block tridiagonal matrix and use the method provided by Golub and Van Loan (1996, p. 174). The final fine tuning of the solution method for this class of problems is a topic of current research, but the important point is that, whatever final procedure is selected, the computational cost will be linear in N as in DP instead of cubic in N as in dense least squares.

Furthermore, if we wish to see the connection to the DP solution, we can proceed as follows. Substitute $\Pi(N) = P_f$ as in (1.11) of Chapter 1 and consider the last three-equation block of the matrix appearing in (6.4)

$$\begin{bmatrix} & R & B' & \\ A & B & & -I \\ & & -I & \Pi(N) \end{bmatrix} \begin{bmatrix} x(N-1) \\ u(N-1) \\ \lambda(N) \\ x(N) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

We can eliminate this small set of equations and solve for $u(N-1)$, $\lambda(N)$, $x(N)$ in terms of $x(N-1)$, resulting in

$$\begin{bmatrix} u(N-1) \\ \lambda(N) \\ x(N) \end{bmatrix} = \begin{bmatrix} -(B'\Pi(N)B+R)^{-1}B'\Pi(N)A \\ \Pi(N)(I-B(B'\Pi(N)B+R)^{-1}B'\Pi(N))A \\ (I-B(B'\Pi(N)B+R)^{-1}B'\Pi(N))A \end{bmatrix} x(N-1)$$

Notice that in terms of the Riccati matrix, we also have the relationship

$$A'\lambda(N) = \Pi(N-1)x(N-1) - Qx(N-1)$$

We then proceed to the next to last block of three equations

$$\begin{bmatrix} & R & B' & & \\ A & B & & -I & \\ & & -I & Q & A' \end{bmatrix} \begin{bmatrix} x(N-2) \\ u(N-2) \\ \lambda(N-1) \\ x(N-1) \\ u(N-1) \\ \lambda(N) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Note that the last equation gives

$$\lambda(N-1) = Qx(N-1) + A'\lambda(N) = \Pi(N-1)x(N-1)$$

Using this relationship and continuing on to solve for $x(N-1)$, $\lambda(N-1)$, $u(N-2)$ in terms of $x(N-2)$ gives

$$\begin{bmatrix} u(N-2) \\ \lambda(N-1) \\ x(N-1) \end{bmatrix} = \begin{bmatrix} -(B'\Pi(N-1)B+R)^{-1}B'\Pi(N-1)A \\ \Pi(N-1)(I-B(B'\Pi(N-1)B+R)^{-1}B'\Pi(N-1))A \\ (I-B(B'\Pi(N-1)B+R)^{-1}B'\Pi(N-1))A \end{bmatrix} x(N-2)$$

Continuing on through each previous block of three equations produces the Riccati iteration and feedback gains of (1.10)–(1.13). The other unknowns, the multipliers, are simply

$$\lambda(k) = \Pi(k)x(k) \qquad k = 1, 2, \ldots, N$$

so the cost to go at each stage is simply $x(k)'\lambda(k)$, and we see the nice connection between the Lagrange multipliers and the cost of the LQR control problem.

6.1.2 Stability of Suboptimal MPC

When using distributed MPC, it may be necessary or convenient to implement the control without solving the complete optimization. We then have a form of suboptimal MPC, which was first considered in Chapter 2, Section 2.7. Before adding the complexity of the distributed version, we wish to further develop a few features of suboptimal MPC in the centralized, single-player setting. These same features arise in the distributed, many-player setting as we discuss subsequently.

We consider a specific variation of suboptimal MPC in which a starting guess is available from the control trajectory at the previous time and we take a fixed number of steps of an optimization algorithm. The exact nature of the optimization method is not essential, but we do restrict the method so that each iteration is feasible and decreases the value of the cost function. To initialize the suboptimal controller, we are given an initial state $x(0) = x_0$, and we generate an initial control sequence $\mathbf{u}(0) = \mathbf{h}(x_0)$. We consider input constraints $u(i) \in \mathbb{U} \subseteq \mathbb{R}^m$, $i \in \mathbb{I}_{0:N-1}$, which we also write as $\mathbf{u} \in \mathbb{U}^N \subseteq \mathbb{R}^N$. As in Chapter 2 we denote the set of feasible states as $\mathcal{X}_N$. These are the states for which the initial control sequence $\mathbf{h}(x_0)$ is well defined. The suboptimal MPC algorithm is as follows.

Algorithm 6.1 (Suboptimal MPC (simplified))**.** Set current state $x = x_0$, current control sequence, $\mathbf{u} = \mathbf{h}(x_0)$, current warm start $\widetilde{\mathbf{u}} = \mathbf{u}$. Then repeat

1. Obtain current measurement of state x.

2. The controller performs some number of iterations of a feasible path optimization algorithm to obtain an improved control sequence $\mathbf{u}$ such that $V_N(x, \mathbf{u}(0)) \leq V_N(x, \tilde{\mathbf{u}}(0))$.

3. Inject the first element of the input sequence $\mathbf{u}$.

4. Compute the next warm start.

$$\tilde{\mathbf{u}}^+ = (u(1), u(2), \ldots, u(N-1), 0)$$

This warm start is a simplified version of the one considered in Chapter 2, in which the final control move in the warm start was determined by the control law $\kappa_f(x)$. In distributed MPC it is simpler to use zero for the final control move in the warm start. We establish later in the chapter that the system cost function $V(x, \mathbf{u})$ satisfies the following properties for the form of suboptimal MPC generated by distributed MPC. There exist constants $a, b, c > 0$ such that

$$a\,|(x, \mathbf{u})|^2 \leq V(x, \mathbf{u}) \leq b\,|(x, \mathbf{u})|^2$$
$$V(x^+, \mathbf{u}^+) - V(x, \mathbf{u}) \leq -c\,|(x, u(0))|^2$$

These properties are similar to those required for a valid Lyapunov function. The difference is that the cost decrease here does not depend on the size of $\mathbf{u}$, but only x and the first element of $\mathbf{u}$, $u(0)$. This cost decrease is sufficient to establish that $x(k)$ and $u(k)$ converge to zero, but allows the possibility that $\mathbf{u}(k)$ is large even though $x(k)$ is small. That fact prevents us from establishing the solution $x(k) = 0$ for all k is Lyapunov stable. We can establish that the solution $x(k) = 0$ for all k is Lyapunov stable at $k = 0$ only. We cannot establish uniform Lyapunov stability nor Lyapunov stability for any $k > 0$. The problem is not that our proof technique is deficient. There is no reason to *expect* that the solution $x(k) = 0$ for all k is Lyapunov stable for suboptimal MPC. The lack of Lyapunov stability of $x(k) = 0$ for all k is a subtle issue and warrants some discussion. To make these matters more precise, consider the following standard definitions of Lyapunov stability at time k and uniform Lyapunov stability (Vidyasagar, 1993, p. 136).

Definition 6.2 (Lyapunov stability)**.** The zero solution $x(k) = 0$ for all k is stable (in the sense of Lyapunov) at $k = k_0$ if for any $\varepsilon > 0$ there exists a $\delta(k_0, \varepsilon) > 0$ such that

$$|x(k_0)| < \delta \implies |x(k)| < \varepsilon \quad \forall k \geq k_0 \tag{6.5}$$

Lyapunov stability is defined at a time k_0. Uniform stability is the concept that guarantees that the zero solution is not losing stability with time. For a uniformly stable zero solution, δ in Definition 6.2 is *not* a function of k_0, so that (6.5) holds for all k_0.

Definition 6.3 (Uniform Lyapunov stability)**.** The zero solution $x(k) = 0$ for all k is uniformly stable (in the sense of Lyapunov) if for any $\varepsilon > 0$ there exists a $\delta(\varepsilon) > 0$ such that

$$|x(k_0)| < \delta \implies |x(k)| < \varepsilon \quad \forall k \geq k_0 \quad \forall k_0$$

Exercise 6.6 gives an example of a linear system for which $x(k)$ converges exponentially to zero with increasing k for all $x(0)$, but the zero solution $x(k) = 0$ for all k is Lyapunov stable only at $k = 0$. It is not uniformly Lyapunov stable nor Lyapunov stable for any $k > 0$. Without further restrictions, suboptimal MPC admits this same type of behavior.

To ensure uniform Lyapunov stability, we add requirements to suboptimal MPC beyond obtaining only a cost decrease. Here we impose the constraint

$$|\mathbf{u}| \leq d\,|x| \qquad x \in r\mathcal{B}$$

in which $d, r > 0$. This type of constraint is also included somewhat indirectly by the suboptimal control approach discussed in Section 2.7. In that arrangement, this constraint is implied by the first case in (2.29), which leads to Proposition 2.44. For simplicity, in this chapter we instead include the constraint explicitly in the distributed MPC optimization problem. Both approaches provide (uniform) Lyapunov stability of the solution $x(k) = 0$ for all k.

The following lemma summarizes the conditions we use later in the chapter for establishing exponential stability of distributed MPC. A similar lemma establishing asymptotic stability of suboptimal MPC was given by Scokaert, Mayne, and Rawlings (1999) (Theorem 1).

First we recall the definition of exponential stability.

Definition 6.4 (Exponential stability)**.** Let $\mathbb{X}$ be positive invariant set for $x^+ = f(x)$. Then the origin is exponentially stable in $\mathbb{X}$ for $x^+ = f(x)$ if there exists $c > 0$ and $0 < \gamma < 1$ such that for each $x \in \mathbb{X}$

$$|\phi(i;x)| \leq c\,|x|\,\gamma^i$$

for all $i \geq \mathbb{I}_{\geq 0}$.

Consider next the suboptimal MPC controller. Let the system satisfy $(x^+, \mathbf{u}^+) = (f(x,\mathbf{u}), g(x,\mathbf{u}))$ with initial sequence $\mathbf{u}(0) = \mathbf{h}(x(0))$. The controller constraints are $x(i) \in \mathbb{X} \subseteq \mathbb{R}^n$ for all $i \in \mathbb{I}_{0:N}$ and $u(i) \in \mathbb{U} \subseteq \mathbb{R}^m$ for all $i \in \mathbb{I}_{0:N-1}$. Let $\mathcal{X}_N$ denote the set of states for which the MPC controller is feasible.

Lemma 6.5 (Exponential stability of suboptimal MPC)**.** *Assume that the suboptimal MPC system satisfies the following inequalities with $r, a, b, c > 0$*

$$\begin{aligned} a\,|(x,\mathbf{u})|^2 \le V(x,\mathbf{u}) \le b\,|(x,\mathbf{u})|^2 \qquad & x \in \mathcal{X}_N \quad \mathbf{u} \in \mathbb{U}^N \\ V(x^+,\mathbf{u}^+) - V(x,\mathbf{u}) \le -c\,|(x,u(0))|^2 \qquad & x \in \mathcal{X}_N \quad \mathbf{u} \in \mathbb{U}^N \\ |\mathbf{u}| \le d\,|x| \qquad & x \in r\mathcal{B} \end{aligned}$$

Then the origin is exponentially stable for the closed-loop system under suboptimal MPC with region of attraction $\mathcal{X}_N$ if either of the following additional assumptions holds

(a) $\mathbb{U}$ is compact. In this case, $\mathcal{X}_N$ may be unbounded.

(b) $\mathcal{X}_N$ is compact. In this case $\mathbb{U}$ may be unbounded.

Proof. First we show that the origin of the extended state $(x,\mathbf{u})$ is exponentially stable for $x(0) \in \mathcal{X}_N$.

(a) For the case $\mathbb{U}$ compact, we have $|\mathbf{u}| \le d\,|x|\,, x \in r\mathcal{B}$. Consider the optimization

$$\max_{\mathbf{u}\in\mathbb{U}^N} |\mathbf{u}| = s > 0$$

The solution exists by the Weierstrass theorem since $\mathbb{U}$ is compact, which implies $\mathbb{U}^N$ is compact. Then we have $|\mathbf{u}| \le (s/r)\,|x|$ for $x \in \mathcal{X}_N \setminus r\mathcal{B}$, so we have $|\mathbf{u}| \le d'\,|x|$ for $x \in \mathcal{X}_N$ in which $d' = \max(d, s/r)$.

(b) For the case $\mathcal{X}_N$ compact, consider the optimization

$$\max_{x\in\mathcal{X}_N} V(x,\mathbf{h}(x)) = \bar{V} > 0$$

The solution exists because $\mathcal{X}_N$ is compact and $\mathbf{h}(\cdot)$ and $V(\cdot)$ are continuous. Define the compact set $\bar{\mathbb{U}}$ by

$$\bar{\mathbb{U}} = \{\mathbf{u} \mid V(x,\mathbf{u}) \le \bar{V}, \quad x \in \mathcal{X}_N\}$$

The set is bounded because $V(x,\mathbf{u}) \ge a\,|(x,\mathbf{u})|^2 \ge a\,|\mathbf{u}|^2$. The set is closed because V is continuous. The significance of this set is that for

all $k \geq 0$ and all $x \in \mathcal{X}_N$, $\mathbf{u}(k) \in \bar{\mathbb{U}}$. Therefore we have established that $\mathcal{X}_N$ compact implies $\mathbf{u}(k)$ evolves in a compact set as in the previous case when $\mathbb{U}$ is assumed compact. Using the same argument as in that case, we have established that there exists $d' > 0$ such that $|\mathbf{u}| \leq d' |x|$ for all $x \in \mathcal{X}_N$.

For the two cases, we therefore have established for all $x \in \mathcal{X}_N$, $\mathbf{u} \in \mathbb{U}^N$ (case (a)) or $\mathbf{u} \in \bar{\mathbb{U}}$ (case (b))

$$|(x, \mathbf{u})| \leq |x| + |\mathbf{u}| \leq |x| + d' |x| \leq (1 + d') |x|$$

which gives $|x| \geq c' |(x, \mathbf{u})|$ with $c' = 1/(1 + d') > 0$. Hence, there exists $a_3 = c(c')^2$ such that $V(x^+, \mathbf{u}^+) - V(x, \mathbf{u}) \leq -a_3 |(x, \mathbf{u})|^2$ for all $x \in \mathcal{X}_N$. Therefore the extended state $(x, \mathbf{u})$ satisfies the standard conditions of an exponential stability Lyapunov function (see Theorem B.19 in Appendix B) with $a_1 = a, a_2 = b, a_3 = c(c')^2, \sigma = 2$ for $(x, \mathbf{u}) \in \mathcal{X}_N \times \mathbb{U}^N$ (case (a)) or $\mathcal{X}_N \times \bar{\mathbb{U}}$ (case (b)). Therefore for all $x(0) \in \mathcal{X}_N$, $k \geq 0$

$$|(x(k), \mathbf{u}(k))| \leq \alpha |(x(0), \mathbf{u}(0))| \gamma^k$$

in which $\alpha > 0$ and $0 < \gamma < 1$.

Finally we remove the input sequence and establish that the origin for the state (rather than the extended state) is exponentially stable for the closed-loop system. We have for all $x(0) \in \mathcal{X}_N$ and $k \geq 0$

$$\begin{aligned} |x(k)| &\leq |(x(k), \mathbf{u}(k))| \leq \alpha |(x(0), \mathbf{u}(0))| \gamma^k \\ &\leq \alpha(|x(0)| + |\mathbf{u}(0)|)\gamma^k \leq \alpha(1 + d') |x(0)| \gamma^k \\ &\leq \alpha' |x(0)| \gamma^k \end{aligned}$$

in which $\alpha' = \alpha(1 + d') > 0$, and we have established exponential stability of the origin on the feasible set $\mathcal{X}_N$. ■

Exercises 6.7 and 6.8 explore what to conclude about exponential stability when both $\mathbb{U}$ and $\mathcal{X}_N$ are unbounded.

We also consider later in the chapter the effects of state estimation error on the closed-loop properties of distributed MPC. For analyzing stability under perturbations, the following lemma is useful. Here e plays the role of estimation error.

Lemma 6.6 (Global asymptotic stability and exponential convergence with mixed powers of norm)**.** *Consider a dynamic system*

$$(x^+, e^+) = f(x, e)$$

with a zero steady-state solution, $f(0,0) = (0,0)$. Assume there exists a function $V : \mathbb{R}^{n+m} \to \mathbb{R}_{\geq 0}$ that satisfies the following for all $(x,e) \in \mathbb{R}^n \times \mathbb{R}^m$

$$a(|x|^\sigma + |e|^\gamma) \leq V((x,e)) \leq b(|x|^\sigma + |e|^\gamma) \tag{6.6}$$

$$V(f(x,e)) - V((x,e)) \leq -c(|x|^\sigma + |e|^\gamma) \tag{6.7}$$

with constants $a, b, c, \sigma, \gamma > 0$. Then the following holds for all $(x(0), e(0))$ and $k \in \mathbb{I}_{\geq 0}$

$$|x(k), e(k)| \leq \delta(|x(0), e(0)|)\lambda^k$$

with $\lambda < 1$ and $\delta(\cdot) \in \mathcal{K}_\infty$.

The proof of this lemma is discussed in Exercise 6.9. We also require a converse theorem for exponential stability.

Lemma 6.7 (Converse theorem for exponential stability)**.** *If the zero steady-state solution of $x^+ = f(x)$ is globally exponentially stable, then there exists Lipschitz continuous $V : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ that satisfies the following: there exist constants $a, b, c, \sigma > 0$, such that for all $x \in \mathbb{R}^n$*

$$a\,|x|^\sigma \leq V(x) \leq b\,|x|^\sigma$$

$$V(f(x)) - V(x) \leq -c\,|x|^\sigma$$

Moreover, any $\sigma > 0$ is valid, and the constant c can be chosen as large as one wishes.

The proof of this lemma is discussed in Exercise B.3.

6.2 Unconstrained Two-Player Game

To introduce clearly the concepts and notation required to analyze distributed MPC, we start with a two-player game. We then generalize to an M-player game in the next section.

Let (A_{11}, B_{11}, C_{11}) be a minimal state space realization of the (u_1, y_1) input-output pair. Similarly, let (A_{12}, B_{12}, C_{12}) be a minimal state space realization of the (u_2, y_1) input-output pair. The dimensions are $u_1 \in \mathbb{R}^{m_1}$, $y_1 \in \mathbb{R}^{p_1}$, $x_{11} \in \mathbb{R}^{n_{11}}$, $x_{12} \in \mathbb{R}^{n_{12}}$ with $n_1 = n_{11} + n_{12}$. Output y_1 can then be represented as the following, possibly nonminimal, state space model

$$\begin{bmatrix} x_{11} \\ x_{12} \end{bmatrix}^+ = \begin{bmatrix} A_{11} & 0 \\ 0 & A_{12} \end{bmatrix} \begin{bmatrix} x_{11} \\ x_{12} \end{bmatrix} + \begin{bmatrix} B_{11} \\ 0 \end{bmatrix} u_1 + \begin{bmatrix} 0 \\ B_{12} \end{bmatrix} u_2$$

$$y_1 = \begin{bmatrix} C_{11} & C_{12} \end{bmatrix} \begin{bmatrix} x_{11} \\ x_{12} \end{bmatrix}$$

Proceeding in an analogous fashion with output y_2 and inputs u_1 and u_2, we model y_2 with the following state space model

$$\begin{bmatrix} x_{22} \\ x_{21} \end{bmatrix}^+ = \begin{bmatrix} A_{22} & 0 \\ 0 & A_{21} \end{bmatrix} \begin{bmatrix} x_{22} \\ x_{21} \end{bmatrix} + \begin{bmatrix} B_{22} \\ 0 \end{bmatrix} u_2 + \begin{bmatrix} 0 \\ B_{21} \end{bmatrix} u_1$$
$$y_2 = \begin{bmatrix} C_{22} & C_{21} \end{bmatrix} \begin{bmatrix} x_{22} \\ x_{21} \end{bmatrix}$$

We next define player one's local cost functions

$$V_1(x_1(0), \mathbf{u}_1, \mathbf{u}_2) = \sum_{k=0}^{N-1} \ell_1(x_1(k), u_1(k)) + V_{1f}(x_1(N))$$

in which

$$x_1 = \begin{bmatrix} x_{11} \\ x_{12} \end{bmatrix}$$

Note that the first local objective is affected by the second player's inputs through the model evolution of x_1, i.e., through the x_{12} states. We choose the stage cost to account for the first player's inputs and outputs

$$\ell_1(x_1, u_1) = (1/2)(y_1' \overline{Q}_1 y_1 + u_1' R_1 u_1)$$
$$\ell_1(x_1, u_1) = (1/2)(x_1' Q_1 x_1 + u_1' R_1 u_1)$$

in which

$$Q_1 = C_1' \overline{Q}_1 C_1 \qquad C_1 = \begin{bmatrix} C_{11} & C_{12} \end{bmatrix}$$

Motivated by the warm start to be described later, for stable systems, we choose the terminal penalty to be the infinite horizon cost to go under zero control

$$V_{1f}(x_1(N)) = (1/2) x_1'(N) P_{1f} x_1(N)$$

We choose P_{1f} as the solution to the following Lyapunov equation assuming A_1 is stable

$$A_1' P_{1f} A_1 - P_{1f} = -Q_1 \tag{6.8}$$

We proceed analogously to define player two's local objective function and penalties

$$V_2(x_2(0), \mathbf{u}_1, \mathbf{u}_2) = \sum_{k=0}^{N-1} \ell_2(x_2(k), u_2(k)) + V_{2f}(x_2(N))$$

In centralized control and the cooperative game, the two players share a common objective, which can be considered to be the overall plant objective

$$V(x_1(0), x_2(0), \mathbf{u}_1, \mathbf{u}_2) = \rho_1 V_1(x_1(0), \mathbf{u}_1, \mathbf{u}_2) + \rho_2 V_2(x_2(0), \mathbf{u}_2, \mathbf{u}_1)$$

in which the parameters ρ_1, ρ_2 are used to specify the relative weights of the two subsystems in the overall plant objective. Their values are restricted so $\rho_1, \rho_2 > 0$, $\rho_1 + \rho_2 = 1$ so that both local objectives must have some nonzero effect on the overall plant objective.

6.2.1 Centralized Control

Centralized control requires the solution of the systemwide control problem. It can be stated as

$$\begin{aligned} \min_{\mathbf{u}_1, \mathbf{u}_2} & V(x_1(0), x_2(0), \mathbf{u}_1, \mathbf{u}_2) \\ \text{s.t. } x_1^+ &= A_1 x_1 + \overline{B}_{11} u_1 + \overline{B}_{12} u_2 \\ x_2^+ &= A_2 x_2 + \overline{B}_{22} u_2 + \overline{B}_{21} u_1 \end{aligned}$$

in which

$$A_1 = \begin{bmatrix} A_{11} & 0 \\ 0 & A_{12} \end{bmatrix} \qquad A_2 = \begin{bmatrix} A_{22} & 0 \\ 0 & A_{21} \end{bmatrix}$$

$$\overline{B}_{11} = \begin{bmatrix} B_{11} \\ 0 \end{bmatrix} \qquad \overline{B}_{12} = \begin{bmatrix} 0 \\ B_{12} \end{bmatrix} \qquad \overline{B}_{21} = \begin{bmatrix} 0 \\ B_{21} \end{bmatrix} \qquad \overline{B}_{22} = \begin{bmatrix} B_{22} \\ 0 \end{bmatrix}$$

This optimal control problem is more complex than all of the distributed cases to follow because the decision variables include both $\mathbf{u}_1$ and $\mathbf{u}_2$. Because the performance is optimal, centralized control is a natural benchmark against which to compare the distributed cases: cooperative, noncooperative, and decentralized MPC. The plantwide stage cost and terminal cost can be expressed as quadratic functions of the subsystem states and inputs

$$\begin{aligned} \ell(x, u) &= (1/2)(x'Qx + u'Ru) \\ V_f(x) &= (1/2)x'P_f x \end{aligned}$$

in which

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \qquad u = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \qquad Q = \begin{bmatrix} \rho_1 Q_1 & 0 \\ 0 & \rho_2 Q_2 \end{bmatrix}$$

$$R = \begin{bmatrix} \rho_1 R_1 & 0 \\ 0 & \rho_2 R_2 \end{bmatrix} \qquad P_f = \begin{bmatrix} \rho_1 P_{1f} & 0 \\ 0 & \rho_2 P_{2f} \end{bmatrix} \tag{6.9}$$

and we have the standard MPC problem considered in Chapters 1 and 2

$$\begin{aligned} &\min_{\mathbf{u}} V(x(0), \mathbf{u}) \\ &\text{s.t. } x^+ = Ax + Bu \end{aligned} \tag{6.10}$$

in which

$$A = \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} \qquad B = \begin{bmatrix} \overline{B}_{11} & \overline{B}_{12} \\ \overline{B}_{21} & \overline{B}_{22} \end{bmatrix} \tag{6.11}$$

Given the terminal penalty in (6.8), stability of the closed-loop centralized system is guaranteed for all choices of system models and tuning parameters subject to the usual stabilizability assumption on the system model.

6.2.2 Decentralized Control

Centralized and decentralized control define the two extremes in distributing the decision making in a large-scale system. Centralized control has full information and optimizes the full control problem over all decision variables. Decentralized control, on the other hand, optimizes only the local objectives and has no information about the actions of the other subsystems. Player one's objective function is

$$V_1(x_1(0), \mathbf{u}_1) = \sum_{k=0}^{N-1} \ell_1(x_1(k), u_1(k)) + V_{1f}(x_1(N))$$

We then have player one's decentralized control problem

$$\begin{aligned} &\min_{\mathbf{u}_1} V_1(x_1(0), \mathbf{u}_1) \\ &\text{s.t. } x_1^+ = A_1 x_1 + \overline{B}_{11} u_1 \end{aligned}$$

We know the optimal solution for this kind of LQ problem is a linear feedback law

$$u_1^0 = K_1 x_1(0)$$

Notice that in decentralized control, player one's model does not account for the inputs of player two, and already contains model error. In the decentralized problem, player one requires no information about player two. The communication overhead for decentralized control is therefore minimal, which is an implementation advantage, but the resulting performance may be quite poor for systems with reasonably

strong coupling. We compute an optimal K_1 for system one $(A_1, \overline{B}_{11}, Q_1, R_1)$ and optimal K_2 for system 2. The closed-loop system evolution is then

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}^+ = \begin{bmatrix} A_1 + \overline{B}_{11}K_1 & \overline{B}_{12}K_2 \\ \overline{B}_{21}K_1 & A_2 + \overline{B}_{22}K_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

and we know only that $A_{11} + \overline{B}_{11}K_1$ and $A_{22} + \overline{B}_{22}K_2$ are stable matrices. Obviously the stability of the closed-loop, decentralized system is fragile and depends in a sensitive way on the sizes of the interaction terms $\overline{B}_{12}$ and $\overline{B}_{21}$ and feedback gains K_1, K_2.

6.2.3 Noncooperative Game

In the noncooperative game, player one optimizes $V_1(x_1(0), \mathbf{u}_1, \mathbf{u}_2)$ over $\mathbf{u}_1$ and player two optimizes $V_2(x_2(0), \mathbf{u}_1, \mathbf{u}_2)$ over $\mathbf{u}_2$. From player one's perspective, player two's planned inputs $\mathbf{u}_2$ are known disturbances affecting player one's output through the dynamic model. Part of player one's optimal control problem is therefore to compensate for player two's inputs with his optimal $\mathbf{u}_1$ sequence in order to optimize his local objective V_1. Similarly, player two considers player one's inputs as a known disturbance and solves an optimal control problem that removes their effect in his local objective V_2. Because this game is noncooperative ($V_1 \neq V_2$), the struggle between players one and two can produce an outcome that is bad for both of them as we show subsequently. Notice that unlike decentralized control, there is no model error in the noncooperative game. Player one knows exactly the effect of the actions of player two and vice versa. Any poor nominal performance is caused by the noncooperative game, not model error.

Summarizing the noncooperative control problem statement, player one's model is

$$x_1^+ = A_1 x_1 + \overline{B}_{11} u_1 + \overline{B}_{12} u_2$$

and player one's objective function is

$$V_1(x_1(0), \mathbf{u}_1, \mathbf{u}_2) = \sum_{k=0}^{N-1} \ell_1(x_1(k), u_1(k)) + V_{1f}(x_1(N))$$

Note that V_1 here depends on $\mathbf{u}_2$ because the state trajectory $x_1(k)$, $k \geq 1$ depends on $\mathbf{u}_2$ as shown in player one's dynamic model. We then have player one's noncooperative control problem

$$\min_{\mathbf{u}_1} V_1(x_1(0), \mathbf{u}_1, \mathbf{u}_2)$$
$$\text{s.t. } x_1^+ = A_1 x_1 + \overline{B}_{11} u_1 + \overline{B}_{12} u_2$$

Solution to player one's optimal control problem. We now solve player one's optimal control problem. Proceeding as in Section 6.1.1 we define

$$\mathbf{z} = \begin{bmatrix} u_1(0) \\ x_1(1) \\ \vdots \\ u_1(N-1) \\ x_1(N) \end{bmatrix} \qquad H = \operatorname{diag}\left(\begin{bmatrix} R_1 & Q_1 & \cdots & R_1 & P_{1f} \end{bmatrix}\right)$$

and can express player one's optimal control problem as

$$\min_{\mathbf{z}} (1/2)(\mathbf{z}' H \mathbf{z} + x_1(0)' Q_1 x_1(0))$$
$$\text{s.t. } D\mathbf{z} = d$$

in which

$$D = -\begin{bmatrix} \overline{B}_{11} & -I & & & & & \\ & A_1 & \overline{B}_{11} & -I & & & \\ & & & \ddots & & & \\ & & & & A_1 & \overline{B}_{11} & -I \end{bmatrix}$$

$$d = \begin{bmatrix} A_1 x_1(0) + \overline{B}_{12} u_2(0) \\ \overline{B}_{12} u_2(1) \\ \vdots \\ \overline{B}_{12} u_2(N-1) \end{bmatrix}$$

We then apply (1.57) to obtain

$$\begin{bmatrix} H & -D' \\ -D & 0 \end{bmatrix} \begin{bmatrix} \mathbf{z} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} 0 \\ -\widetilde{A}_1 \end{bmatrix} x_1(0) + \begin{bmatrix} 0 \\ -\widetilde{B}_{12} \end{bmatrix} \mathbf{u}_2 \tag{6.12}$$

in which we have defined

$$\boldsymbol{\lambda} = \begin{bmatrix} \lambda(1) \\ \lambda(2) \\ \vdots \\ \lambda(N) \end{bmatrix} \qquad \widetilde{A}_1 = \begin{bmatrix} A_1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \qquad \widetilde{B}_{12} = \begin{bmatrix} \overline{B}_{12} & & & \\ & \overline{B}_{12} & & \\ & & \ddots & \\ & & & \overline{B}_{12} \end{bmatrix}$$

Solving this equation and picking out the rows of $\mathbf{z}$ corresponding to the elements of $\mathbf{u}_1$ gives

$$\mathbf{u}_1^0 = K_1 x_1(0) + L_1 \mathbf{u}_2$$

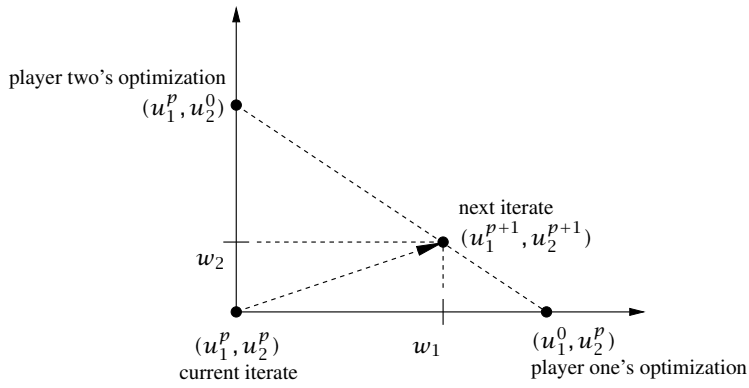


Figure 6.1: Convex step from (u_1^p, u_2^p) to (u_1^{p+1}, u_2^{p+1}); the parameters w_1, w_2 with $w_1 + w_2 = 1$ determine location of next iterate on line joining the two players' optimizations: (u_1^0, u_2^p) and (u_1^p, u_2^0).

and we see player one's optimal decision depends linearly on his initial state, but also on *player two's decision.* This is the key difference between decentralized control and noncooperative control. In noncooperative control, player two's decisions are communicated to player one and player one accounts for them in optimizing the local objective.

Convex step. Let $p \in \mathbb{I}_{\geq 0}$ denote the integer-valued iteration in the optimization problem. Looking ahead to the M-player game, we do not take the full step, but a convex combination of the current optimal solution, $\mathbf{u}_1^0$, and the current iterate, $\mathbf{u}_1^p$

$$\mathbf{u}_1^{p+1} = w_1 \mathbf{u}_1^0 + (1 - w_1)\mathbf{u}_1^p \qquad 0 < w_1 < 1$$

This iteration is displayed in Figure 6.1. Notice we have chosen a distributed optimization of the Gauss-Jacobi type (see Bertsekas and Tsitsiklis, 1997, pp.219–223).

We place restrictions on the systems under consideration before analyzing stability of the controller.

Assumption 6.8 (Unconstrained two-player game)**.**

(a) All subsystems, $A_{ij}, i = 1, 2, j = 1, 2$, are stable.

(b) The controller penalties Q_1, Q_2, R_1, R_2 are positive definite.

The assumption of stable models is purely for convenience of exposition. We treat unstable, stabilizable systems in Section 6.3.

Convergence of the players' iteration. To understand the convergence of the two players' iterations, we express both players' moves as follows

$$\begin{aligned}\mathbf{u}_1^{p+1} &= w_1\mathbf{u}_1^0 + (1-w_1)\mathbf{u}_1^p\\ \mathbf{u}_2^{p+1} &= w_2\mathbf{u}_2^0 + (1-w_2)\mathbf{u}_2^p\\ 1 &= w_1 + w_2 \qquad 0 < w_1, w_2 < 1\end{aligned}$$

or

$$\begin{bmatrix}\mathbf{u}_1\\ \mathbf{u}_2\end{bmatrix}^{p+1} = \begin{bmatrix}w_1 I & 0\\ 0 & w_2 I\end{bmatrix}\begin{bmatrix}\mathbf{u}_1^0\\ \mathbf{u}_2^0\end{bmatrix} + \begin{bmatrix}(1-w_1)I & 0\\ 0 & (1-w_2)I\end{bmatrix}\begin{bmatrix}\mathbf{u}_1\\ \mathbf{u}_2\end{bmatrix}^p$$

The optimal control for each player is

$$\begin{bmatrix}\mathbf{u}_1^0\\ \mathbf{u}_2^0\end{bmatrix} = \begin{bmatrix}K_1 & 0\\ 0 & K_2\end{bmatrix}\begin{bmatrix}x_1(0)\\ x_2(0)\end{bmatrix} + \begin{bmatrix}0 & L_1\\ L_2 & 0\end{bmatrix}\begin{bmatrix}\mathbf{u}_1\\ \mathbf{u}_2\end{bmatrix}^p$$

Substituting the optimal control into the iteration gives

$$\begin{bmatrix}\mathbf{u}_1\\ \mathbf{u}_2\end{bmatrix}^{p+1} = \underbrace{\begin{bmatrix}w_1K_1 & 0\\ 0 & w_2K_2\end{bmatrix}}_{\overline{K}}\begin{bmatrix}x_1(0)\\ x_2(0)\end{bmatrix} + \underbrace{\begin{bmatrix}(1-w_1)I & w_1L_1\\ w_2L_2 & (1-w_2)I\end{bmatrix}}_{L}\begin{bmatrix}\mathbf{u}_1\\ \mathbf{u}_2\end{bmatrix}^p$$

Finally writing this equation in the plantwide notation, we express the iteration as

$$\mathbf{u}^{p+1} = \overline{K}x(0) + L\mathbf{u}^p$$

The convergence of the two players' control iteration is governed by the eigenvalues of L. If L is stable, the control sequence converges to

$$\mathbf{u}^\infty = (I-L)^{-1}\overline{K}x(0) \qquad |\lambda| < 1 \text{ for } \lambda \in \text{eig}(L)$$

in which

$$\begin{aligned}(I-L)^{-1}\overline{K} &= \begin{bmatrix}w_1 I & -w_1L_1\\ -w_2L_2 & w_2 I\end{bmatrix}^{-1}\begin{bmatrix}w_1K_1 & 0\\ 0 & w_2K_2\end{bmatrix}\\ (I-L)^{-1}\overline{K} &= \begin{bmatrix}I & -L_1\\ -L_2 & I\end{bmatrix}^{-1}\begin{bmatrix}K_1 & 0\\ 0 & K_2\end{bmatrix}\end{aligned}$$

Note that the weights w_1, w_2 do not appear in the converged input sequence. The $\mathbf{u}_1^\infty$, $\mathbf{u}_2^\infty$ pair have the equilibrium property that neither player can improve his position given the other player's current decision. This point is called a Nash equilibrium (Başar and Olsder, 1999, p. 4). Notice that the distributed MPC game does not have a Nash equilibrium if the eigenvalues of L are on or outside the unit circle. If the controllers have sufficient time during the control system's sample time to iterate to convergence, then the effect of the initial control sequence is removed by using the converged control sequence. If the iteration has to be stopped before convergence, the solution is

$$\mathbf{u}^{p+1} = L^p \mathbf{u}^{[0]} + \sum_{j=0}^{p-1} L^j \overline{K} x(0) \qquad 0 \le p$$

in which $\mathbf{u}^{[0]}$ is the $p = 0$ (initial) input sequence. We use the brackets with $p = 0$ to distinguish this initial input sequence from an optimal input sequence.

Stability of the closed-loop system. We assume the Nash equilibrium is stable and there is sufficient computation time to iterate to convergence.

We require a matrix of zeros and ones to select the first move from the input sequence for injection into the plant. For the first player, the required matrix is

$$\begin{aligned} u_1(0) &= E_1 \mathbf{u}_1 \\ E_1 &= \begin{bmatrix} I_{m_1} & 0_{m_1} & \dots & 0_{m_1} \end{bmatrix} \qquad m_1 \times m_1 N \text{ matrix} \end{aligned}$$

The closed-loop system is then

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}^+ = \underbrace{\begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix}}_{A} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \underbrace{\begin{bmatrix} \overline{B}_{11} & \overline{B}_{12} \\ \overline{B}_{21} & \overline{B}_{22} \end{bmatrix}}_{B} \underbrace{\begin{bmatrix} E_1 & 0 \\ 0 & E_2 \end{bmatrix} \begin{bmatrix} I & -L_1 \\ -L_2 & I \end{bmatrix}^{-1} \begin{bmatrix} K_1 & 0 \\ 0 & K_2 \end{bmatrix}}_{K} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Using the plantwide notation for this equation and defining the feedback gain K gives

$$x^+ = (A + BK)x$$

The stability of the closed loop with converged, noncooperative control is therefore determined by the eigenvalues of $(A + BK)$.

We next present three simple examples to show that (i) the Nash equilibrium may not be stable (L is unstable), (ii) the Nash equilibrium may be stable but the closed loop is unstable (L is stable, $A + BK$ is unstable), and (iii) the Nash equilibrium may be stable and the closed loop is stable (L is stable, $A + BK$ is stable). Which situation arises depends in a nonobvious way on all of the problem data: A_1, A_2, $\overline{B}_{11}$, $\overline{B}_{12}$, $\overline{B}_{21}$, $\overline{B}_{22}$, Q_1, Q_2, P_{1f}, P_{2f}, R_1, R_2, w_1, w_2, N. One has to examine the eigenvalues of L and $A+BK$ for each application of interest to know how the noncooperative distributed MPC is going to perform. Even for a fixed dynamic model, when changing tuning parameters such as Q, P_f, R, w, one has to examine eigenvalues of L and $A + BK$ to know the effect on the closed-loop system. This is the main drawback of the noncooperative game. In many control system design methods, such as all forms of MPC presented in Chapter 2, closed-loop properties such as exponential stability are guaranteed for the *nominal* system for all choices of performance tuning parameters. Noncooperative distributed MPC does not have this feature and a stability analysis is required. We show in the next section that cooperative MPC does not suffer from this drawback, at the cost of slightly more information exchange.

Example 6.9: Nash equilibrium is unstable

Consider the following transfer function matrix for a simple two-input two-output system

$$\begin{bmatrix} y_1(s) \\ y_2(s) \end{bmatrix} = \begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix} \begin{bmatrix} u_1(s) \\ u_2(s) \end{bmatrix}$$

in which

$$G(s) = \begin{bmatrix} \dfrac{1}{s^2 + 2(0.2)s + 1} & \dfrac{0.5}{0.225s + 1} \\ \dfrac{-0.5}{(0.5s + 1)(0.25s + 1)} & \dfrac{1.5}{0.75s^2 + 2(0.8)(0.75)s + 1} \end{bmatrix}$$

Obtain discrete time models (A_{ij}, B_{ij}, C_{ij}) for each of the four transfer functions $G_{ij}(s)$ using a sample time of $T = 0.2$ and zero-order holds on the inputs. Set the control cost function parameters to be

$$\overline{Q}_1 = \overline{Q}_2 = 1 \qquad \overline{P}_{1f} = \overline{P}_{2f} = 0 \qquad R_1 = R_2 = 0.01$$
$$N = 30 \qquad w_1 = w_2 = 0.5$$

Compute the eigenvalues of the L matrix for this system using noncooperative MPC. Show the Nash equilibrium is unstable and the closed-loop system is therefore unstable. Discuss why this system is problematic for noncooperative control.

Solution

For this problem L is a 60×60 matrix ($N(m_1 + m_2)$). The magnitudes of the largest eigenvalues are

$$\left|\operatorname{eig}(L)\right| = \begin{bmatrix} 1.11 & 1.11 & 1.03 & 1.03 & 0.914 & 0.914 & \cdots \end{bmatrix}$$

The noncooperative iteration does not converge. The steady-state gains for this system are

$$G(0) = \begin{bmatrix} 1 & 0.5 \\ -0.5 & 1.5 \end{bmatrix}$$

and we see that the diagonal elements are reasonably large compared to the nondiagonal elements. So the *steady-state* coupling between the two systems is relatively weak. The dynamic coupling is unfavorable, however. The response of y_1 to u_2 is more than four times faster than the response of y_1 to u_1. The faster input is the disturbance and the slower input is used for control. Likewise the response of y_2 to u_1 is three times faster than the response of y_2 to u_2. Also in the second loop, the faster input is the disturbance and the slower input is used for control. These pairings are unfavorable dynamically, and that fact is revealed in the instability of L and lack of a Nash equilibrium for the noncooperative dynamic regulation problem. □

Example 6.10: Nash equilibrium is stable but closed loop is unstable

Switch the outputs for the previous example and compute the eigenvalues of L and $(A+BK)$ for the noncooperative distributed MPC regulator for the system

$$G(s) = \begin{bmatrix} \dfrac{-0.5}{(0.5s+1)(0.25s+1)} & \dfrac{1.5}{0.75s^2 + 2(0.8)(0.75)s + 1} \\ \dfrac{1}{s^2 + 2(0.2)s + 1} & \dfrac{0.5}{0.225s + 1} \end{bmatrix}$$

Show in this case that the Nash equilibrium is stable, but the noncooperative regulator destabilizes the system. Discuss why this system is problematic for noncooperative control.

Solution

For this case the largest magnitude eigenvalues of L are

$$|\operatorname{eig}(L)| = \begin{bmatrix} 0.63 & 0.63 & 0.62 & 0.62 & 0.59 & 0.59 & \cdots \end{bmatrix}$$

and we see the Nash equilibrium for the noncooperative game is stable. So we have removed the first source of closed-loop instability by switching the input-output pairings of the two subsystems. There are seven states in the complete system model, and the magnitudes of the eigenvalues of the closed-loop regulator $(A + BK)$ are

$$|\operatorname{eig}(A + BK)| = \begin{bmatrix} 1.03 & 1.03 & 0.37 & 0.37 & 0.77 & 0.77 & 0.04 \end{bmatrix}$$

which also gives an unstable closed-loop system. We see the distributed noncooperative regulator has destabilized a stable open-loop system. The problem with this pairing is the steady-state gains are now

$$G(0) = \begin{bmatrix} -0.5 & 1.5 \\ 1 & 0.5 \end{bmatrix}$$

If one computes any steady-state interaction measure, such as the relative gain array (RGA), we see the new pairings are poor from a steady-state interaction perspective

$$\text{RGA} = \begin{bmatrix} 0.14 & 0.86 \\ 0.86 & 0.14 \end{bmatrix}$$

Neither pairing of the inputs and outputs is closed-loop stable with noncooperative distributed MPC.

Decentralized control with this pairing is discussed in Exercise 6.10. □

Example 6.11: Nash equilibrium is stable and the closed loop is stable

Next consider the system

$$G(s) = \begin{bmatrix} \dfrac{1}{s^2 + 2(0.2)s + 1} & \dfrac{0.5}{0.9s + 1} \\ \dfrac{-0.5}{(2s+1)(s+1)} & \dfrac{1.5}{0.75s^2 + 2(0.8)(0.75)s + 1} \end{bmatrix}$$

Compute the eigenvalues of L and $A + BK$ for this system. What do you conclude about noncooperative distributed MPC for this system?

Solution

This system is not difficult to handle with distributed control. The gains are the same as in the original pairing in Example 6.9, and the steady-state coupling between the two subsystems is reasonably weak. Unlike Example 6.9, however, the responses of y_1 to u_2 and y_2 to u_1 have been slowed so they are not faster than the responses of y_1 to u_1 and y_2 to u_2, respectively. Computing the eigenvalues of L and $A+BK$ for noncooperative control gives

$$|\operatorname{eig}(L)| = \begin{bmatrix} 0.61 & 0.61 & 0.59 & 0.59 & 0.56 & 0.56 & 0.53 & 0.53 \cdots \end{bmatrix}$$

$$|\operatorname{eig}(A+BK)| = \begin{bmatrix} 0.88 & 0.88 & 0.74 & 0.67 & 0.67 & 0.53 & 0.53 \end{bmatrix}$$

The Nash equilibrium is stable since L is stable, and the closed loop is stable since both L and $A+BK$ are stable. □

These examples reveal the simple fact that communicating the actions of the other controllers does not guarantee acceptable closed-loop behavior. If the coupling of the subsystems is weak enough, both dynamically and in steady state, then the closed loop is stable. In this sense, noncooperative MPC has few advantages over completely decentralized control, which has this same basic property.

We next show how to obtain much better closed-loop properties while maintaining the small size of the distributed control problems.

6.2.4 Cooperative Game

In the cooperative game, the two players share a common objective, which can be considered to be the overall plant objective

$$V(x_1(0), x_2(0), \mathbf{u}_1, \mathbf{u}_2) = \rho_1 V_1(x_1(0), \mathbf{u}_1, \mathbf{u}_2) + \rho_2 V_2(x_2(0), \mathbf{u}_2, \mathbf{u}_1)$$

in which the parameters ρ_1, ρ_2 are used to specify the relative weights of the two subsystems in the overall plant objective. In the cooperative problem, each player keeps track of *how his input affects the other player's output* as well as his own output. We can implement this cooperative game in several ways. The implementation leading to the simplest notation is to combine x_1 and x_2 into a single model

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}^+ = \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} \overline{B}_{11} \\ \overline{B}_{21} \end{bmatrix} u_1 + \begin{bmatrix} \overline{B}_{12} \\ \overline{B}_{22} \end{bmatrix} u_2$$

and then express player one's stage cost as

$$\ell_1(x_1, x_2, u_1) = \frac{1}{2}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}' \begin{bmatrix} \rho_1 Q_1 & 0 \\ 0 & \rho_2 Q_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \frac{1}{2} u_1' (\rho_1 R_1) u_1 + \text{const.}$$

$$V_{1f}(x_1, x_2) = \frac{1}{2}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}' \begin{bmatrix} \rho_1 P_{1f} & 0 \\ 0 & \rho_2 P_{2f} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Notice that u_2 does not appear because the contribution of u_2 to the stage cost cannot be affected by player one, and can therefore be neglected. The cost function is then expressed as

$$V(x_1(0), x_2(0), \mathbf{u}_1, \mathbf{u}_2) = \sum_{k=0}^{N-1} \ell_1(x_1(k), x_2(k), u_1(k)) + V_{1f}(x_1(N), x_2(N))$$

Player one's optimal control problem is

$$\min_{\mathbf{u}_1} V(x_1(0), x_2(0), \mathbf{u}_1, \mathbf{u}_2)$$

$$\text{s.t. } \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}^+ = \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} \overline{B}_{11} \\ \overline{B}_{21} \end{bmatrix} u_1 + \begin{bmatrix} \overline{B}_{12} \\ \overline{B}_{22} \end{bmatrix} u_2$$

Note that this form is identical to the noncooperative form presented previously if we redefine the terms (noncooperative $\longrightarrow$ cooperative)

$$x_1 \to \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad A_1 \to \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} \quad \overline{B}_{11} \to \begin{bmatrix} \overline{B}_{11} \\ \overline{B}_{21} \end{bmatrix} \quad \overline{B}_{12} \to \begin{bmatrix} \overline{B}_{12} \\ \overline{B}_{22} \end{bmatrix}$$

$$Q_1 \to \begin{bmatrix} \rho_1 Q_1 & 0 \\ 0 & \rho_2 Q_2 \end{bmatrix} \qquad R_1 \to \rho_1 R_1 \quad P_{1f} \to \begin{bmatrix} \rho_1 P_{1f} & 0 \\ 0 & \rho_2 P_{2f} \end{bmatrix}$$

Any computational program written to solve either the cooperative or noncooperative optimal control problem can be used to solve the other.

Eliminating states $\mathbf{x}_2$. An alternative implementation is to remove states $x_2(k), k \geq 1$ from player one's optimal control problem by substituting the dynamic model of system two. This implementation reduces the size of the dynamic model because only states x_1 are retained. This reduction in model size may be important in applications with many players. The removal of states $x_2(k), k \geq 1$ also introduces linear terms into player one's objective function. We start by using the

dynamic model for x_2 to obtain

$$\begin{bmatrix} x_2(1) \\ x_2(2) \\ \vdots \\ x_2(N) \end{bmatrix} = \begin{bmatrix} A_2 \\ A_2^2 \\ \vdots \\ A_2^N \end{bmatrix} x_2(0) + \begin{bmatrix} \overline{B}_{21} & & & \\ A_2\overline{B}_{21} & \overline{B}_{21} & & \\ \vdots & \vdots & \ddots & \\ A_2^{N-1}\overline{B}_{21} & A_2^{N-2}\overline{B}_{21} & \dots & \overline{B}_{21} \end{bmatrix} \begin{bmatrix} u_1(0) \\ u_1(1) \\ \vdots \\ u_1(N-1) \end{bmatrix} + \begin{bmatrix} \overline{B}_{22} & & & \\ A_2\overline{B}_{22} & \overline{B}_{22} & & \\ \vdots & \vdots & \ddots & \\ A_2^{N-1}\overline{B}_{22} & A_2^{N-2}\overline{B}_{22} & \dots & \overline{B}_{22} \end{bmatrix} \begin{bmatrix} u_2(0) \\ u_2(1) \\ \vdots \\ u_2(N-1) \end{bmatrix}$$

Using more compact notation, we have

$$\mathbf{x}_2 = \mathcal{A}_2 x_2(0) + \mathcal{B}_{21}\mathbf{u}_1 + \mathcal{B}_{22}\mathbf{u}_2$$

We can use this relation to replace the cost contribution of $\mathbf{x}_2$ with linear and quadratic terms in $\mathbf{u}_1$ as follows

$$\sum_{k=0}^{N-1} x_2(k)'Q_2x_2(k) + x_2(N)'P_{2f}x_2(N) =$$
$$\mathbf{u}_1' \left[\mathcal{B}_{21}'\mathcal{Q}_2\mathcal{B}_{21}\right]\mathbf{u}_1 + 2\left[x_2(0)'\mathcal{A}_2' + \mathbf{u}_2'\mathcal{B}_{22}'\right]\mathcal{Q}_2\mathcal{B}_{21}\,\mathbf{u}_1 + \text{constant}$$

in which

$$\mathcal{Q}_2 = \text{diag}\left(\begin{bmatrix} Q_2 & Q_2 & \dots & P_{2f} \end{bmatrix}\right) \qquad Nn_2 \times Nn_2 \text{ matrix}$$

and the constant term contains products of $x_2(0)$ and $\mathbf{u}_2$, which are constant with respect to player one's decision variables and can therefore be neglected.

Next we insert the new terms created by eliminating $\mathbf{x}_2$ into the cost function. Assembling the cost function gives

$$\min_{\mathbf{z}} (1/2)\mathbf{z}'\widetilde{H}\mathbf{z} + h'\mathbf{z}$$
$$\text{s.t. } D\mathbf{z} = d$$

and (1.57) again gives the necessary and sufficient conditions for the optimal solution

$$\begin{bmatrix} \widetilde{H} & -D' \\ -D & 0 \end{bmatrix} \begin{bmatrix} \mathbf{z} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} 0 \\ -\widetilde{A}_1 \end{bmatrix} x_1(0) + \begin{bmatrix} -\widetilde{A}_2 \\ 0 \end{bmatrix} x_2(0) + \begin{bmatrix} -\widetilde{B}_{22} \\ -\widetilde{B}_{12} \end{bmatrix} \mathbf{u}_2 \quad (6.13)$$

in which

$$\widetilde{H} = H + E'\mathcal{B}_{21}'\mathcal{Q}_2\mathcal{B}_{21}E \qquad \widetilde{B}_{22} = E'\mathcal{B}_{21}'\mathcal{Q}_2\mathcal{B}_{22} \qquad \widetilde{A}_2 = E'\mathcal{B}_{21}'\mathcal{Q}_2\mathcal{A}_2$$
$$E = I_N \otimes \begin{bmatrix} I_{m_1} & 0_{m_1,n_1} \end{bmatrix}$$

See also Exercise 6.13 for details on constructing the padding matrix E. Comparing the cooperative and noncooperative dynamic games, (6.13) and (6.12), we see the cooperative game has made three changes: (i) the quadratic penalty H has been modified, (ii) the effect of $x_2(0)$ has been included with the term $\widetilde{A}_2$, and (iii) the influence of $\mathbf{u}_2$ has been modified with the term $\widetilde{B}_{22}$. Notice that the size of the vector $\mathbf{z}$ has not changed, and we have accomplished the goal of keeping player one's dynamic model in the cooperative game the same size as his dynamic model in the noncooperative game.

Regardless of the implementation choice, the cooperative optimal control problem is no more complex than the noncooperative game considered previously. The extra information required by player one in the cooperative game is $x_2(0)$. Player one requires $\mathbf{u}_2$ in both the cooperative and noncooperative games. Only in decentralized control does player one not require player two's input sequence $\mathbf{u}_2$. The other extra required information, $A_2, B_{21}, Q_2, R_2, P_{2f}$, are fixed parameters and making their values available to player one is a minor communication overhead.

Proceeding as before, we solve this equation for $\mathbf{z}^0$ and pick out the rows corresponding to the elements of $\mathbf{u}_1^0$ giving

$$\mathbf{u}_1^0(x(0), \mathbf{u}_2) = \begin{bmatrix} K_{11} & K_{12} \end{bmatrix} \begin{bmatrix} x_1(0) \\ x_2(0) \end{bmatrix} + L_1\mathbf{u}_2$$

Combining the optimal control laws for each player gives

$$\begin{bmatrix} \mathbf{u}_1^0 \\ \mathbf{u}_2^0 \end{bmatrix} = \begin{bmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{bmatrix} \begin{bmatrix} x_1(0) \\ x_2(0) \end{bmatrix} + \begin{bmatrix} 0 & L_1 \\ L_2 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \end{bmatrix}^p$$

in which the gain matrix multiplying the state is a full matrix for the cooperative game. Substituting the optimal control into the iteration gives

$$\begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \end{bmatrix}^{p+1} = \underbrace{\begin{bmatrix} w_1K_{11} & w_1K_{12} \\ w_2K_{21} & w_2K_{22} \end{bmatrix}}_{\overline{K}} \begin{bmatrix} x_1(0) \\ x_2(0) \end{bmatrix} + \underbrace{\begin{bmatrix} (1-w_1)I & w_1L_1 \\ w_2L_2 & (1-w_2)I \end{bmatrix}}_{L} \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \end{bmatrix}^p$$

Finally writing this equation in the plantwide notation, we express the iteration as

$$\mathbf{u}^{p+1} = \overline{K}x(0) + L\mathbf{u}^p$$

Exponential stability of the closed-loop system. In the case of cooperative control, we consider the closed-loop system with a finite number of iterations, p. With finite iterations, distributed MPC becomes a form of *suboptimal* MPC as discussed in Sections 6.1.2 and 2.7. To analyze the behavior of the cooperative controller with a finite number of iterations, we require the cost decrease achieved by a single iteration, which we derive next. First we write the complete system evolution as in (6.10)

$$x^+ = Ax + Bu$$

in which A and B are defined in (6.11). We can then use (6.3) to express the overall cost function

$$V(x(0),\mathbf{u}) = (1/2)x'(0)(Q + \mathcal{A}'\mathcal{Q}\mathcal{A})x(0) + \mathbf{u}'(\mathcal{B}'\mathcal{Q}\mathcal{A})x(0) + (1/2)\mathbf{u}'H_{\mathbf{u}}\mathbf{u}$$

in which $\mathcal{A}$ and $\mathcal{B}$ are given in (6.1), the cost penalties $\mathcal{Q}$ and $\mathcal{R}$ are given in (6.2) and (6.9), and

$$H_{\mathbf{u}} = \mathcal{B}'\mathcal{Q}\mathcal{B} + \mathcal{R}$$

The overall cost is a positive definite quadratic function in $\mathbf{u}$ because R_1 and R_2 are positive definite, and therefore so are $\mathcal{R}_1$, $\mathcal{R}_2$, and $\mathcal{R}$.

The iteration in the two players' moves satisfies

$$\begin{aligned}(\mathbf{u}_1^{p+1},\mathbf{u}_2^{p+1}) &= \left((w_1\mathbf{u}_1^0 + (1-w_1)\mathbf{u}_1^p), (w_2\mathbf{u}_2^0 + (1-w_2)\mathbf{u}_2^p)\right)\\ &= (w_1\mathbf{u}_1^0, (1-w_2)\mathbf{u}_2^p) + ((1-w_1)\mathbf{u}_1^p, w_2\mathbf{u}_2^0)\\ (\mathbf{u}_1^{p+1},\mathbf{u}_2^{p+1}) &= w_1(\mathbf{u}_1^0,\mathbf{u}_2^p) + w_2(\mathbf{u}_1^p,\mathbf{u}_2^0) \end{aligned} \tag{6.14}$$

Exercise 6.18 analyzes the cost decrease for a convex step with a positive definite quadratic function and shows

$$V(x(0),\mathbf{u}_1^{p+1},\mathbf{u}_2^{p+1}) = V(x(0),\mathbf{u}_1^p,\mathbf{u}_2^p) - \frac{1}{2}\left[\mathbf{u}^p - \mathbf{u}^0(x(0))\right]' P\left[\mathbf{u}^p - \mathbf{u}^0(x(0))\right] \tag{6.15}$$

in which $P > 0$ is given by

$$P = H_{\mathbf{u}}D^{-1}\widetilde{H}D^{-1}H_{\mathbf{u}} \qquad \widetilde{H} = D - N$$

$$D = \begin{bmatrix} w_1^{-1}H_{\mathbf{u},11} & 0 \\ 0 & w_2^{-1}H_{\mathbf{u},22}\end{bmatrix} \qquad N = \begin{bmatrix} -w_1^{-1}w_2H_{\mathbf{u},11} & H_{\mathbf{u},12} \\ H_{\mathbf{u},21} & -w_1w_2^{-1}H_{\mathbf{u},22}\end{bmatrix}$$

and $H_{\mathbf{u}}$ is partitioned for the two players' input sequences. Notice that the cost decrease achieved in a single iteration is quadratic in the distance from the optimum. An important conclusion is that *each iteration in the cooperative game reduces the systemwide cost.* This cost reduction is the key property that gives cooperative MPC its excellent convergence properties, as we show next.

The two players' warm starts at the next sample are given by

$$\begin{aligned}\tilde{\mathbf{u}}_1^+ &= (u_1(1), u_1(2), \ldots, u_1(N-1), 0)\\ \tilde{\mathbf{u}}_2^+ &= (u_2(1), u_2(2), \ldots, u_2(N-1), 0)\end{aligned}$$

We define the following linear time-invariant functions g_1^p and g_2^p as the outcome of applying the control iteration procedure p times

$$\begin{aligned}\mathbf{u}_1^p &= g_1^p(x_1, x_2, \mathbf{u}_1, \mathbf{u}_2)\\ \mathbf{u}_2^p &= g_2^p(x_1, x_2, \mathbf{u}_1, \mathbf{u}_2)\end{aligned}$$

in which $p \geq 0$ is an integer, x_1 and x_2 are the states, and $\mathbf{u}_1, \mathbf{u}_2$ are the input sequences from the previous sample, used to generate the warm start for the iteration. Here we consider p to be constant with time, but Exercise 6.20 considers the case in which the controller iterations may vary with sample time. The system evolution is then given by

$$\begin{aligned} x_1^+ &= A_1 x_1 + \overline{B}_{11} u_1 + \overline{B}_{12} u_2 & x_2^+ &= A_2 x_2 + \overline{B}_{21} u_1 + \overline{B}_{22} u_2\\ \mathbf{u}_1^+ &= g_1^p(x_1, x_2, \mathbf{u}_1, \mathbf{u}_2) & \mathbf{u}_2^+ &= g_2^p(x_1, x_2, \mathbf{u}_1, \mathbf{u}_2) \end{aligned} \tag{6.16}$$

By the construction of the warm start, $\tilde{\mathbf{u}}_1^+, \tilde{\mathbf{u}}_2^+$, we have

$$\begin{aligned} V(x_1^+, x_2^+, \tilde{\mathbf{u}}_1^+, \tilde{\mathbf{u}}_2^+) = V(x_1, x_2, \mathbf{u}_1, \mathbf{u}_2) &- \rho_1 \ell_1(x_1, u_1) - \rho_2 \ell_2(x_2, u_2)\\ &+ (1/2)\rho_1 x_1(N)' \left[A_1' P_{1f} A_1 - P_{1f} + Q_1\right] x_1(N)\\ &+ (1/2)\rho_2 x_2(N)' \left[A_2' P_{2f} A_2 - P_{2f} + Q_2\right] x_2(N)\end{aligned}$$

From our choice of terminal penalty satisfying (6.8), the last two terms are zero giving

$$\begin{aligned} V(x_1^+, x_2^+, \tilde{\mathbf{u}}_1^+, \tilde{\mathbf{u}}_2^+) = V(x_1, x_2, \mathbf{u}_1, \mathbf{u}_2)&\\ - \rho_1 \ell_1(x_1, u_1) - \rho_2 \ell_2(x_2, u_2)&\end{aligned} \tag{6.17}$$

No optimization, $p = 0$. If we do no further optimization, then we have $\mathbf{u}_1^+ = \tilde{\mathbf{u}}_1^+$, $\mathbf{u}_2^+ = \tilde{\mathbf{u}}_2^+$, and the equality

$$V(x_1^+, x_2^+, \mathbf{u}_1^+, \mathbf{u}_2^+) = V(x_1, x_2, \mathbf{u}_1, \mathbf{u}_2) - \rho_1 \ell_1(x_1, u_1) - \rho_2 \ell_2(x_2, u_2)$$

The input sequences add a zero at each sample until $\mathbf{u}_1 = \mathbf{u}_2 = 0$ at time $k = N$. The system decays exponentially under zero control and the closed loop is exponentially stable.

Further optimization, $p \geq 1$. We next consider the case in which optimization is performed. Equation 6.15 then gives

$$V(x_1^+, x_2^+, \mathbf{u}_1^+, \mathbf{u}_2^+) \leq V(x_1^+, x_2^+, \tilde{\mathbf{u}}_1^+, \tilde{\mathbf{u}}_2^+) - \left[\tilde{\mathbf{u}}^+ - \mathbf{u}^0(x^+)\right]' P \left[\tilde{\mathbf{u}}^+ - \mathbf{u}^0(x^+)\right] \qquad p \geq 1$$

with equality holding for $p = 1$. Using this result in (6.17) gives

$$V(x_1^+, x_2^+, \mathbf{u}_1^+, \mathbf{u}_2^+) \leq V(x_1, x_2, \mathbf{u}_1, \mathbf{u}_2) - \rho_1 \ell_1(x_1, u_1) - \rho_2 \ell_2(x_2, u_2) - \left[\tilde{\mathbf{u}}^+ - \mathbf{u}^0(x^+)\right]' P \left[\tilde{\mathbf{u}}^+ - \mathbf{u}^0(x^+)\right]$$

Since V is bounded below by zero and ℓ_1 and ℓ_2 are positive functions, we conclude the time sequence $V(x_1(k), x_2(k), \mathbf{u}_1(k), \mathbf{u}_2(k))$ converges. and therefore $x_1(k)$, $x_2(k)$, $u_1(k)$, and $u_2(k)$ converge to zero. Moreover, since $P > 0$, the last term implies that $\tilde{\mathbf{u}}^+$ converges to $\mathbf{u}^0(x^+)$, which converges to zero because x^+ converges to zero. Therefore, the entire input sequence $\mathbf{u}$ converges to zero. Because the total system evolution is a linear time-invariant system, the convergence is exponential. Even though we are considering here a form of *suboptimal* MPC, we do not require an additional inequality constraint on $\mathbf{u}$ because the problem considered here is *unconstrained* and the iterations satisfy (6.15).

6.2.5 Tracking Nonzero Setpoints

For tracking nonzero setpoints, we compute steady-state targets as discussed in Section 1.5. The steady-state input-output model is given by

$$y_s = G u_s \qquad G = C(I - A)^{-1} B$$

in which G is the steady-state gain of the system. The two subsystems are denoted

$$\begin{bmatrix} y_{1s} \\ y_{2s} \end{bmatrix} = \begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix} \begin{bmatrix} u_{1s} \\ u_{2s} \end{bmatrix}$$

For simplicity, we assume that the targets are chosen to be the measurements ($H = I$). Further, we assume that both local systems are square, and that the local targets can be reached exactly with the local

inputs. This assumption means that G_{11} and G_{22} are square matrices of full rank. We remove all of these assumptions when we treat the constrained two-player game in the next section. If there is model error, integrating disturbance models are required as discussed in Chapter 1. We discuss these later.

The target problem also can be solved with any of the four approaches discussed so far. We consider each.

Centralized case. The centralized problem gives in one shot both inputs required to meet both output setpoints

$$\begin{aligned} u_s &= G^{-1} y_{\text{sp}} \\ y_s &= y_{\text{sp}} \end{aligned}$$

Decentralized case. The decentralized problem considers only the diagonal terms and computes the following steady inputs

$$u_s = \begin{bmatrix} G_{11}^{-1} & \\ & G_{22}^{-1} \end{bmatrix} y_{\text{sp}}$$

Notice these inputs produce offset in both output setpoints

$$y_s = \begin{bmatrix} I & G_{12}G_{22}^{-1} \\ G_{21}G_{11}^{-1} & I \end{bmatrix} y_{\text{sp}}$$

Noncooperative case. In the noncooperative game, each player attempts to remove offset in only its outputs. Player one solves the following problem

$$\begin{aligned} \min_{u_1} &(y_1 - y_{1sp})' \overline{Q}_1 (y_1 - y_{1sp}) \\ &\text{s.t. } y_1 = G_{11}u_1 + G_{12}u_2 \end{aligned}$$

Because the target can be reached exactly, the optimal solution is to find u_1 such that $y_1 = y_{1sp}$, which gives

$$u_{1s}^0 = G_{11}^{-1}\left(y_{1sp} - G_{12}u_2^p\right)$$

Player two solves the analogous problem. If we iterate on the two players' solutions, we obtain

$$\begin{bmatrix} u_{1s} \\ u_{2s} \end{bmatrix}^{p+1} = \underbrace{\begin{bmatrix} w_1 G_{11}^{-1} & \\ & w_2 G_{22}^{-1} \end{bmatrix}}_{\overline{K}_s} \begin{bmatrix} y_{1sp} \\ y_{2sp} \end{bmatrix} + \underbrace{\begin{bmatrix} w_2 I & -w_1 G_{11}^{-1} G_{12} \\ -w_2 G_{22}^{-1} G_{21} & w_1 I \end{bmatrix}}_{L_s} \begin{bmatrix} u_{1s} \\ u_{2s} \end{bmatrix}^{p}$$

This iteration can be summarized by

$$u_s^{p+1} = \overline{K}_s y_{\text{sp}} + L_s u_s^p$$

If L_s is stable, this iteration converges to

$$u_s^{\infty} = (I - L_s)^{-1} \overline{K}_s y_{\text{sp}}$$
$$u_s^{\infty} = G^{-1} y_{\text{sp}}$$

and we have no offset. We already have seen that we cannot expect the dynamic noncooperative iteration to converge. The next several examples explore the issue of whether we can expect at least the steady-state iteration to be stable.

Cooperative case. In the cooperative case, both players work on minimizing the offset in both outputs. Player one solves

$$\min_{u_1} (1/2) \begin{bmatrix} y_1 - y_{1sp} \\ y_2 - y_{2sp} \end{bmatrix}' \begin{bmatrix} \rho_1 \overline{Q}_1 & \\ & \rho_2 \overline{Q}_2 \end{bmatrix} \begin{bmatrix} y_1 - y_{1sp} \\ y_2 - y_{2sp} \end{bmatrix}$$
$$\text{s.t.} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} G_{11} \\ G_{21} \end{bmatrix} u_1 + \begin{bmatrix} G_{12} \\ G_{22} \end{bmatrix} u_2$$

We can write this in the general form

$$\min_{r_s} (1/2) r_s' H r_s + h' r_s$$
$$\text{s.t. } D r_s = d$$

in which

$$r_s = \begin{bmatrix} y_{1s} \\ y_{2s} \\ u_{1s} \end{bmatrix} \qquad H = \begin{bmatrix} \rho_1 \overline{Q}_1 & & \\ & \rho_2 \overline{Q}_2 & \\ & & 0 \end{bmatrix} \qquad h = \begin{bmatrix} -Q y_{\text{sp}} \\ 0 \end{bmatrix}$$

$$D = \begin{bmatrix} I & -G_1 \end{bmatrix} \qquad d = G_2 u_2 \qquad G_1 = \begin{bmatrix} G_{11} \\ G_{12} \end{bmatrix} \qquad G_2 = \begin{bmatrix} G_{12} \\ G_{22} \end{bmatrix}$$

We can then solve the linear algebra problem

$$\begin{bmatrix} H & -D' \\ -D & 0 \end{bmatrix} \begin{bmatrix} r_s \\ \lambda_s \end{bmatrix} = - \begin{bmatrix} h \\ d \end{bmatrix}$$

and identify the linear gains between the optimal u_{1s} and the setpoint y_{sp} and player two's input u_{2s}

$$u_{1s}^0 = K_{1s} y_{\text{sp}} + L_{1s} u_{2s}^p$$

Combining the optimal control laws for each player gives

$$\begin{bmatrix} u_{1s}^0 \\ u_{2s}^0 \end{bmatrix} = \begin{bmatrix} K_{1s} \\ K_{2s} \end{bmatrix} y_{\text{sp}} + \begin{bmatrix} 0 & L_{1s} \\ L_{2s} & 0 \end{bmatrix} \begin{bmatrix} u_{1s} \\ u_{2s} \end{bmatrix}^p$$

Substituting the optimal control into the iteration gives

$$\begin{bmatrix} u_{1s} \\ u_{2s} \end{bmatrix}^{p+1} = \underbrace{\begin{bmatrix} w_1 K_{1s} \\ w_2 K_{2s} \end{bmatrix}}_{\overline{K}_s} y_{\text{sp}} + \underbrace{\begin{bmatrix} (1-w_1)I & w_1 L_{1s} \\ w_2 L_{2s} & (1-w_2)I \end{bmatrix}}_{L_s} \begin{bmatrix} u_{1s} \\ u_{2s} \end{bmatrix}^p$$

Finally writing this equation in the plantwide notation, we express the iteration as

$$u_s^{p+1} = \overline{K}_s y_{\text{sp}} + L_s u_s^p$$

As we did with the cooperative regulation problem, we can analyze the optimization problem to show that this iteration is always stable and converges to the centralized target. Next we explore the use of these approaches in some illustrative examples.

Example 6.12: Stability and offset in the distributed target calculation

Consider the following two-input, two-output system with steady-state gain matrix and setpoint

$$\begin{bmatrix} y_{1s} \\ y_{2s} \end{bmatrix} = \begin{bmatrix} -0.5 & 1.0 \\ 2.0 & 1.0 \end{bmatrix} \begin{bmatrix} u_{1s} \\ u_{2s} \end{bmatrix} \qquad \begin{bmatrix} y_{1sp} \\ y_{2sp} \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

(a) Show the first 10 iterations of the noncooperative and cooperative steady-state cases starting with the decentralized solution as the initial guess.

Describe the differences. Compute the eigenvalues of L for the cooperative and noncooperative cases. Discuss the relationship

between these eigenvalues and the result of the iteration calculations.

Mark also the solution to the centralized and decentralized cases on your plots.

(b) Switch the pairings and repeat the previous part. Explain your results.

Solution

(a) The first 10 iterations of the noncooperative steady-state calculation are shown in Figure 6.2. Notice the iteration is unstable and the steady-state target does not converge. The cooperative case is shown in Figure 6.3. This case is stable and the iterations converge to the centralized target and achieve zero offset. The magnitudes of the eigenvalues of L_s for the noncooperative (nc) and cooperative (co) cases are given by

$$|\text{eig}(L_{snc})| = \{1.12, 1.12\} \qquad |\text{eig}(L_{sco})| = \{0.757, 0.243\}$$

Stability of the iteration is determined by the magnitudes of the eigenvalues of L_s.

(b) Reversing the pairings leads to the following gain matrix in which we have reversed the labels of the outputs for the two systems

$$\begin{bmatrix} y_{1s} \\ y_{2s} \end{bmatrix} = \begin{bmatrix} 2.0 & 1.0 \\ -0.5 & 1.0 \end{bmatrix} \begin{bmatrix} u_{1s} \\ u_{2s} \end{bmatrix}$$

The first 10 iterations of the noncooperative and cooperative controllers are shown in Figures 6.4 and 6.5. For this pairing, the noncooperative case also converges to the centralized target. The eigenvalues are given by

$$|\text{eig}(L_{snc})| = \{0.559, 0.559\} \qquad |\text{eig}(L_{sco})| = \{0.757, 0.243\}$$

The eigenvalues of the cooperative case are unaffected by the reversal of pairings. □

Given the stability analysis of the simple unconstrained two-player game, we remove from further consideration two options we have been discussing to this point: noncooperative control and decentralized control. We next further develop the theory of cooperative MPC and compare its performance to centralized MPC in more general and challenging situations.

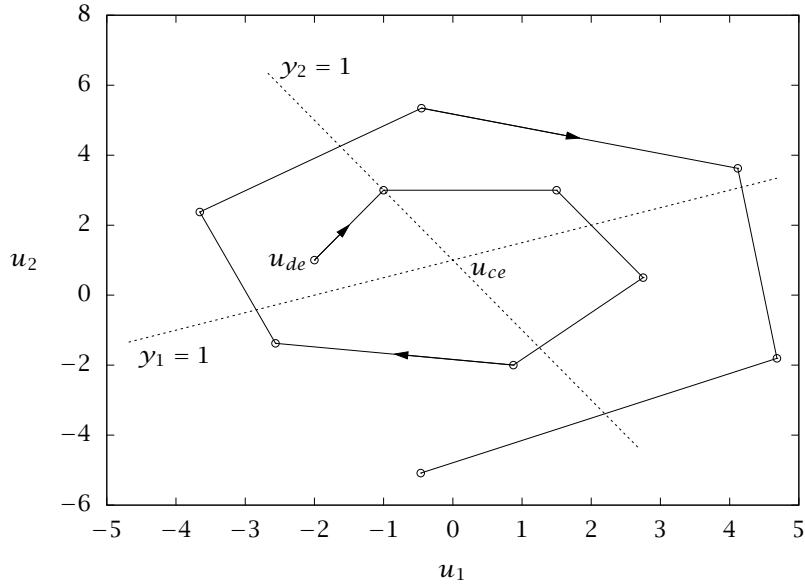


Figure 6.2: Ten iterations of noncooperative steady-state calculation, $u^{[0]} = u_{de}$; iterations are unstable, $u^p \to \infty$.

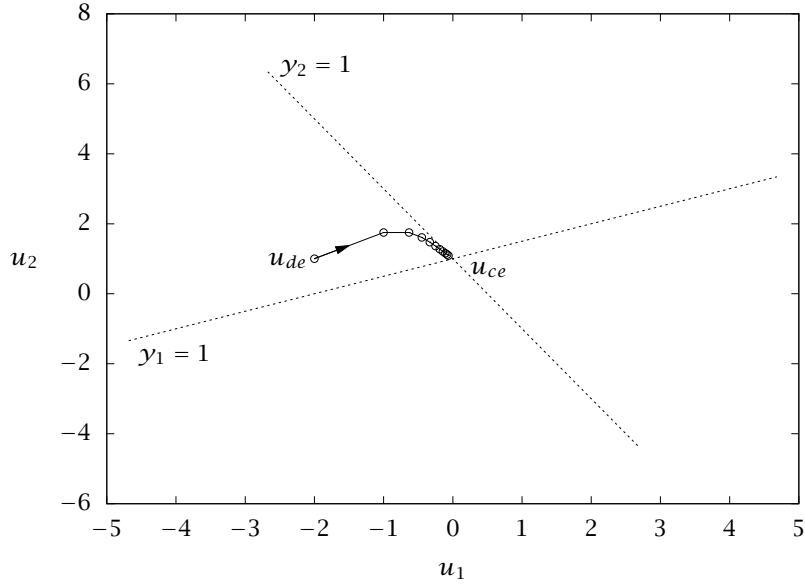


Figure 6.3: Ten iterations of cooperative steady-state calculation, $u^{[0]} = u_{de}$; iterations are stable, $u^p \to u_{ce}$.

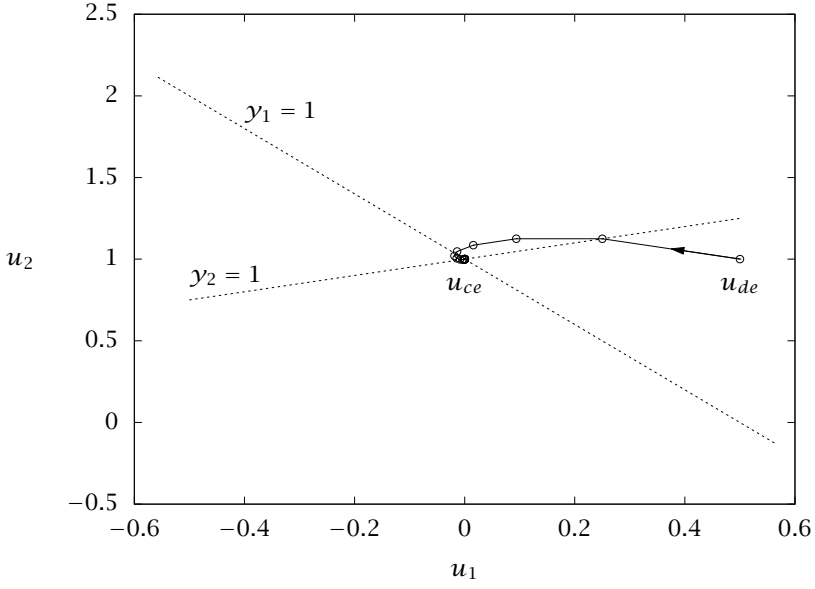


Figure 6.4: Ten iterations of noncooperative steady-state calculation, $u^{[0]} = u_{de}$; iterations are stable with reversed pairing.

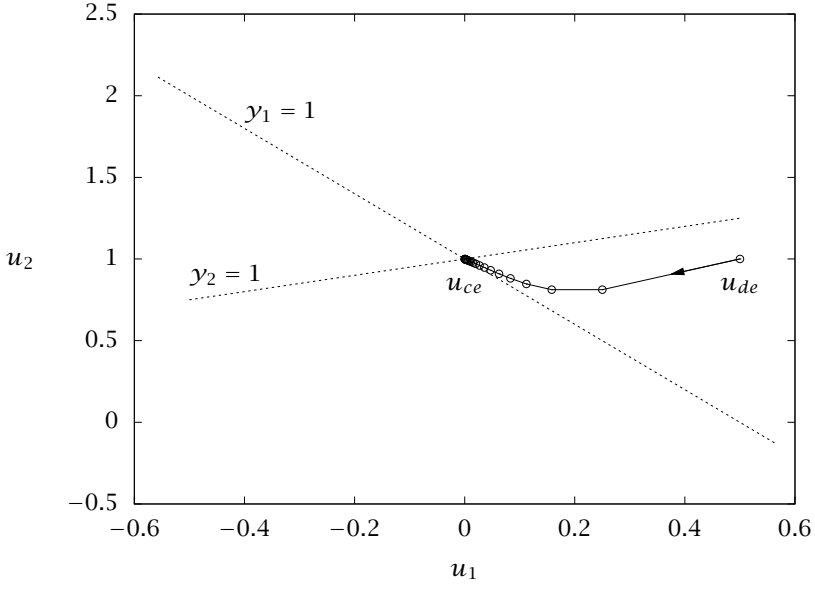


Figure 6.5: Ten iterations of cooperative steady-state calculation, $u^{[0]} = u_{de}$; iterations are stable with reversed pairing.

6.2.6 State Estimation

Given output measurements, we can express the state estimation problem also in distributed form. Player one uses local measurements of y_1 and knowledge of both inputs u_1 and u_2 to estimate state x_1

$$\hat{x}_1^+ = A_1\hat{x}_1 + \overline{B}_{11}u_1 + \overline{B}_{12}u_2 + L_1(y_1 - C_1\hat{x}_1)$$

Defining estimate error to be $e_1 = x_1 - \hat{x}_1$ gives

$$e_1^+ = (A_1 - L_1C_1)e_1$$

Because all the subsystems are stable, we know L_1 exists so that $A_1 - L_1C_1$ is stable and player one's local estimator is stable. The estimate error for the two subsystems is then given by

$$\begin{bmatrix} e_1 \\ e_2 \end{bmatrix}^+ = \begin{bmatrix} A_{L1} & \\ & A_{L2} \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \end{bmatrix} \tag{6.18}$$

in which $A_{Li} = A_i - L_iC_i$.

Closed-loop stability. The dynamics of the estimator are given by

$$\begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix}^+ = \begin{bmatrix} A_1 & \\ & A_2 \end{bmatrix} \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix} + \begin{bmatrix} \overline{B}_{11} & \overline{B}_{12} \\ \overline{B}_{21} & \overline{B}_{22} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} + \begin{bmatrix} L_1C_1 & \\ & L_2C_2 \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \end{bmatrix}$$

In the control law we use the state estimate in place of the state, which is unmeasured and unknown. We consider two cases.

Converged controller. In this case the distributed control law converges to the centralized controller, and we have

$$\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{bmatrix} \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix}$$

The closed-loop system evolves according to

$$\begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix}^+ = \left\{ \begin{bmatrix} A_1 & \\ & A_2 \end{bmatrix} + \begin{bmatrix} \overline{B}_{11} & \overline{B}_{12} \\ \overline{B}_{21} & \overline{B}_{22} \end{bmatrix} \begin{bmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{bmatrix} \right\} \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix} + \begin{bmatrix} L_1C_1 & \\ & L_2C_2 \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \end{bmatrix}$$

The $A+BK$ term is stable because this term is the same as in the stabilizing centralized controller. The perturbation is exponentially decaying because the distributed estimators are stable. Therefore $\hat{x}$ goes to zero exponentially, which, along with e going to zero exponentially, implies x goes to zero exponentially.

Finite iterations. Here we use the state plus input sequence description given in (6.16), which, as we have already noted, is a linear time-invariant system. With estimate error, the system equation is

$$\begin{bmatrix} \hat{x}_1^+ \\ \hat{x}_2^+ \\ \mathbf{u}_1^+ \\ \mathbf{u}_2^+ \end{bmatrix} = \begin{bmatrix} A_1\hat{x}_1 + \overline{B}_{11}u_1 + \overline{B}_{12}u_2 \\ A_2\hat{x}_2 + \overline{B}_{21}u_1 + \overline{B}_{22}u_2 \\ g_1^p(\hat{x}_1, \hat{x}_2, \mathbf{u}_1, \mathbf{u}_2) \\ g_2^p(\hat{x}_1, \hat{x}_2, \mathbf{u}_1, \mathbf{u}_2) \end{bmatrix} + \begin{bmatrix} L_1C_1e_1 \\ L_2C_2e_2 \\ 0 \\ 0 \end{bmatrix}$$

Because there is again only one-way coupling between the estimate error evolution, (6.18), and the system evolution given above, the composite system is exponentially stable.

6.3 Constrained Two-Player Game

Now that we have introduced most of the notation and the fundamental ideas, we consider more general cases. Because we are interested in establishing stability properties of the controlled systems, we focus exclusively on *cooperative distributed MPC* from this point forward. In this section we consider convex input constraints on the two players. We assume output constraints have been softened with exact soft constraints and added to the objective function, so do not consider output constraints explicitly. The input constraints break into two significant categories: coupled and uncoupled constraints. We treat each of these in turn.

We also allow unstable systems and replace Assumption 6.8 with the following more general restrictions on the systems and controller parameters.

Assumption 6.13 (Constrained two-player game)**.**

(a) The systems $(\underline{A}_i, \underline{B}_i)$, $i = 1, 2$ are stabilizable, in which $\underline{A}_i = \text{diag}(A_{1i}, A_{2i})$ and $\underline{B}_i = \left[\begin{smallmatrix} B_{1i} \\ B_{2i} \end{smallmatrix}\right]$.

(b) The systems (A_i, C_i), $i = 1, 2$ are detectable.

(c) The input penalties R_1, R_2 are positive definite, and the state penalties Q_1, Q_2 are semidefinite.

(d) The systems (A_1, Q_1) and (A_2, Q_2) are detectable.

(e) The horizon is chosen sufficiently long to zero the unstable modes, $N \geq \max_{i \in \mathbb{I}_{1:2}} \underline{n}_i^u$, in which $\underline{n}_i^u$ is the number of unstable modes of $\underline{A}_i$, i.e., number of $\lambda \in \text{eig}(\underline{A}_i)$ such that $|\lambda| \geq 1$.

Assumption (b) implies that we have L_i such that $(A_i - L_i C_i), i = 1, 2$ is stable. Note that the stabilizable and detectable conditions of Assumption 6.13 are automatically satisfied if we obtain the state space models from a minimal realization of the input/output models for (u_i, y_j), $i, j = 1, 2$.

Unstable modes. To handle unstable systems, we add constraints to zero the unstable modes at the end of the horizon. To set up this constraint, consider the real Schur decomposition of A_{ij} for $i, j \in \mathbb{I}_{1:2}$

$$A_{ij} = \begin{bmatrix} S_{ij}^s & S_{ij}^u \end{bmatrix} \begin{bmatrix} A_{ij}^s & - \\ & A_{ij}^u \end{bmatrix} \begin{bmatrix} S_{ij}^{s\prime} \\ S_{ij}^{u\prime} \end{bmatrix} \tag{6.19}$$

in which A_{ij}^s is upper triangular and stable, and A_{ij}^u is upper triangular with all unstable eigenvalues.[3] Given the Schur decomposition (6.19), we define the matrices

$$\begin{aligned} S_i^s &= \text{diag}(S_{i1}^s, S_{i2}^s) & A_i^s &= \text{diag}(A_{i1}^s, A_{i2}^s) \quad i \in \mathbb{I}_{1:2} \\ S_i^u &= \text{diag}(S_{i1}^u, S_{i2}^u) & A_i^u &= \text{diag}(A_{i1}^u, A_{i2}^u) \quad i \in \mathbb{I}_{1:2} \end{aligned}$$

These matrices satisfy the Schur decompositions

$$A_i = \begin{bmatrix} S_i^s & S_i^u \end{bmatrix} \begin{bmatrix} A_i^s & - \\ & A_i^u \end{bmatrix} \begin{bmatrix} S_i^{s\prime} \\ S_i^{u\prime} \end{bmatrix} \quad i \in \mathbb{I}_{1:2}$$

We further define the matrices Σ_1, Σ_2 as the solutions to the Lyapunov equations

$$A_1^{s\prime} \Sigma_1 A_1^s - \Sigma_1 = -S_1^{s\prime} Q_1 S_1^s \qquad A_2^{s\prime} \Sigma_2 A_2^s - \Sigma_2 = -S_2^{s\prime} Q_2 S_2^s \tag{6.20}$$

We then choose the terminal penalty for each subsystem to be the cost to go under zero control

$$P_{1f} = S_1^s \Sigma_1 S_1^{s\prime} \qquad P_{2f} = S_2^s \Sigma_2 S_2^{s\prime}$$

[3] If A_{ij} is stable, then there is no A_{ij}^u and S_{ij}^u.

6.3.1 Uncoupled Input Constraints

We consider convex input constraints of the following form

$$Hu(k) \le h \qquad k = 0, 1, \ldots, N$$

Defining convex set $\mathbb{U}$

$$\mathbb{U} = \{u | Hu \le h\}$$

we express the input constraints as

$$u(k) \in \mathbb{U} \qquad k = 0, 1, \ldots, N$$

We drop the time index and indicate the constraints are applied over the entire input sequence using the notation $\mathbf{u} \in \mathbb{U}$. In the uncoupled constraint case, the two players' inputs must satisfy

$$\mathbf{u}_1 \in \mathbb{U}_1 \qquad \mathbf{u}_2 \in \mathbb{U}_2$$

in which $\mathbb{U}_1$ and $\mathbb{U}_2$ are convex subsets of $\mathbb{R}^{m_1}$ and $\mathbb{R}^{m_2}$, respectively. The constraints are termed *uncoupled* because there is no interaction or coupling of the inputs in the constraint relation. Player one then solves the following constrained optimization

$$\begin{aligned}
&\min_{\mathbf{u}_1} V(x_1(0), x_2(0), \mathbf{u}_1, \mathbf{u}_2) \\
\text{s.t. } &\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}^+ = \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} \overline{B}_{11} \\ \overline{B}_{21} \end{bmatrix} u_1 + \begin{bmatrix} \overline{B}_{12} \\ \overline{B}_{22} \end{bmatrix} u_2 \\
&\mathbf{u}_1 \in \mathbb{U}_1 \\
&S_{j1}^{u\,\prime} x_{j1}(N) = 0 \quad j \in \mathbb{I}_{1:2} \\
&|\mathbf{u}_1| \le d_1(|x_{11}(0)| + |x_{21}(0)|) \quad x_{11}(0), x_{21}(0) \in r\mathcal{B}
\end{aligned}$$

in which we include the system's hard input constraints, the stability constraint on the unstable modes, and the Lyapunov stability constraints. Exercise 6.22 discusses how to write the constraint $|\mathbf{u}_1| \le d_1 |x_1(0)|$ as a set of linear inequalities on $\mathbf{u}_1$. Similarly, player two solves

$$\begin{aligned}
&\min_{\mathbf{u}_2} V(x_1(0), x_2(0), \mathbf{u}_1, \mathbf{u}_2) \\
\text{s.t. } &\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}^+ = \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} \overline{B}_{11} \\ \overline{B}_{21} \end{bmatrix} u_1 + \begin{bmatrix} \overline{B}_{12} \\ \overline{B}_{22} \end{bmatrix} u_2 \\
&\mathbf{u}_2 \in \mathbb{U}_2 \\
&S_{j2}^{u\,\prime} x_{j2}(N) = 0 \quad j \in \mathbb{I}_{1:2} \\
&|\mathbf{u}_2| \le d_2(|x_{21}(0)| + |x_{22}(0)|) \quad x_{12}(0), x_{22}(0) \in r\mathcal{B}
\end{aligned}$$

We denote the solutions to these problems as

$$\mathbf{u}_1^0(x_1(0), x_2(0), \mathbf{u}_2) \qquad \mathbf{u}_2^0(x_1(0), x_2(0), \mathbf{u}_1)$$

The feasible set $\mathcal{X}_N$ for the unstable system is the set of states for which the unstable modes can be brought to zero in N moves while satisfying the input constraints.

Given an initial iterate, $(\mathbf{u}_1^p, \mathbf{u}_2^p)$, the next iterate is defined to be

$$(\mathbf{u}_1, \mathbf{u}_2)^{p+1} = w_1(\mathbf{u}_1^0(x_1(0), x_2(0), \mathbf{u}_2^p), \mathbf{u}_2^p) + w_2(\mathbf{u}_1^p, \mathbf{u}_2^0(x_1(0), x_2(0), \mathbf{u}_1^p))$$

To reduce the notational burden we denote this as

$$(\mathbf{u}_1, \mathbf{u}_2)^{p+1} = w_1(\mathbf{u}_1^0, \mathbf{u}_2^p) + w_2(\mathbf{u}_1^p, \mathbf{u}_2^0)$$

and the functional dependencies of $\mathbf{u}_1^0$ and $\mathbf{u}_2^0$ should be kept in mind.

This procedure provides three important properties, which we establish next.

1. The iterates are feasible: $(\mathbf{u}_1, \mathbf{u}_2)^p \in (\mathbb{U}_1, \mathbb{U}_2)$ implies $(\mathbf{u}_1, \mathbf{u}_2)^{p+1} \in (\mathbb{U}_1, \mathbb{U}_2)$. This follows from convexity of $\mathbb{U}_1$, $\mathbb{U}_2$ and the convex combination of the feasible points $(\mathbf{u}_1^p, \mathbf{u}_2^p)$ and $(\mathbf{u}_1^0, \mathbf{u}_2^0)$ to make $(\mathbf{u}_1, \mathbf{u}_2)^{p+1}$.

2. The cost decreases on iteration: $V(x_1(0), x_2(0), (\mathbf{u}_1, \mathbf{u}_2)^{p+1}) \leq V(x_1(0), x_2(0), (\mathbf{u}_1, \mathbf{u}_2)^p)$ for all $x_1(0)$, $x_2(0)$, and for all feasible $(\mathbf{u}_1, \mathbf{u}_2)^p \in (\mathbb{U}_1, \mathbb{U}_2)$. The systemwide cost satisfies the following inequalities

$$\begin{aligned} V(x(0), \mathbf{u}_1^{p+1}, \mathbf{u}_2^{p+1}) &= V\left(x(0), \left(w_1(\mathbf{u}_1^0, \mathbf{u}_2^p) + w_2(\mathbf{u}_1^p, \mathbf{u}_2^0)\right)\right) \\ &\leq w_1 V(x(0), (\mathbf{u}_1^0, \mathbf{u}_2^p)) + w_2 V(x(0), (\mathbf{u}_1^p, \mathbf{u}_2^0)) \\ &\leq w_1 V(x(0), (\mathbf{u}_1^p, \mathbf{u}_2^p)) + w_2 V(x(0), (\mathbf{u}_1^p, \mathbf{u}_2^p)) \\ &= V(x(0), \mathbf{u}_1^p, \mathbf{u}_2^p) \end{aligned}$$

   The first equality follows from (6.14). The next inequality follows from convexity of V. The next follows from optimality of $\mathbf{u}_1^0$ and $\mathbf{u}_2^0$, and the last follows from $w_1 + w_2 = 1$. Because the cost is bounded below, the cost iteration converges.

3. The converged solution of the cooperative problem is equal to the optimal solution of the centralized problem. Establishing this property is discussed in Exercise 6.26.

Exponential stability of the closed-loop system. We next consider the closed-loop system. The two players' warm starts at the next sample are as defined previously

$$\widetilde{\mathbf{u}}_1^+ = (u_1(1), u_1(2), \ldots, u_1(N-1), 0)$$
$$\widetilde{\mathbf{u}}_2^+ = (u_2(1), u_2(2), \ldots, u_2(N-1), 0)$$

We define again the functions g_1^p, g_2^p as the outcome of applying the control iteration procedure p times

$$\mathbf{u}_1^p = g_1^p(x_1, x_2, \mathbf{u}_1, \mathbf{u}_2)$$
$$\mathbf{u}_2^p = g_2^p(x_1, x_2, \mathbf{u}_1, \mathbf{u}_2)$$

The important difference between the previous unconstrained and this constrained case is that the functions g_1^p, g_2^p are nonlinear due to the input constraints. The system evolution is then given by

$$x_1^+ = A_1 x_1 + \overline{B}_{11} u_1 + \overline{B}_{12} u_2 \qquad x_2^+ = A_2 x_2 + \overline{B}_{21} u_1 + \overline{B}_{22} u_2$$
$$\mathbf{u}_1^+ = g_1^p(x_1, x_2, \mathbf{u}_1, \mathbf{u}_2) \qquad \mathbf{u}_2^+ = g_2^p(x_1, x_2, \mathbf{u}_1, \mathbf{u}_2)$$

We have the following cost using the warm start at the next sample

$$\begin{aligned} V(x_1^+, x_2^+, \widetilde{\mathbf{u}}_1^+, \widetilde{\mathbf{u}}_2^+) = V(x_1, x_2, \mathbf{u}_1, \mathbf{u}_2) - \rho_1 \ell_1(x_1, u_1) - \rho_2 \ell_2(x_2, u_2) \\ +(1/2)\rho_1 x_1(N)' \left[A_1' P_{1f} A_1 - P_{1f} + Q_1 \right] x_1(N) \\ +(1/2)\rho_2 x_2(N)' \left[A_2' P_{2f} A_2 - P_{2f} + Q_2 \right] x_2(N) \end{aligned}$$

Using the Schur decomposition (6.19) and the constraints $S_{ji}^{u\prime} x_{ji}(N) = 0$ for $i, j \in \mathbb{I}_{1:2}$, the last two terms can be written as

$$\begin{aligned} (1/2)\rho_1 x_1(N)' S_1^s \left[A_1^{s\prime} \Sigma_1 A_1^s - \Sigma_1 + S_1^{s\prime} Q_1 S_1^s \right] S_1^{s\prime} x_1(N) \\ +(1/2)\rho_2 x_2(N)' S_2^s \left[A_2^{s\prime} \Sigma_2 A_2^s - \Sigma_2 + S_2^{s\prime} Q_2 S_2^s \right] S_2^{s\prime} x_2(N) \end{aligned}$$

These terms are zero because of (6.20). Using this result and applying the iteration for the controllers gives

$$V(x_1^+, x_2^+, \mathbf{u}_1^+, \mathbf{u}_2^+) \le V(x_1, x_2, \mathbf{u}_1, \mathbf{u}_2) - \rho_1 \ell_1(x_1, u_1) - \rho_2 \ell_2(x_2, u_2)$$

The Lyapunov stability constraints give (see also Exercise 6.28)

$$|(\mathbf{u}_1, \mathbf{u}_2)| \le 2 \max(d_1, d_2) \, |(x_1, x_2)| \qquad (x_1, x_2) \in r\mathcal{B}$$

Given the cost decrease and this constraint on the size of the input sequence, we satisfy the conditions of Lemma 6.5, and conclude the solution $x(k) = 0$ for all k is exponentially stable on all of $\mathcal{X}_N$ if either $\mathcal{X}_N$ is compact or $\mathbb{U}$ is compact.

6.3.2 Coupled Input Constraints

By contrast, in the coupled constraint case, the constraints are of the form

$$H_1\mathbf{u}_1 + H_2\mathbf{u}_2 \le h \quad \text{or} \quad (\mathbf{u}_1, \mathbf{u}_2) \in \mathbb{U} \tag{6.21}$$

These constraints represent the players sharing some common resource. An example would be different subsystems in a chemical plant drawing steam or some other utility from a single plantwide generation plant. The total utility used by the different subsystems to meet their control objectives is constrained by the generation capacity.

The players solve the same optimization problems as in the uncoupled constraint case, with the exception that both players' input constraints are given by (6.21). This modified game provides only two of the three properties established for the uncoupled constraint case. These are

1. The iterates are feasible: $(\mathbf{u}_1, \mathbf{u}_2)^p \in \mathbb{U}$ implies $(\mathbf{u}_1, \mathbf{u}_2)^{p+1} \in \mathbb{U}$. This follows from convexity of $\mathbb{U}$ and the convex combination of the feasible points $(\mathbf{u}_1^p, \mathbf{u}_2^p)$ and $(\mathbf{u}_1^0, \mathbf{u}_2^0)$ to make $(\mathbf{u}_1, \mathbf{u}_2)^{p+1}$.

2. The cost decreases on iteration: $V(x_1(0), x_2(0), (\mathbf{u}_1, \mathbf{u}_2)^{p+1}) \le V(x_1(0), x_2(0), (\mathbf{u}_1, \mathbf{u}_2)^p)$ for all $x_1(0)$, $x_2(0)$, and for all feasible $(\mathbf{u}_1, \mathbf{u}_2)^p \in \mathbb{U}$. The systemwide cost satisfies the same inequalities established for the uncoupled constraint case giving

$$V(x(0), \mathbf{u}_1^{p+1}, \mathbf{u}_2^{p+1}) \le V(x(0), \mathbf{u}_1^p, \mathbf{u}_2^p)$$

   Because the cost is bounded below, the cost iteration converges.

The converged solution of the cooperative problem is *not* equal to the optimal solution of the centralized problem, however. We have lost property 3 of the uncoupled case. To see how the convergence property is lost, consider Figure 6.6. Region $\mathbb{U}$ is indicated by the triangle and its interior. Consider point u^p on the boundary of $\mathbb{U}$. Neither player one nor player two can improve upon the current point u^p so the iteration has converged. But the converged point is clearly not the optimal point, u_{ce}.

Because of property 2, the nominal stability properties for the coupled and uncoupled cases are identical. The differences arise when the performance of cooperative control is compared to the benchmark of centralized control. Improving the performance of cooperative control in the case of coupled constraints is therefore a topic of current

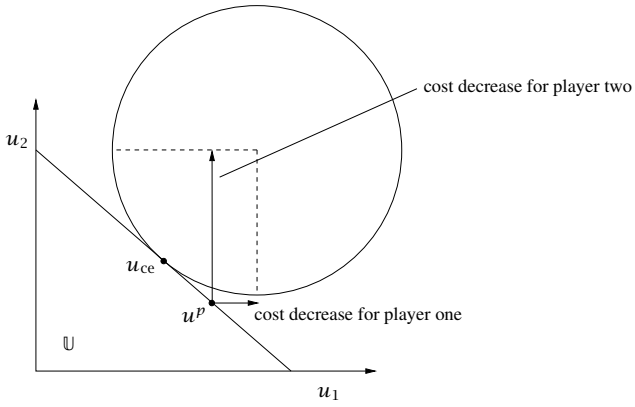


Figure 6.6: Cooperative control stuck on the boundary of $\mathbb{U}$ under coupled constraints; $u^{p+1} = u^p \neq u_{ce}$.

research. Current approaches include adding another player to the game, whose sole objective is to parcel out the coupled resource to the other players in a way that achieves optimality on iteration. This approach also makes sense from an engineering perspective because it is commonplace to design a dedicated control system for managing a shared resource such as steam or power among many plant units. The design of this single unit's control system is a reasonably narrow and well-defined task compared to the design of a centralized controller for the entire plant.

6.3.3 Exponential Convergence with Estimate Error

Consider next the constrained system evolution with estimate error

$$\begin{bmatrix} \hat{x}^+ \\ \mathbf{u}^+ \\ e^+ \end{bmatrix} = \begin{bmatrix} A\hat{x} + \overline{B}_1 u_1 + \overline{B}_2 u_2 + Le \\ g^p(\hat{x}, \mathbf{u}) \\ A_L e \end{bmatrix} \tag{6.22}$$

The estimate error is globally exponentially stable so we know from Lemma 6.7 that there exists a Lipschitz continuous Lyapunov function $J(\cdot)$ such that for all $e \in \mathbb{R}^n$

$$\begin{aligned} \overline{a}\,|e| \leq J(e) \leq \overline{b}\,|e| \\ J(e^+) - J(e) \leq -\overline{c}\,|e| \end{aligned}$$

in which $\overline{b} > 0$, $\overline{a} > 0$, and we can choose constant $\overline{c} > 0$ as large as desired. In the subsequent development, we require this Lyapunov function to be based on the first power of the norm rather than the usual square of the norm to align with Lipschitz continuity of the Lyapunov function. From the stability of the solution $x(k) = 0$ for all k for the *nominal* system, the cost function $V(\hat{x}, \mathbf{u})$ satisfies for all $\hat{x} \in \mathcal{X}_N$, $\mathbf{u} \in \mathbb{U}^N$

$$\begin{gathered}
\widetilde{a} \left|(\hat{x}, \mathbf{u})\right|^2 \leq V(\hat{x}, \mathbf{u}) \leq \widetilde{b} \left|(\hat{x}, \mathbf{u})\right|^2 \\
V(A\hat{x} + \overline{B}_1 u_1 + \overline{B}_2 u_2, \mathbf{u}^+) - V(\hat{x}, \mathbf{u}) \leq -\widetilde{c} \left|\hat{x}\right|^2 \\
|\mathbf{u}| \leq d \left|\hat{x}\right| \qquad \hat{x} \in \widetilde{r}\mathcal{B}
\end{gathered}$$

in which $\widetilde{a}, \widetilde{b}, \widetilde{c}, \widetilde{r} > 0$. We propose $W(\hat{x}, \mathbf{u}, e) = V(\hat{x}, \mathbf{u}) + J(e)$ as a Lyapunov function candidate for the perturbed system. We next derive the required properties of $W(\cdot)$ to establish exponential stability of the solution $(x(k), e(k)) = 0$. From the definition of $W(\cdot)$ we have for all $(\hat{x}, \mathbf{u}, e) \in \mathcal{X}_N \times \mathbb{U}^N \times \mathbb{R}^n$

$$\begin{aligned}
\widetilde{a} \left|(\hat{x}, \mathbf{u})\right|^2 + \overline{a} \left|e\right| \leq W(\hat{x}, \mathbf{u}, e) \leq \widetilde{b} \left|(\hat{x}, \mathbf{u})\right|^2 + \overline{b} \left|e\right| \\
a(\left|(\hat{x}, \mathbf{u})\right|^2 + |e|) \leq W(\hat{x}, \mathbf{u}, e) \leq b(\left|(\hat{x}, \mathbf{u})\right|^2 + |e|)
\end{aligned} \tag{6.23}$$

in which $a = \min(\widetilde{a}, \overline{a}) > 0$, $b = \max(\widetilde{b}, \overline{b})$. Next we compute the cost change

$$W(\hat{x}^+, \mathbf{u}^+, e^+) - W(\hat{x}, \mathbf{u}, e) = V(\hat{x}^+, \mathbf{u}^+) - V(\hat{x}, \mathbf{u}) + J(e^+) - J(e)$$

The Lyapunov function V is quadratic in $(x, \mathbf{u})$ and therefore Lipschitz continuous on bounded sets. Therefore, for all $\hat{x}, u_1, u_2, \mathbf{u}^+, e$ in some bounded set

$$\left| V(A\hat{x} + \overline{B}_1 u_1 + \overline{B}_2 u_2 + Le, \mathbf{u}^+) - V(A\hat{x} + \overline{B}_1 u_1 + \overline{B}_2 u_2, \mathbf{u}^+) \right| \leq L_V \left|Le\right|$$

in which L_V is the Lipschitz constant for V with respect to its first argument. Using the system evolution we have

$$V(\hat{x}^+, \mathbf{u}^+) \leq V(A\hat{x} + \overline{B}_1 u_1 + \overline{B}_2 u_2, \mathbf{u}^+) + L'_V \left|e\right|$$

in which $L'_V = L_V \left|L\right|$. Subtracting $V(\hat{x}, \mathbf{u})$ from both sides gives

$$V(\hat{x}^+, \mathbf{u}^+) - V(\hat{x}, \mathbf{u}) \leq -\widetilde{c} \left|\hat{x}\right|^2 + L'_V \left|e\right|$$

Substituting this result into the equation for the change in W gives

$$\begin{aligned} W(\hat{x}^+, \mathbf{u}^+, e^+) - W(\hat{x}, \mathbf{u}, e) &\le -\widetilde{c}\,|\hat{x}|^2 + L'_V\,|e| - \overline{c}\,|e| \\ &\le -\widetilde{c}\,|\hat{x}|^2 - (\overline{c} - L'_V)\,|e| \\ W(\hat{x}^+, \mathbf{u}^+, e^+) - W(\hat{x}, \mathbf{u}, e) &\le -c(|\hat{x}|^2 + |e|) \end{aligned} \tag{6.24}$$

in which we choose $\overline{c} > L'_V$, which is possible because we may choose $\overline{c}$ as large as we wish, and $c = \min(\widetilde{c}, \overline{c} - L'_V) > 0$. Notice this step is what motivated using the first power of the norm in $J(\cdot)$. Lastly, we require the constraint

$$|\mathbf{u}| \le d\,|\hat{x}| \qquad \hat{x} \in \widetilde{r}\mathcal{B} \tag{6.25}$$

Lemma 6.14 (Global asymptotic stability and exponential convergence of perturbed system)**.** *If either $\mathcal{X}_N$ or $\mathbb{U}$ is compact, there exist $\lambda < 1$ and $\delta(\cdot) \in \mathcal{K}_\infty$ such that the combined system* (6.22) *satisfies for all $(x(0), e(0))$ and $k \ge 0$*

$$|x(k), e(k)| \le \delta(|x(0), e(0)|)\lambda^k$$

The proof is based on the properties (6.23), (6.24), and (6.25) of function $W(\hat{x}, \mathbf{u}, e)$, and is basically a combination of the proofs of Lemmas 6.5 and 6.6. The region of attraction is the set of states and initial estimate errors for which the unstable modes of the two subsystems can be brought to zero in N moves while satisfying the respective input constraints. If both subsystems are stable, for example, the region of attraction is $(x, e) \in \mathcal{X}_N \times \mathbb{R}^n$.

6.3.4 Disturbance Models and Zero Offset

Integrating disturbance model. As discussed in Chapter 1, we model the disturbance with an integrator to remove steady offset. The augmented models for the local systems are

$$\begin{bmatrix} x_i \\ d_i \end{bmatrix}^+ = \begin{bmatrix} A_i & B_{di} \\ 0 & I \end{bmatrix} \begin{bmatrix} x_i \\ d_i \end{bmatrix} + \begin{bmatrix} \overline{B}_{i1} \\ 0 \end{bmatrix} u_1 + \begin{bmatrix} \overline{B}_{i2} \\ 0 \end{bmatrix} u_2$$
$$y_i = \begin{bmatrix} C_i & C_{di} \end{bmatrix} \begin{bmatrix} x_i \\ d_i \end{bmatrix} \qquad i = 1, 2$$

We wish to estimate both x_i and d_i from measurements y_i. To ensure this goal is possible, we make the following restriction on the disturbance models.

Assumption 6.15 (Disturbance models)**.**

$$\operatorname{rank}\begin{bmatrix} I - A_i & -B_{di} \\ C_i & C_{di} \end{bmatrix} = n_i + p_i \qquad i = 1, 2$$

It is always possible to satisfy this assumption by proper choice of B_{di}, C_{di}. From Assumption 6.13 (b), (A_i, C_i) is detectable, which implies that the first n_i columns of the square $(n_i + p_i) \times (n_i + p_i)$ matrix in Assumption 6.15 are linearly independent. Therefore the columns of $\left[\begin{smallmatrix} -B_{di} \\ C_{di} \end{smallmatrix}\right]$ can be chosen so that the entire matrix has rank $n_i + p_i$. Assumption 6.15 is equivalent to detectability of the following augmented system.

Lemma 6.16 (Detectability of distributed disturbance model)**.** *Consider the augmented systems*

$$\widetilde{A}_i = \begin{bmatrix} A_i & B_{di} \\ 0 & I \end{bmatrix} \qquad \widetilde{C}_i = \begin{bmatrix} C_i & C_{di} \end{bmatrix} \qquad i = 1, 2$$

The augmented systems $(\widetilde{A}_i, \widetilde{C}_i), i = 1, 2$ *are detectable if and only if Assumption 6.15 is satisfied.*

Proving this lemma is discussed in Exercise 6.29. The detectability assumption then establishes the existence of $\widetilde{L}_i$ such that $(\widetilde{A}_i - \widetilde{L}_i \widetilde{C}_i)$, $i = 1, 2$ are stable and the local integrating disturbances can be estimated from the local measurements.

Centralized target problem. We can solve the target problem at the plantwide level or as a distributed target problem at the subunit controller level. Consider first the centralized target problem with the disturbance model discussed in Chapter 1, (1.45)

$$\min_{x_s, u_s} \frac{1}{2} \left| u_s - u_{\mathrm{sp}} \right|^2_{R_s} + \frac{1}{2} \left| Cx_s + C_d \hat{d}(k) - y_{\mathrm{sp}} \right|^2_{Q_s}$$

subject to

$$\begin{bmatrix} I - A & -B \\ HC & 0 \end{bmatrix} \begin{bmatrix} x_s \\ u_s \end{bmatrix} = \begin{bmatrix} B_d \hat{d}(k) \\ r_{\mathrm{sp}} - HC_d \hat{d}(k) \end{bmatrix}$$
$$Eu_s \le e$$

in which we have removed the state inequality constraints to be consistent with the regulator problem. We denote the solution to this problem $(x_s(k), u_s(k))$. Notice first that the solution of the target problem

depends only on the disturbance estimate, $\hat{d}(k)$, and not the solution of the control problem. So we can analyze the behavior of the target by considering only the exponential convergence of the estimator. We restrict the plant disturbance d so that the target problem is feasible, and denote the solution to the target problem for the plant disturbance, $\hat{d}(k) = d$, as (x_s^*, u_s^*). Because the estimator is exponentially stable, we know that $\hat{d}(k) \to d$ as $k \to \infty$. Because the target problem is a positive definite quadratic program (QP), we know the solution is Lipschitz continuous on bounded sets in the term $\hat{d}(k)$, which appears linearly in the objective function and the right-hand side of the equality constraint. Therefore, if we also restrict the initial disturbance estimate error so that the target problem remains feasible for all time, we know $(x_s(k), u_s(k)) \to (x_s^*, u_s^*)$ and the rate of convergence is exponential.

Distributed target problem. Consider next the cooperative approach, in which we assume the input inequality constraints are uncoupled. In the constrained case, we try to set things up so each player solves a local target problem

$$\min_{x_{1s},u_{1s}} \frac{1}{2}\begin{bmatrix} y_{1s}-y_{1sp} \\ y_{2s}-y_{2sp} \end{bmatrix}' \begin{bmatrix} Q_{1s} & \\ & Q_{2s} \end{bmatrix} \begin{bmatrix} y_{1s}-y_{1sp} \\ y_{2s}-y_{2sp} \end{bmatrix} + \frac{1}{2}\begin{bmatrix} u_{1s}-u_{1sp} \\ u_{2s}-u_{2sp} \end{bmatrix}' \begin{bmatrix} R_{1s} & \\ & R_{2s} \end{bmatrix} \begin{bmatrix} u_{1s}-u_{1sp} \\ u_{2s}-u_{2sp} \end{bmatrix}$$

subject to

$$\begin{bmatrix} I-A_1 & & -\overline{B}_{11} & -\overline{B}_{12} \\ & I-A_2 & -\overline{B}_{21} & -\overline{B}_{22} \\ H_1C_1 & & & \\ & H_2C_2 & & \end{bmatrix} \begin{bmatrix} x_{1s} \\ x_{2s} \\ u_{1s} \\ u_{2s} \end{bmatrix} = \begin{bmatrix} B_{d1}\hat{d}_1(k) \\ B_{d2}\hat{d}_2(k) \\ r_{1sp} - H_1C_{d1}\hat{d}_1(k) \\ r_{2sp} - H_2C_{d2}\hat{d}_2(k) \end{bmatrix}$$

$$E_1 u_{1s} \le e_1$$

in which

$$y_{1s} = C_1 x_{1s} + C_{d1}\hat{d}_1(k) \qquad y_{2s} = C_2 x_{2s} + C_{d2}\hat{d}_2(k) \tag{6.27}$$

But here we run into several problems. First, the constraints to ensure zero offset in both players' controlled variables are not feasible with only the u_{1s} decision variables. We require also u_{2s}, which is not available to player one. We can consider deleting the zero offset condition for player two's controlled variables, the last equality constraint. But

if we do that for both players, then the two players have *different and coupled* equality constraints. That is a path to instability as we have seen in the noncooperative target problem. To resolve this issue, we move the controlled variables to the objective function, and player one solves instead the following

$$\min_{x_{1s},u_{1s}} \frac{1}{2}\begin{bmatrix} H_1y_{1s}-r_{1sp} \\ H_2y_{2s}-r_{2sp}\end{bmatrix}' \begin{bmatrix} T_{1s} & \\ & T_{2s}\end{bmatrix}\begin{bmatrix} H_1y_{1s}-r_{1sp} \\ H_2y_{2s}-r_{2sp}\end{bmatrix}$$

subject to (6.27) and

$$\begin{bmatrix} I-A_1 & & -\overline{B}_{11} & -\overline{B}_{12} \\ & I-A_2 & -\overline{B}_{21} & -\overline{B}_{22}\end{bmatrix}\begin{bmatrix} x_{1s} \\ x_{2s} \\ u_{1s} \\ u_{2s}\end{bmatrix} = \begin{bmatrix} B_{d1}\hat{d}_1(k) \\ B_{d2}\hat{d}_2(k)\end{bmatrix}$$

$$E_1u_{1s} \le e_1 \tag{6.28}$$

The equality constraints for the two players appear coupled when written in this form. Coupled constraints admit the potential for the optimization to become stuck on the boundary of the feasible region, and not achieve the centralized target solution after iteration to convergence. But Exercise 6.30 discusses how to show that the equality constraints are, in fact, uncoupled. Also, the distributed target problem as expressed here may not have a unique solution when there are more manipulated variables than controlled variables. In such cases, a regularization term using the input setpoint can be added to the objective function. The controlled variable penalty can be converted to a linear penalty with a large penalty weight to ensure exact satisfaction of the controlled variable setpoint.

If the input inequality constraints are coupled, however, then the distributed target problem may indeed become stuck on the boundary of the feasible region and not eliminate offset in the controlled variables. If the input inequality constraints are coupled, we recommend using the centralized approach to computing the steady-state target. As discussed above, the centralized target problem eliminates offset in the controlled variables as long as it remains feasible given the disturbance estimates.

Zero offset. Finally we establish the zero offset property. As described in Chapter 1, the regulator is posed in deviation variables

$$\tilde{\hat{x}}(k) = \hat{x}(k) - x_s(k) \qquad \tilde{u}(k) = u(k) - u_s(k) \qquad \tilde{\mathbf{u}} = \mathbf{u} - u_s(k)$$

in which the notation $\mathbf{u} - u_s(k)$ means to subtract $u_s(k)$ from each element of the $\mathbf{u}$ sequence. Player one then solves

$$\begin{aligned}
&\min_{\tilde{\mathbf{u}}_1} V(\tilde{x}_1(0), \tilde{x}_2(0), \tilde{\mathbf{u}}_1, \tilde{\mathbf{u}}_2) \\
\text{s.t. } \begin{bmatrix} \tilde{x}_1 \\ \tilde{x}_2 \end{bmatrix}^+ &= \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} \begin{bmatrix} \tilde{x}_1 \\ \tilde{x}_2 \end{bmatrix} + \begin{bmatrix} \overline{B}_{11} \\ \overline{B}_{21} \end{bmatrix} \tilde{u}_1 + \begin{bmatrix} \overline{B}_{12} \\ \overline{B}_{22} \end{bmatrix} \tilde{u}_2 \\
\tilde{\mathbf{u}}_1 &\in \mathbb{U}_1 \ominus u_s(k) \\
S_{1u}' \tilde{x}_1(N) &= 0 \\
|\tilde{\mathbf{u}}_1| &\le d_1 |\tilde{x}_1(0)|
\end{aligned}$$

Notice that because the input constraint is shifted by the input target, we must retain feasibility of the regulation problem by restricting also the plant disturbance and its initial estimate error. If the two players' regulation problems remain feasible as the estimate error converges to zero, we have exponential stability of the zero solution from Lemma 6.14. Therefore we conclude

$$\begin{aligned}
&(\tilde{x}(k), \tilde{u}(k)) \to (0,0) && \text{Lemma 6.14} \\
\implies\ &(\hat{x}(k), u(k)) \to (x_s(k), u_s(k)) && \text{definition of deviation variables} \\
\implies\ &(\hat{x}(k), u(k)) \to (x_s^*, u_s^*) && \text{target problem convergence} \\
\implies\ &x(k) \to x_s^* && \text{estimator stability} \\
\implies\ &r(k) \to r_{\text{sp}} && \text{target equality constraint}
\end{aligned}$$

and we have *zero offset* in the plant controlled variable $r = Hy$. The rate of convergence of $r(k)$ to r_{sp} is also exponential. As we saw here, this convergence depends on maintaining feasibility in both the target problem and the regulation problem at all times.

6.4 Constrained M-Player Game

We have set up the constrained two-player game so that the approach generalizes naturally to the M-player game. We do not have a lot of work left to do to address this general case. Recall $\mathbb{I}_{1:M}$ denotes the set

of integers $\{1, 2, \ldots, M\}$. We define the following systemwide variables

$$x(0) = \begin{bmatrix} x_1(0) \\ x_2(0) \\ \vdots \\ x_M(0) \end{bmatrix} \qquad \mathbf{u} = \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \vdots \\ \mathbf{u}_M \end{bmatrix} \qquad B_i = \begin{bmatrix} \overline{B}_{1i} \\ \overline{B}_{2i} \\ \vdots \\ \overline{B}_{Mi} \end{bmatrix} \quad \underline{B}_i = \begin{bmatrix} B_{1i} \\ B_{2i} \\ \vdots \\ B_{Mi} \end{bmatrix} \quad i \in \mathbb{I}_{1:M}$$

$$V(x(0), \mathbf{u}) = \sum_{j \in \mathbb{I}_{1:M}} \rho_j V_j(x_j(0), \mathbf{u})$$

Each player solves a similar optimization, so for $i \in \mathbb{I}_{1:M}$

$$\begin{aligned} &\min_{\mathbf{u}_i} V(x(0), \mathbf{u}) \\ &\text{s.t. } x^+ = Ax + \sum_{j \in \mathbb{I}_{1:M}} B_j u_j \\ &\mathbf{u}_i \in \mathbb{U}_i \\ &S_{ji}^{u\prime} x_{ji}(N) = 0 \quad j \in \mathbb{I}_{1:M} \\ &|\mathbf{u}_i| \le d_i \sum_{j \in \mathbb{I}_{1:M}} \left| x_{ji}(0) \right| \qquad \text{if } x_{ji}(0) \in r\mathcal{B},\ j \in \mathbb{I}_{1:M} \end{aligned}$$

This optimization can be expressed as a quadratic program, whose constraints and linear cost term depend affinely on parameter x. The warm start for each player at the next sample is generated from purely local information

$$\widetilde{\mathbf{u}}_i^+ = (u_i(1), u_i(2), \ldots, u_i(N-1), 0) \quad i \in \mathbb{I}_{1:M}$$

The controller iteration is given by

$$\mathbf{u}^{p+1} = \sum_{j \in \mathbb{I}_{1:M}} w_j \left(\mathbf{u}_1^p, \ldots, \mathbf{u}_j^0, \ldots, \mathbf{u}_M^p \right)$$

in which $\mathbf{u}_i^0 = \mathbf{u}_i^0 \left(x(0), \mathbf{u}_{j \in \mathbb{I}_{1:M}, j \neq i}^p \right)$. The plantwide cost function then satisfies for any $p \ge 0$ (see Exercise 6.26)

$$\begin{aligned} V(x^+, \mathbf{u}^+) &\le V(x, \mathbf{u}) - \sum_{j \in \mathbb{I}_{1:M}} \rho_j \ell_j(x_j, u_j) \\ |\mathbf{u}| &\le d\,|x| \qquad x \in r\mathcal{B} \end{aligned}$$

For the M-player game, we generalize Assumption 6.13 of the two-player game to the following.

Assumption 6.17 (Constrained M-player game)**.**

(a) The systems $(\underline{A}_i, \underline{B}_i)$, $i \in \mathbb{I}_{1:M}$ are stabilizable, in which $\underline{A}_i = \text{diag}(A_{1i}, A_{2i}, \cdots, A_{Mi})$.

(b) The systems (A_i, C_i), $i \in \mathbb{I}_{1:M}$ are detectable.

(c) The input penalties $R_i, i \in \mathbb{I}_{1:M}$ are positive definite, and $Q_i, i \in \mathbb{I}_{1:M}$ are semidefinite.

(d) The systems (A_i, Q_i), $i \in \mathbb{I}_{1:M}$ are detectable.

(e) The horizon is chosen sufficiently long to zero the unstable modes; $N \geq \max_{i \in \mathbb{I}_{1:M}}(\underline{n}_i^u)$, in which $\underline{n}_i^u$ is the number of unstable modes of $\underline{A}_i$.

(f) Zero offset. For achieving zero offset, we augment the models with integrating disturbances such that

$$\text{rank} \begin{bmatrix} I - A_i & -B_{di} \\ C_i & C_{di} \end{bmatrix} = n_i + p_i \qquad i \in \mathbb{I}_{1:M}$$

Applying Theorem 6.5 then establishes exponential stability of the solution $x(k) = 0$ for all k. The region of attraction is the set of states for which the unstable modes of each subsystem can be brought to zero in N moves, while satisfying the respective input constraints. These conclusions apply regardless of how many iterations of the players' optimizations are used in the control calculation. Although the closed-loop system is exponentially stable for both coupled and uncoupled constraints, the converged distributed controller is equal to the centralized controller only for the case of uncoupled constraints.

The exponential stability of the regulator implies that the states and inputs of the constrained M-player system converge to the steady-state target. The steady-state target can be calculated as a centralized or distributed problem. We assume the centralized target has a feasible, zero offset solution for the true plant disturbance. The initial state of the plant and the estimate error must be small enough that feasibility of the target is maintained under the nonzero estimate error.

6.5 Nonlinear Distributed MPC

In the nonlinear case, the usual model comes from physical principles and conservation laws of mass, energy, and momentum. The state has a physical meaning and the measured outputs usually are a subset of

the state. We assume the model is of the form

$$\frac{dx_1}{dt} = f_1(x_1, x_2, u_1, u_2) \qquad y_1 = C_1 x_1$$
$$\frac{dx_2}{dt} = f_2(x_1, x_2, u_1, u_2) \qquad y_2 = C_2 x_2$$

in which C_1, C_2 are matrices of zeros and ones selecting the part of the state that is measured in subsystems one and two. We generally cannot avoid state x_2 dependence in the differential equation for x_1. But often only a small subset of the entire state x_2 appears in f_1, and vice versa. The reason in chemical process systems is that the two subsystems are generally coupled through a small set of process streams transferring mass and energy between the systems. These connecting streams isolate the coupling between the two systems and reduce the influence to a small part of the entire state required to describe each system.

Given these physical system models of the subsystems, the overall plant model is

$$\frac{dx}{dt} = f(x, u) \qquad y = Cx$$

with

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \qquad u = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \qquad f = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} \qquad y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} \qquad C = \begin{bmatrix} C_1 & \\ & C_2 \end{bmatrix}$$

6.5.1 Nonconvexity

The basic difficulty in both the theory and application of nonlinear MPC is the nonconvexity in the control objective function caused by the nonlinear dynamic model. This difficulty applies even to centralized nonlinear MPC as discussed in Section 2.7, and motivates the development of suboptimal MPC. In the distributed case, nonconvexity causes extra difficulties. As an illustration, consider the simple two-player, nonconvex game depicted in Figure 6.7. The cost function is

$$V(u_1, u_2) = e^{-2u_1} - 2e^{-u_1} + e^{-2u_2} - 2e^{-u_2} + a\exp(-\beta((u_1 + 0.2)^2 + (u_2 + 0.2)^2))$$

in which $a = 1.1$ and $\beta = 0.4$. Each player optimizes the cooperative objective starting at ① and produces the points (u_1^0, u_2^p), denoted ②,

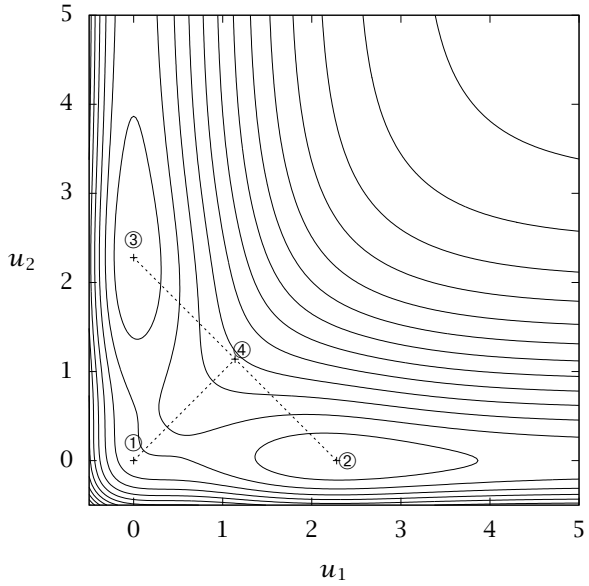


Figure 6.7: Cost contours for a two-player, nonconvex game; cost *increases* for the convex combination of the two players' optimal points.

and (u_1^p, u_2^0), denoted ③. Consider taking a convex combination of the two players' optimal points for the next iterate

$$(u_1^{p+1}, u_2^{p+1}) = w_1(u_1^0, u_2^p) + w_2(u_1^p, u_2^0) \qquad w_1 + w_2 = 1, \quad w_1, w_2 \geq 0$$

We see in Figure 6.7 that this iterate causes the objective function to *increase* rather than decrease for most values of w_1, w_2. For $w_1 = w_2 = 1/2$, we see clearly from the contours that V at point ④ is greater than V at point ①.

The possibility of a cost increase leads to the possibility of closed-loop instability and precludes developing even a nominal control theory for this simple approach, which was adequate for the convex, linear plant case.[4] In the centralized MPC problem, this nonconvexity issue can be addressed in the optimizer, which can move both inputs simultaneously and always avoid a cost increase. One can of course consider

[4]This point marked the state of affairs at the time of publication of the first edition of this text. The remaining sections summarize one approach that addresses the nonconvexity problem (Stewart, Wright, and Rawlings, 2011).

adding another player to the game who has access to more systemwide information. This player takes the optimization results of the individual players and determines a search direction and step length that achieve a cost decrease for the overall system. This player is often known as a coordinator. The main drawback of this approach is that the design of the coordinator may not be significantly simpler than the design of the centralized controller.

Rather than design a coordinator, we instead let each player evaluate the effect of taking a combination of all the players' optimal moves. The players can then easily find an effective combination that leads to a cost decrease. We describe one such algorithm in the next section, which we call the *distributed gradient algorithm.*

6.5.2 Distributed Algorithm for Nonconvex Functions

We consider the problem

$$\min_{u} V(u) \quad \text{s.t.} \quad u \in \mathbb{U} \tag{6.29}$$

in which $u \in \mathbb{R}^m$ and $V : \mathbb{R}^m \to \mathbb{R}_{\geq 0}$ is twice continuously differentiable and not necessarily convex. We assume $\mathbb{U}$ is closed, convex, and can be expressed as $\mathbb{U} = \mathbb{U}_1 \times \cdots \times \mathbb{U}_M$ with $\mathbb{U}_i \in \mathbb{R}^{m_i}$ for all $i \in \mathbb{I}_{1:M}$. We solve approximately the following subproblems at iterate $p \geq 0$ for all $i \in \mathbb{I}_{1:M}$

$$\min_{u_i \in \mathbb{U}_i} V(u_i, u_{-i}^p)$$

in which $u_{-i} = (u_1, \ldots, u_{i-1}, u_{i+1}, \ldots, u_M)$. Let $\overline{u}_i^p$ denote the approximate solution to these optimizations. We compute the approximate solutions via the standard technique of line search with gradient projection. At iterate $p \geq 0$

$$\overline{u}_i^p = \mathcal{P}_i(u_i^p - \nabla_i V(u^p)) \tag{6.30}$$

in which $\nabla_i V(u^p)$ is the ith component of $\nabla V(u^p)$ and $\mathcal{P}_i(\cdot)$ denotes projection onto the set $\mathbb{U}_i$. Define the step $v_i^p = \overline{u}_i^p - u_i^p$. The stepsize α_i^p is chosen as follows; each suboptimizer initializes the stepsize with $\overline{\alpha}_i$, and then uses backtracking until α_i^p satisfies the Armijo rule (Bertsekas, 1999, p.230)

$$V(u^p) - V(u_i^p + \alpha_i^p v_i^p, u_{-i}^p) \geq -\sigma \alpha_i^p \nabla_i V(u^p)' v_i^p \tag{6.31}$$

in which $\sigma \in (0,1)$. After all suboptimizers finish backtracking, they exchange proposed steps. Each suboptimizer forms a candidate step

$$u_i^{p+1} = u_i^p + w_i \alpha_i^p v_i^p \quad \forall i \in \mathbb{I}_{1:M} \tag{6.32}$$

and checks the following inequality

$$V(u^{p+1}) \leq \sum_{i \in \mathbb{I}_{1:M}} w_i V(u_i^p + \alpha_i^p v_i^p, u_{-i}^p) \tag{6.33}$$

with $\sum_{i \in \mathbb{I}_{1:M}} w_i = 1$ and $w_i > 0$ for all $i \in \mathbb{I}_{1:M}$. If condition (6.33) is not satisfied, then we remove the direction with the least cost improvement, $i_{\max} = \arg\max_i \{V(u_i^p + \alpha_i^p v_i^p, u_{-i}^p)\}$, by setting $w_{i_{\max}}$ to zero and repartitioning the remaining w_i so that they sum to one. The candidate step (6.32) is recalculated and condition (6.33) is checked again. This process is repeated until (6.33) is satisfied. It may happen that condition (6.33) is satisfied with only a single direction. The distributed algorithm thus eliminates poor suboptimizer steps and ensures that the objective function decreases at each iterate, even for nonconvex objective functions. The proposed algorithm has the following properties.

Lemma 6.18 (Distributed gradient algorithm properties)**.** *The distributed gradient projection algorithm has the following properties.*

(a) (Feasibility.) Given a feasible initial condition, the iterates u^p are feasible for all $p \geq 0$.

(b) (Objective decrease.) The objective function decreases at every iterate: $V(u^{p+1}) \leq V(u^p)$.

(c) (Convergence.) Every accumulation point of the sequence $(u^p)_{p \geq 0}$ is a stationary point.

The proof of Lemma 6.18 is given in Stewart et al. (2011). Note that the test of inequality (6.33) does not require a coordinator. At each iteration the subsystems exchange the solutions of the gradient projection. Because each subsystem has access to the plantwide model, they can evaluate the objection function, and the algorithm can be run independently on each controller. This computation is likely a smaller overhead than a coordinating optimization.

Figure 6.8 shows the results of applying the proposed distributed gradient algorithm to the previous example. The problem has two global minima located at $(0.007, 2.28)$ and $(2.28, 0,007)$, and a local

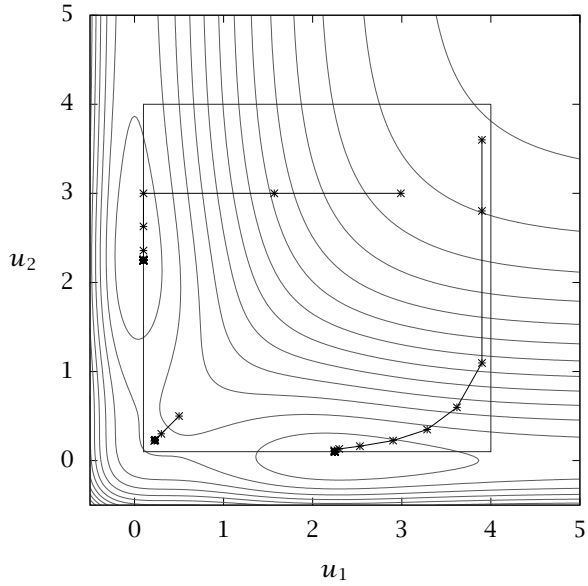


Figure 6.8: Nonconvex function optimized with the distributed gradient algorithm. Iterations converge to local minima from all starting points.

minimum at $(0.23, 0.23)$. The inputs are constrained: $0.1 \leq u_i \leq 4$ for $i \in \mathbb{I}_{1:2}$. The algorithm is initialized at three different starting points $(0.5, 0.5)$, $(3.9, 3.6)$, and $(2.99, 3)$. From Figure 6.8 we see that each of these starting points converges to a different local minimum.

6.5.3 Distributed Nonlinear Cooperative Control

Next we design a controller based on the distributed optimization algorithm. For simplicity of presentation, we assume that the plant consists of two subsystems. We consider the standard MPC cost function for each system $i \in \mathbb{I}_{1:2}$

$$V_i(x(0), \mathbf{u}_1, \mathbf{u}_2) = \sum_{k=0}^{N-1} \ell_i(x_i(k), u_i(k)) + V_{if}(x(N))$$

with $\ell_i(x_i, u_i)$ denoting the stage cost, $V_{if}(x)$ the terminal cost of system i, and $x_i(i) = \phi_i(k; x_i, \mathbf{u}_1, \mathbf{u}_2)$. Because x_i is a function of both u_1

and u_2, V_i is a function of both $\mathbf{u}_1$ and $\mathbf{u}_2$. As in the case for linear plants, we define the plantwide objective

$$V(x_1(0), x_2(0), \mathbf{u}_1, \mathbf{u}_2) = \rho_1 V_1(x(0), \mathbf{u}_1, \mathbf{u}_2) + \rho_2 V_2(x(0), \mathbf{u}_1, \mathbf{u}_2)$$

in which $\rho_1, \rho_2 > 0$ are weighting factors. To simplify notation we use $V(x, \mathbf{u})$ to denote the plantwide objective. Similarly we define the system stage cost and terminal cost as the combined stage costs $\ell(x, u) := \rho_1 \ell_1(x_1, u_1) + \rho_2 \ell_2(x_2, u_2)$, and terminal costs $V_f(x) := \rho_1 V_{1f}(x) + \rho_2 V_{2f}(x)$. Each subsystem has constraints of the form

$$u_1(k) \in \mathbb{U}_1 \qquad u_2(k) \in \mathbb{U}_2 \quad k \in \mathbb{I}_{0:N-1}$$

in which each $\mathbb{U}_i \in \mathbb{R}^{m_i}$ is compact, convex, and contains the origin. Finally, we define the terminal region $\mathbb{X}_f$ to be a sublevel set of V_f.

$$\mathbb{X}_f = \{x \mid V_f(x) \le a\}$$

for some $a > 0$.

We next modify the standard stability assumption to account for the distributed nature of the problem.

Assumption 6.19 (Basic stability assumption (distributed))**.** $V_f(\cdot)$, $\mathbb{X}_f$, and $\ell(\cdot)$ have the following properties.

(a) For all $x \in \mathbb{X}_f$, there exists (u_1, u_2) (such that $(x, u_1, u_2) \in \mathbb{R}^n \times U_1 \times U_2$) satisfying

$$f(x, u_1, u_2) \in \mathbb{X}_f$$
$$V_f(f(x, u_1, u_2)) - V_f(x) \le -\ell(x, u_1, u_2)$$

(b) For each $i \in \mathbb{I}_{1:2}$, there exist $\mathcal{K}_\infty$ functions $\alpha_i(\cdot)$, and $\alpha_f(\cdot)$ satisfying

$$\ell_i(x_i, u_i) \ge \alpha_i(|x_i|) \qquad \forall (x_i, u_i) \in \mathcal{X}_N \times \mathbb{U}_i$$
$$V_f(x) \le \alpha_f(|x|) \qquad \forall x \in \mathbb{X}_f$$

This assumption implies that there exist local controllers $\kappa_{if} : \mathbb{X}_f \to \mathbb{U}_i$ for all $i \in \mathbb{I}_{1:2}$ such that for all $x \in \mathbb{X}_f$

$$V_f\big(f(x, \kappa_{1f}(x), \kappa_{2f}(x))\big) - V_f(x) \le -\ell(x, \kappa_{1f}(x), \kappa_{2f}(x)) \qquad (6.34)$$

with $f(x, \kappa_{1f}(x), \kappa_{2f}(x)) \in \mathbb{X}_f$. Each terminal controller $\kappa_{if}(\cdot)$ may be found offline.

Removing the terminal constraint in suboptimal MPC. To show stability, we require that $\phi(N;x,\mathbf{u}) \in \mathbb{X}_f$. But the terminal constraint on the state shows up as a coupled input constraint in each subsystem's optimization problem. As we have already discussed, coupled input constraints may prevent the distributed algorithm from converging to the optimal plantwide control (Stewart, Venkat, Rawlings, Wright, and Pannocchia, 2010). The terminal constraint can be removed from the control problem by modifying the terminal penalty, however, as we demonstrate next.

For some $\beta \geq 1$, we define the objective function

$$V^{\beta}(x,\mathbf{u}) = \sum_{k=0}^{N-1} \ell(x(k),u(k)) + \beta V_f(x(N)) \tag{6.35}$$

and the set of admissible initial $(x,\mathbf{u})$ as

$$\mathbb{Z}_0 = \{(x,\mathbf{u}) \in \mathbb{X} \times \mathbb{U}^N \mid V^{\beta}(x,\mathbf{u}) \leq \overline{V},\ \phi(N;x,\mathbf{u}) \in \mathbb{X}_f\} \tag{6.36}$$

in which $\overline{V} > 0$ is an arbitrary constant and $\mathbb{X} = \mathbb{R}^n$. The set of initial states $\mathbb{X}_0$ is the projection of $\mathbb{Z}_0$ onto $\mathbb{X}$

$$\mathbb{X}_0 = \{x \in \mathbb{X} \mid \exists \mathbf{u} \text{ such that } (x,\mathbf{u}) \in \mathbb{Z}_0\}$$

We have the following result.

Proposition 6.20 (Terminal constraint satisfaction)**.** *Let* $\{(x(k),\mathbf{u}(k)) \mid k \in \mathbb{I}_{\geq 0}\}$ *denote the set of states and control sequences generated by the suboptimal system. There exists a* $\overline{\beta} > 1$ *such that for all* $\beta \geq \overline{\beta}$*, if* $(x(0),\mathbf{u}(0)) \in \mathbb{Z}_0$*, then* $(x(k),\mathbf{u}(k)) \in \mathbb{Z}_0$ *with* $\phi(N;x(k),\mathbf{u}(k)) \in \mathbb{X}_f$ *for all* $k \in \mathbb{I}_{\geq 0}$*.*

The proof of this proposition is given in Stewart et al. (2011). We are now ready to define the cooperative control algorithm for nonlinear systems.

Cooperative control algorithm. Let $x(0)$ be the initial state and $\widetilde{\mathbf{u}} \in \mathbb{U}$ be the initial feasible input sequence for the cooperative MPC algorithm such that $\phi(N;x(0),\widetilde{\mathbf{u}}) \in \mathbb{X}_f$. At each iterate p, an approximate solu-

tion of the following optimization problem is computed

$$\begin{aligned}
&\min_{\mathbf{u}} \; V(x_1(0), x_2(0), \mathbf{u}_1, \mathbf{u}_2) \\
&\text{s.t.} \quad x_1^+ = f_1(x_1, x_2, u_1, u_2) \\
&\qquad x_2^+ = f_2(x_1, x_2, u_1, u_2) \\
&\qquad \mathbf{u}_i \in \mathbb{U}_i^N \quad \forall i \in \mathbb{I}_{1:2} \\
&\qquad |\mathbf{u}_i| \le \delta_i(|x_i(0)|) \quad \text{if } x(0) \in \mathcal{B}r \quad \forall i \in \mathbb{I}_{1:2}
\end{aligned} \tag{6.37}$$

in which $\delta_i(\cdot) \in \mathcal{K}_\infty$ and $r > 0$ can be chosen as small as desired. We can express (6.37) in the form of (6.29) by eliminating the model equality constraints. To implement distributed control, we simply use the distributed gradient algorithm to solve (6.37).

Denote the solution returned by the algorithm as $\mathbf{u}^{\bar{p}}(x, \tilde{\mathbf{u}})$. The first element of the sequence, denoted $\kappa^{\bar{p}}(x(0)) = u^{\bar{p}}(0; x(0), \tilde{\mathbf{u}})$, is injected into the plant. To reinitialize the algorithm at the next sample time, we compute the warm start

$$\begin{aligned}
\tilde{\mathbf{u}}_1^+ &= \{u_1(1), u_1(2), \ldots, u_1(N-1), \kappa_{1f}(x(N))\} \\
\tilde{\mathbf{u}}_2^+ &= \{u_2(1), u_2(2), \ldots, u_2(N-1), \kappa_{2f}(x(N))\}
\end{aligned}$$

in which $x(N) = \phi(N; x(0), \mathbf{u}_1, \mathbf{u}_2)$. We expect that it is not possible to solve (6.37) to optimality in the available sample time, and the distributed controller is therefore a form of suboptimal MPC. The properties of the closed-loop system are therefore analyzed using suboptimal MPC theory.

6.5.4 Stability of Distributed Nonlinear Cooperative Control

We first show that the plantwide objective function decreases between sampling times. Let $(x, \mathbf{u})$ be the state and input sequence at some time. Using the warm start as the initial condition at the next sample time, we have

$$\begin{aligned}
V(x^+, \tilde{\mathbf{u}}^+) = V(x, \mathbf{u}) &- \rho_1 \ell_1(x_1, u_1) - \rho_2 \ell_2(x_2, u_2) \\
&- \rho_1 V_{1f}(x(N)) - \rho_2 V_{2f}(x(N)) \\
+ \rho_1 \ell_1(x_1(N), \kappa_{1f}(x(N))) &+ \rho_2 \ell_2(x_2(N), \kappa_{2f}(x(N))) \\
+ \rho_1 V_{1f}\Big(f_1(x_1(N), x_2(N), \kappa_{1f}(x(N)), \kappa_{2f}(x(N)))\Big) & \\
+ \rho_2 V_{2f}\Big(f_2(x_1(N), x_2(N), \kappa_{1f}(x(N)), \kappa_{2f}(x(N)))\Big) &
\end{aligned}$$

From (6.34) of the stability assumption, we have that

$$V(x^+, \tilde{\mathbf{u}}^+) \leq V(x, \mathbf{u}) - \rho_1 \ell_1(x_1, u_1) - \rho_2 \ell_2(x_2, u_2)$$

By Lemma 6.18(b), the objective function cost only decreases from this warm start, so that

$$V(x^+, \mathbf{u}^+) \leq V(x, \mathbf{u}) - \rho_1 \ell_1(x_1, u_1) - \rho_2 \ell_2(x_2, u_2)$$

and we have the required cost decrease of a Lyapunov function

$$V(x^+, \mathbf{u}^+) - V(x, \mathbf{u}) \leq -\alpha(|(x, u)|) \tag{6.38}$$

in which $\alpha(|(x,u)|) = \rho_1 \alpha_1(|(x_1,u_1)|) + \rho_2 \alpha_2(|(x_2,u_2)|)$.

We can now state the main result. Let $\mathcal{X}_N$ be the admissible set of initial states for which the control optimization (6.37) is feasible.

Theorem 6.21 (Asymptotic stability)**.** *Let Assumptions 2.2, 2.3, and 6.19 hold, and let $V(\cdot) \leftarrow V^{\bar{p}}(\cdot)$ from Proposition 6.20. Then for every $x(0) \in \mathcal{X}_N$, the origin is asymptotically stable for the closed-loop system $x^+ = f(x, \kappa^{\bar{p}}(x))$.*

The proof follows, with minor modification, the proof that suboptimal MPC is asymptotically stable in Theorem 2.48. As in the previous sections, the controller has been presented for the case of two subsystems, but can be extended to any finite number of subsystems.

We conclude the discussion of nonlinear distributed MPC by revisiting the unstable nonlinear example system presented in Stewart et al. (2011).

Example 6.22: Nonlinear distributed control

We consider the unstable nonlinear system

$$\begin{aligned} x_1^+ &= x_1^2 + x_2 + u_1^3 + u_2 \\ x_2^+ &= x_1 + x_2^2 + u_1 + u_2^3 \end{aligned}$$

with initial condition $(x_1, x_2) = (3, -3)$. The control objective is to stabilize the system and regulate the states to the origin. We use a standard quadratic stage cost

$$\begin{aligned} \ell_1(x_1, u_1) &= \frac{1}{2}(x_1' Q_1 x_1 + u_1' R_1 u_1) \\ \ell_2(x_2, u_2) &= \frac{1}{2}(x_2' Q_2 x_2 + u_2' R_2 u_2) \end{aligned}$$

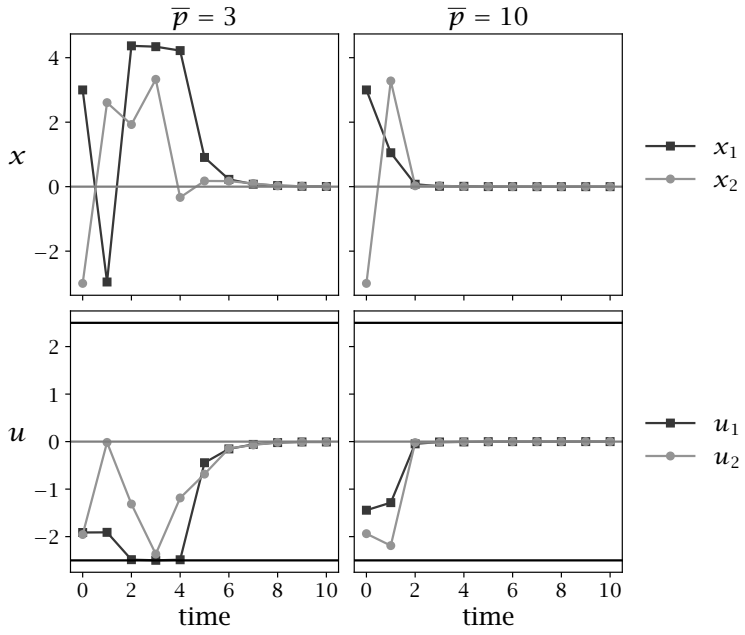


Figure 6.9: Closed-loop state and control evolution with $(x_1(0), x_2(0)) = (3, -3)$. Setting $\overline{p} = 10$ approximates the centralized controller.

with $Q_1, Q_2 > 0$ and $R_1, R_2 > 0$. This stage cost gives the objective function

$$V(x, \mathbf{u}) = \frac{1}{2} \sum_{k=0}^{N-1} x(k)' Q x(k) + u(k)' R u(k) + V_f(x(N))$$

in which $Q = \text{diag}(Q_1, Q_2)$, $R = \text{diag}(R_1, R_2)$. The terminal penalty is defined in the standard way for centralized MPC; we linearize the system at the steady state, and design an LQ controller for the linearized system. The terminal region is then a sublevel set of the terminal penalty chosen small enough to satisfy the input constraints. We use the following parameter values in the simulation study

$$Q = I \quad R = I \quad N = 2 \quad \overline{p} = 3 \quad \mathbb{U}_i = [-2.5, 2.5] \quad \forall i \in \mathbb{I}_{1:2}$$

Figure 6.9 shows that the controller is stabilizing for as few as $\overline{p} = 3$ iterations. Increasing the maximum number of iterations can signifi-

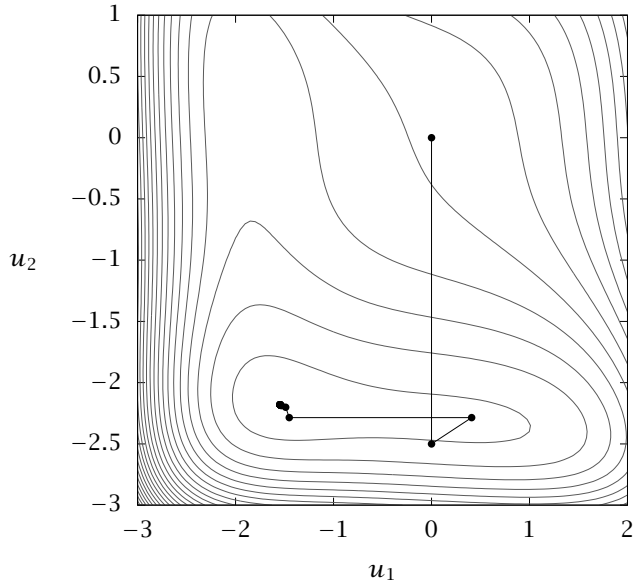


Figure 6.10: Contours of $V(x(0), \mathbf{u}_1, \mathbf{u}_2)$ with $N = 1$ at $k = 0$, $(x_1(0), x_2(0)) = (3, -3)$. Iterations of the subsystem controllers with initial condition $(u_1^0, u_2^0) = (0, 0)$.

cantly improve the performance. Figure 6.9 shows the performance improvement for $\overline{p} = 10$, which is close to the centralized MPC performance. To see the difficulty in optimizing the nonconvex objective function, iterations of the initial control optimization are shown in Figure 6.10 for the $N = 1$ case. Clearly the distributed optimization method is able to efficiently handle this nonconvex objective with only a few iterations. □

6.6 Notes

At least three different fields have contributed substantially to the material presented in this chapter. We attempt here to point out briefly what each field has contributed, and indicate what literature the interested reader may wish to consult for further pursuing this and related subjects.

Game theory. Game theory emerged in the mid-1900s to analyze situations in which multiple players follow a common set of rules but have their own and different objectives that they try to optimize in competition with each other. Von Neumann and Morgenstern introduced the classic text on this subject, "Theory of Games and Economic Behavior," in 1944. A principle aim of game theory since its inception was to model and understand human *economic* behavior, especially as it arises in a capitalistic, free-market system. For that reason, much of the subsequent game theory literature was published in economics journals rather than systems theory journals. This field has contributed richly to the ideas and vocabulary used in this chapter to describe distributed control. For example, the game in which players have different objectives is termed *noncooperative*. The equilibrium of a noncooperative game is known as a *Nash equilibrium* (Nash, 1951). The Nash equilibrium is usually not Pareto optimal, which means that the outcomes for all players can be improved simultaneously from the Nash solution. A comprehensive overview of the game theory literature, especially the parts relevant to control theory, is provided by Başar and Olsder (1999, Chapter 1), which is a highly recommended reference. Analyzing the equilibria of a noncooperative game is usually more complex than the cooperative game (optimal control problem). The closed-loop properties of a receding horizon implementation of any of these game theory solutions is not addressed in game theory. That topic is addressed by control theory.

Distributed optimization. The optimization community has extensively studied the issue of solving large-scale optimization problems using distributed optimization methods. The primary motivation in this field is to exploit parallel computing hardware and distributed data communication networks to solve large optimization problems faster. Bertsekas and Tsitsiklis provide an excellent and comprehensive overview of this field, focusing on numerical algorithms for implementing the distributed approaches. The important questions that are addressed in designing a distributed optimization are: task allocation, communication, and synchronization (Bertsekas and Tsitsiklis, 1997, Chapter 1).

These basic concepts arise in distributed problems of all types, and therefore also in the distributed MPC problem, which provides good synergy between these fields. But one should also note the structural distinctions between distributed optimization and distributed MPC. The primary obstacle to implementing centralized MPC for large-scale plants

is not *computational* but *organizational*. The agents considered in distributed MPC are usually existing MPC systems already built for units or subsystems within an existing large-scale process. The plant management often is seeking to improve the plant performance by better coordinating the behavior of the different agents already in operation. Ignoring these structural constraints and treating the distributed MPC problem purely as a form of distributed optimization, ignores aspects of the design that are critical for successful industrial application (Rawlings and Stewart, 2008).

Control theory. Researchers have long studied the issue of how to distribute control tasks in a complex large-scale plant (Mesarović, Macko, and Takahara, 1970; Sandell Jr., Varaiya, Athans, and Safonov, 1978). The centralized controller and decentralized controller define two limiting design extremes. Centralized control accounts for all possible interactions, large and small, whereas decentralized control ignores them completely. In decentralized control the local agents have no knowledge of each others' actions. It is well known that the nominal closed-loop system behavior under decentralized control can be arbitrarily poor (unstable) if the system interactions are not small. The following reviews provide general discussion of this and other performance issues involving decentralized control (Šiljak, 1991; Lunze, 1992; Larsson and Skogestad, 2000; Cui and Jacobsen, 2002).

The next level up in design complexity from decentralized control is noncooperative control. In this framework, the agents have interaction models and communicate at each iteration (Jia and Krogh, 2002; Motee and Sayyar-Rodsari, 2003; Dunbar and Murray, 2006). The advantage of noncooperative control over decentralized control is that the agents have accurate knowledge of the effects of all other agents on their local objectives. The basic issue to analyze and understand in this setup is the competition between the agents. Characterizing the noncooperative equilibrium is the subject of noncooperative game theory, and the impact of using that solution for feedback control is the subject of control theory. For example, Dunbar (2007) shows closed-loop stability for an extension of noncooperative MPC described in (Dunbar and Murray, 2006) that handles systems with interacting subsystem dynamics. The key assumptions are the existence of a stabilizing *decentralized* feedback law valid near the origin, and an inequality condition limiting the coupling between the agents.

Cooperative MPC was introduced by Venkat, Rawlings, and Wright (2007). They show that a receding horizon implementation of a coop-

erative game with any number of iterates of the local MPC controllers leads to closed-loop stability for linear dynamics. Venkat, Rawlings, and Wright (2006a,b) show that state estimation errors (output instead of state feedback) do not change the system closed-loop stability if the estimators are also asymptotically stable. Most of the theoretical results on cooperative MPC of linear systems given in this chapter are presented in Venkat (2006) using an earlier, different notation. If implementable, this form of distributed MPC clearly has the best control properties. Although one can easily modify the agents' objective functions in a single large-scale process owned by a single company, this kind of modification may not be possible in other situations in which competing interests share critical infrastructure.

The requirements of the many different classes of applications continue to create exciting opportunities for continued research in this field. An excellent recent review provides a useful taxonomy of the different features of the different approaches (Scattolini, 2009). A recent text compiles no less than *35* different approaches to distributed MPC from more than *80* contributors (Maestre and Negenborn, 2014). The growth in the number and diversity of applications of distributed MPC shows no sign of abating.

6.7 Exercises

Exercise 6.1: Three looks at solving the LQ problem (LQP)

In the following exercise, you will write three codes to solve the LQR using Octave or MATLAB. The objective function is the LQR with mixed term

$$V = \frac{1}{2}\sum_{k=0}^{N-1}\left(x(k)'Qx(k) + u(k)'Ru(k) + 2x(k)'Mu(k)\right) + (1/2)x(N)'P_f x(N)$$

First, implement the method described in Section 6.1.1 in which you eliminate the state and solve the problem for the decision variable

$$\mathbf{u} = (u(0), u(1), \ldots, u(N-1))$$

Second, implement the method described in Section 6.1.1 in which you do *not* eliminate the state and solve the problem for

$$\mathbf{z} = (u(0), x(1), u(1), x(2), \ldots, u(N-1), x(N))$$

Third, use backward dynamic programming (DP) and the Riccati iteration to compute the closed-form solution for $u(k)$ and $x(k)$.

(a) Let

$$A = \begin{bmatrix} 4/3 & -2/3 \\ 1 & 0 \end{bmatrix} \qquad B = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \qquad C = \begin{bmatrix} -2/3 & 1 \end{bmatrix} \qquad x(0) = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$Q = C'C + 0.001I \qquad P_f = \Pi \qquad R = 0.001 \qquad M = 0$$

in which the terminal penalty, P_f is set equal to Π, the steady-state cost to go. Compare the three solutions for $N = 5$. Plot $x(k)$, $u(k)$ versus time for the closed-loop system.

(b) Let $N = 50$ and repeat. Do any of the methods experience numerical problems generating an accurate solution? Plot the condition number of the matrix that is inverted in the first two methods versus N.

(c) Now consider the following unstable system

$$A = \begin{bmatrix} 27.8 & -82.6 & 34.6 \\ 25.6 & -76.8 & 32.4 \\ 40.6 & -122.0 & 51.9 \end{bmatrix} \qquad B = \begin{bmatrix} 0.527 & 0.548 \\ 0.613 & 0.530 \\ 1.06 & 0.828 \end{bmatrix} \qquad x(0) = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

Consider regulator tuning parameters and constraints

$$Q = I \qquad P_f = \Pi \qquad R = I \qquad M = 0$$

Repeat parts (a) and (b) for this system. Do you lose accuracy in any of the solution methods? What happens to the condition number of $H(N)$ and $S(N)$ as N becomes large? Which methods are still accurate for this case? Can you explain what happened?

Exercise 6.2: LQ as least squares

Consider the standard LQP

$$\min_{\mathbf{u}} V = \frac{1}{2}\sum_{k=0}^{N-1}\left(x(k)'Qx(k) + u(k)'Ru(k)\right) + (1/2)x(N)'P_f x(N)$$

subject to

$$x^+ = Ax + Bu$$

(a) Set up the dense Hessian least squares problem for the LQP with a horizon of three, $N = 3$. Eliminate the state equations and write out the objective function in terms of only the decision variables $u(0), u(1), u(2)$.

(b) What are the conditions for an optimum, i.e., what linear algebra problem do you solve to compute $u(0), u(1), u(2)$?

Exercise 6.3: Lagrange multiplier method

Consider the general least squares problem

$$\min_x V(x) = \frac{1}{2}x'Hx + \text{const}$$

subject to

$$Dx = d$$

(a) What is the Lagrangian L for this problem? What is the dimension of the Lagrange multiplier vector, λ?

(b) What are necessary and sufficient conditions for a solution to the optimization problem?

(c) Apply this approach to the LQP of Exercise 6.2 using the equality constraints to represent the model equations. What are H, D, d for the LQP?

(d) Write out the linear algebra problem to be solved for the optimum.

(e) Contrast the two different linear algebra problems in these two approaches. Which do you want to use when N is large and why?

Exercise 6.4: Reparameterizing an unstable system

Consider again the LQR problem with cross term

$$\min_{\mathbf{u}} V = \frac{1}{2}\sum_{k=0}^{N-1}\left(x(k)'Qx(k) + u(k)'Ru(k) + 2x(k)'Mu(k)\right) + (1/2)x(N)'P_f x(N)$$

subject to

$$x^+ = Ax + Bu$$

and the three approaches of Exercise 6.1.

1. The method described in Section 6.1.1 in which you eliminate the state and solve the problem for the decision variable

$$\mathbf{u} = (u(0), u(1), \ldots, u(N-1))$$

2. The method described in Section 6.1.1 in which you do *not* eliminate the state and solve the problem for

$$\mathbf{z} = (u(0), x(1), u(1), x(2), \ldots, u(N-1), x(N))$$

3. The method of DP and the Riccati iteration to compute the closed-form solution for $u(k)$ and $x(k)$.

(a) You found that unstable A causes numerical problems in the first method using large horizons. So let's consider a fourth method. Reparameterize the input in terms of a state feedback gain via

$$u(k) = Kx(k) + v(k)$$

in which K is chosen so that $A + BK$ is a stable matrix. Consider the matrices in a transformed LQP

$$\min_{\mathbf{v}} V = \frac{1}{2}\sum_{k=0}^{N-1}\left(x(k)'\widetilde{Q}x(k) + v(k)'\widetilde{R}v(k) + 2x(k)'\widetilde{M}v(k)\right) + (1/2)x(N)'\widetilde{P}_f x(N)$$

subject to $x^+ = \widetilde{A}x + \widetilde{B}v$.

What are the matrices $\widetilde{A}, \widetilde{B}, \widetilde{Q}, \widetilde{P}_f, \widetilde{R}, \widetilde{M}$ such that the two problems give the same solution (state trajectory)?

(b) Solve the following problem using the first method and the fourth method and describe differences between the two solutions. Compare your results to the DP approach. Plot $x(k)$ and $u(k)$ versus k.

$$A = \begin{bmatrix} 27.8 & -82.6 & 34.6 \\ 25.6 & -76.8 & 32.4 \\ 40.6 & -122.0 & 51.9 \end{bmatrix} \qquad B = \begin{bmatrix} 0.527 & 0.548 \\ 0.613 & 0.530 \\ 1.06 & 0.828 \end{bmatrix} \qquad x(0) = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

Consider regulator tuning parameters and constraints

$$Q = P_f = I \qquad R = I \qquad M = 0 \qquad N = 50$$

Exercise 6.5: Recursively summing quadratic functions

Consider generalizing Example 1.1 to an N-term sum. Let the N-term sum of quadratic functions be defined as

$$V(N, x) = \frac{1}{2}\sum_{i=1}^{N}(x - x(i))'X_i(x - x(i))$$

in which $x, x(i) \in \mathbb{R}^n$ are real n-vectors and $X_i \in \mathbb{R}^{n\times n}$ are positive definite matrices.

(a) Show that $V(N, x)$ can be found recursively

$$V(N, x) = (1/2)(x - v(N))'H(N)(x - v(N)) + \text{constant}$$

in which $v(i)$ and $H(i)$ satisfy the recursion

$$H(i+1) = H_i + X_{i+1} \qquad v(i+1) = H^{-1}(i+1)\left(H_i v_i + X_{i+1}x(i+1)\right)$$
$$H_1 = X_1 \qquad v_1 = x_1$$

Notice the recursively defined $v(m)$ and $H(m)$ provide the solutions and the Hessian matrices of the sequence of optimization problems

$$\min_x V(m, x) \qquad 1 \le m \le N$$

(b) Check your answer by solving the equivalent, but larger dimensional, constrained least squares problem (see Exercise 1.16)

$$\min_z (z - z_0)' \widetilde{H} (z - z_0)$$

subject to

$$Dz = 0$$

in which $z, z_0 \in \mathbb{R}^{nN}$, $\widetilde{H} \in \mathbb{R}^{nN \times nN}$ is a block diagonal matrix, $D \in \mathbb{R}^{n(N-1) \times nN}$

$$z_0 = \begin{bmatrix} x(1) \\ \vdots \\ x(N-1) \\ x(N) \end{bmatrix} \qquad \widetilde{H} = \begin{bmatrix} X_1 & & & \\ & \ddots & & \\ & & X_{N-1} & \\ & & & X_N \end{bmatrix} \qquad D = \begin{bmatrix} I & -I & & \\ & \ddots & \ddots & \\ & & I & -I \end{bmatrix}$$

(c) Compare the size and number of matrix inverses required for the two approaches.

Exercise 6.6: Why call the Lyapunov stability *nonuniform*?

Consider the following linear system

$$\begin{aligned} w^+ &= Aw \qquad & w(0) &= Hx(0) \\ x &= Cw \end{aligned}$$

with solution $w(k) = A^k w(0) = A^k H x(0)$, $x(k) = CA^k H x(0)$. Notice that $x(0)$ completely determines both $w(k)$ and $x(k)$, $k \geq 0$. Also note that zero is a solution, i.e., $x(k) = 0, k \geq 0$ satisfies the model.

(a) Consider the following case

$$A = \rho \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \qquad H = \begin{bmatrix} 0 \\ -1 \end{bmatrix} \qquad C = \begin{bmatrix} 1 & -1 \end{bmatrix}$$

$$\rho = 0.925 \qquad \theta = \pi/4 \qquad x(0) = 1$$

Plot the solution $x(k)$. Does $x(k)$ converge to zero? Does $x(k)$ achieve zero exactly for finite $k > 0$?

(b) Is the zero solution $x(k) = 0$ Lyapunov stable? State your definition of Lyapunov stability, and prove your answer. Discuss how your answer is consistent with the special case considered above.

Exercise 6.7: Exponential stability of suboptimal MPC with unbounded feasible set

Consider again Lemma 6.5 when both $\mathbb{U}$ and $\mathcal{X}_N$ are unbounded. Show that the suboptimal MPC controller is exponentially stable on the following sets.

(a) Any sublevel set of $V(x, \mathbf{h}(x))$

(b) Any compact subset of $\mathcal{X}_N$

Exercise 6.8: A refinement to the warm start

Consider the following refinement to the warm start in the suboptimal MPC strategy. First add the requirement that the initialization strategy satisfies the following bound

$$\mathbf{h}(x) \le \bar{d}\,|x| \qquad x \in \mathcal{X}_N$$

in which $\bar{d} > 0$. Notice that all initializations considered in the chapter satisfy this requirement.

Then, at time k and state x, in addition to the shifted input sequence from time $k-1$, $\widetilde{\mathbf{u}}$, evaluate the initialization sequence applied to the current state, $\mathbf{u} = \mathbf{h}(x)$. Select whichever of these two input sequence has lower cost as the warm start for time k. Notice also that this refinement makes the constraint

$$|\mathbf{u}| \le d\,|x| \qquad x \in r\mathcal{B}$$

redundant, and it can be removed from the MPC optimization.

Prove that this refined suboptimal strategy is exponentially stabilizing on the set $\mathcal{X}_N$. Notice that with this refinement, we do not have to assume that $\mathcal{X}_N$ is bounded or that $\mathbb{U}$ is bounded.

Exercise 6.9: Global asymptotic stability and exponential convergence with mixed powers of the norm

Prove Lemma 6.6.

Hints: exponential convergence can be established as in standard exponential stability theorems. To establish Lyapunov stability, notice that $|x(0)| \le |(x(0), e(0))|$ and $|e(0)| \le |(x(0), e(0))|$ and that $(\cdot)^\alpha$ for $\alpha > 0$ is a $\mathcal{K}_\infty$ function.

Exercise 6.10: Decentralized control of Examples 6.9–6.11

Apply decentralized control to the systems in Examples 6.9–6.11. Which of these systems are closed-loop unstable with decentralized control? Compare this result to the result for noncooperative MPC.

Exercise 6.11: Cooperative control of Examples 6.9–6.11

Apply cooperative MPC to the systems in Examples 6.9–6.11. Are any of these systems closed-loop unstable? Compare the closed-loop eigenvalues of converged cooperative control to centralized MPC, and discuss any differences.

Exercise 6.12: Adding norms

Establish the following result used in the proof of Lemma 6.14. Given that $w \in \mathbb{R}^m$, $e \in \mathbb{R}^n$

$$\frac{1}{\sqrt{2}}(|w| + |e|) \le |(w, e)| \le |w| + |e| \qquad \forall w, e$$

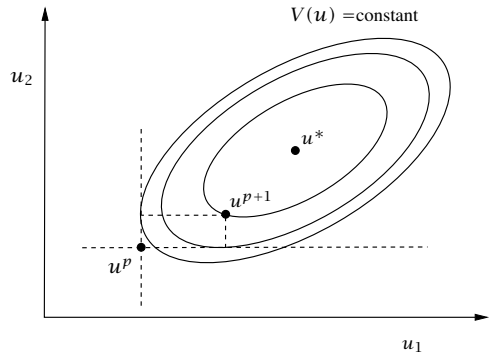


Figure 6.11: Optimizing a quadratic function in one set of variables at a time.

Exercise 6.13: Padding matrices

Given a vector $\mathbf{z}$ and subvector $\mathbf{u}$

$$\mathbf{z} = \begin{bmatrix} u(0) \\ x(1) \\ u(1) \\ x(2) \\ \vdots \\ u(N-1) \\ x(N) \end{bmatrix} \qquad \mathbf{u} = \begin{bmatrix} u(0) \\ u(1) \\ \vdots \\ u(N-1) \end{bmatrix} \qquad x \in \mathbb{R}^n \qquad u \in \mathbb{R}^m$$

and quadratic function of $\mathbf{u}$

$$(1/2)\mathbf{u}'H\mathbf{u} + h'\mathbf{u}$$

Find the corresponding quadratic function of $\mathbf{z}$ so that

$$(1/2)\mathbf{z}'H_z\mathbf{z} + h_z'\mathbf{z} = (1/2)\mathbf{u}'H\mathbf{u} + h'\mathbf{u} \qquad \forall \mathbf{z}, \mathbf{u}$$

Hint: first find the padding matrix E such that $\mathbf{u} = E\mathbf{z}$.

Exercise 6.14: A matrix inverse

Compute the four partitioned elements in the two-player feedback gain $(I - L)^{-1}\overline{K}$

$$\mathbf{u}^\infty = (I - L)^{-1}\overline{K}x(0) \qquad |\operatorname{eig}(L)| < 1$$

in which

$$(I - L)^{-1}\overline{K} = \begin{bmatrix} I & -L_1 \\ -L_2 & I \end{bmatrix}^{-1} \begin{bmatrix} K_1 & 0 \\ 0 & K_2 \end{bmatrix}$$

Exercise 6.15: Optimizing one variable at a time

Consider the positive definite quadratic function partitioned into two sets of variables

$$V(u) = (1/2)u'Hu + c'u + d$$

$$V(u_1,u_2) = (1/2)\begin{bmatrix}u_1' & u_2'\end{bmatrix}\begin{bmatrix}H_{11} & H_{12}\\ H_{21} & H_{22}\end{bmatrix}\begin{bmatrix}u_1\\ u_2\end{bmatrix} + \begin{bmatrix}c_1' & c_2'\end{bmatrix}\begin{bmatrix}u_1\\ u_2\end{bmatrix} + d$$

in which $H > 0$. Imagine we wish to optimize this function by first optimizing over the u_1 variables holding u_2 fixed and then optimizing over the u_2 variables holding u_1 fixed as shown in Figure 6.11. Let's see if this procedure, while not necessarily efficient, is guaranteed to converge to the optimum.

(a) Given an initial point (u_1^p, u_2^p), show that the next iteration is

$$\begin{aligned} u_1^{p+1} &= -H_{11}^{-1}\left(H_{12}u_2^p + c_1\right)\\ u_2^{p+1} &= -H_{22}^{-1}\left(H_{21}u_1^p + c_2\right) \end{aligned} \tag{6.39}$$

The procedure can be summarized as

$$u^{p+1} = Au^p + b \tag{6.40}$$

in which the iteration matrix A and constant b are given by

$$A = \begin{bmatrix}0 & -H_{11}^{-1}H_{12}\\ -H_{22}^{-1}H_{21} & 0\end{bmatrix} \qquad b = \begin{bmatrix}-H_{11}^{-1}c_1\\ -H_{22}^{-1}c_2\end{bmatrix} \tag{6.41}$$

(b) Establish that the optimization procedure converges by showing the iteration matrix is stable

$$|\operatorname{eig}(A)| < 1$$

(c) Given that the iteration converges, show that it produces the same solution as

$$u^* = -H^{-1}c$$

Exercise 6.16: Monotonically decreasing cost

Consider again the iteration defined in Exercise 6.15.

(a) Prove that the cost function is monotonically decreasing when optimizing one variable at a time

$$V(u^{p+1}) < V(u^p) \qquad \forall\, u^p \neq -H^{-1}c$$

(b) Show that the following expression gives the size of the decrease

$$V(u^{p+1}) - V(u^p) = -(1/2)(u^p - u^*)'P(u^p - u^*)$$

in which

$$P = HD^{-1}\widetilde{H}D^{-1}H \qquad \widetilde{H} = D - N \qquad D = \begin{bmatrix}H_{11} & 0\\ 0 & H_{22}\end{bmatrix} \qquad N = \begin{bmatrix}0 & H_{12}\\ H_{21} & 0\end{bmatrix}$$

and $u^* = -H^{-1}c$ is the optimum.

Hint: to simplify the algebra, first change coordinates and move the origin of the coordinate system to u^*.

Exercise 6.17: One variable at a time with convex step

Consider Exercise 6.15 but with the convex step for the iteration

$$\begin{bmatrix} u_1^{p+1} \\ u_2^{p+1} \end{bmatrix} = w_1 \begin{bmatrix} u_1^0(u_2^p) \\ u_2^p \end{bmatrix} + w_2 \begin{bmatrix} u_1^p \\ u_2^0(u_1^p) \end{bmatrix} \qquad 0 \le w_1, w_2 \quad w_1 + w_2 = 1$$

(a) Show that the iteration for the convex step is also of the form

$$u^{p+1} = Au^p + b$$

and the A matrix and b vector for this case are

$$A = \begin{bmatrix} w_2 I & -w_1 H_{11}^{-1} H_{12} \\ -w_2 H_{22}^{-1} H_{21} & w_1 I \end{bmatrix} \qquad b = \begin{bmatrix} -w_1 H_{11}^{-1} & \\ & -w_2 H_{22}^{-1} \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \end{bmatrix}$$

(b) Show that A is stable.

(c) Show that this iteration also converges to $u^* = -H^{-1}c$.

Exercise 6.18: Monotonically decreasing cost with convex step

Consider again the problem of optimizing one variable at a time with the convex step given in Exercise 6.17.

(a) Prove that the cost function is monotonically decreasing

$$V(u^{p+1}) < V(u^p) \qquad \forall u^p \neq -H^{-1}c$$

(b) Show that the following expression gives the size of the decrease

$$V(u^{p+1}) - V(u^p) = -(1/2)(u^p - u^*)' P (u^p - u^*)$$

in which

$$P = HD^{-1}\tilde{H}D^{-1}H \qquad \tilde{H} = D - N$$

$$D = \begin{bmatrix} w_1^{-1} H_{11} & 0 \\ 0 & w_2^{-1} H_{22} \end{bmatrix} \qquad N = \begin{bmatrix} -w_1^{-1} w_2 H_{11} & H_{12} \\ H_{21} & -w_1 w_2^{-1} H_{22} \end{bmatrix}$$

and $u^* = -H^{-1}c$ is the optimum.

Hint: to simplify the algebra, first change coordinates and move the origin of the coordinate system to u^*.

Exercise 6.19: Splitting more than once

Consider the generalization of Exercise 6.15 in which we repeatedly decompose a problem into one-variable-at-a-time optimizations. For a three-variable problem we have the three optimizations

$$u_1^{p+1} = \arg\min_{u_1} V(u_1, u_2^p, u_3^p)$$

$$u_2^{p+1} = \arg\min_{u_2} V(u_1^p, u_2, u_3^p) \qquad u_3^{p+1} = \arg\min_{u_3} V(u_1^p, u_2^p, u_3)$$

Is it true that

$$V(u_1^{p+1}, u_2^{p+1}, u_3^{p+1}) \le V(u_1^p, u_2^p, u_3^p)$$

Hint: you may wish to consider the following example, $V(u) = (1/2)u'Hu + c'u$, in which

$$H = \begin{bmatrix} 2 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 2 \end{bmatrix} \qquad c = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} \qquad u^p = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}$$

Exercise 6.20: Time-varying controller iterations

We let $p_k \geq 0$ be a time-varying integer-valued index representing the iterations applied in the controller at time k.

$$\begin{aligned}
x_1(k+1) &= A_1 x_1(k) + \overline{B}_{11} u_1(0;k) + \overline{B}_{12} u_2(0;k) \\
x_2(k+1) &= A_2 x_2(k) + \overline{B}_{21} u_1(0;k) + \overline{B}_{22} u_2(0;k) \\
\mathbf{u}_1(k+1) &= g_1^{p_k}(x_1(k), x_2(k), \mathbf{u}_1(k), \mathbf{u}_2(k)) \\
\mathbf{u}_2(k+1) &= g_2^{p_k}(x_1(k), x_2(k), \mathbf{u}_1(k), \mathbf{u}_2(k))
\end{aligned}$$

Notice the system evolution is time-varying even though the models are time invariant because we allow a time-varying sequence of controller iterations.

Show that cooperative MPC is exponentially stabilizing for any $p_k \geq 0$ sequence.

Exercise 6.21: Stable interaction models

In some industrial applications it is preferable to partition the plant so that there are no unstable connections between subsystems. Any inputs u_j that have unstable connections to outputs y_i should be included in the ith subsystem inputs. Allowing an unstable connection between two subsystems may not be robust to faults and other kinds of system failures.[5] To implement this design idea in the two-player case, we replace Assumption 6.13 (b) with the following

Modified Assumption 6.13 (Constrained two-player game)**.**

(b) The interaction models $A_{ij}, i \neq j$ are stable.

Prove that Modified Assumption 6.13 (b) implies Assumption 6.13 (b). It may be helpful to first prove the following lemma.

Lemma 6.23 (Local detectability)**.** *Given partitioned system matrices*

$$\boldsymbol{A} = \begin{bmatrix} A & 0 \\ 0 & A_s \end{bmatrix} \qquad \boldsymbol{C} = \begin{bmatrix} C & C_s \end{bmatrix}$$

in which A_s is stable, the system $(\boldsymbol{A}, \boldsymbol{C})$ is detectable if and only if the system (A, C) is detectable.

Hint: use the Hautus lemma as the test for detectability.

Next show that this lemma and Modified Assumption 6.13 (b) establishes the distributed detectability assumption, Assumption 6.13 (b).

Exercise 6.22: Norm constraints as linear inequalities

Consider the quadratic program (QP) in decision variable u with parameter x

$$\begin{aligned}
&\min_u (1/2) u' H u + x' D u \\
&\text{s.t. } Eu \leq Fx
\end{aligned}$$

[5] We are not considering the common instability of base-level inventory management in this discussion. It is assumed that level control in storage tanks (integrators) is maintained at all times with simple, local level controllers. The internal unit flowrates dedicated for inventory management are not considered available inputs in the MPC problem.

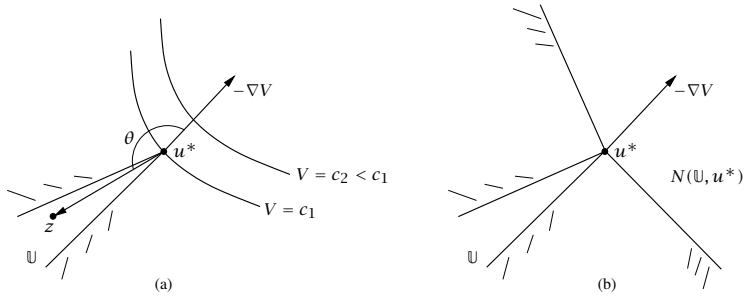


Figure 6.12: (a) Optimality of u^* means the angle between $-\nabla V$ and any point z in the feasible region must be greater than 90° and less than 270°. (b) The same result restated: u^* is optimal if and only if the negative gradient is in the normal cone to the feasible region at u^*, $-\nabla V|_{u^*} \in N(\mathbb{U}, u^*)$.

in which $u \in \mathbb{R}^m$, $x \in \mathbb{R}^n$, and $H > 0$. The parameter x appears linearly (affinely) in the cost function and constraints. Assume that we wish to add a norm constraint of the following form

$$|u|_\alpha \leq c\,|x|_\alpha \qquad \alpha = 2, \infty$$

(a) If we use the infinity norm, show that this problem can be posed as an equivalent QP with additional decision variables, and the cost function and constraints remain linear (affine) in parameter x. How many decision variables and constraints are added to the problem?

(b) If we use the two norm, show that this problem can be approximated by a QP whose solution does satisfy the constraints, but the solution may be suboptimal compared to the original problem.

Exercise 6.23: Steady-state noncooperative game

Consider again the steady-state target problem for the system given in Example 6.12.

(a) Resolve the problem for the choice of convex step parameters $w_1 = 0.2$, $w_2 = 0.8$. Does the iteration for noncooperative control converge? Plot the iterations for the noncooperative and cooperative cases.

(b) Repeat for the convex step $w_1 = 0.8, w_2 = 0.2$. Are the results identical to the previous part? If not, discuss any differences.

(c) For what choices of w_1, w_2 does the target iteration converge using noncooperative control for the target calculation?

Exercise 6.24: Optimality conditions for constrained optimization

Consider the convex quadratic optimization problem

$$\min_u V(u) \qquad \text{subject to} \quad u \in \mathbb{U}$$

in which V is a convex quadratic function and $\mathbb{U}$ is a convex set. Show that u^* is an optimal solution if and only if

$$\langle z - u^*, -\nabla V|_{u^*} \rangle \quad \leq 0 \qquad \forall z \in \mathbb{U} \tag{6.42}$$

Figure 6.12(a) depicts this condition for $u \in \mathbb{R}^2$. This condition motivates defining the normal cone (Rockafellar, 1970) to $\mathbb{U}$ at u^* as follows

$$N(\mathbb{U}, u^*) = \{y \mid \langle z - u^*, y - u^* \rangle \leq 0 \quad \forall z \in \mathbb{U}\}$$

The optimality condition can be stated equivalently as u^* is an optimal point if and only if the negative gradient is in the normal cone to $\mathbb{U}$ at u^*

$$-\nabla V|_{u^*} \in N(\mathbb{U}, u^*)$$

This condition and the normal cone are depicted in Figure 6.12(b).

Exercise 6.25: Partitioned optimality conditions with constraints

Consider a partitioned version of the constrained optimization problem of Exercise 6.24 with uncoupled constraints

$$\min_{u_1, u_2} V(u_1, u_2) \qquad \text{subject to} \quad u_1 \in \mathbb{U}_1 \qquad u_2 \in \mathbb{U}_2$$

in which V is a quadratic function and $\mathbb{U}_1$ and $\mathbb{U}_2$ are convex and nonempty.

(a) Show that (u_1^*, u_2^*) is an optimal solution if and only if

$$\begin{aligned} \langle z_1 - u_1^*, -\nabla_{u_1} V|_{(u_1^*, u_2^*)} \rangle \quad &\leq 0 \qquad \forall z_1 \in \mathbb{U}_1 \\ \langle z_2 - u_2^*, -\nabla_{u_2} V|_{(u_1^*, u_2^*)} \rangle \quad &\leq 0 \qquad \forall z_2 \in \mathbb{U}_2 \end{aligned} \tag{6.43}$$

(b) Extend the optimality conditions to cover the case

$$\min_{u_1, \ldots, u_M} V(u_1, \ldots, u_M) \qquad \text{subject to} \quad u_j \in \mathbb{U}_j \quad j = 1, \ldots, M$$

in which V is a quadratic function and the $\mathbb{U}_j$ are convex and nonempty.

Exercise 6.26: Constrained optimization of M variables

Consider an optimization problem with M variables and uncoupled constraints

$$\min_{u_1, u_2, \ldots, u_M} V(u_1, u_2, \ldots, u_M) \qquad \text{subject to} \quad u_l \in \mathbb{U}_j \quad j = 1, \ldots, M$$

in which V is a strictly convex function. Assume that the feasible region is convex and nonempty and denote the unique optimal solution as $(u_1^*, u_2^*, \ldots, u_M^*)$ having cost $V^* = V(u_1^*, \ldots, u_M^*)$. Denote the M one-variable-at-a-time optimization problems at iteration k

$$z_j^{p+1} = \arg\min_{u_j} V(u_1^p, \ldots, u_j, \ldots, u_M^p) \qquad \text{subject to } u_j \in \mathbb{U}_j$$

Then define the next iterate to be the following convex combination of the previous and new points

$$\begin{aligned} u_j^{p+1} &= \alpha_j^p z_j^{p+1} + (1 - \alpha_j^p) u_j^p \qquad j = 1, \ldots, M \\ \varepsilon \leq \alpha_j^p &< 1 \qquad 0 < \varepsilon \qquad j = 1, \ldots, M, \quad p \geq 1 \end{aligned}$$

$$\sum_{j=1}^{M} \alpha_j^p = 1, \qquad p \geq 1$$

Prove the following results.

(a) Starting with any feasible point, $(u_1^0, u_2^0, \ldots, u_M^0)$, the iterations $(u_1^p, u_2^p, \ldots, u_M^p)$ are feasible for $p \geq 1$.

(b) The objective function decreases monotonically from any feasible initial point

$$V(u_1^{p+1}, \ldots, u_M^{p+1}) \leq V(u_1^p, \ldots, u_M^p) \qquad \forall\, u_l^0 \in \mathbb{U}_j, j = 1, \ldots, M, \quad p \geq 1$$

(c) The cost sequence $V(u_1^p, u_2^p, \ldots, u_M^p)$ converges to the optimal cost V^* from any feasible initial point.

(d) The sequence $(u_1^p, u_2^p, \ldots, u_M^p)$ converges to the optimal solution $(u_1^*, u_2^*, \ldots, u_M^*)$ from any feasible initial point.

Exercise 6.27: The constrained two-variable special case

Consider the special case of Exercise 6.26 with $M = 2$

$$\min_{u_1, u_2} V(u_1, u_2) \qquad \text{subject to} \quad u_1 \in \mathbb{U}_1 \qquad u_2 \in \mathbb{U}_2$$

in which V is a strictly positive quadratic function. Assume that the feasible region is convex and nonempty and denote the unique optimal solution as (u_1^*, u_2^*) having cost $V^* = V(u_1^*, u_2^*)$. Consider the two one-variable-at-a-time optimization problems at iteration k

$$u_1^{p+1} = \arg\min_{u_1} V(u_1, u_2^p) \qquad u_2^{p+1} = \arg\min_{u_2} V(u_1^p, u_2)$$
$$\text{subject to } u_1 \in \mathbb{U}_1 \qquad \text{subject to } u_2 \in \mathbb{U}_2$$

We know from Exercise 6.15 that taking the full step in the unconstrained problem with $M = 2$ achieves a cost decrease. We know from Exercise 6.19 that taking the full step for an unconstrained problem with $M \geq 3$ does *not* provide a cost decrease in general. We know from Exercise 6.26 that taking a reduced step in the constrained problem for all M achieves a cost decrease. That leaves open the case of a full step for a constrained problem with $M = 2$.

Does the full step in the constrained case for $M = 2$ guarantee a cost decrease? If so, prove it. If not, provide a counterexample.

Exercise 6.28: Subsystem stability constraints

Show that the following uncoupled subsystem constraints imply an overall system constraint of the same type. The first is suitable for asymptotic stability and the second for exponential stability.

(a) Given $r_1, r_2 > 0$, and functions γ_1 and γ_2 of class $\mathcal{K}$, assume the following constraints are satisfied

$$|\mathbf{u}_1| \leq \gamma_1(|x_1|) \quad x_1 \in r_1\mathcal{B}$$
$$|\mathbf{u}_2| \leq \gamma_2(|x_2|) \quad x_2 \in r_2\mathcal{B}$$

Show that there exists $r > 0$ and function γ of class $\mathcal{K}$ such that

$$|(\mathbf{u}_1, \mathbf{u}_2)| \leq \gamma(|(x_1, x_2)|) \qquad (x_1, x_2) \in r\mathcal{B} \tag{6.44}$$

(b) Given $r_1, r_2 > 0$, and constants $c_1, c_2, \sigma_1, \sigma_2 > 0$, assume the following constraints are satisfied

$$|\mathbf{u}_1| \leq c_1 |x_1|^{\sigma_1} \quad x_1 \in r_1 \mathcal{B}$$
$$|\mathbf{u}_2| \leq c_2 |x_2|^{\sigma_2} \quad x_2 \in r_2 \mathcal{B}$$

Show that there exists $r > 0$ and function $c, \sigma > 0$ such that

$$|(\mathbf{u}_1, \mathbf{u}_2)| \leq c |(x_1, x_2)|^{\sigma} \qquad (x_1, x_2) \in r\mathcal{B} \tag{6.45}$$

Exercise 6.29: Distributed disturbance detectability

Prove Lemma 6.16.

Hint: use the Hautus lemma as the test for detectability.

Exercise 6.30: Distributed target problem and uncoupled constraints

Player one's distributed target problem in the two-player game is given in (6.28)

$$\min_{x_{11s}, x_{21s}, u_{1s}} (1/2) \begin{bmatrix} H_1 y_{1s} - z_{1sp} \\ H_2 y_{2s} - z_{2sp} \end{bmatrix}' \begin{bmatrix} T_{1s} & \\ & T_{2s} \end{bmatrix} \begin{bmatrix} H_1 y_{1s} - z_{1sp} \\ H_2 y_{2s} - z_{2sp} \end{bmatrix}$$

subject to

$$\begin{bmatrix} I - A_1 & & -\overline{B}_{11} & -\overline{B}_{12} \\ & I - A_2 & -\overline{B}_{21} & -\overline{B}_{22} \end{bmatrix} \begin{bmatrix} x_{1s} \\ x_{2s} \\ u_{1s} \\ u_{2s} \end{bmatrix} = \begin{bmatrix} B_{1d} \hat{d}_1(k) \\ B_{2d} \hat{d}_2(k) \end{bmatrix}$$

$$E_1 u_{1s} \leq e_1$$

Show that the constraints can be expressed so that the target problem constraints are uncoupled.

Bibliography

T. Başar and G. J. Olsder. *Dynamic Noncooperative Game Theory*. SIAM, Philadelphia, 1999.

D. P. Bertsekas. *Nonlinear Programming*. Athena Scientific, Belmont, MA, second edition, 1999.

D. P. Bertsekas and J. N. Tsitsiklis. *Parallel and Distributed Computation*. Athena Scientific, Belmont, Massachusetts, 1997.

A. E. Bryson and Y. Ho. *Applied Optimal Control*. Hemisphere Publishing, New York, 1975.

H. Cui and E. W. Jacobsen. Performance limitations in decentralized control. *J. Proc. Cont.*, 12:485–494, 2002.

W. B. Dunbar. Distributed receding horizon control of dynamically coupled nonlinear systems. *IEEE Trans. Auto. Cont.*, 52(7):1249–1263, 2007.

W. B. Dunbar and R. M. Murray. Distributed receding horizon control with application to multi-vehicle formation stabilization. *Automatica*, 42(4):549–558, 2006.

G. H. Golub and C. F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, Baltimore, Maryland, third edition, 1996.

D. Jia and B. H. Krogh. Min-max feedback model predictive control for distributed control with communication. In *Proceedings of the American Control Conference*, pages 4507–4512, Anchorage,Alaska, May 2002.

T. Larsson and S. Skogestad. Plantwide control- A review and a new design procedure. *Mod. Ident. Control*, 21(4):209–240, 2000.

J. Lunze. *Feedback Control of Large Scale Systems*. Prentice-Hall, London, U.K., 1992.

J. M. Maestre and R. R. Negenborn. *Distributed Model Predictive Control Made Easy*. Springer Netherlands, 2014.

M. Mesarović, D. Macko, and Y. Takahara. *Theory of hierarchical, multilevel systems*. Academic Press, New York, 1970.

N. Motee and B. Sayyar-Rodsari. Optimal partitioning in distributed model predictive control. In *Proceedings of the American Control Conference*, pages 5300–5305, Denver,Colorado, June 2003.

J. Nash. Noncooperative games. *Ann. Math.*, 54:286–295, 1951.

J. B. Rawlings and B. T. Stewart. Coordinating multiple optimization-based controllers: New opportunities and challenges. *J. Proc. Cont.*, 18:839–845, 2008.

R. T. Rockafellar. *Convex Analysis*. Princeton University Press, Princeton, N.J., 1970.

N. R. Sandell Jr., P. Varaiya, M. Athans, and M. Safonov. Survey of decentralized control methods for large scale systems. *IEEE Trans. Auto. Cont.*, 23(2):108–128, 1978.

R. Scattolini. Architectures for distributed and hierarchical model predictive control - a review. *J. Proc. Cont.*, 19(5):723–731, May 2009.

P. O. M. Scokaert, D. Q. Mayne, and J. B. Rawlings. Suboptimal model predictive control (feasibility implies stability). *IEEE Trans. Auto. Cont.*, 44(3):648–654, March 1999.

D. D. Šiljak. *Decentralized Control of Complex Systems*. Academic Press, London, 1991.

B. T. Stewart, A. N. Venkat, J. B. Rawlings, S. J. Wright, and G. Pannocchia. Cooperative distributed model predictive control. *Sys. Cont. Let.*, 59:460–469, 2010.

B. T. Stewart, S. J. Wright, and J. B. Rawlings. Cooperative distributed model predictive control for nonlinear systems. *J. Proc. Cont.*, 21(5):698–704, June 2011.

A. N. Venkat. *Distributed Model Predictive Control: Theory and Applications*. PhD thesis, University of Wisconsin–Madison, October 2006.

A. N. Venkat, J. B. Rawlings, and S. J. Wright. Stability and optimality of distributed, linear MPC. Part 1: state feedback. Technical Report 2006–03, TWMCC, Department of Chemical and Biological Engineering, University of Wisconsin–Madison (Available at http://jbrwww.che.wisc.edu/tech-reports.html), October 2006a.

A. N. Venkat, J. B. Rawlings, and S. J. Wright. Stability and optimality of distributed, linear MPC. Part 2: output feedback. Technical Report 2006–04, TWMCC, Department of Chemical and Biological Engineering, University of Wisconsin–Madison (Available at http://jbrwww.che.wisc.edu/tech-reports.html), October 2006b.

A. N. Venkat, J. B. Rawlings, and S. J. Wright. Distributed model predictive control of large-scale systems. In *Assessment and Future Directions of Nonlinear Model Predictive Control*, pages 591–605. Springer, 2007.

M. Vidyasagar. *Nonlinear Systems Analysis*. Prentice-Hall, Inc., Englewood Cliffs, New Jersey, second edition, 1993.

J. von Neumann and O. Morgenstern. *Theory of Games and Economic Behavior*. Princeton University Press, Princeton and Oxford, 1944.

S. J. Wright. Applying new optimization algorithms to model predictive control. In J. C. Kantor, C. E. García, and B. Carnahan, editors, *Chemical Process Control–V*, pages 147–155. CACHE, AIChE, 1997.

7

Explicit Control Laws for Constrained Linear Systems

7.1 Introduction

In preceding chapters we show how model predictive control (MPC) can be derived for a variety of control problems with constraints. It is interesting to recall the major motivation for MPC; solution of a *feedback* optimal control problem for constrained and/or nonlinear systems to obtain a stabilizing control *law* is often prohibitively difficult. MPC sidesteps the problem of determining a control *law* $\kappa(\cdot)$ by determining, instead, at each state x encountered, a control *action* $u = \kappa(x)$ by solving a mathematical programming problem. This procedure, if repeated at *every* state x, yields an implicit control *law* $\kappa(\cdot)$ that solves the original feedback problem. In many cases, determining an explicit control law is impractical while solving a mathematical programming problem online for a given state is possible; this fact has led to the wide-scale adoption of MPC in the chemical process industry.

Some of the control problems for which MPC has been extensively used, however, have recently been shown to be amenable to analysis, at least for relatively simple systems. One such problem is control of linear discrete time systems with polytopic constraints, for which determination of a stabilizing control law was thought in the past to be prohibitively difficult. It has been shown that it is possible, in principle, to determine a stabilizing control law for some of these control problems. This result is often referred to as *explicit MPC* because it yields an explicit control law in contrast to MPC that yields a control action for each encountered state, thereby *implicitly* defining a control law. There are two objections to this terminology. First, determination of control laws for a wide variety of control problems has been the prime concern of control theory since its birth and certainly before the advent of MPC,

an important tool in this endeavor being dynamic programming (DP). The new result shows that classical control-theoretic tools, such as DP, can be successfully applied to a wider range of problems than was previously thought possible. MPC is a useful method for implementing an implicit control law that can, in principle, be explicitly determined using control-theoretic tools.

Second, some authors using this terminology have, perhaps inadvertently, implied that these results can be employed in place of conventional MPC. This is far from the truth, since only relatively simple problems, far simpler than those routinely solved in MPC applications, can be solved. That said, the results may be useful in applications where models with low state dimension are sufficiently accurate and where it is important that the control be rapidly computed. A previously determined control law may yield the control action more rapidly than solving an optimal control problem. Potential applications include vehicle control.

In the next section we give a few simple examples of parametric programming. In subsequent sections we show how the solutions to parametric linear and quadratic programs may be obtained, and also show how these solutions may be used to solve optimal control problems when the system is linear, the cost quadratic or affine, and the constraints polyhedral.

7.2 Parametric Programming

A conventional optimization problem has the form $V^0 = \min_u \{V(u) \mid u \in \mathcal{U}\}$ where u is the "decision" variable, $V(u)$ is the cost to be minimized, and $\mathcal{U}$ is the constraint set. The solution to a conventional optimization is a *point* or *set* in $\mathcal{U}$; the value V^0 of the problem satisfies $V^0 = V(u^0)$ where u^0 is a minimizer. A simple example of such a problem is $V^0 = \min_u \{a + bu + (1/2)cu^2 \mid u \in [-1, 1]\}$ where the solution is required for only *one* value of the parameters a, b and c. The solution to this problem $u^0 = -b/c$ if $|b/c| \leq 1$, $u^0 = -1$ if $b/c \geq 1$ and $u^0 = 1$ if $b/c \leq -1$. This may be written more compactly as $u^0 = -\text{sat}(b/c)$ where $\text{sat}(\cdot)$ is the saturation function. The corresponding value is $V^0 = a - b^2/2c$ if $|b/c| \leq 1$, $V^0 = a - b + c^2/2$ if $b/c \geq 1$ and $V^0 = a + b + c^2/2$ if $b/c \leq -1$.

A parametric programming problem $\mathbb{P}(x)$ on the other hand, takes the form $V^0(x) = \min_u \{V(x, u) \mid u \in \mathcal{U}(x)\}$ where x is a *parameter* so that the optimization problem, and its solution, depend on the

value of the parameter. Hence, the solution to a parametric programming problem $\mathbb{P}(x)$ is not a point or set but a *function* $x \mapsto u^0(x)$ that may be set valued; similarly the value of the problem is a function $x \mapsto V^0(x)$. At each x, the minimizer $u^0(x)$ may be a point or a set. Optimal control problems often take this form, with x being the state, and u, in open-loop discrete time optimal control, being a control sequence; $u^0(x)$, the optimal control sequence, is a function of the initial state. In state feedback optimal control, necessary when uncertainty is present, DP is employed yielding a sequence of parametric optimization problems in each of which x is the state and u a control action; see Chapter 2. The programming problem in the first paragraph of this section may be regarded as a parametric programming problem with the parameter $x := (a, b, c)$, $V(x, u) := (x_1 + x_2 u + (1/2)x_3 u^2/2)$ and $\mathcal{U}(x) := [-1, 1]$; $\mathcal{U}(x)$, in this example, does not depend on x. The solution to this problem yields the functions $u^0(\cdot)$ and $V^0(\cdot)$ defined by $u^0(x) = -\text{sat}(x_2/x_3)$ and $V^0(x) = V(x, u^0(x)) = x_1 + x_2 u^0(x) + (x_3/2)(u^0(x))^2$.

Because the minimizer and value of a parametric programming problem are *functions* rather than points or sets, we would not, in general, expect to be able to compute a solution. Surprisingly, parametric programs may be solved when the cost function $V(\cdot)$ is affine ($V(x, u) = a + b'x + c'u$) or quadratic ($V(x, u) = (1/2)x'Qx + x'Su + (1/2)u'Ru$) and $\mathcal{U}(x)$ is defined by a set of affine inequalities: $\mathcal{U}(x) = \{u \mid Mu \leq Nx + p\}$. The parametric constraint $u \in \mathcal{U}(x)$ may be conveniently expressed as $(x, u) \in \mathbb{Z}$ where $\mathbb{Z}$ is a subset of (x, u)-space which we will take to be $\mathbb{R}^n \times \mathbb{R}^m$; for each x

$$\mathcal{U}(x) = \{u \mid (x, u) \in \mathbb{Z}\}$$

We assume that $x \in \mathbb{R}^n$ and $u \in \mathbb{R}^m$. Let $\mathcal{X} \subset \mathbb{R}^n$ be defined by

$$\mathcal{X} := \{x \mid \exists u \text{ such that } (x, u) \in \mathbb{Z}\} = \{x \mid \mathcal{U}(x) \neq \emptyset\}$$

The set $\mathcal{X}$ is the domain of $V^0(\cdot)$ and $u^0(\cdot)$ and is thus the set of points x for which a feasible solution of $\mathbb{P}(x)$ exists; it is the projection of $\mathbb{Z}$ (which is a set in (x, u)-space) onto x-space. See Figure 7.1, which illustrates $\mathbb{Z}$ and $\mathcal{U}(x)$ for the case when $\mathcal{U}(x) = \{u \mid Mu \leq Nx + p\}$; the set $\mathbb{Z}$ is thus defined by $\mathbb{Z} := \{(x, u) \mid Mu \leq Nx + p\}$. In this case, both $\mathbb{Z}$ and $\mathcal{U}(x)$ are polyhedral.

Before proceeding to consider parametric linear and quadratic programming, some simple examples may help the reader to appreciate the underlying ideas. Consider first a very simple parametric linear

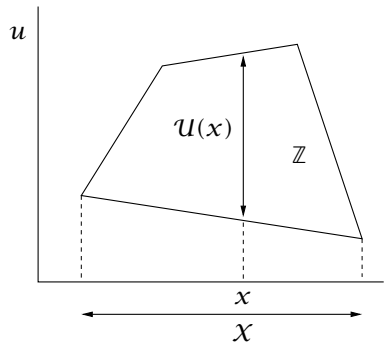


Figure 7.1: The sets $\mathbb{Z}$, $\mathcal{X}$, and $\mathcal{U}(x)$.

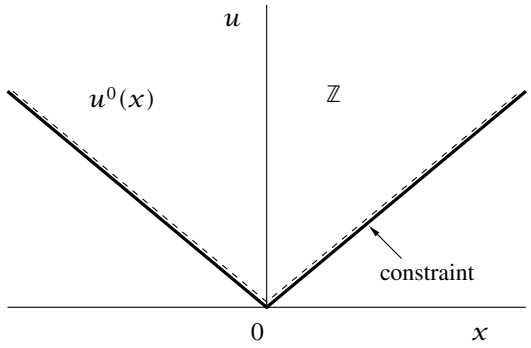


Figure 7.2: Parametric linear program.

program $\min_u\{V(x,u) \mid (x,u) \in \mathbb{Z}\}$ where $V(x,u) := x + u$ and $\mathbb{Z} := \{(x,u) \mid u + x \geq 0,\ u - x \geq 0\}$ so that $\mathcal{U}(x) = \{u \mid u \geq -x,\ u \geq x\}$. The problem is illustrated in Figure 7.2. The set $\mathbb{Z}$ is the region lying above the two solid lines $u = -x$ and $u = x$, and is convex. The gradient $\nabla_u V(x,u) = 1$ everywhere, so the solution, at each x, to the parametric program is the smallest u in $\mathcal{U}(x)$, i.e., the smallest u lying above the two lines $u = -x$ and $u = x$. Hence $u^0(x) = -x$ if $x \leq 0$ and $u^0(x) = x$ if $x \geq 0$, i.e., $u^0(x) = |x|$; the graph of $u^0(\cdot)$ is the dashed line in Figure 7.2. Both $u^0(\cdot)$ and $V^0(\cdot)$, in which $V^0(x) = x + u^0(x)$, are *piecewise affine*, being affine in each of the two regions $X_1 := \{x \mid x \leq 0\}$ and $X_2 := \{x \mid x \geq 0\}$.

Next consider an unconstrained parametric quadratic program (QP) $\min_u V(x,u)$ where $V(x,u) := (1/2)(x-u)^2 + u^2/2$. The problem is

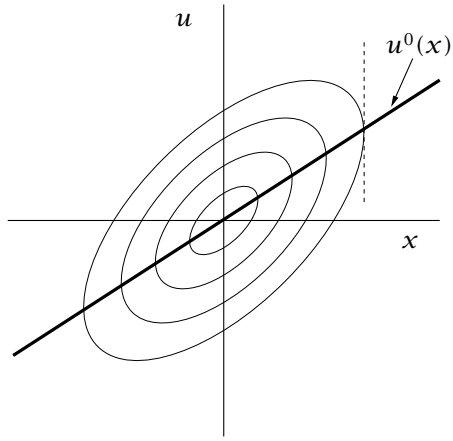


Figure 7.3: Unconstrained parametric quadratic program.

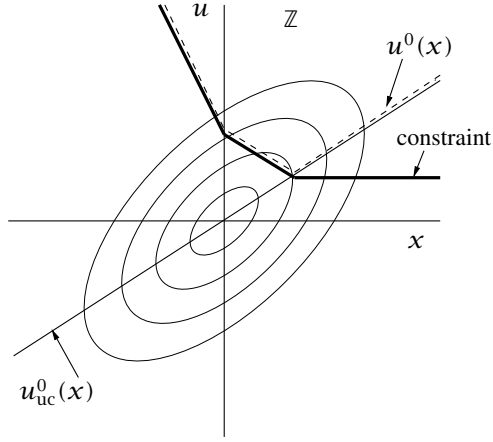


Figure 7.4: Parametric quadratic program.

illustrated in Figure 7.3. For each $x \in \mathbb{R}$, $\nabla_u V(x,u) = -x + 2u$ and $\nabla_{uu} V(x,u) = 2$ so that $u^0(x) = x/2$ and $V^0(x) = x^2/4$. Hence $u^0(\cdot)$ is affine and $V^0(\cdot)$ is quadratic in $\mathbb{R}$.

We now add the constraint set $\mathbb{Z} := \{(x,u) \mid u \geq 1,\ u + x/2 \geq 2,\ u + x \geq 2\}$; see Figure 7.4. The solution is defined on three regions, $X_1 := (-\infty, 0]$, $X_2 := [0,2]$, and $X_3 := [2,\infty)$. From the preceding example, the unconstrained minimum is achieved at $u^0_{\text{uc}}(x) = x/2$ shown by the solid straight line in Figure 7.4. Since $\nabla_u V(x,u) = -x + 2u$, $\nabla_u V(x,$

$u) > 0$ for all $u > u^0_{uc}(x) = x/2$. Hence, in X_1, $u^0(x)$ lies on the boundary of $\mathbb{Z}$ and satisfies $u^0(x) = 2 - x$. Similarly, in X_2, $u^0(x)$ lies on the boundary of $\mathbb{Z}$ and satisfies $u^0(x) = 2 - x/2$. Finally, in X_3, $u^0(x) = u^0_{uc}(x) = x/2$, the unconstrained minimizer, and lies in the interior of $\mathbb{Z}$ for $x > 1$. The third constraint $u \geq 2 - x$ is active in X_1, the second constraint $u \geq 2 - x/2$ is active in X_2, while no constraints are active in X_3. Hence the minimizer $u^0(\cdot)$ is piecewise affine, being affine in each of the regions X_1, X_2 and X_3. Since $V^0(x) = (1/2)\left|x - u^0(x)\right|^2 + u^0(x)^2/2$, the value function $V^0(\cdot)$ is piecewise quadratic, being quadratic in each of the regions X_1, X_2 and X_3.

We require, in the sequel, the following definitions.

Definition 7.1 (Polytopic (polyhedral) partition)**.** A set $\mathcal{P} = \{\mathbb{Z}_i \mid i \in \mathcal{I}\}$, for some index set $\mathcal{I}$, is called a polytopic (polyhedral) partition of the polytopic (polyhedral) set $\mathbb{Z}$ if $\mathbb{Z} = \cup_{i\in\mathcal{I}}\mathbb{Z}_i$ and the sets $\mathbb{Z}_i$, $i \in \mathcal{I}$, are polytopes (polyhedrons) with nonempty interiors (relative to $\mathbb{Z}$)[1] that are nonintersecting: $\text{int}(\mathbb{Z}_i) \cap \text{int}(\mathbb{Z}_j) = \emptyset$ if $i \neq j$.

Definition 7.2 (Piecewise affine function)**.** A function $f : \mathbb{Z} \to \mathbb{R}^m$ is said to be piecewise affine on a polytopic (polyhedral) partition $\mathcal{P} = \{\mathbb{Z}_i \mid i \in \mathcal{I}\}$ if it satisfies, for some K_i, k_i, $i \in \mathcal{I}$, $f(x) = K_i x + k_i$ for all $x \in \mathbb{Z}_i$, all $i \in \mathcal{I}$. Similarly, a function $f : \mathbb{Z} \to \mathbb{R}$ is said to be piecewise quadratic on a polytopic (polyhedral) partition $\mathcal{P} = \{\mathbb{Z}_i \mid i \in \mathcal{I}\}$ if it satisfies, for some Q_i, r_i, and s_i, $i \in \mathcal{I}$, $f(x) = (1/2)x'Q_i x + r_i' x + s_i$ for all $x \in \mathbb{Z}_i$, all $i \in \mathcal{I}$.

Note the piecewise affine and piecewise quadratic functions defined this way are not necessarily continuous and may, therefore, be set valued at the intersection of the defining polyhedrons. An example is the piecewise affine function $f(\cdot)$ defined by

$$\begin{aligned} f(x) &:= -x - 1 \quad x \in (-\infty, 0] \\ &:= x + 1 \quad x \in [0, \infty) \end{aligned}$$

This function is set valued at $x = 0$ where it has the value $f(0) = \{-1, 1\}$. We shall mainly be concerned with continuous piecewise affine and piecewise quadratic functions.

We now generalize the points illustrated by our example above and consider, in turn, parametric quadratic programming and parametric

[1] The interior of a set $S \subseteq \mathbb{Z}$ relative to the set $\mathbb{Z}$ is the set $\{z \in S \mid \varepsilon(z)\mathcal{B} \cap \text{aff}(\mathbb{Z}) \subseteq \mathbb{Z}$ for some $\varepsilon > 0\}$ where $\text{aff}(\mathbb{Z})$ is the intersection of all affine sets containing $\mathbb{Z}$.

linear programming and their application to optimal control problems. We deal with parametric quadratic programming first because it is more widely used and because, with reasonable assumptions, the minimizer is unique making the underlying ideas somewhat simpler to follow.

7.3 Parametric Quadratic Programming

7.3.1 Preliminaries

The parametric QP $\mathbb{P}(x)$ is defined by

$$V^0(x) = \min_u \{V(x,u) \mid (x,u) \in \mathbb{Z}\}$$

where $x \in \mathbb{R}^n$ and $u \in \mathbb{R}^m$. The cost function $V(\cdot)$ is defined by

$$V(x,u) := (1/2)x'Qx + u'Sx + (1/2)u'Ru + q'x + r'u + c$$

and the polyhedral constraint set $\mathbb{Z}$ is defined by

$$\mathbb{Z} := \{(x,u) \mid Mx \le Nu + p\}$$

where $M \in \mathbb{R}^{r\times n}$, $N \in \mathbb{R}^{r\times m}$ and $p \in \mathbb{R}^r$; thus $\mathbb{Z}$ is defined by r affine inequalities. Let $u^0(x)$ denote the solution of $\mathbb{P}(x)$ if it exists, i.e., if $x \in \mathcal{X}$, the domain of $V^0(\cdot)$; thus

$$u^0(x) := \arg\min_u \{V(x,u) \mid (x,u) \in \mathbb{Z}\}$$

The solution $u^0(x)$ is unique if $V(\cdot)$ is strictly convex in u; this is the case if R is positive definite. Let the matrix $\mathcal{Q}$ be defined by

$$\mathcal{Q} := \begin{bmatrix} Q & S' \\ S & R \end{bmatrix}$$

For simplicity we assume the following in the sequel.

Assumption 7.3 (Strict convexity)**.** The matrix $\mathcal{Q}$ is positive definite.

Assumption 7.3 implies that both R and Q are positive definite. The cost function $V(\cdot)$ may be written in the form

$$V(x,u) = (1/2)(x,u)'\mathcal{Q}(x,u) + q'x + r'u + c$$

where, as usual, the vector (x,u) is regarded as a column vector $(x', u')'$ in algebraic expressions. The parametric QP may also be expressed as

$$V^0(x) := \min_u \{V(x,u) \mid u \in \mathcal{U}(x)\}$$

where the parametric constraint set $\mathcal{U}(x)$ is defined by

$$\mathcal{U}(x) := \{u \mid (x,u) \in \mathbb{Z}\} = \{u \in \mathbb{R}^m \mid Mu \leq Nx + p\}$$

For each x the set $\mathcal{U}(x)$ is polyhedral. The domain $\mathcal{X}$ of $V^0(\cdot)$ and $u^0(\cdot)$ is defined by

$$\mathcal{X} := \{x \mid \exists u \in \mathbb{R}^m \text{ such that } (x,u) \in \mathbb{Z}\} = \{x \mid \mathcal{U}(x) \neq \emptyset\}$$

For all $(x,u) \in \mathbb{Z}$, let the index set $I(x,u)$ specify the constraints that are *active* at (x,u), i.e.,

$$I(x,u) := \{i \in \mathbb{I}_{1:r} \mid M_i u = N_i x + p_i\}$$

where M_i, N_i, and p_i denote, respectively, the ith row of M, N, and p. Similarly, for any matrix or vector A and any index set I, A_I denotes the matrix or vector with rows A_i, $i \in I$. For any $x \in \mathcal{X}$, the indices set $I^0(x)$ specifies the constraints that are active at $(x, u^0(x))$, namely

$$I^0(x) := I(x, u^0(x)) = \{i \in \mathbb{I}_{1:r} \mid M_i u^0(x) = N_i x + p_i\}$$

Since $u^0(x)$ is unique, $I^0(x)$ is well defined. Thus $u^0(x)$ satisfies the equation

$$M_x^0 u = N_x^0 x + p_x^0$$

where

$$M_x^0 := M_{I^0(x)},\ N_x^0 := N_{I^0(x)},\ p_x^0 := p_{I^0(x)} \tag{7.1}$$

7.3.2 Preview

We show in the sequel that $V^0(\cdot)$ is piecewise quadratic and $u^0(\cdot)$ piecewise affine on a polyhedral partition of $\mathcal{X}$, the domain of both these functions. To do this, we take an arbitrary point x in $\mathcal{X}$, and show that $u^0(x)$ is the solution of an *equality* constrained QP $\mathbb{P}(x) : \min_u \{V(x, u) \mid M_x^0 u = N_x^0 x + p_x^0\}$ in which the equality constraint is $M_x^0 u = N_x^0 x + p_x^0$. We then show that there is a polyhedral region $R_x^0 \subset \mathcal{X}$ in which x lies and such that, for all $w \in R_x^0$, $u^0(w)$ is the solution of the equality constrained QP $\mathbb{P}(w) : \min_u \{V(w,u) \mid M_x^0 u = N_x^0 w + p_x^0\}$ in which the equality constraints are the same as those for $\mathbb{P}(x)$. It follows that $u^0(\cdot)$ is affine and $V^0(\cdot)$ is quadratic in R_x^0. We then show that there are only a finite number of such polyhedral regions so that $u^0(\cdot)$ is piecewise affine, and $V^0(\cdot)$ piecewise quadratic, on a polyhedral partition of $\mathcal{X}$. To carry out this program, we require a suitable characterization of optimality. We develop this in the next subsection. Some readers may prefer to jump to Proposition 7.8, which gives the optimality condition we employ in the sequel.

7.3.3 Optimality Condition for a Convex Program

Necessary and sufficient conditions for nonlinear optimization problems are developed in Section C.2 of Appendix C. Since we are concerned here with a relatively simple optimization problem where the cost is convex and the constraint set polyhedral, we give a self-contained exposition that uses the concept of a *polar cone*.

Definition 7.4 (Polar cone)**.** The *polar cone* of a cone $C \subseteq \mathbb{R}^n$ is the cone C^* defined by

$$C^* := \{g \in \mathbb{R}^n \mid \langle g, h\rangle \leq 0 \ \forall h \in C\}$$

We recall that a set $C \subseteq \mathbb{R}^n$ is a cone if $0 \in C$ and that $h \in C$ implies $\lambda h \in C$ for all $\lambda > 0$. A cone C is said to be *generated* by $\{a_i \mid i \in I\}$ where I is an index set if $C = \sum_{i\in I}\{\mu_i a_i \mid \mu_i \geq 0, i \in I\}$ in which case we write $C = \text{cone}\{a_i \mid i \in I\}$. We need the following result.

Proposition 7.5 (Farkas's lemma)**.** *Suppose C is a polyhedral cone defined by*

$$C := \{h \mid Ah \leq 0\} = \{h \mid \langle a_i, h\rangle \leq 0 \mid i \in \mathbb{I}_{1:m}\}$$

in which, for each i, a_i is the ith row of A. Then

$$C^* = \text{cone}\{a_i \mid i \in \mathbb{I}_{1:m}\}$$

A proof of this result is given in Section C.2 of Appendix C; that $g \in \text{cone}\{a_i \mid i \in \mathbb{I}_{1:m}\}$ implies $\langle g, h\rangle \leq 0$ for all $h \in C$ is easily shown. An illustration of Proposition 7.5 is given in Figure 7.5.

Next we make use of a standard necessary and sufficient condition of optimality for optimization problems in which the cost is convex and differentiable and the constraint set is convex.

Proposition 7.6 (Optimality conditions for convex set)**.** *Suppose, for each $x \in \mathcal{X}$, $u \mapsto V(x,u)$ is convex and differentiable and $\mathcal{U}(x)$ is convex. Then u is optimal for* $\min_u\{V(x,u) \mid u \in \mathcal{U}(x)\}$ *if and only if*

$$u \in \mathcal{U}(x) \text{ and } \langle \nabla_u V(x,u), v - u\rangle \geq 0 \quad \forall v \in \mathcal{U}(x)$$

Proof. This Proposition appears as Proposition C.9 in Appendix C where a proof is given. ∎

In our case $\mathcal{U}(x)$, $x \in \mathcal{X}$, is polyhedral and is defined by

$$\mathcal{U}(x) := \{v \in \mathbb{R}^m \mid Mv \leq Nx + p\} \tag{7.2}$$

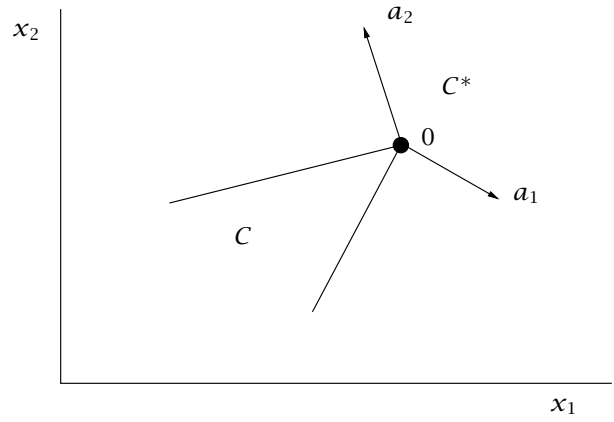


Figure 7.5: Polar cone.

so $v \in \mathcal{U}(x)$ if and only if, for all $u \in \mathcal{U}(x)$, $v - u \in \mathcal{U}(x) - \{u\} := \{v - u \mid v \in \mathcal{U}(x)\}$. With $h := v - u$

$$\mathcal{U}(x)-\{u\} = \left\{ h \in \mathbb{R}^m \;\middle|\; \begin{array}{ll} M_i h \le 0, & i \in I(x,u) \\ M_j h < N_j x + p_j - M_j u, & j \in \mathbb{I}_{1:r} \setminus I(x,u) \end{array} \right\}$$

since $M_i u = N_i x + p_i$ for all $i \in I(x,u)$. For each $z = (x,u) \in \mathbb{Z}$, let $C(x,u)$ denote the cone of feasible directions[2] $h = v - u$ at u, i.e., $C(x,u)$ is defined by

$$C(x,u) := \{h \in \mathbb{R}^m \mid M_i h \le 0,\ i \in I(x,u)\}$$

Clearly

$$\mathcal{U}(x)-\{u\} = C(x,u) \cap \{h \in \mathbb{R}^m \mid M_i h < N_i x + p_i - M_i u,\ i \in \mathbb{I}_{1:r} \setminus I(x,u)\}$$

so that $\mathcal{U}(x) - \{u\} \subseteq C(x,u)$; for any $(x,u) \in \mathbb{Z}$, any $h \in C(x,u)$, there exists an $\alpha > 0$ such that $u + \alpha h \in \mathcal{U}(x)$. Proposition 7.6 may be expressed as: u is optimal for $\min_u \{V(x,u) \mid u \in \mathcal{U}(x)\}$ if and only if

$$u \in \mathcal{U}(x) \text{ and } \langle \nabla_u V(x,u), h \rangle \ge 0 \quad \forall h \in \mathcal{U}(x) - \{u\}$$

We may now state a modified form of Proposition 7.6.

[2] A direction h at u is feasible if there exists an $\varepsilon > 0$ such that $u + \lambda h \in U(x)$ for all $\lambda \in [0, \varepsilon]$.

Proposition 7.7 (Optimality conditions in terms of polar cone)**.** *Suppose for each $x \in \mathcal{X}$, $u \mapsto V(x,\cdot)$ is convex and differentiable, and $\mathcal{U}(x)$ is defined by* (7.2)*. Then u is optimal for* $\min_u\{V(x,u) \mid u \in \mathcal{U}(x)\}$ *if and only if*

$$u \in \mathcal{U}(x) \text{ and } \langle \nabla_u V(x,u), h\rangle \geq 0 \quad \forall h \in C(x,u)$$

Proof. We show that the condition $\langle \nabla_u V(x,u), h\rangle \geq 0$ for all $h \in C(x,u)$ is equivalent to the condition $\langle \nabla_u V(x,u), h\rangle \geq 0$ for all $h \in \mathcal{U}(x) - \{u\}$ employed in Proposition 7.6. (i) Since $\mathcal{U}(x) - \{u\} \subseteq C(x,u)$, $\langle \nabla_u V(x,u), h\rangle \geq 0$ for all $h \in C(x,u)$ implies $\langle \nabla_u V(x,u), h\rangle \geq 0$ for all $h \in \mathcal{U}(x) - \{u\}$. (ii) $\langle \nabla_u V(x,u), h\rangle \geq 0$ for all $h \in \mathcal{U}(x) - \{u\}$ implies $\langle \nabla_u V(x,u), \alpha h\rangle \geq 0$ for all $h \in \mathcal{U}(x) - \{u\}$, all $\alpha > 0$. But, for any $h^* \in C(x,u)$, there exists an $\alpha \geq 1$ such that $h^* = \alpha h$ with $h := (1/\alpha)h^* \in \mathcal{U}(x) - \{u\}$. Hence $\langle \nabla_u V(x,u), h^*\rangle = \langle \nabla_u V(x,u), \alpha h\rangle \geq 0$ for all $h^* \in C(x,u)$. ∎

We now make use of Proposition 7.7 to obtain the optimality condition in the form we use in the sequel. For all $(x,u) \in \mathbb{Z}$, let $C^*(x,u)$ denote the polar cone to $C(x,u)$.

Proposition 7.8 (Optimality conditions for linear inequalities)**.** *Suppose, for each $x \in \mathcal{X}$, $u \mapsto V(x,u)$ is convex and differentiable, and $\mathcal{U}(x)$ is defined by* (7.2)*. Then u is optimal for* $\min_u\{V(x,u) \mid u \in \mathcal{U}(x)\}$ *if and only if*

$$u \in \mathcal{U}(x) \text{ and } -\nabla_u V(x,u) \in C^*(x,u) = \text{cone}\{M_i' \mid i \in I(x,u)\}$$

Proof. The desired result follows from a direct application of Proposition 7.5 to Proposition 7.7. ∎

Note that $C(x,u)$ and $C^*(x,u)$ are both cones so that each set contains the origin. In particular, $C^*(x,u)$ is generated by the gradients of the constraints active at $z = (x,u)$, and may be defined by a set of affine inequalities: for each $z \in \mathbb{Z}$, there exists a matrix L_z such that

$$C^*(x,u) = C^*(z) = \{g \in \mathbb{R}^m \mid L_z g \leq 0\}$$

The importance of this result for us lies in the fact that the necessary and sufficient condition for optimality is satisfaction of two polyhedral constraints, $u \in \mathcal{U}(x)$ and $-\nabla_u V(x,u) \in C^*(x,u)$. Proposition 7.8 may also be obtained by direct application of Proposition C.12 of Appendix C; $C^*(x,u)$ may be recognized as $\mathcal{N}_{\mathcal{U}(x)}(u)$, the regular normal cone to the set $\mathcal{U}(x)$ at u.

7.3.4 Solution of the Parametric Quadratic Program

For the parametric programming problem $\mathbb{P}(x)$, the parametric cost function is

$$V(x,u) := (1/2)x'Qx + u'Sx + (1/2)u'Ru + q'x + r'u + c$$

and the parametric constraint set is

$$\mathcal{U}(x) := \{u \mid Mu \le Nx + p\}$$

Hence, the cost gradient is

$$\nabla_u V(x,u) = Ru + Sx + r$$

in which, because of Assumption 7.3, R is positive definite. Hence, the necessary and sufficient condition for the optimality of u for the parametric QP $\mathbb{P}(x)$ is

$$\begin{aligned} &Mu \le Nx + p \\ &-(Ru + Sx + r) \in C^*(x,u) \end{aligned}$$

in which $C^*(x,u) = \text{cone}\{M_i' \mid i \in I(x,u)\}$, the cone generated by the gradients of the active constraints, is polyhedral. We cannot use this characterization of optimality directly to solve the parametric programming problem since $I(x,u)$ and, hence, $C^*(x,u)$, varies with (x, u). Given any $x \in \mathcal{X}$, however, there exists the possibility of a region containing x such that $I^0(x) \subseteq I^0(w)$ for all w in this region. We make use of this observation as follows. It follows from the definition of $I^0(x)$ that the unique solution $u^0(x)$ of $\mathbb{P}(x)$ satisfies the equation

$$\begin{aligned} &M_i u = N_i x + p_i, \quad i \in I^0(x), \text{ i.e.,} \\ &M_x^0 u = N_x^0 x + p_x^0 \end{aligned}$$

where M_x^0, N_x^0, and p_x^0 are defined in (7.1). Hence $u^0(x)$ is the solution of the equality constrained problem

$$V^0(x) = \min_u \{V(x,u) \mid M_x^0 u = N_x^0 x + p_x^0\}$$

If the active constraint set remains constant near the point x or, more precisely, if $I^0(x) \subseteq I^0(w)$ for all w in some region in $\mathbb{R}^n$ containing x, then, for all w in this region, $u^0(w)$ satisfies the equality constraint

$M_x^0 u = N_x^0 w + p_x^0$. This motivates us to consider the simple equality constrained problem $\mathbb{P}_x(w)$ defined by

$$V_x^0(w) = \min_u \{V(w,u) \mid M_x^0 u = N_x^0 w + p_x^0\}$$
$$u_x^0(w) = \arg\min_u \{V(w,u) \mid M_x^0 u = N_x^0 w + p_x^0\}$$

The subscript x indicates that the equality constraints in $\mathbb{P}_x(w)$ depend on x. Problem $\mathbb{P}_x(w)$ is an optimization problem with a quadratic cost function and linear equality constraints and is, therefore, easily solved; see the exercises at the end of this chapter. Its solution is

$$V_x^0(w) = (1/2)w'Q_x w + r_x' w + s_x \tag{7.3}$$
$$u_x^0(w) = K_x w + k_x \tag{7.4}$$

for all w such that $I^0(w) = I^0(x)$ where $Q_x \in \mathbb{R}^{n\times n}$, $r_x \in \mathbb{R}^n$, $s_x \in \mathbb{R}$, $K_x \in \mathbb{R}^{m\times n}$ and $k_x \in \mathbb{R}^m$ are easily determined. Clearly, $u_x^0(x) = u^0(x)$; but, is $u_x^0(w)$, the optimal solution to $\mathbb{P}_x(w)$, the optimal solution $u^0(w)$ to $\mathbb{P}(w)$ in some region containing x and, if it is, what is the region? Our optimality condition answers this question. For all $x \in \mathcal{X}$, let the region R_x^0 be defined by

$$R_x^0 := \left\{ w \;\middle|\; \begin{matrix} u_x^0(w) \in \mathcal{U}(w) \\ -\nabla_u V(w, u_x^0(w)) \in C^*(x, u^0(x)) \end{matrix} \right\} \tag{7.5}$$

Because of the equality constraint $M_x^0 u = N_x^0 w + p_x^0$ in problem $\mathbb{P}_x(w)$, it follows that $I(w, u_x^0(w)) \supseteq I(x, u^0(x))$, and that $C(w, u_x^0(w)) \subseteq C(x, u^0(x))$ and $C^*(w, u_x^0(w)) \supseteq C^*(x, u^0(x))$ for all $w \in R_x^0$. Hence $w \in R_x^0$ implies $u_x^0(w) \in \mathcal{U}(w)$ and $-\nabla_u V(w, u_x^0(w)) \in C^*(w, u_x^0(w))$ for all $w \in R_x^0$ which, by Proposition 7.8, is a necessary and sufficient condition for $u_x^0(w)$ to be optimal for $\mathbb{P}(w)$. In fact, $I(w, u_x^0(w)) = I(x, u^0(x))$ so that $C^*(w, u_x^0(w)) = C^*(x, u^0(x))$ for all w in the interior of R_x^0. The obvious conclusion of this discussion is the following.

Proposition 7.9 (Solution of $\mathbb{P}(w)$, $w \in R_x^0$)**.** *For any $x \in \mathcal{X}$, $u_x^0(w)$ is optimal for $\mathbb{P}(w)$ for all $w \in R_x^0$.*

The constraint $u_x^0(w) \in \mathcal{U}(w)$ may be expressed as

$$M(K_x w + k_x) \leq Nw + p$$

which is an affine inequality in w. Similarly, since $\nabla_u V(w,u) = Ru + Sw + r$ and since $C^*(x, u^0(x)) = \{g \mid L_x^0 g \leq 0\}$ where $L_x^0 = L_{(x,u^0(x))}$, the constraint $-\nabla_u V(x, u_x^0(w)) \in C(x, u^0(x))$ may be expressed as

$$-L_x^0(R(K_x w + k_x) + Sw + r) \leq 0$$

which is also an affine inequality in the variable w. Thus, for each x, there exists a matrix F_x and vector f_x such that

$$R_x^0 = \{w \mid F_x w \le f_x\}$$

so that R_x^0 is polyhedral. Since $u_x^0(x) = u^0(x)$, it follows that $u_x^0(x) \in \mathcal{U}(x)$ and $-\nabla_u V(x, u_x^0(x)) \in C^*(x, u^0(x))$ so that $x \in R_x^0$.

Our next task is to bound the number of distinct regions R_x^0 that exist as we permit x to range over $\mathcal{X}$. We note, from its definition, that R_x^0 is determined, through the constraint $M_x^0 u = N_x^0 w + p_x^0$ in $\mathbb{P}_x(w)$, through $u_x^0(\cdot)$ and through $C^*(x, u^0(x))$, by $I^0(x)$ so that $R_{x_1}^0 \ne R_{x_2}^0$ implies that $I^0(x_1) \ne I^0(x_2)$. Since the number of subsets of $\{1, 2, \ldots, p\}$ is finite, the number of distinct regions R_x^0 as x ranges over $\mathcal{X}$ is finite. Because each $x \in \mathcal{X}$ lies in the set R_x^0, there exists a discrete set of points $X \subset \mathcal{X}$ such that $\mathcal{X} = \cup\{R_x^0 \mid x \in X\}$. We have proved the following.

Proposition 7.10 (Piecewise quadratic (affine) cost (solution))**.**

(a) There exists a set X of a finite number of points in $\mathcal{X}$ such that $\mathcal{X} = \cup\{R_x^0 \mid x \in X\}$ and $\{R_x^0 \mid x \in X\}$ is a polyhedral partition of $\mathcal{X}$.

(b) The value function $V^0(\cdot)$ of the parametric piecewise QP $\mathbb{P}$ is piecewise quadratic in $\mathcal{X}$, being quadratic and equal to $V_x^0(\cdot)$, defined in (7.3) *in each polyhedron R_x, $x \in X$. Similarly, the minimizer $u^0(\cdot)$ is piecewise affine in $\mathcal{X}$, being affine and equal to $u_x^0(\cdot)$ defined in* (7.4) *in each polyhedron R_x^0, $x \in X$.*

Example 7.11: Parametric QP

Consider the example in Section 7.2. This may be expressed as

$$V^0(x) = \min_u V(x, u), \quad V(x, u) := \{(1/2)x^2 - ux + u^2 \mid Mu \le Nx + p\}$$

where

$$M = \begin{bmatrix} -1 \\ -1 \\ -1 \end{bmatrix} \quad N = \begin{bmatrix} 0 \\ 1/2 \\ 1 \end{bmatrix} \quad p = \begin{bmatrix} -1 \\ -2 \\ -2 \end{bmatrix}$$

At $x = 1$, $u^0(x) = 3/2$ and $I^0(x) = \{2\}$. The equality constrained optimization problem $\mathbb{P}_x(w)$ is

$$V_x^0(w) = \min_u \{(1/2)w^2 - uw + u^2 \mid -u = (1/2)w - 2\}$$

so that $u^0(w) = 2 - w/2$. Hence

$$R_x^0 := \left\{ w \;\middle|\; \begin{matrix} Mu_x^0(w) \le Nw + p(w) \\ -\nabla_u V(w, u_x^0(w)) \in C^*(x, u^0(x)) \end{matrix} \right\}$$

Since $M_2 = -1$, $C^*(x) = \text{cone}\{M_i' \mid i \in I^0(x)\} = \text{cone}\{M_2'\} = \{h \in \mathbb{R} \mid h \le 0\}$; also

$$\nabla_u V(w, u_x^0(w)) = -w + 2u^0(w) = -w + 2(2 - w/2) = -2w + 4$$

so that R_x^0 is defined by the following inequalities

$$\begin{aligned} (1/2)w - 2 &\le -1 && \text{or } w \le 2 \\ (1/2)w - 2 &\le (1/2)w - 2 && \text{or } w \in \mathbb{R} \\ (1/2)w - 2 &\le w - 2 && \text{or } w \ge 0 \\ 2w - 4 &\le 0 && \text{or } w \le 2 \end{aligned}$$

which reduces to $w \in [0, 2]$ so $R_x^0 = [0, 2]$ when $x = 1$; $[0, 2]$ is the set X_2 determined in Section 7.2. □

Example 7.12: Explicit optimal control

We return to the MPC problem presented in Example 2.5 of Chapter 2

$$\begin{aligned} V^0(x, \mathbf{u}) &= \min_{\mathbf{u}} \{V(x, \mathbf{u}) \mid \mathbf{u} \in \mathcal{U}\} \\ V(x, \mathbf{u}) &:= (3/2)x^2 + [2x, x]\mathbf{u} + (1/2)\mathbf{u}' H \mathbf{u} \\ H &:= \begin{bmatrix} 3 & 1 \\ 1 & 2 \end{bmatrix} \\ \mathcal{U} &:= \{\mathbf{u} \mid M\mathbf{u} \le p\} \end{aligned}$$

where

$$M := \begin{bmatrix} 1 & 0 \\ -1 & 0 \\ 0 & 1 \\ 0 & -1 \end{bmatrix} \qquad p := \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

It follows from the solution to Example 2.5 that

$$u^0(2) = \begin{bmatrix} -1 \\ -(1/2) \end{bmatrix}$$

and $I^0(x) = \{2\}$. The equality constrained optimization problem at $x = 2$ is

$$V_x^0(w) = \min_{\mathbf{u}} \{(3/2)w^2 + 2wu_1 + wu_2 + (1/2)\mathbf{u}' H \mathbf{u} \mid u_1 = -1\}$$

so that

$$u_x^0(w) = \begin{bmatrix} -1 \\ (1/2) - (1/2)w \end{bmatrix}$$

Hence $u_x^0(2) = [-1, -1/2]' = u^0(2)$ as expected. Since $M_x^0 = M_2 = [-1, 0]$, $C^*(x, u^0(x)) = \{g \in \mathbb{R}^2 \mid g_1 \le 0\}$. Also

$$\nabla_{\mathbf{u}} V(w, \mathbf{u}) = \begin{bmatrix} 2w + 3u_1 + u_2 \\ w + u_1 + 2u_2 \end{bmatrix}$$

so that

$$\nabla_{\mathbf{u}} V(w, \mathbf{u}_x^0(w)) = \begin{bmatrix} (3/2)w - (5/2) \\ 0 \end{bmatrix}$$

Hence R_x^0, $x = 2$ is the set of w satisfying the following inequalities

$$\begin{aligned} (1/2) - (1/2)w \le 1 \quad &\text{or } w \ge -1 \\ (1/2) - (1/2)w \ge -1 \ &\text{or } w \le 3 \\ -(3/2)w + (5/2) \le 0 \quad &\text{or } w \ge (5/3) \end{aligned}$$

which reduces to $w \in [5/3, 3]$; hence $R_x^0 = [5/3, 3]$ when $x = 2$ as shown in Example 2.5. □

7.3.5 Continuity of $V^0(\cdot)$ and $u^0(\cdot)$

Continuity of $V^0(\cdot)$ and $u^0(\cdot)$ follows from Theorem C.34 in Appendix C. We present here a simpler proof based on the above analysis. We use the fact that the parametric quadratic problem is strictly convex, i.e., for each $x \in \mathcal{X}$, $u \mapsto V(x, u)$ is strictly convex and $\mathcal{U}(x)$ is convex, so that the minimizer $u^0(x)$ is unique as shown in Proposition C.8 of Appendix C.

Let $X = \{x_i \mid i \in \mathbb{I}_{1:I}\}$ denote the set defined in Proposition 7.10(a). For each $i \in \mathbb{I}_{i:I}$, let $R_i := R_{x_i}^0$, $V_i(\cdot) := V_{x_i}^0(\cdot)$ and $u_i(\cdot) := u_{x_i}^0(\cdot)$. From Proposition 7.10, $u^0(x) = u_i(x)$ for each $x \in R_i$, each $i \in \mathbb{I}_{1:I}$ so that $u^0(\cdot)$ is affine and hence continuous in the interior of each R_i, and also continuous at any point x on the boundary of $\mathcal{X}$ such that x lies in a single region R_i. Consider now a point x lying in the intersection of several regions, $x \in \cap_{i \in J} R_i$, where J is a subset of $\mathbb{I}_{1:I}$. Then, by Proposition 7.10, $u_i(x) = u^0(x)$ for all $x \in \cap_{i \in J} R_i$, all $i \in J$. Each $u_i(\cdot)$ is affine and, therefore, continuous, so that $u^0(\cdot)$ is continuous in $\cap_{i \in J} R_i$. Hence $u^0(\cdot)$ is continuous in $\mathcal{X}$. Because $V(\cdot)$ is continuous and $u^0(\cdot)$ is continuous in $\mathcal{X}$, the value function $V^0(\cdot)$ defined by $V^0(x) = V(x, u^0(x))$ is also continuous in $\mathcal{X}$. Let S denote any bounded subset of $\mathcal{X}$.

Then, since $V^0(x) = V_i(x) = (1/2)x'Q_ix + r_i'x + s_i$ for all $x \in R_i$, all $i \in \mathbb{I}_{1:I}$ where $Q_i := Q_{x_i}$, $r_i := r_{x_i}$ and $s_i := s_{x_i}$, it follows that $V^0(\cdot)$ is Lipschitz continuous in each set $R_i \cap S$ and, hence, Lipschitz continuous in $\mathcal{X} \cap S$. We have proved the following.

Proposition 7.13 (Continuity of cost and solution)**.** *The value function $V^0(\cdot)$ and the minimizer $u^0(\cdot)$ are continuous in $\mathcal{X}$. Moreover, the value function and the minimizer are Lipschitz continuous on bounded sets.*

7.4 Constrained Linear Quadratic Control

We now show how parametric quadratic programming may be used to solve the optimal receding horizon control problem when the system is linear, the constraints polyhedral, and the cost is quadratic. The system is described, as before, by

$$x^+ = Ax + Bu \tag{7.6}$$

and the constraints are, as before

$$x \in \mathbb{X} \qquad u \in \mathbb{U} \tag{7.7}$$

where $\mathbb{X}$ is a polyhedron containing the origin in its interior and $\mathbb{U}$ is a polytope also containing the origin in its interior. There may be a terminal constraint of the form

$$x(N) \in \mathbb{X}_f \tag{7.8}$$

where $\mathbb{X}_f$ is a polyhedron containing the origin in its interior. The cost is

$$V_N(x, \mathbf{u}) = \left[\sum_{i=0}^{N-1} \ell(x(i), u(i)) \right] + V_f(x(N)) \tag{7.9}$$

in which, for all i, $x(i) = \phi(i; x, \mathbf{u})$, the solution of (7.6) at time i if the initial state at time 0 is x and the control sequence is $\mathbf{u} := (u(0), u(1), \ldots, u(N-1))$. The functions $\ell(\cdot)$ and $V_f(\cdot)$ are quadratic

$$\ell(x, u) := (1/2)x'Qx + (1/2)u'Ru, \quad V_f(x) := (1/2)x'Q_fx \tag{7.10}$$

The state and control constraints (7.7) induce, via the difference equation (7.6), an implicit constraint $(x, \mathbf{u}) \in \mathcal{Z}$ where

$$\mathcal{Z} := \{(x, \mathbf{u}) \mid x(i) \in \mathbb{X},\ u(i) \in \mathbb{U}, i \in \mathbb{I}_{0:N-1}, x(N) \in \mathbb{X}_f\} \tag{7.11}$$

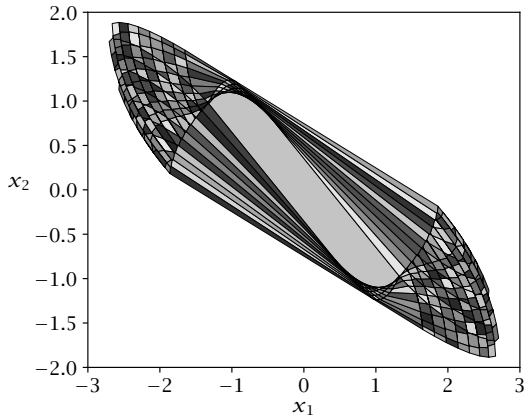


Figure 7.6: Regions R_x, $x \in X$ for a second-order example; after Mayne and Raković (2003).

where, for all i, $x(i) = \phi(i; x, \mathbf{u})$. It is easily seen that $\mathbb{Z}$ is polyhedral since, for each i, $x(i) = A^i x + M_i \mathbf{u}$ for some matrix M_i in $\mathbb{R}^{n \times Nm}$; here $\mathbf{u}$ is regarded as the column vector $\begin{bmatrix} u(0)' & u(1)' & \cdots & u(N-1)' \end{bmatrix}'$. Clearly $x(i) = \phi(i; x, \mathbf{u})$ is linear in $(x, \mathbf{u})$. The constrained linear optimal control problem may now be defined by

$$V_N^0(x) = \min_{\mathbf{u}} \{V_N(x, \mathbf{u}) \mid (x, \mathbf{u}) \in \mathbb{Z}\}$$

Using the fact that for each i, $x(i) = A^i x + M_i \mathbf{u}$, it is possible to determine matrices $\mathbf{Q} \in \mathbb{R}^{n \times n}$, $\mathbf{R} \in \mathbb{R}^{Nm \times Nm}$, and $\mathbf{S} \in \mathbb{R}^{Nm \times n}$ such that

$$V_N(x, \mathbf{u}) = (1/2) x' \mathbf{Q} x' + (1/2) \mathbf{u}' \mathbf{R} \mathbf{u} + \mathbf{u}' \mathbf{S} x \tag{7.12}$$

Similarly, as shown above, there exist matrices $\mathbf{M}$, $\mathbf{N}$ and a vector $\mathbf{p}$ such that

$$\mathbb{Z} = \{(x, \mathbf{u}) \mid \mathbf{M}\mathbf{u} \leq \mathbf{N}x + \mathbf{p}\} \tag{7.13}$$

This is precisely the parametric problem studied in Section 7.3, so that the solution $\mathbf{u}^0(x)$ to $\mathbb{P}(x)$ is piecewise affine on a polytopic partition $\mathcal{P} = \{R_x \mid x \in X\}$ of $\mathcal{X}$ the projection of $\mathbb{Z} \subset \mathbb{R}^n \times \mathbb{R}^{Nm}$ onto $\mathbb{R}^n$, being affine in each of the constituent polytopes of $\mathcal{P}$. The receding horizon control law is $x \mapsto u^0(0; x)$, the first element of $\mathbf{u}^0(x)$. An example is shown in Figure 7.6.

7.5 Parametric Piecewise Quadratic Programming

The dimension of the decision variable $\mathbf{u}$ in the constrained linear quadratic control problem discussed in Section 7.4 is Nm which is large. It may be better to employ dynamic programming by solving a sequence of problems $\mathbb{P}_1, \mathbb{P}_2, \ldots, \mathbb{P}_N$. Although $\mathbb{P}_1$ is a conventional parametric QP, each problem $\mathbb{P}_i$, $i = 2, 3, \ldots, N$, has the form

$$V_i^0(x) = \min_u \{V_{i-1}^0(Ax + Bu) + \ell(x, u) \mid u \in \mathbb{U}, Ax + Bu \in \mathcal{X}_{i-1}\}$$

in which $V_{i-1}^0(\cdot)$ is piecewise quadratic and $\mathcal{X}_{i-1}$ is polyhedral. The decision variable u in each problem $\mathbb{P}_i$ has dimension m. But each problem $\mathbb{P}_i(x)$, $x \in \mathcal{X}_i$, is a parametric piecewise QP rather than a conventional parametric QP. Hence a method for solving parametric piecewise quadratic programming problems is required if dynamic programming is employed to obtain a parametric solution to $\mathbb{P}_N$. Readers not concerned with this extension should proceed to Section 7.7.

The parametric QP $\mathbb{P}(x)$ is defined, as before, by

$$V^0(x) = \min_u \{V(x, u) \mid (x, u) \in \mathbb{Z}\} \tag{7.14}$$

where $x \in \mathcal{X} \subset \mathbb{R}^n$ and $u \in \mathbb{R}^m$, but now the cost function $V(\cdot)$ is assumed to be continuous, strictly convex, and piecewise quadratic on a polytopic partition $\mathcal{P} = \{\mathbb{Z}_i \mid i \in \mathcal{I}\}$ of the set $\mathbb{Z}$ so that

$$V(z) = V_i(z) = (1/2)z'\mathcal{Q}_i z + s_i' z + c_i$$

for all $z \in \mathbb{Z}_i$, all $i \in \mathcal{I}$ where $\mathcal{I}$ is an index set.[3] In (7.14), the matrix Q_i and the vector s_i have the structure

$$\mathcal{Q}_i = \begin{bmatrix} Q_i & S_i' \\ S_i & R_i \end{bmatrix} \qquad s_i = \begin{bmatrix} q_i \\ r_i \end{bmatrix}$$

so that for all $i \in \mathcal{I}$

$$V_i(x, u) = (1/2)x'Q_i x + u'S_i x + (1/2)u'R_i u + q_i' x + r_i' u + c$$

For each x, the function $u \mapsto V_i(x, u)$ is quadratic and depends on x. The constraint set $\mathbb{Z}$ is defined, as above, by

$$\mathbb{Z} := \{(x, u) \mid Mu \leq Nx + p\}$$

[3] Note that in this section the subscript i denotes partition i rather than "time to go."

Let $u^0(x)$ denote the solution of $\mathbb{P}(x)$, i.e.,

$$u^0(x) = \arg\min_u \{V(x,u) \mid (x,u) \in \mathbb{Z}\}$$

The solution $u^0(x)$ is unique if $V(\cdot)$ is strictly convex in u at each x; this is the case if each R_i is positive definite. The parametric piecewise QP may also be expressed, as before, as

$$V^0(x) = \min_u \{V(x,u) \mid u \in \mathcal{U}(x)\}$$
$$u^0(x) = \arg\min_u \{V(x,u) \mid u \in \mathcal{U}(x)\}$$

where the parametric constraint set $\mathcal{U}(x)$ is defined by

$$\mathcal{U}(x) := \{u \mid (x,u) \in \mathbb{Z}\} = \{u \mid Mu \leq Nx + p\}$$

Let $\mathcal{X} \subset \mathbb{R}^n$ be defined by

$$\mathcal{X} := \{x \mid \exists u \text{ such that } (x,u) \in \mathbb{Z}\} = \{x \mid \mathcal{U}(x) \neq \emptyset\}$$

The set $\mathcal{X}$ is the domain of $V^0(\cdot)$ and of $u^0(\cdot)$ and is thus the set of points x for which a feasible solution of $\mathbb{P}(x)$ exists; it is the projection of $\mathbb{Z}$, which is a set in (x,u)-space, onto x-space as shown in Figure 7.1. We make the following assumption in the sequel.

Assumption 7.14 (Continuous, piecewise quadratic function)**.** The function $V(\cdot)$ is continuous, strictly convex, and piecewise quadratic on the polytopic partition $\mathcal{P} = \{\mathbb{Z}_i \mid i \in \mathcal{I} := \mathbb{I}_{1:q}\}$ of the polytope $\mathbb{Z}$ in $\mathbb{R}^n \times \mathbb{R}^m$; $V(x,u) = V_i(x,u)$ where $V_i(\cdot)$ is a positive definite quadratic function of (x,u) for all $(x,u) \in \mathbb{Z}_i$, all $i \in \mathcal{I}$, and q is the number of constituent polytopes in $\mathcal{P}$.

The assumption of continuity places restrictions on the quadratic functions $V_i(\cdot)$, $i \in \mathcal{I}$. For example, we must have $V_i(z) = V_j(z)$ for all $z \in \mathbb{Z}_i \cap \mathbb{Z}_j$. Assumption 7.14 implies that the piecewise quadratic programming problem $\mathbb{P}(x)$ satisfies the hypotheses of Theorem C.34 so that the value function $V^0(\cdot)$ is continuous. It follows from Assumption 7.14 and Theorem C.34 that $V^0(\cdot)$ is strictly convex and continuous and that the minimizer $u^0(\cdot)$ is continuous. Assumption 7.14 implies that Q_i is positive definite for all $i \in \mathcal{I}$. For each x, let the set $\mathcal{U}(x)$ be defined by

$$\mathcal{U}(x) := \{u \mid (x,u) \in \mathbb{Z}\}$$

Thus $\mathcal{U}(x)$ is the set of admissible u at x, and $\mathbb{P}(x)$ may be expressed in the form $V^0(x) = \min_u \{V(x,u) \mid u \in \mathcal{U}(x)\}$.

For each $i \in \mathcal{I}$, we define an "artificial" problem $\mathbb{P}_i(x)$ as follows

$$V_i^0(x) := \min_u \{V_i(x,u) \mid (x,u) \in \mathbb{Z}_i\}$$
$$u_i^0(x) := \arg\min_u \{V_i(x,u) \mid (x,u) \in \mathbb{Z}_i\}$$

The cost $V_i(x,u)$ in the above equations may be replaced by $V(x,u)$ since $V(x,u) = V_i(x,u)$ in $\mathbb{Z}_i$. The problem is artificial because it includes constraints (the boundaries of $\mathbb{Z}_i$) that are not necessarily constraints of the original problem. We introduce this problem because it helps us to understand the solution of the original problem. For each $i \in \mathbb{I}_{1:p}$, let the set $\mathcal{U}_i(x)$ be defined as follows

$$\mathcal{U}_i(x) := \{u \mid (x,u) \in \mathbb{Z}_i\}$$

Thus the set $\mathcal{U}_i(x)$ is the set of admissible u at x, and problem $\mathbb{P}_i(x)$ may be expressed as $V_i^0(x) := \min_u \{V_i(x,u) \mid u \in \mathcal{U}_i(x)\}$; the set $\mathcal{U}_i(x)$ is polytopic. For each i, problem $\mathbb{P}_i(x)$ may be recognized as a standard parametric QP discussed in Section 7.4. Because of the piecewise nature of $V(\cdot)$, we require another definition.

Definition 7.15 (Active polytope (polyhedron))**.** A polytope (polyhedron) $\mathbb{Z}_i$ in a polytopic (polyhedral) partition $\mathcal{P} = \{\mathbb{Z}_i \mid i \in \mathcal{I}\}$ of a polytope (polyhedron) $\mathbb{Z}$ is said to be *active* at $z \in \mathbb{Z}$ if $z = (x,u) \in \mathbb{Z}_i$. The index set specifying the polytopes active at $z \in \mathbb{Z}$ is

$$S(z) := \{i \in \mathcal{I} \mid z \in \mathbb{Z}_i\}$$

A polytope $\mathbb{Z}_i$ in a polytopic partition $\mathcal{P} = \{\mathbb{Z}_i \mid i \in \mathcal{I}\}$ of a polytope $\mathbb{Z}$ is said to be *active* for problem $\mathbb{P}(x)$) if $(x, u^0(x)) \in \mathbb{Z}_i$. The index set specifying polytopes active at $(x, u^0(x))$ is $S^0(x)$ defined by

$$S^0(x) := S(x, u^0(x)) = \{i \in \mathcal{I} \mid (x, u^0(x)) \in \mathbb{Z}_i\}$$

Because we know how to solve the "artificial" problems $\mathbb{P}_i(x)$, $i \in \mathcal{I}$ that are parametric quadratic programs, it is natural to ask if we can recover the solution of the original problem $\mathbb{P}(x)$ from the solutions to these simpler problems. This question is answered by the following proposition.

Proposition 7.16 (Solving $\mathbb{P}$ using $\mathbb{P}_i$)**.** *For any $x \in \mathcal{X}$, u is optimal for $\mathbb{P}(x)$ if and only if u is optimal for $\mathbb{P}_i(x)$ for all $i \in S(x,u)$.*

Proof. (i) Suppose u is optimal for $\mathbb{P}(x)$ but, contrary to what we wish to prove, there exists an $i \in S(x,u) = S^0(x)$ such that u is not optimal for $\mathbb{P}_i(x)$. Hence there exists a $v \in \mathbb{R}^m$ such that $(x,v) \in \mathbb{Z}_i$ and $V(x,v) = V_i(x,v) < V_i(x,u) = V(x,u) = V^0(x)$, a contradiction of the optimality of u for $\mathbb{P}(x)$. (ii) Suppose u is optimal for $\mathbb{P}_i(x)$ for all $i \in S(x,u)$ but, contrary to what we wish to prove, u is not optimal for $\mathbb{P}(x)$. Hence $V^0(x) = V(x,u^0(x)) < V(x,u)$. If $u^0(x) \in \mathbb{Z}^{(x,u)} := \cup_{i \in S(x,u)} \mathbb{Z}_i$, we have a contradiction of the optimality of u in $\mathbb{Z}^{(x,u)}$. Assume then that $u^0(x) \in \mathbb{Z}_j$, $j \notin S(x,u)$; for simplicity, assume further that $\mathbb{Z}_j$ is adjacent to $\mathbb{Z}^{(x,u)}$. Then, there exists a $\lambda \in (0,1]$ such that $u^\lambda := u + \lambda(u^0(x) - u) \in \mathbb{Z}^{(x,u)}$; if not, $j \in S(x,u)$, a contradiction. Since $V(\cdot)$ is strictly convex, $V(x,u^\lambda) < V(x,u)$, which contradicts the optimality of u in $\mathbb{Z}^{(x,u)}$. The case when $\mathbb{Z}_j$ is not adjacent to $\mathbb{Z}^{(x,u)}$ may be treated similarly. ∎

To obtain a parametric solution, we proceed as before. We select a point $x \in \mathcal{X}$ and obtain the solution $u^0(x)$ to $\mathbb{P}(x)$ using a standard algorithm for convex programs. The solution $u^0(x)$ satisfies an equality constraint $E_x u = F_x x + g_x$, which we employ to define, for any $w \in \mathcal{X}$ near x an easily solved equality constrained optimization problem $\mathbb{P}_x(w)$ that is derived from the problems $\mathbb{P}_i(x)$, $i \in S^0(x)$. Finally, we show that the solution to this simple problem is also a solution to the original problem $\mathbb{P}(w)$ at all w in a polytope $R_x \subset \mathcal{X}$ in which x lies.

For each $i \in \mathcal{I}$, $\mathbb{Z}_i$ is defined by

$$\mathbb{Z}_i := \{(x,u) \mid M^i u \le N^i x + p^i\}$$

Let M^i_j, N^i_j and q^i_j denote, respectively, the jth row of M^i, N^i and q^i, and let $I_i(x,u)$ and $I^0_i(x)$, defined by

$$I_i(x,u) := \{j \mid M^i_j u = N^i_j x + p^i_j\}, \quad I^0_i(x) := I_i(x,u^0_i(x))$$

denote, respectively, the active constraint set at $(x,u) \in \mathbb{Z}_i$ and the active constraint set for $\mathbb{P}_i(x)$. Because we now use subscript i to specify $\mathbb{Z}_i$, we change our notation slightly and now let $C_i(x,u)$ denote the cone of first-order feasible variations for $\mathbb{P}_i(x)$ at $u \in \mathcal{U}_i(x)$, i.e.,

$$C_i(x,u) := \{h \in \mathbb{R}^m \mid M^i_j h \le 0 \ \forall j \in I_i(x,u)\}$$

Similarly, we define the polar cone $C^*_i(x,u)$ of the cone $C_i(x,u)$ at

$h = 0$ by

$$C_i^*(x,u) := \{v \in \mathbb{R}^m \mid v'h \le 0 \ \forall h \in C_i(x,u)\}$$
$$= \left\{ \sum_{j \in I_i(x,u)} (M_j^i)'\lambda_j \;\middle|\; \lambda_j \ge 0, j \in I_i(x,u) \right\}$$

As shown in Proposition 7.7, a necessary and sufficient condition for the optimality of u for problem $\mathbb{P}_i(x)$ is

$$-\nabla_u V_i(x,u) \in C_i^*(x,u), \quad u \in \mathcal{U}_i(x) \tag{7.15}$$

If u lies in the interior of $\mathcal{U}_i(x)$ so that $I_i^0(x) = \emptyset$, condition (7.15) reduces to $\nabla_u V_i(x,u) = 0$. For any $x \in \mathcal{X}$, the solution $u^0(x)$ of the piecewise parametric program $\mathbb{P}(x)$ satisfies

$$M_j^i u = N_j^i x + p_j^i, \quad \forall j \in I_i^0(x), \ \forall i \in S^0(x) \tag{7.16}$$

To simplify our notation, we rewrite the equality constraint (7.16) as

$$E_x u = F_x x + g_x$$

where the subscript x denotes the fact that the constraints are precisely those constraints that are active for the problems $\mathbb{P}_i(x)$, $i \in S^0(x)$. The fact that $u^0(x)$ satisfies these constraints and is, therefore, the unique solution of the optimization problem

$$V^0(x) = \min_u \{V(x,u) \mid E_x u = F_x x + g_x\}$$

motivates us to define the equality constrained problem $\mathbb{P}_x(w)$ for $w \in \mathcal{X}$ near x by

$$V_x^0(w) = \min_u \{V_x(w,u) \mid E_x u = F_x w + g_x\}$$

where $V_x(w,u) := V_i(w,u)$ for all $i \in S^0(x)$ and is, therefore, a positive definite quadratic function of (x,u). The notation $V_x^0(w)$ denotes the fact that the parameter in the parametric problem $\mathbb{P}_x(w)$ is now w but the data for the problem, namely (E_x, F_x, g_x), is derived from the solution $u^0(x)$ of $\mathbb{P}(x)$ and is, therefore, x-dependent. Problem $\mathbb{P}_x(w)$ is a simple equality constrained problem in which the cost $V_x(\cdot)$ is quadratic and the constraints $E_x u = F_x w + g_x$ are linear. Let $V_x^0(w)$ denote the value of $\mathbb{P}_x(w)$ and $u_x^0(w)$ its solution. Then

$$V_x^0(w) = (1/2)w'Q_x w + r_x' w + s_x$$
$$u_x^0(w) = K_x w + k_x \tag{7.17}$$

where Q_x, r_x, s_x, K_x and k_x are easily determined. It is easily seen that $u_x^0(x) = u^0(x)$ so that $u_x^0(x)$ is optimal for $\mathbb{P}(x)$. Our hope is that $u_x^0(w)$ is optimal for $\mathbb{P}(w)$ for all w in some neighborhood R_x of x. We now show this is the case.

Proposition 7.17 (Optimality of $u_x^0(w)$ in R_x)**.** *Let x be an arbitrary point in $\mathcal{X}$. Then*

(a) $u^0(w) = u_x^0(w)$ and $V^0(w) = V_x^0(w)$ for all w in the set R_x defined by

$$R_x := \left\{ w \in \mathbb{R}^n \;\middle|\; \begin{array}{c} u_x^0(w) \in \mathcal{U}_i(w) \;\; \forall i \in S^0(x) \\ -\nabla_u V_i(w, u_x^0(w)) \in C_i^*(x, u^0(x)) \;\; \forall i \in S^0(x) \end{array} \right\}$$

(b) R_x is a polytope

(c) $x \in R_x$

Proof.

(a) Because of the equality constraint 7.16 it follows that $I_i(w, u_x(w)) \supseteq I_i(x, u^0(x))$ and that $S(w, u_x^0(w)) \supseteq S(x, u^0(x))$ for all $i \in S(x, u^0(x)) = S^0(x)$, all $w \in R_x$. Hence $C_i(w, u_x^0(w)) \subseteq C_i(x, u^0(x))$, which implies $C_i^*(w, u_x^0(w)) \supseteq C_i^*(x, u^0(x))$ for all $i \in S(x, u^0(x)) \subseteq S(w, u_x^0(w))$. It follows from the definition of R_x that $u_x^0(w) \in \mathcal{U}_i(w)$ and that $-\nabla_u V_i(w, u_x^0(w)) \in C_i^*(w, u_x^0(w))$ for all $i \in S(w, u_x^0(w))$. Hence $u = u_x^0(w)$ satisfies necessary and sufficient for optimality for $\mathbb{P}_i(w)$ for all $i \in S(w, u)$, all $w \in R_x$ and, by Proposition 7.16, necessary and sufficient conditions of optimality for $\mathbb{P}(w)$ for all $w \in R_x$. Hence $u_x^0(w) = u^0(w)$ and $V_x^0(w) = V^0(w)$ for all $w \in R_x$.

(b) That R_x is a polytope follows from the facts that the functions $w \mapsto u_x^0(w)$ and $w \mapsto \nabla_u V_i(w, u_x^0(w))$ are affine, the sets $\mathbb{Z}_i$ are polytopic and the sets $C_i^0(x, u^0(x))$ are polyhedral; hence $(w, u_x^0(w)) \in \mathbb{Z}_i$ is a polytopic constraint and $-\nabla_u V_i(w, u_x^0(w)) \in C_i^*(x, u^0(x))$ a polyhedral constraint on w.

(c) That $x \in R_x$ follows from Proposition 7.16 and the fact that $u_x^0(x) = u^0(x)$. ■

Reasoning as in the proof of Proposition 7.10, we obtain the following.

Proposition 7.18 (Piecewise quadratic (affine) solution)**.** *There exists a finite set of points X in $\mathcal{X}$ such that $\{R_x \mid x \in X\}$ is a polytopic partition of $\mathcal{X}$. The value function $V^0(\cdot)$ for $\mathbb{P}(x)$ is strictly convex and*

piecewise quadratic and the minimizer $u^0(\cdot)$ is piecewise affine in $\mathcal{X}$ being equal, respectively, to the quadratic function $V_x^0(\cdot)$ and the affine function $u_x^0(\cdot)$ in each region R_x, $x \in \mathcal{X}$.

7.6 DP Solution of the Constrained LQ Control Problem

A disadvantage in the procedure described in Section 7.4 for determining the piecewise affine receding horizon control law is the dimension Nm of the decision variable $\mathbf{u}$. It seems natural to inquire whether or not dynamic programming (DP), which replaces a multistage decision problem by a sequence of relatively simple single-stage problems, provides a simpler solution. We answer this question by showing how DP may be used to solve the constrained linear quadratic (LQ) problem discussed in Section 7.4. For all $j \in \mathbb{I}_{1:N}$, let $V_j^0(\cdot)$, the optimal value function at time-to-go j, be defined by

$$\begin{aligned} V_j^0(x) &:= \min_u \{V_j(x,u) \mid (x,\mathbf{u}) \in \mathbb{Z}_j\} \\ V_j(x,\mathbf{u}) &:= \sum_{i=0}^{j-1} \ell(x(i),u(i)) + V_f(x(j)) \\ \mathbb{Z}_j &:= \{(x,\mathbf{u}) \mid x(i) \in \mathbb{X}, u(i) \in \mathbb{U}, i \in \mathbb{I}_{0:j-1},\ x(j) \in \mathbb{X}_f\} \end{aligned}$$

with $x(i) := \phi(i;x,\mathbf{u})$; $V_j^0(\cdot)$ is the value function for $\mathbb{P}_j(x)$. As shown in Chapter 2, the constrained DP recursion is

$$V_{j+1}^0(x) = \min_u \{\ell(x,u) + V_j^0(f(x,u)) \mid u \in \mathbb{U}, f(x,u) \in \mathcal{X}_j\} \quad (7.18)$$

$$\mathcal{X}_{j+1} = \{x \in \mathbb{X} \mid \exists\ u \in \mathbb{U} \text{ such that } f(x,u) \in \mathcal{X}_j\} \quad (7.19)$$

where $f(x,u) := Ax + Bu$ with boundary condition

$$V_0^0(\cdot) = V_f(\cdot), \quad \mathcal{X}_0 = \mathbb{X}_f$$

The minimizer of (7.18) is $\kappa_{j+1}(x)$. In the equations, the subscript j denotes time to go, so that current time $i = N - j$. For each j, $\mathcal{X}_j$ is the domain of the value function $V_j^0(\cdot)$ and of the control law $\kappa_j(\cdot)$, and is the set of states that can be steered to the terminal set $\mathbb{X}_f$ in j steps or less by an admissible control that satisfies the state and control constraints. The time-invariant receding horizon control law for horizon j is $\kappa_j(\cdot)$ whereas the optimal policy for problem $\mathbb{P}_j(x)$ is $\{\kappa_j(\cdot), \kappa_{j-1}(\cdot), \ldots, \kappa_1(\cdot)\}$. The data of the problem are identical to the data in Section 7.4.

We know from Section 7.4 that $V_j^0(\cdot)$ is continuous, strictly convex and piecewise quadratic, and that $\kappa_j(\cdot)$ is continuous and piecewise affine on a polytopic partition $\mathcal{P}_{\mathcal{X}_j}$ of $\mathcal{X}_j$. Hence the function $(x,u) \mapsto V(x,u) := \ell(x,u) + V_j^0(Ax+Bu)$ is continuous, strictly convex and piecewise quadratic on a polytopic partition $\mathcal{P}_{\mathbb{Z}_{j+1}}$ of the polytope $\mathbb{Z}_{j+1}$ defined by

$$\mathbb{Z}_{j+1} := \{(x,u) \mid x \in \mathbb{X}, u \in \mathbb{U}, Ax+Bu \in \mathcal{X}_j\}$$

The polytopic partition $\mathcal{P}_{\mathbb{Z}_{j+1}}$ of $\mathbb{Z}_{j+1}$ may be computed as follows: if X is a constituent polytope of $\mathcal{X}_j$, then, from (7.19), the corresponding constituent polytope of $\mathcal{P}_{\mathbb{Z}_{j+1}}$ is the polytope Z defined by

$$Z := \{z = (x,u) \mid x \in \mathbb{X}, u \in \mathbb{U}, Ax+Bu \in X\}$$

Thus Z is defined by a set of linear inequalities; also $\ell(x,u) + V_j^0(f(x,u))$ is quadratic on Z. Thus the techniques of Section 7.5 can be employed for its solution, yielding the piecewise quadratic value function $V_{j+1}^0(\cdot)$, the piecewise affine control law $\kappa_{j+1}(\cdot)$, and the polytopic partition $\mathcal{P}_{\mathcal{X}_{j+1}}$ on which $V_{j+1}^0(\cdot)$ and $\kappa_{j+1}(\cdot)$ are defined. Each problem (7.18) is much simpler than the problem considered in Section 7.4 since m, the dimension of u, is much less than Nm, the dimension of $\mathbf{u}$. Thus, the DP solution is preferable to the direct method described in Section 7.4.

7.7 Parametric Linear Programming

7.7.1 Preliminaries

The parametric linear program $\mathbb{P}(x)$ is

$$V^0(x) = \min_u \{V(x,u) \mid (x,u) \in \mathbb{Z}\}$$

where $x \in \mathcal{X} \subset \mathbb{R}^n$ and $u \in \mathbb{R}^m$, the cost function $V(\cdot)$ is defined by

$$V(x,u) = q'x + r'u$$

and the constraint set $\mathbb{Z}$ is defined by

$$\mathbb{Z} := \{(x,u) \mid Mu \leq Nx + p\}$$

Let $u^0(x)$ denote the solution of $\mathbb{P}(x)$, i.e.,

$$u^0(x) = \arg\min_u \{V(x,u) \mid (x,u) \in \mathbb{Z}\}$$

The solution $u^0(x)$ may be set valued. The parametric linear program (LP) may also be expressed as

$$V^0(x) = \min_u \{V(x,u) \mid u \in \mathcal{U}(x)\}$$

where, as before, the parametric constraint set $\mathcal{U}(x)$ is defined by

$$\mathcal{U}(x) := \{u \mid (x,u) \in \mathbb{Z}\} = \{u \mid Mu \le Nx + p\}$$

Also, as before, the domain of $V^0(\cdot)$ and $u^0(\cdot)$, i.e., the set of points x for which a feasible solution of $\mathbb{P}(x)$ exists, is the set $\mathcal{X}$ defined by

$$\mathcal{X} := \{x \mid \exists u \text{ such that } (x,u) \in \mathbb{Z}\} = \{x \mid \mathcal{U}(x) \neq \emptyset\}$$

The set $\mathcal{X}$ is the projection of $\mathbb{Z}$ (which is a set in (x,u)-space) onto x-space; see Figure 7.1. We assume in the sequel that the problem is well posed, i.e., for each $x \in \mathcal{X}$, $V^0(x) > -\infty$. This excludes problems like $V^0(x) = \inf_u \{x + u \mid -x \le 1, x \le 1\}$ for which $V^0(x) = -\infty$ for all $x \in \mathcal{X} = [-1,1]$. Let $\mathbb{I}_{1:p}$ denote, as usual, the index set $\{1,2,\ldots,p\}$. For all $(x,u) \in \mathbb{Z}$, let $I(x,u)$ denote the set of active constraints at (x,u), i.e.,

$$I(x,u) := \{i \in \mathbb{I}_{1:p} \mid M_i u = N_i x + p_i\}$$

where A_i denotes the ith row of any matrix (or vector) A. Similarly, for any matrix A and any index set I, A_I denotes the matrix with rows A_i, $i \in I$. If, for any $x \in \mathcal{X}$, $u^0(x)$ is unique, the set $I^0(x)$ of constraints active at $(x, u^0(x))$ is defined by

$$I^0(x) := I(x, u^0(x))$$

When $u^0(x)$ is unique, it is a vertex (a face of dimension zero) of the polyhedron $\mathcal{U}(x)$ and is the *unique* solution of

$$M_x^0 u = N_x^0 x + p_x^0$$

where

$$M_x^0 := M_{I^0(x)},\ N_x^0 := N_{I^0(x)},\ p_x^0 := p_{I^0(x)}$$

In this case, the matrix M_x^0 has rank m.

Any face F of $\mathcal{U}(x)$ with dimension $d \in \{1,2,\ldots,m\}$ satisfies $M_i u = N_i x + p_i$ for all $i \in I_F$, all $u \in F$ for some index set $I_F \subseteq \mathbb{I}_{1:p}$. The matrix M_{I_F} with rows M_i, $i \in I_F$, has rank $m - d$, and the face F is defined by

$$F := \{u \mid M_i u = N_i x + p_i, i \in I_F\} \cap \mathcal{U}(x)$$

When $u^0(x)$ is not unique, it is a face of dimension $d \geq 1$ and the set $I^0(x)$ of active constraints is defined by

$$I^0(x) := \{i \mid M_i u = N_i x + p_i \ \forall u \in u^0(x)\} = \{i \mid i \in I(x,u) \ \forall u \in u^0(x)\}$$

The set $\{u \mid M_i u = N_i x + p_i, \ i \in I^0(x)\}$ is a hyperplane in which $u^0(x)$ lies. See Figure 7.7 where $u^0(x_1)$ is unique, a vertex of $\mathcal{U}(x_1)$, and $I^0(x_1) = \{2,3\}$. If, in Figure 7.7, $r = -e_1$, then $u^0(x_1) = F_2(x_1)$, a face of dimension 1; $u^0(x_1)$ is, therefore, set valued. Since $u \in \mathbb{R}^m$ where $m = 2$, $u^0(x_1)$ is a facet, i.e., a face of dimension $m - 1 = 1$. Thus $u^0(x_1)$ is a set defined by $u^0(x_1) = \{u \mid M_1 u \leq N_1 x_1 + p_1, \ M_2 u = N_2 x_1 + p_2, \ M_3 u \leq N_3 x_1 + p_3\}$. At each $z = (x,u) \in \mathbb{Z}$, i.e., for each (x,u) such that $x \in \mathcal{X}$ and $u \in \mathcal{U}(x)$, the cone $C(z) = C(x,u)$ of first-order feasible variations is defined, as before, by

$$C(z) := \{h \in \mathbb{R}^m \mid M_i h \leq 0, \ i \in I(z)\} = \{h \in \mathbb{R}^m \mid M_{I(z)} h \leq 0\}$$

If $I(z) = I(x,u) = \emptyset$ (no constraints are active), $C(z) = \mathbb{R}^m$ (all variations are feasible).

Since $u \mapsto V(x,\cdot)$ is convex and differentiable, and $\mathcal{U}(x)$ is polyhedral for all x, the parametric LP $\mathbb{P}(x)$ satisfies the assumptions of Proposition 7.8. Hence, repeating Proposition 7.8 for convenience, we have

Proposition 7.19 (Optimality conditions for parametric LP)**.** *A necessary and sufficient condition for u to be a minimizer for the parametric LP $\mathbb{P}(x)$ is*

$$u \in \mathcal{U}(x) \text{ and } -\nabla_u V(x,u) \in C^*(x,u)$$

where $\nabla_u V(x,u) = r$ and $C^*(x,u)$ is the polar cone of $C(x,u)$.

An important difference between this result and that for the parametric QP is that $\nabla_u V(x,u) = r$ and, therefore, does not vary with x or u. We now use this result to show that both $V^0(\cdot)$ and $u^0(\cdot)$ are piecewise affine. We consider the simple case when $u^0(x)$ is unique for all $x \in \mathcal{X}$.

7.7.2 Minimizer $u^0(x)$ Is Unique for all $x \in \mathcal{X}$

Before proceeding to obtain the solution to a parametric LP when the minimizer $u^0(x)$ is unique for each $x \in \mathcal{X}$, we look first at the simple example illustrated in Figure 7.7, which shows the constraint set $\mathcal{U}(x)$

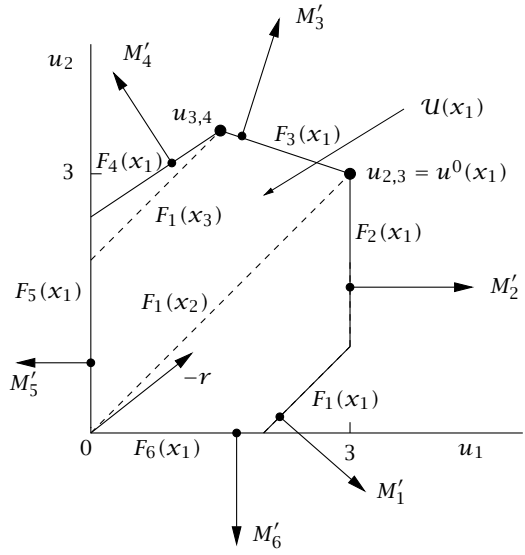


Figure 7.7: Solution to a parametric LP.

for various values of the parameter x in the interval $[x_1, x_3]$. The set $\mathcal{U}(x_1)$ has six faces: $F_1(x_1), F_2(x_1), F_3(x_1), F_4(x_1), F_5(x_1)$, and $F_6(x_1)$. Face $F_1(x)$ lies in the hyperplane $\mathcal{H}_1(x)$ that varies linearly with x; each face $F_i(x)$, $i = 2, \ldots, 6$, lies in the hyperplane $\mathcal{H}_i$ that does *not* vary with x. All the faces vary with x as shown so that $\mathcal{U}(x_2)$ has four faces: $F_1(x_2)$, $F_3(x_2)$, $F_4(x_2)$, and $F_5(x_2)$; and $\mathcal{U}(x_3)$ has three faces: $F_1(x_3)$, $F_4(x_3)$, and $F_5(x_3)$. The face $F_1(x)$ is shown for three values of x: $x = x_1$ (the bold line), and $x = x_2$ and $x = x_3$ (dotted lines). It is apparent that for $x \in [x_1, x_2]$, $u^0(x) = u_{2,3}$ in which $u_{2,3}$ is the intersection of $\mathcal{H}_2$ and $\mathcal{H}_3$, and $u^0(x_3) = u_{3,4}$, in which $u_{3,4}$ is the intersection of $\mathcal{H}_3$ and $\mathcal{H}_4$. It can also be seen that $u^0(x)$ is unique for all $x \in \mathcal{X}$.

We now return to the general case. Suppose, for some $\in \mathcal{X}$, $u^0(x)$ is the unique solution of $\mathbb{P}(x)$; $u^0(x)$ is the unique solution of

$$M_x^0 u = N_x^0 x + p_x^0$$

It follows that $u^0(x)$ is the trivial solution of the simple equality constrained problem defined by

$$V^0(x) = \min_u \{V(x, u) \mid M_x^0 u = N_x^0 x + p_x^0\} \tag{7.20}$$

The solution $u^0(x)$ of this equality constrained problem is trivial because it is determined entirely by the equality constraints; the cost plays no part.

The optimization problem (7.20) motivates us, as in parametric quadratic programming, to consider, for any parameter w "close" to x, the simpler equality constrained problem $\mathbb{P}_x(w)$ defined by

$$V_x^0(w) = \min_u \{V(w,u) \mid M_x^0 u = N_x^0 w + p_x^0\}$$
$$u_x^0(w) = \arg\min_u \{V(w,u) \mid M_x^0 u = N_x^0 w + p_x^0\}$$

Let $u_x^0(w)$ denote the solution of $\mathbb{P}_x(w)$. Because, for each $x \in \mathcal{X}$, the matrix M_x^0 has full rank m, there exists an index set I_x such that $M_{I_x} \in \mathbb{R}^{m\times m}$ is invertible. Hence, for each w, $u_x^0(w)$ is the unique solution of

$$M_{I_x} u = N_{I_x} w + p_{I_x}$$

so that for all $x \in \mathcal{X}$, all $w \in \mathbb{R}^m$

$$u_x^0(w) = K_x w + k_x \tag{7.21}$$

where $K_x := (M_{I_x})^{-1} N_{I_x}$ and $k_x := (M_{I_x})^{-1} p_{I_x}$. In particular, $u^0(x) = u_x^0(x) = K_x x + k_x$. Since $V_x^0(x) = V_x(x, u_x^0(w)) = q'x + r' u_x^0(w)$, it follows that

$$V_x^0(x) = (q' + r' K_x) x + r' k_x$$

for all $x \in \mathcal{X}$, all $w \in \mathbb{R}^m$. Both $V_x^0(\cdot)$ and $u_x^0(\cdot)$ are affine in x.

It follows from Proposition 7.19 that $-r \in C^*(x, u^0(x)) = \text{cone}\{M_i' \mid i \in I^0(x) = I(x, u^0(x))\} = \text{cone}\{M_i' \mid i \in I_x\}$. Since $\mathbb{P}_x(w)$ satisfies the conditions of Proposition 7.8, we may proceed as in Section 7.3.4 and define, for each $x \in \mathcal{X}$, the set R_x^0 as in (7.5)

$$R_x^0 := \left\{ w \in \mathbb{R}^n \;\middle|\; \begin{array}{c} u_x^0(w) \in \mathcal{U}(w) \\ -\nabla_u V(w, u_x^0(w)) \in C^*(x, u^0(x)) \end{array} \right\}$$

It then follows, as shown in Proposition 7.9, that for any $x \in \mathcal{X}$, $u_x^0(w)$ is optimal for $\mathbb{P}(w)$ for all $w \in R_x^0$. Because $\mathbb{P}(w)$ is a parametric LP, however, rather than a parametric QP, it is possible to simplify the definition of R_x^0. We note that $\nabla_u V(w, u_x^0(w)) = r$ for all $x \in \mathcal{X}$, all $w \in \mathbb{R}^m$. Also, it follows from Proposition 7.8, since $u^0(x)$ is optimal for $\mathbb{P}(x)$, that $-\nabla_u V(x, u^0(x)) = -r \in C^*(x)$ so that the second condition in the definition above for R_x^0 is automatically satisfied. Hence

we may simplify our definition for R_x^0; for the parametric LP, R_x^0 may be defined by

$$R_x^0 := \{w \in \mathbb{R}^n \mid u_x^0(w) \in \mathcal{U}(w)\} \tag{7.22}$$

Because $u_x^0(\cdot)$ is affine, it follows from the definition of $\mathcal{U}(w)$ that R_x^0 is polyhedral. The next result follows from the discussion in Section 7.3.4.

Proposition 7.20 (Solution of $\mathbb{P}$)**.** *For any $x \in \mathcal{X}$, $u_x^0(w)$ is optimal for $\mathbb{P}(w)$ for all w in the set R_x^0 defined in* (7.22)*.*

Finally, the next result characterizes the solution of the parametric LP $\mathbb{P}(x)$ when the minimizer is unique.

Proposition 7.21 (Piecewise affine cost and solution)**.**

(a) There exists a finite set of points X in $\mathcal{X}$ such that $\{R_x^0 \mid x \in X\}$ is a polyhedral partition of $\mathcal{X}$.

(b) The value function $V^0(\cdot)$ for $\mathbb{P}(x)$ and the minimizer $u^0(\cdot)$ are piecewise affine in $\mathcal{X}$ being equal, respectively, to the affine functions $V_x^0(\cdot)$ and $u_x^0(\cdot)$ in each region R_x, $x \in \mathcal{X}$.

(c) The value function $V^0(\cdot)$ and the minimizer $u^0(\cdot)$ are continuous in $\mathcal{X}$.

Proof. The proof of parts (a) and (b) follows, apart from minor changes, the proof of Proposition 7.10. The proof of part (c) uses the fact that $u^0(x)$ is unique, by assumption, for all $x \in \mathcal{X}$ and is similar to the proof of Proposition 7.13. ■

7.8 Constrained Linear Control

The previous results on parametric linear programming may be applied to obtain the optimal receding horizon control law when the system is linear, the constraints polyhedral, and the cost linear as is done in a similar fashion in Section 7.4 where the cost is quadratic. The optimal control problem is therefore defined as in Section 7.4, except that the stage cost $\ell(\cdot)$ and the terminal cost $V_f(\cdot)$ are now defined by

$$\ell(x, u) := q'x + r'u \qquad V_f(x) := q_f'x$$

As in Section 7.4, the optimal control problem $\mathbb{P}_N(x)$ may be expressed as

$$V_N^0(x) = \min_{\mathbf{u}} \{V_N(x, \mathbf{u}) \mid \mathbf{M}\mathbf{u} \leq \mathbf{N}x + \mathbf{p}\}$$

where, now

$$V_N(x, \mathbf{u}) = \mathbf{q}'x + \mathbf{r}'\mathbf{u}$$

Hence the problem has the same form as that discussed in Section 7.7 and may be solved as shown there.

It is possible, using a simple transcription, to use the solution of $\mathbb{P}_N(x)$ to solve the optimal control problem when the stage cost and terminal cost are defined by

$$\ell(x, u) := |Qx|_p + |Ru|_p, \quad V_f(x) := \left|Q_f x\right|_p$$

where $|\cdot|_p$ denotes the p-norm and p is either 1 or ∞.

7.9 Computation

Our main purpose above was to establish the structure of the solution of parametric linear or QPs and, hence, of the solutions of constrained linear optimal control problems when the cost is quadratic or linear. We have not presented algorithms for solving these problem although; there is now a considerable literature on this topic. One of the earliest algorithms (Serón, De Doná, and Goodwin, 2000) is enumeration based: checking every active set to determine if it defines a non-empty region in which the optimal control is affine. There has recently been a return to this approach because of its effectiveness in dealing with systems with relatively high state dimension but a low number of constraints (Feller, Johansen, and Olaru, 2013). The enumeration based procedures can be extended to solve mixed-integer problems. While the early algorithms for parametric linear and quadratic programming have exponential complexity, most later algorithms are based on a linear complementarity formulation and execute in polynomial time in the number of regions; they also use symbolic perturbation to select a unique and continuous solution when one exists (Columbano, Fukudu, and Jones, 2009). Some research has been devoted to obtaining approximate solutions with lower complexity but guaranteed properties such as stability (Borrelli, Bemporad, and Morari, 2017, Chapter 13).

Toolboxes for solving parametric linear and quadratic programming problems include the The Multi-Parametric Toolbox in MATLAB and MPT3 described in (Herceg, Kvasnica, Jones, and Morari, 2013).

A feature of parametric problems is that state dimension is not a reliable indicator of complexity. There exist problems with two states that require over 10^5 regions and problems with 80 states that require

only hundreds of regions. While problems with state dimension less than, say, 4 can be expected to have reasonable complexity, higher dimension problems may or may not have manageable complexity.

7.10 Notes

Early work on parametric programming, e.g., (Dantzig, Folkman, and Shapiro, 1967) and (Bank, Guddat, Klatte, Kummer, and Tanner, 1983), was concerned with the sensitivity of optimal solutions to parameter variations. Solutions to the parametric linear programming problem were obtained relatively early (Gass and Saaty, 1955) and (Gal and Nedoma, 1972). Solutions to parametric QPs were obtained in (Serón et al., 2000) and (Bemporad, Morari, Dua, and Pistikopoulos, 2002) and applied to the determination of optimal control laws for linear systems with polyhedral constraints. Since then a large number of papers on this topic have appeared, many of which are reviewed in (Alessio and Bemporad, 2009). Most papers employ the Kuhn-Tucker conditions of optimality in deriving the regions R_x, $x \in X$. Use of the polar cone condition was advocated in (Mayne and Raković, 2002) in order to focus on the geometric properties of the parametric optimization problem and avoid degeneracy problems. Section 7.5, on parametric piecewise quadratic programming, is based on (Mayne, Raković, and Kerrigan, 2007). The example in Section 7.4 was first computed by Raković (Mayne and Raković, 2003). That results from parametric linear and quadratic programming can be employed, instead of maximum theorems, to establish continuity of $u^0(\cdot)$ and, hence, of $V^0(\cdot)$, was pointed out by Bemporad et al. (2002) and Borrelli (2003, p. 37).

Much research has been devoted to obtaining reliable algorithms; see the survey papers (Alessio and Bemporad, 2009) and (Jones, Barić, and Morari, 2007) and the references therein. Jones (2017, Chapter 13) provides a useful review of approximate explicit control laws of specified complexity that nevertheless guarantee stability and recursive feasibility.

7.11 Exercises

Exercise 7.1: QP with equality constraints

Obtain the solution u^0 and the value V^0 of the equality constrained optimization problem $V^0 = \min_u\{V(u) \mid h(u) = 0\}$ where $V(u) = (1/2)u'Ru + r'u + c$ and $h(u) := Mu - p$.

Exercise 7.2: Parametric QP with equality constraints

Show that the solution $u^0(x)$ and the value $V^0(x)$ of the parametric optimization problem $V^0(x) = \min_u\{V(x,u) \mid h(x,u) = 0\}$ where $V(x,u) := (1/2)x'Qx + u'Sx + (1/2)u'Ru + q'x + r'u + c$ and $h(x,u) := Mu - Nx - p$ have the form $u^0(x) = Kx + k$ and $V^0(x) = (1/2)x'\bar{Q}x + \bar{q}'x + s$. Determine $\bar{Q}$, $\bar{q}$, s, K, and k.

Exercise 7.3: State and input trajectories in constrained LQ problem

For the constrained linear quadratic problem defined in Section 7.4, show that $\mathbf{u} := (u(0), u(1), \ldots, u(N-1))$ and $\mathbf{x} := (x(0), x(1), \ldots, x(N))$, where $x(0) = x$ and $x(i) = \phi(i; x, \mathbf{u})$, $i = 0, 1, \ldots, N$, satisfy

$$\mathbf{x} = \mathbf{F}x + \mathbf{G}\mathbf{u}$$

and determine the matrices $\mathbf{F}$ and $\mathbf{G}$; in this equation $\mathbf{u}$ and $\mathbf{x}$ are column vectors. Hence show that $V_N(x, \mathbf{u})$ and $\mathbb{Z}$, defined respectively in (7.9) and (7.11), satisfy (7.12) and (7.13), and determine $\mathbf{Q}$, $\mathbf{R}$, $\mathbf{M}$, $\mathbf{N}$, and $\mathbf{p}$.

Exercise 7.4: The parametric LP with unique minimizer

For the example of Figure 7.7, determine $u^0(x)$, $V^0(x)$, $I^0(x)$, and $C^*(x)$ for all x in the interval $[x_1, x_3]$. Show that $-r$ lies in $C^*(x)$ for all x in $[x_1, x_3]$.

Exercise 7.5: Cost function and constraints in constrained LQ control problem

For the constrained linear control problem considered in Section 7.8, determine the matrices $\mathbf{M}$, $\mathbf{N}$, and $\mathbf{p}$ that define the constraint set $\mathbb{Z}$, and the vectors $\mathbf{q}$ and $\mathbf{r}$ that define the cost $V_N(\cdot)$.

Exercise 7.6: Cost function in constrained linear control problem

Show that $|x|_p$, $p = 1$ and $p = \infty$, may be expressed as $\max_j\{s_j'x \mid j \in J\}$ and determine s_i, $i \in I$ for the two cases $p = 1$ and $p = \infty$. Hence show that the optimal control problem in Section 7.8 may be expressed as

$$V_N^0(x) = \min_{\mathbf{v}}\{V_N(x, \mathbf{v}) \mid \mathbf{M}\mathbf{v} \leq \mathbf{N}x + \mathbf{p}\}$$

where, now, $\mathbf{v}$ is a column vector whose components are $u(0), u(1), \ldots, u(N-1)$, $\ell_x(0)$, $\ell_x(1), \ldots, \ell_x(N)$, $\ell_u(0), \ell_u(1), \ldots, \ell_u(N-1)$ and f; the cost $V_N(x, \mathbf{v})$ is now defined by

$$V_N(x, \mathbf{v}) = \sum_{i=0}^{N-1} (\ell_x(i) + \ell_u(i)) + f$$

Finally, $\mathbf{Mv} \le \mathbf{N}x + \mathbf{p}$ now specifies the constraints $u(i) \in \mathbb{U}$ and $x(i) \in \mathbb{X}$, $|Ru(i)|_p \le \ell_u(i)$, $|Qx(i)|_p \le \ell_x(i)$, $i = 0, 1, \ldots, N-1$, $x(N) \in \mathbb{X}_f$, and $\left|Q_f x(N)\right| \le f$. As before, $\mathbf{x}^+ = \mathbf{Fx} + \mathbf{Gu}$.

Exercise 7.7: Is QP constraint qualification relevant to MPC?

Continuity properties of the MPC control law are often used to establish robustness properties of MPC such as robust asymptotic stability. In early work on continuity properties of linear model MPC, Scokaert, Rawlings, and Meadows (1997) used results on continuity of QPs with respect to parameters to establish MPC stability under perturbations. For example, Hager (1979) considered the following QP

$$\min_u (1/2)u'Hu + h'u + c$$

subject to

$$Du \le d$$

and established that the QP solution u^0 and cost V^0 are Lipschitz continuous in the data of the QP, namely the parameters H, h, D, d. To establish this result Hager (1979) made the following assumptions.

- The solution is unique for all H, h, D, d in a chosen set of interest.
- The rows of D corresponding to the constraints active at the solution are linearly independent. The assumption of linear independence of active constraints is a form of *constraint qualification*.

(a) First we show that some form of constraint qualification is required to establish continuity of the QP solution with respect to matrix D. Consider the following QP example that does not satisfy Hager's constraint qualification assumption.

$$H = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \qquad D = \begin{bmatrix} 1 & 1 \\ -1 & -1 \end{bmatrix} \qquad d = \begin{bmatrix} 1 \\ -1 \end{bmatrix} \qquad h = \begin{bmatrix} -1 \\ -1 \end{bmatrix} \qquad c = 1$$

Find the solution u^0 for this problem.

Next perturb the D matrix to

$$D = \begin{bmatrix} 1 & 1 \\ -1+\epsilon & -1 \end{bmatrix}$$

in which $\epsilon > 0$ is a small perturbation. Find the solution to the perturbed problem. Are V^0 and u^0 continuous in parameter D for this QP? Draw a sketch of the feasible region and cost contours for the original and perturbed problems. What happens to the feasible set when D is perturbed?

(b) Next consider MPC control of the following system with state inequality constraint and no input constraints

$$A = \begin{bmatrix} -1/4 & 1 \\ -1 & 1/2 \end{bmatrix} \qquad B = \begin{bmatrix} 1 & 1 \\ -1 & -1 \end{bmatrix} \qquad x(k) \le \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad k \in \mathbb{I}_{0:N}$$

Using a horizon $N = 1$, eliminate the state $x(1)$ and write out the MPC QP for the input $u(0)$ in the form given above for $Q = R = I$ and zero terminal penalty. Find an initial condition x_0 such that the MPC constraint matrix D and vector d are identical to those given in the previous part. Is this $x_0 \in \mathcal{X}_N$?

Are the rows of the matrix of active constraints linearly independent in this MPC QP on the set $\mathcal{X}_N$? Are the MPC control law $\kappa_N(x)$ and optimal value function $V_N^0(x)$ Lipschitz continuous on the set $\mathcal{X}_N$ for this system? Explain the reason if these two answers differ.

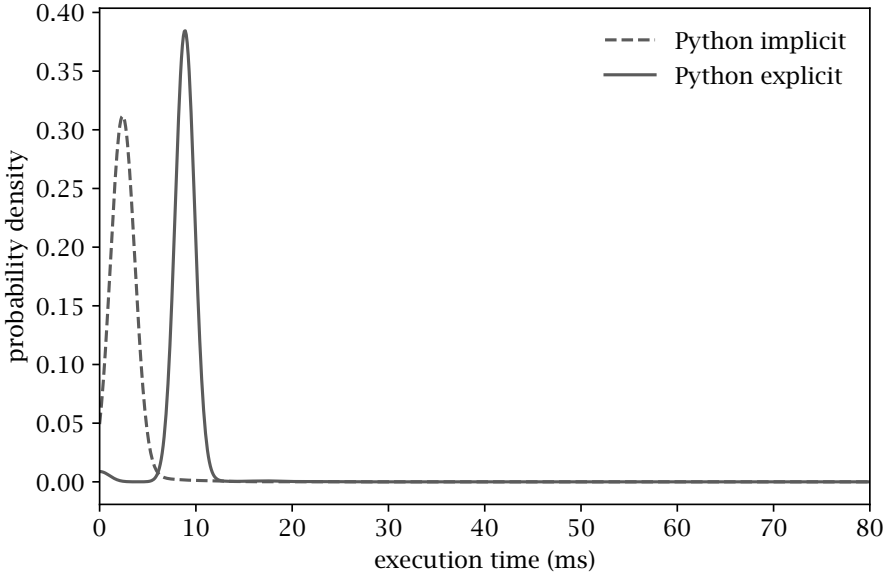


Figure 7.8: Solution times for explicit and implicit MPC for $N = 20$. Plot shows kernel density estimate for 10,000 samples using a Gaussian kernel ($\sigma = 1$ ms).

Exercise 7.8: Explicit versus implicit

Using the system from Figure 7.6, find the explicit control law for horizon $N = 20$ (you should find 1719 regions). Implement a simple lookup function for the explicit control law. Randomly sample a large number of points (≥ 1000) from $\mathcal{X}_N$ and compare execution times for explicit MPC (via the lookup function) and implicit MPC (via solving a QP). Which method is better? Example results are shown in Figure 7.8, although your times may vary significantly. How could you improve your lookup function?

Exercise 7.9: Cascaded MPC and PID

Consider a Smart Tank$^{\text{TM}}$ of liquid whose height h evolves according to

$$\tau \frac{dh}{dt} + h = Kq, \quad \tau = 10, K = 1$$

with q the (net) inflow. The tank is Smart$^{\text{TM}}$ in that it has an integrated PI controller that computes

$$q = K_c \left(h_{\text{sp}} - h + \frac{1}{\tau_c} \epsilon \right)$$
$$\epsilon = \int h_{\text{sp}} - h \, dt$$

so that the height of the tank returns to h_{sp} automatically. Unfortunately, the controller parameters are not very Smart$^{\text{TM}}$, as they are fixed permanently at $K_c = 1/2$ and $\tau_c = 1$.

(a) Simulate the closed-loop behavior of the system starting from $h = -1$, $\epsilon = 0$ with $h_{\text{sp}} \equiv 0$.

(b) Design an MPC controller to choose h_{sp}. As a cost function take

$$\ell(h, \epsilon, q, h_{\text{sp}}) = 5(h^2 + \epsilon^2) + q^2 + 10h_{\text{sp}}^2$$

so that the controller drives the system to $h = \epsilon = 0$. Choose $\Delta = 1$. How does performance compare to the previous case? How much storage (i.e., how many floating-point numbers must be stored) to implement this controller?

(c) Add the constraint $q \in [-0.2, 0.2]$ to the MPC formulation, and design an explicit MPC controller valid for $h \in [-5, 5]$ and $\epsilon \in [-10, 10]$ (use `solvempqp.m` from Figure 7.6, and add constraints $Ep \leq e$ to only search the region of interest). How large does N have to be so that the full region is covered? How much storage is needed to implement this controller?

Exercise 7.10: Explicit economic MPC for electricity arbitrage

Electricity markets are often subject to real-time pricing, whereby the cost of purchasing electricity varies with time. Suppose that you have a large battery that allows you to buy electricity at one time and then sell it back to the grid at another. We can model this as a simple integrator system

$$x^+ = x + u$$

with x representing the amount of stored energy in the tank, and u giving the amount of electricity that is purchased for the battery $u > 0$ or discharged from the battery and sold back to the grid ($u < 0$). We wish to find an explicit control law based on the initial condition $x(0)$ a known forecast of electricity prices $c(0), c(1), \ldots, c(N-1)$.

(a) To start, suppose that u is constrained to the interval $[-1, 1]$ but x is unconstrained. A reasonable optimization problem is

$$\begin{aligned} \min_{\mathbf{u}} \quad & \sum_{k=0}^{N-1} c(k)u(k) + 0.1u(k)^2 \\ \text{s.t.} \quad & x(k+1) = x(k) + u(k) \\ & u(k) \in [-1, 1] \end{aligned}$$

where the main component of the objective function is the cost of electricity purchase/sale with a small penalty added to discourage larger transactions. By removing the state evolution equation, formulate an explicit quadratic programming problem with N variables (the $u(k)$) and $N+1$ parameters ($x(0)$ and the price forecast $c(k)$). What is a theoretical upper bound on the number of regions in the explicit control law? Assuming that $x(0) \in [-10, 10]$ and each $c(k) \in [-1, -1]$, find the explicit control law for a few small values of N. (Consider using `solvempqp.m` from Figure 7.6; you will need to add constraints $Ep \leq e$ on the parameter vector to make sure the regions are bounded.) How many regions do you find?

(b) To make the problem more realistic, we add the constraint $x(k) \in [-10, 10]$ to the optimization, as well as an additional penalty on stored inventory. The optimization problem is then

$$\begin{aligned} \min_{\mathbf{u}} \quad & \sum_{k=0}^{N-1} c(k)u(k) + 0.1u(k)^2 + 0.01x(k)^2 \\ \text{s.t.} \quad & x(k+1) = x(k) + u(k) \\ & u(k) \in [-1, 1] \\ & x(k) \in [-10, 10] \end{aligned}$$

Repeat the previous part but using the new optimization problem.

(c) Suppose you wish to solve this problem with a 7-day horizon and a 1-hour time step. Can you use the explicit solution of either formulation? (Hint: for comparison, there are roughly 10^{80} atoms in the observable universe.)

Bibliography

A. Alessio and A. Bemporad. A survey on explicit model predictive control. In L. Magni, D. Raimondo, and F. Allgöwer, editors, *Nonlinear Model Predictive Control - Towards New Challenging Applications*, pages 345–369. Springer Berlin / Heidelberg, 2009.

B. Bank, J. Guddat, D. Klatte, B. Kummer, and K. Tanner. *Non-linear parametric optimization*. Birkhäuser Verlag, Basel, Boston, Stuttgart, 1983.

A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos. The explicit linear quadratic regulator for constrained systems. *Automatica*, 38(1):3–20, 2002.

F. Borrelli. *Constrained Optimal Control of Linear and Hybrid Systems*. Springer-Verlag Berlin Heidelberg, 2003.

F. Borrelli, A. Bemporad, and M. Morari. *Predictive Control for Linear and Hybrid Systems*. Cambridge University Press, 2017.

S. Columbano, K. Fukudu, and C. N. Jones. An output sensitive algorithm for multiparametric LCPs with sufficient matrices. *Polyhedral Computation*, 48: 73, 2009.

G. B. Dantzig, J. Folkman, and N. Z. Shapiro. On the continuity of the minimum set of a continuous function. *J. Math. Anal. Appl.*, 17(3):519–548, 1967.

C. Feller, T. A. Johansen, and S. Olaru. An improved algorithm for combinatorial multi-parametric quadratic programming. *Automatica*, 49(5):1370–1376, 2013.

T. Gal and J. Nedoma. Multiparametric linear programming. *Management Science*, 18(7):406–422, 1972.

S. I. Gass and T. L. Saaty. The computational algorithm for the parametric objective function. *Naval Research Logistics Quarterly*, 2:39–45, 1955.

W. W. Hager. Lipschitz continuity for constrained processes. *SIAM J. Cont. Opt.*, 17(3):321–338, 1979.

M. Herceg, M. Kvasnica, C. N. Jones, and M. Morari. Multi-Parametric Toolbox 3.0. In *Proc. of the European Control Conference*, pages 502–510, Zürich, Switzerland, July 17–19 2013. `http://control.ee.ethz.ch/~mpt`.

C. N. Jones. Approximate receding horizon control. In F. Borrelli, A. Bemporad, and M. Morari, editors, *Predictive Control for Linear and Hybrid Systems*, pages 277–300. Cambridge University Press, 2017.

C. N. Jones, M. Barić, and M. Morari. Multiparametric linear programming with applications in control. *Eur. J. Control*, 13:152–170, 2007.

D. Q. Mayne and S. V. Raković. Optimal control of constrained piecewise affine discrete-time systems using reverse transformation. In *Proceedings of the IEEE 2002 Conference on Decision and Control*, volume 2, pages 1546 – 1551 vol.2, Las Vegas, USA, 2002.

D. Q. Mayne and S. V. Raković. Optimal control of constrained piecewise affine discrete-time systems. *Comp. Optim. Appl.*, 25(1-3):167–191, 2003.

D. Q. Mayne, S. V. Raković, and E. C. Kerrigan. Optimal control and piecewise parametric programming. In *Proceedings of the European Control Conference 2007*, pages 2762–2767, Kos, Greece, July 2–5 2007.

P. O. M. Scokaert, J. B. Rawlings, and E. S. Meadows. Discrete-time stability with perturbations: Application to model predictive control. *Automatica*, 33(3): 463–470, 1997.

M. M. Serón, J. A. De Doná, and G. C. Goodwin. Global analytical model predictive control with input constraints. In *Proceedings of the 39th IEEE Conference on Decision and Control*, pages 154–159, Sydney, Australia, December 2000.

8
Numerical Optimal Control

8.1 Introduction

Numerical optimal control methods are at the core of every model predictive control implementation, and algorithmic choices strongly affect the reliability and performance of the resulting MPC controller. The aim of this chapter is to explain some of the most widely used algorithms for the numerical solution of optimal control problems. Before we start, recall that the ultimate aim of the computations in MPC is to find a numerical approximation of the optimal feedback control $u^0(x_0)$ for a given current state x_0. This state x_0 serves as initial condition for an optimal control problem, and $u^0(x_0)$ is obtained as the first control of the trajectory that results from the numerical solution of the optimal control problem. Due to a multitude of approximations, the feedback law usually is not exact. Some of the reasons are the following.

- The system model is only an approximation of the real plant.
- The horizon length is finite instead of infinite.
- The system's differential equation is discretized.
- The optimization problem is not solved exactly.

While the first two of the above are discussed in Chapters 2 and 3 of this book, the last two are due to the numerical solution of the optimal control problems arising in model predictive control and are the focus of this chapter. We argue throughout the chapter that it is not a good idea to insist that the finite horizon MPC problem shall be solved exactly. First, it usually is impossible to solve a simulation or optimization problem without any numerical errors, due to finite precision arithmetic and finite computation time. Second, it might not even be desirable to solve the problem as exactly as possible, because the necessary computations might lead to large feedback delays or an excessive use of CPU resources. Third, in view of the other errors that are necessarily introduced in the modeling process and in the MPC problem

formulation, errors due to an inexact numerical solution do not significantly change the closed-loop performance, at least as long as they are smaller than the other error sources. Thus, the optimal choice of a numerical method for MPC should be based on a trade-off between accuracy and computation time. There are, however, tremendous differences between different numerical choices, and it turns out that some methods, compared to others, can have significantly lower computational cost for achieving the same accuracy. Also, reliability is an issue, as some methods might more often fail to find an approximate solution than other methods. Thus, the aim of this chapter is to give an overview of the necessary steps toward the numerical solution of the MPC problem, and to discuss the properties of the different choices that can be made in each step.

8.1.1 Discrete Time Optimal Control Problem

When working in a discrete time setting, the MPC optimization problem that needs to be solved numerically in each time step, for a given system state x_0, can be stated as follows. For ease of notation, we introduce the sequence of future control inputs on the prediction horizon, $\mathbf{u} := (u(0), u(1), \ldots, u(N-1))$, as well as the predicted state trajectories $\mathbf{x} := (x(0), x(1), \ldots, x(N))$.

$$\begin{aligned}
\underset{\mathbf{x},\mathbf{u}}{\text{minimize}} \quad & \sum_{k=0}^{N-1} \ell(x(k), u(k)) + V_f(x(N)) && && \text{(8.1a)}\\
\text{subject to} \quad & x(0) = x_0 && && \text{(8.1b)}\\
& x(k+1) = f(x(k), u(k)), && k = 0, 1, \ldots, N-1 && \text{(8.1c)}\\
& (x(k), u(k)) \in \mathbb{Z}, && k = 0, 1, \ldots, N-1 && \text{(8.1d)}\\
& x(N) \in \mathbb{X}_f && && \text{(8.1e)}
\end{aligned}$$

We call the above optimization problem $\mathbb{P}_N(x_0)$ to indicate its dependence on the parameter x_0, and denote the resulting optimal value function by $V_N(x_0)$. The value function $V_N(x_0)$ is mostly of theoretical interest, and is in practice computed only for those values of x_0 that actually arise in the MPC context. In this chapter, we are mostly interested in fast and efficient ways to find an optimal solution, which we denote by $(\mathbf{x}^0(x_0), \mathbf{u}^0(x_0))$. The solution need not be unique for a given problem $\mathbb{P}_N(x_0)$, and in a mathematically correct notation one could only define the set $S^0(x_0)$ of all solutions to $\mathbb{P}_N(x_0)$. Usually one tries to ensure by a proper formulation that the MPC optimization

problems have unique solutions, however, so that the set of solutions is a singleton, $S^0(x_0) = \{(\mathbf{x}^0(x_0), \mathbf{u}^0(x_0))\}$.

A few remarks are in order regarding the statement of the optimization problem (8.1a)-(8.1e). First, as usual in the field of optimization, we list the optimization variables of problem $\mathbb{P}_N(x_0)$ below the word "minimize." Here, they are given by the sequences $\mathbf{x}$ and $\mathbf{u}$. The constraints of the problem appear after the keywords "subject to" and restrict the search for the optimal solution. Let us discuss each of them briefly: constraint (8.1b) ensures that the trajectory $\mathbf{x} = (x(0), \dots)$ starts at x_0, and uniquely determines $x(0)$. Constraints (8.1c) ensure that the state and control trajectories obey the system dynamics for all time steps $k = 0, \dots, N-1$. If in addition to $x(0)$ one would also fix the controls $\mathbf{u}$, the whole state trajectory $\mathbf{x}$ would be uniquely determined by these constraints. Constraints (8.1d) shall ensure that the state control pairs $(x(k), u(k))$ are contained in the set $\mathbb{Z}$ at each time step k. Finally, the terminal state constraint (8.1e) requires the final state to be in a given terminal set $\mathbb{X}_f$. The set of all variables $(\mathbf{x}, \mathbf{u})$ that satisfy all constraints (8.1b)-(8.1e) is called the *feasible set*. Note that the feasible set is the intersection of all constraint sets defined by the individual constraints.

8.1.2 Convex Versus Nonconvex Optimization

The most important dividing line in the field of optimization is between convex and nonconvex optimization problems. If an optimization problem is convex, every local minimum is also a global one. One can reliably solve most convex optimization problems of interest, finding the globally optimal solution in polynomial time. On the other hand, if a problem is not convex, one can usually not find the global minimum. Even if one has accidentally found the global minimum, one usually cannot certify that it is the global minimum. Thus, in nonconvex optimization, one has usually to accept that one is only able to find feasible or locally optimal points. Fortunately, if one has found such a point, one usually is also able to certify that it is a feasible or locally optimal point. But in the worst case, one might not be able to find even a feasible point, without knowing if this is due to the problem being infeasible, or the optimization algorithm being just unable to find points in the feasible set. Thus, the difference between convex and nonconvex has significant implications in practice. To say it in the words of the famous mathematical optimizer R. Tyrrell Rockafellar, "The great watershed in optimization is not between linearity and nonlinearity, but

convexity and nonconvexity."

When is a given optimization problem a *convex optimization problem*? By definition, an optimization problem is convex if its feasible set is a convex set and if its objective function is a convex function. In MPC, we usually have freedom in choosing the objective function, and in most cases one chooses a convex objective function. For example, the sum of quadratic functions of the form $\ell(x,u) = x'Qx + u'Ru$ with positive semidefinite matrices Q and R is a convex function. Usually, one also chooses the terminal cost V_f to be a convex function, so that the objective function is a convex function.

Likewise, one usually chooses the terminal set $\mathbb{X}_f$ to be a convex set. For example, one might choose an ellipsoid $\mathbb{X}_f = \{x \mid x'Px \leq 1\}$ with a positive definite matrix P, which is a convex set. Very often, one is lucky and also has convex constraint sets $\mathbb{Z}$, for example box constraints on $x(k)$ and $u(k)$. The initial-value constraint (8.1b) restricts the variable $x(0)$ to be in the point set $\{x_0\}$, which is convex. Thus, most of the constraints in the MPC optimization problem usually can be chosen to be convex. On the other hand, the constraints (8.1c) reflect the system dynamics $x(k+1) = f(x(k),u(k))$ for all k, and these might or might not describe a convex set. Interestingly, it turns out that they describe a convex set if the system model is linear or affine, i.e., if $f(x(k),u(k)) = Ax(k) + Bu(k) + c$ with matrices A, B and vector c of appropriate dimensions. This follows because the solution set of linear equalities is an affine set, which is convex. Conversely, if the system model is nonlinear, the solution set of the dynamic constraints (8.1c) is most likely not a convex set. Thus, we can formulate a modification of Rockafellar's statement above: in MPC practice, the great watershed between convex and nonconvex optimization problems usually coincides with the division line between linear and nonlinear system models.

One speaks of *linear MPC* if a linear or affine simulation model is used, and of *nonlinear MPC* otherwise. When speaking of linear MPC, one implicitly assumes that all other constraints and the objective function are chosen to be convex, but not necessarily linear. In particular, in linear MPC, the objective function usually is chosen to be convex quadratic. Thus, in the MPC literature, the term *linear MPC* is used as if it coincides with "convex linear MPC." Theoretically possible "nonconvex linear MPC" methods, where the system model is linear but where the cost or constraint sets are not convex, are not of great practical interest. On the other hand, for nonlinear MPC, i.e., when a nonlinear model is used, convexity usually is lost anyway, and there are no

implicit convexity assumptions on the objective and constraints, such that the term *nonlinear MPC* nearly always coincides with "nonconvex nonlinear MPC."

Example 8.1: Nonlinear MPC

We regard a simple MPC optimization problem of the form (8.1) with one dimensional state x and control u, system dynamics $f(x,u) = x + u - 2u^2$, initial value $x_0 = 1$, and horizon length $N = 1$, as follows

$$\begin{aligned} \underset{x(0),x(1),u(0)}{\text{minimize}} \quad & x(0)^2 + u(0)^2 + 10x(1)^2 && \text{(8.2a)} \\ \text{subject to} \quad & x(0) = x_0 && \text{(8.2b)} \\ & x(1) = x(0) + u(0) - 2u(0)^2 && \text{(8.2c)} \\ & -1 \le u(0) \le 1 && \text{(8.2d)} \end{aligned}$$

First, we observe that the optimization problem has a three-dimensional space of optimization variables. To check convexity of the problem, we first regard the objective, which is a sum of positive quadratic functions, thus a convex function. On the other hand, we need to check convexity of the feasible set. The initial-value constraint (8.2b) fixes one of the three variables, thus selects a two-dimensional affine subset in the three-dimensional space. This subset is described by $x(0) = 1$ while $u(0)$ and $x(1)$ remain free. Likewise, the control bounds in (8.2d) cut away all values for $u(0)$ that are less than -1 or more than $+1$, thus, there remains only a straight stripe of width 2 in the affine subset, still extending to infinity in the $x(1)$ direction. This straight two-dimensional stripe still is a convex set. The system equation (8.2c) is a nonlinear constraint that selects a curve out of the stripe, which is visualized on the left of Figure 8.1. This curve is not a convex set, because the connecting lines between two points on the curve are not always contained in the curve. In a formula, the feasible set is given by $\{(x(0), x(1), u(0)) \mid x(0) = 1, u(0) \in [-1,1], x(1) = 1 + u(0) - 2u(0)^2\}$.

Even though the objective function is convex, the fact that the optimization problem has a nonconvex feasible set can lead to different local minima. This is indeed the case in our example. To see this, let us evaluate the objective function on all feasible points and plot it as a function of $u(0)$. This *reduced objective function* $\psi(u)$ can be obtained by inserting $x(0) = 1$ and $x(1) = x(0)+u(0)-2u(0)^2$ into the objective $x(0)^2+u(0)^2+10x(1)^2$, which yields $\psi(u) = 1+u^2+10(1+u-2u^2)^2 = 11 + 20u - 29u^2 - 40u^3 + 40u^4$. This reduced objective is visualized

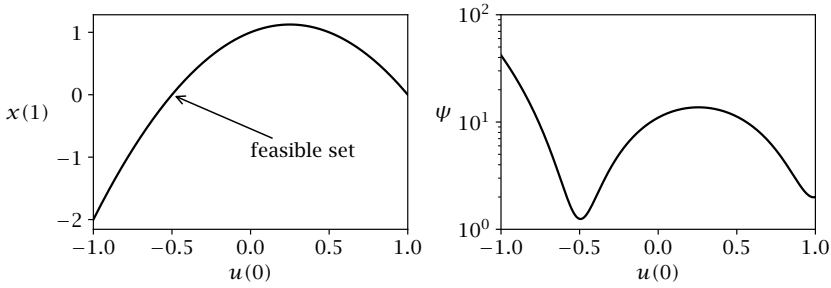


Figure 8.1: Feasible set and reduced objective $\psi(u(0))$ of the nonlinear MPC Example 8.1.

on the right of Figure 8.1 and it can clearly be seen that two different locally optimal solutions exist, only one of which is the globally optimal choice. □

8.1.3 Simultaneous Versus Sequential Optimal Control

The optimal control problem (OCP) (8.1) can be passed to an appropriate optimization routine without any modification. In this case, the optimization variables are given by both, the state trajectory $\mathbf{x}$ as well as the control trajectory $\mathbf{u}$. The pair $(\mathbf{x}, \mathbf{u})$ is consistent with the initial value x_0 and the simulation model if and only if the constraints (8.1b) and (8.1c) are satisfied, which is the case for any feasible solution of the problem. During the optimization calculations, however, these constraints might be violated, and the state trajectory $\mathbf{x}$ might not be a valid simulation corresponding to the controls $\mathbf{u}$. Since the optimization routine has to simultaneously solve the simulation and the optimization problem, one calls this approach the *simultaneous approach to optimal control*.

On the other hand, one could use the constraints (8.1b)-(8.1c) to find the unique feasible state trajectory $\mathbf{x}$ for any given control trajectory $\mathbf{u}$. We denote, as before in Chapter 2, the state $x(k)$ that results from a given initial condition x_0 and a given control trajectory $\mathbf{u} = (u(0), u(1), \ldots, u(N-1))$ by $\phi(k; x_0, \mathbf{u})$. Using this expression, that can be computed by a simple forward simulation routine, we can replace the equalities (8.1b)-(8.1c) by the trivial equalities $x(k) = \phi(k; x_0, \mathbf{u})$ for

$k = 0, 1, \ldots, N$. And these constraints can be used to eliminate the complete state trajectory $\mathbf{x} = (x(0), x(1), \ldots, x(N))$ from the optimization problem. The optimization problem in this reduced variable space is given by

$$\begin{aligned}
\underset{\mathbf{u}}{\text{minimize}} \quad & \sum_{k=0}^{N-1} \ell(\phi(k; x_0, \mathbf{u}), u(k)) + V_f(\phi(N; x_0, \mathbf{u})) && \text{(8.3a)}\\
\text{subject to} \quad & (\phi(k; x_0, \mathbf{u}), u(k)) \in \mathbb{Z}, \quad k = 0, 1, \ldots, N-1 && \text{(8.3b)}\\
& \phi(N; x_0, \mathbf{u}) \in \mathbb{X}_f && \text{(8.3c)}
\end{aligned}$$

If this reduced optimization problem is solved by an iterative optimization routine, in each iteration, one performs a sequence of two steps. First, for given $\mathbf{u}$, the simulation routine computes the state trajectory $\mathbf{x}$, and second, the optimization routine updates the control variables $\mathbf{u}$ to iterate toward an optimal solution. Due to this sequential evaluation of simulation and optimization routines, one calls this approach the *sequential approach to optimal control*. Though the simultaneous and the sequential approach solve equivalent optimization problems, their approach toward finding the solutions is different.

For linear MPC problems, where the system model is linear, the difference between the two approaches regards mostly the sparsity structure of the optimization problem, as discussed in Chapter 6 and in Section 8.8.4. In this case, one usually calls the reduced optimization problem (8.3) the *condensed problem*, and the computational process to generate the data for the condensed problem (8.3) from the data of the original problem (8.1) is called *condensing*. Though the condensed problem has fewer variables, the matrices defining it may have more nonzero entries than the original problem. Which of the two formulations leads to shorter computation times for a given problem depends on the number of states, controls and constraints, the specific sparsity structures, and on the horizon length N. For small N, condensing is typically preferable, while for large N, it is advisable to apply a sparse convex solver to the original problem in the full variable space. Despite the different sparsity structure, and different cost per iteration, many widely used convex optimization algorithms perform identical iterates on both problems, because the eliminated constraints are linear and are exactly respected in each iteration in both the condensed as well as the original problem formulation.

For nonlinear MPC problems, the sequential and simultaneous approach can lead to significantly different optimization iterations. Even

if both problems are addressed with the same optimization algorithm and are initialized with the same initial guess, i.e., the same $\mathbf{u}$ for both, together with the corresponding simulation result $\mathbf{x}$, the optimization iterations typically differ after the first iteration, such that the two formulations can need a significantly different number of iterations to converge; they might even converge to different local solutions or one formulation might converge while the other does not. As a rule of thumb, the sequential approach is preferable if the optimization solver cannot exploit sparsity and the system is stable, while the simultaneous approach is preferable for unstable nonlinear systems, for problems with state constraints, and for systems which need implicit simulation routines.

Example 8.2: Sequential approach

We regard again the simple MPC optimization problem (8.2a), but eliminate the states as a function of $\mathbf{u} = (u(0))$ by $x(0) = \phi(0; x_0, \mathbf{u}) = x_0$ and $x(1) = \phi(1; x_0, \mathbf{u}) = x_0 + u(0) - 2u(0)^2$. The reduced optimization problem in the sequential approach is then given by

$$\underset{u(0)}{\text{minimize}} \quad x_0^2 + u(0)^2 + 10\left(x_0 + u(0) - 2u(0)^2\right)^2 \tag{8.4a}$$

$$\text{subject to} \quad -1 \le u(0) \le 1 \tag{8.4b}$$

□

8.1.4 Continuous Time Optimal Control Problem

In most nonlinear MPC applications and many linear MPC applications, the system dynamics are not given in discrete time but in continuous time, in form of differential equations

$$\frac{dx}{dt} = f_c(x, u)$$

For notational convenience, we usually denote differentiation with respect to time by a dot above the quantity, i.e., we can abbreviate the above equations by $\dot{x} = f_c(x, u)$. Both the state and control trajectories are functions of continuous time, and we denote them by $x(t)$ and $u(t)$. The trajectories need only to be defined on the time horizon of interest, i.e., for all $t \in [0, T]$, where T is the horizon length. If we do not assume any discretization, and if we use the shorthand symbols

$x(\cdot)$ and $u(\cdot)$ to denote the state and control trajectories, the continuous time optimal control problem (OCP) can be formulated as follows

$$\begin{aligned}
\underset{x(\cdot),u(\cdot)}{\text{minimize}} \quad & \int_0^T \ell_c(x(t),u(t))\,dt + V_f(x(T)) && \text{(8.5a)}\\
\text{subject to} \quad & x(0) = x_0 && \text{(8.5b)}\\
& \dot{x}(t) = f_c(x(t),u(t)), \quad t \in [0,T] && \text{(8.5c)}\\
& (x(t),u(t)) \in \mathbb{Z}, \qquad t \in [0,T] && \text{(8.5d)}\\
& x(T) \in \mathbb{X}_f && \text{(8.5e)}
\end{aligned}$$

It is important to note that the continuous time optimal control problem is an infinite-dimensional optimization problem with infinite-dimensional decision variables and an infinite number of constraints, because the time index t runs through infinitely many values $t \in [0,T]$. This is in contrast to discrete time, where the finite number of time indices $k \in \mathbb{I}_{0:N}$ leads to finitely many decision variables and constraints.

There exists a variety of methods to numerically solve continuous time OCPs. What all approaches have in common is that at one point, the infinite-dimensional problem needs to be discretized. One family of methods first formulates what is known as the Hamilton-Jacobi-Bellman (HJB) equation, a partial differential equation for the value function, which depends on both state space and time, and then discretizes and solves it. Unfortunately, due to the "curse of dimensionality," this approach is only practically applicable to systems with small state dimensions, say less than five, or to the special case of unconstrained linear systems with quadratic costs.

A second family of methods, the *indirect methods*, first derive optimality conditions in continuous time by algebraic manipulations that use similar expressions as the HJB equation; they typically result in the formulation of a boundary-value problem (BVP), and only discretize the resulting continuous time BVP at the very end of the procedure. One characterizes the indirect methods often as "first optimize, then discretize." A third class of methods, the *direct methods*, first discretizes the continuous time OCP, to convert it into a finite-dimensional optimization problem. The finite-dimensional optimization problem can then be solved by tailored algorithms from the field of numerical optimization. The direct methods are often characterized as "first discretize, then optimize." These methods are most widely used in MPC applications and are therefore the focus of this chapter.

To sketch the discretization methods, we look at the continuous time optimal control problem (8.5). In a direct method, we replace the continuous index $t \in [0, T]$ by a discrete integer index. For this aim, we can divide the time horizon T into N intervals, each of length $h = \frac{T}{N}$, and evaluate the quantities of interest only for the discrete time points $t = hk$ with $k \in \mathbb{I}_{0:N}$. We use the notation $h\mathbb{I}_{0:N} = \{0, h, 2h, \ldots, Nh\}$, such that we can use the expression "$t \in h\mathbb{I}_{0:N}$" to indicate that t is only considered at these discrete time points. To discretize the OCP, the objective integral is replaced by a Riemann sum, and the time derivative by a finite difference approximation: $\dot{x}(t) \approx \frac{x(t+h)-x(t)}{h}$. As before in discrete time, we denote the sequence of discrete states by $\mathbf{x} = (x(0), x(h), x(2h), \ldots, x(Nh))$ and the sequence of controls by $\mathbf{u} = (u(0), u(h), \ldots, u(Nh-h))$.

$$\begin{aligned}
\underset{\mathbf{x},\mathbf{u}}{\text{minimize}} \quad & \sum_{t \in h\mathbb{I}_{0:N-1}} h\ell_{\mathrm{c}}(x(t), u(t)) \; + \; V_f(x(Nh)) && \text{(8.6a)}\\
\text{subject to} \quad & x(0) = x_0 && \text{(8.6b)}\\
& \frac{x(t+h)-x(t)}{h} = f_{\mathrm{c}}(x(t), u(t)), \quad t \in h\mathbb{I}_{0:N-1} && \text{(8.6c)}\\
& (x(t), u(t)) \in \mathbb{Z}, \qquad\qquad t \in h\mathbb{I}_{0:N-1} && \text{(8.6d)}\\
& x(Nh) \in \mathbb{X}_f && \text{(8.6e)}
\end{aligned}$$

It is easily checked that the constraints (8.6b)-(8.6c) uniquely determine all states $\mathbf{x}$ if the control sequence $\mathbf{u}$ is given. The above problem is exactly in the form of the discrete time optimization problem (8.1), if one uses the definitions $\ell(x, u) := h\ell_{\mathrm{c}}(x, u)$ and $f(x, u) := x + hf_{\mathrm{c}}(x, u)$. This simple way to go from continuous to discrete time, in particular the idea to solve a differential equation $\dot{x} = f_{\mathrm{c}}(x, u)$ by the simple difference method $x^+ = x + hf_{\mathrm{c}}(x, u)$, is originally due to Leonhard Euler (1707–1783), and is therefore called the *Euler integration method*. The Euler method is not the only possible integration method, and in fact, not the most efficient one. Numerical analysts have investigated the simulation of differential equations for more than two centuries, and discovered powerful discretization methods that have much lower computational cost and higher accuracy than the Euler method and are therefore more widely used in practice. These are the topic of the next section.

8.2 Numerical Simulation

The classical task of numerical simulation is the solution of *initial-value problems*. An initial-value problem is characterized by an initial state value x_0 at time 0, and a differential equation $\dot{x} = f(t, x)$ that the solution $x(t)$ should satisfy on the time interval of interest, i.e., for all $t \in [0, T]$ with $T > 0$. In particular, we are interested in computing an approximation of the final state $x(T)$. In this section, we allow an explicit dependence of the *right-hand-side* function $f(t, x)$ on time. To be consistent with the literature in the field of numerical simulation—and deviating from the notation in other chapters of this book—we use t here as the first input argument of $f(t, x)$. The time dependence might in particular be due to a fixed control trajectory $u(t)$, and if a given system is described by the continuous time ODE $\dot{x} = f_c(x, u)$, the time dependent right-hand-side function is defined by $f(t, x) := f_c(x, u(t))$. The choice of the control trajectory $u(t)$ is not the focus in this section, but becomes important later when we treat the solution of optimal control problems. Instead, in this section, we just review results from the field of numerical simulation of ordinary differential equations—which is sometimes also called *numerical integration*—that are most relevant to continuous time optimal control computations.

Throughout this section we consider the following initial-value problem

$$x(0) = x_0, \quad \dot{x}(t) = f(t, x(t)) \quad \text{for} \quad t \in [0, T] \tag{8.7}$$

with a given right-hand-side function $f : [0, T] \times \mathbb{R}^n \to \mathbb{R}^n$. We denote the exact solution, if it exists, by $x(t)$. Existence of a unique solution of the initial-value problem is guaranteed by a classical theorem by Émile Picard (1856–1941) and Ernst Lindelöf (1870–1946), which requires the function f to be continuous with respect to time t and Lipschitz continuous with respect to the state x. Lipschitz continuity is stronger than continuity and requires the existence of a constant $L > 0$ such that the following inequality

$$|f(t, x) - f(t, y)| \leq L |x - y| \tag{8.8}$$

holds for all $t \in [0, T]$ and all $x, y \in \mathbb{R}^n$. In many cases of interest, the function f is not defined on the whole state space, or there might exist no global Lipschitz constant L for all states x and y. Fortunately, a local version of the Picard-Lindelöf Theorem exists that only needs Lipschitz continuity in a neighborhood of the point $(0, x_0)$ and still

ensures the existence of a unique solution $x(t)$ for sufficiently small T. Local Lipschitz continuity is implied by continuous differentiability, which is easy to verify and holds for most functions f arising in practice. In fact, the function f usually is many times differentiable in both its arguments, and often even infinitely many times—for example, in the case of polynomials or other analytic functions. The higher differentiability of f also leads to higher differentiability of the solution trajectory $x(t)$ with respect to t, and is at the basis of the higher-order integration methods that are widely used in practice.

Because all numerical integration methods produce only approximations to the true solution $x(t)$, we use a different symbol for these approximations, namely $\widetilde{x}(t)$. The numerical approximation is usually only exact for the initial value, where we simply set $\widetilde{x}(0) := x_0$. For the final state at time T, we aim to have a small error $E(T) := |\widetilde{x}(T) - x(T)|$, at low computational cost. All integration methods divide the time horizon of interest into smaller intervals, and proceed by making a sequence of integration steps, one per interval. For simplicity, assume that the steps are equidistant, and that in total N steps of size $h = T/N$ are taken. In each step, the integration method makes a *local error*, and the combined effect of the accumulated local errors at time t, i.e., the distance $E(t) = |\widetilde{x}(t) - x(t)|$, is called the *global error*. After the first integrator step, local and global error coincide because the integration starts on the exact trajectory, but in subsequent steps, the global error typically grows while the local errors remain of similar size.

8.2.1 Explicit Runge-Kutta Methods

Let us first investigate the Euler integrator, that iterates according to the update rule

$$\widetilde{x}(t+h) = \widetilde{x}(t) + hf(t, \widetilde{x}(t))$$

starting with $\widetilde{x}(0) = x_0$. Which local error do we make in each step? For local error analysis, we assume that the starting point $\widetilde{x}(t)$ was on an exact trajectory, i.e., equal to $x(t)$, while the result of the integrator step $\widetilde{x}(t+h)$ is different from $x(t+h)$. For the analysis, we assume that the true trajectory $x(t)$ is twice continuously differentiable with bounded second derivatives, which implies that its first-order Taylor series satisfies $x(t+h) = x(t) + h\dot{x}(t) + O(h^2)$, where $O(h^2)$ denotes an arbitrary function whose size shrinks faster than h^2 for $h \to 0$. Since the first derivative is known exactly, $\dot{x}(t) = f(t, x(t))$, and was used in the Euler

integrator, we immediately obtain that $|\widetilde{x}(t+h) - x(t+h)| = O(h^2)$. Because the global error is the accumulated and propagated effect of the local errors, and because the total number of integrator steps grows linearly with $1/h$, one can show that the global error at the end of the interval of interest is of size $1/h\ O(h^2) = O(h)$, i.e., of first order. For this reason one says that the Euler method is a first-order integration method. The Euler integrator is easy to remember and easy to implement, but the number of time steps that are needed to obtain even a moderate accuracy can be reduced significantly if higher-order methods are used.

Like the Euler integrator, all *one-step integration methods* create a discrete time system of the form

$$\widetilde{x}(t+h) = \widetilde{x}(t) + \Phi(t, \widetilde{x}(t), h)$$

Here, the map Φ approximates the integral $\int_t^{t+h} f(\tau, x(\tau))\ d\tau$. If Φ would be equal to this integral, the integration method would be exact, due to the identity

$$x(t+h) - x(t) = \int_t^{t+h} \dot{x}(\tau)\ d\tau = \int_t^{t+h} f(\tau, x(\tau))\ d\tau$$

While the Euler integrator approximates the integral by the expression $\Phi(t, x, h) = hf(t, x(t))$ that has an error of $O(h^2)$ and needs only one evaluation of the function f per step, one can find more accurate approximations by allowing more than one function evaluation per integration step. This idea leads directly to the Runge-Kutta (RK) integration methods, that are named after Carl Runge (1856–1927) and Martin Wilhelm Kutta (1867–1944).

The classical Runge-Kutta method (RK4). One of the most widely used methods invented by Runge and Kutta performs four function evaluations, as follows.

$$\begin{aligned} k_1 &= f(t, x) \\ k_2 &= f(t + h/2, x + (h/2)k_1) \\ k_3 &= f(t + h/2, x + (h/2)k_2) \\ k_4 &= f(t + h, x + hk_3) \\ \Phi &= (h/6)k_1 + (h/3)k_2 + (h/3)k_3 + (h/6)k_4 \end{aligned}$$

It is a fourth-order method, and therefore often abbreviated RK4. Since it is one of the most competitive methods for the accuracies that are

typically needed in applications, the RK4 integrator is one of the most widely used integration methods for simulation of ordinary differential equations. A comparison of the RK4 method with Euler's first-order method and a second-order method named after Karl Heun (1859–1929) is shown in Figure 8.2.

Example 8.3: Integration methods of different order

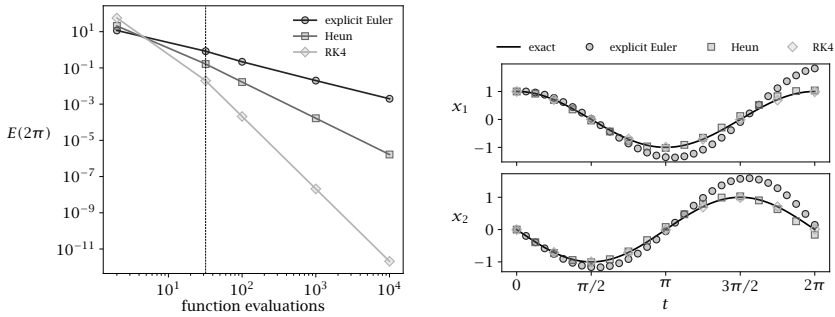
We regard the simulation of the linear ordinary differential equation (ODE)

$$\dot{x} = Ax \qquad \text{with} \qquad A = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$$

over the interval $T = 2\pi$, starting at $x_0 = [1, 0]'$. The analytic solution of this system is known to be $x(t) = \exp(At)x_0 = [\cos(t), -\sin(t)]'$, such that the final state is given by $x(2\pi) = [1, 0]'$. To investigate the performance of different methods, we divide the time horizon into N equal integration intervals of length $h = 2\pi/N$. Note that a Runge-Kutta method with s stages needs in total $M := Ns$ function evaluations. We compare the Euler ($s = 1$), Heun ($s = 2$), and RK4 method ($s = 4$). For each integration method we evaluate the global error at the end of the integration interval, $E(2\pi) = |\widetilde{x}(2\pi) - x(2\pi)|$, and plot it as a function of the number of function evaluations, M, in Figure 8.2. We use a doubly logarithmic scale, i.e., plot $\log(\epsilon)$ versus $\log(M)$, to show the effect of the order. Note that the slope of the higher-order methods is an integer multiple of the slope of the Euler method. Also note that the accuracy for each investigated method cannot exceed a certain base value due to the finite precision arithmetic, and that this limit is reached for the RK4 integrator at approximately $M = 10^5$. After this point, increasing the number of integration steps does not further improve the accuracy. □

The Butcher tableau. A general explicit Runge-Kutta method with s stages performs the following computations in each integration step

$$\begin{aligned}
k_1 &= f(t + c_1 h,\ x \quad\quad\quad\



(a) Accuracy vs. function evaluations. (b) Simulation results for $M = 32$.

Figure 8.2: Performance of different integration methods.

It is important to note that on the right-hand side of each row, only those k_i values are used that are already computed. This property holds for every *explicit* integration method, and makes it possible to explicitly evaluate the first s equations one after the other to obtain all values $k_1, \ldots, k_s$ for the summation in the last line. One usually summarizes the coefficients of a Runge-Kutta method in what is known as a Butcher tableau (after John C. Butcher, born 1933) given by

$$\begin{array}{c|cccc} c_1 & & & & \\ c_2 & a_{21} & & & \\ c_3 & a_{31} & a_{32} & & \\ \vdots & \ddots & \ddots & & \\ c_s & a_{s1} & \cdots & a_{s,s-1} & \\ \hline & b_1 & b_2 & \cdots & b_s \end{array}$$

The Butcher tableau of three popular RK methods is stated below

Euler

$$\begin{array}{c|c} 0 & \\ \hline & 1 \end{array}$$

Heun

$$\begin{array}{c|cc} 0 & & \\ 1 & 1 & \\ \hline & 1/2 & 1/2 \end{array}$$

RK4

$$\begin{array}{c|cccc} 0 & & & & \\ 1/2 & 1/2 & & & \\ 1/2 & 0 & 1/2 & & \\ 1 & 0 & 0 & 1 & \\ \hline & 1/6 & 2/6 & 2/6 & 1/6 \end{array}$$

Note that the b_i coefficients on the bottom always add to one. An interesting fact is that an s-stage explicit Runge-Kutta method can never

have a higher order than s. And only for orders equal or less than four exist explicit Runge-Kutta methods for which the order and the number of stages coincide.

8.2.2 Stiff Equations and Implicit Integrators

Unfortunately, some differential equations cannot reliably be solved by explicit integration methods; it can occur that even if the underlying ODE is stable, the integration method is not. Let us regard the scalar linear ODE

$$\dot{x} = \lambda x$$

with initial condition x_0 as a test case. The exact solution is known to be $x(t) = e^{\lambda t} x_0$. When this ODE is solved by an explicit Euler method, it iterates like $x^+ = x + h\lambda x$ and it is easy to see that the explicit solution is given by $\widetilde{x}(kh) = (1 + h\lambda)^k x_0$. For positive λ, this leads to exponential growth, which is not surprising given that the exact ODE solution grows exponentially. If λ is a large negative number, however, the exact solution $x(t)$ would decay very fast to zero, while the Euler integrator is unstable and oscillates with exponentially growing amplitude if h is larger than $2/(-\lambda)$. A similar observation holds for all explicit integration methods.

The most perturbing fact is that the explicit integration methods are extremely unstable exactly because of the fact that the system is extremely stable. Extremely stable ODEs are called *stiff* equations. For stiff ODE $\dot{x} = f(t, x)$, some of the eigenvalues of the Jacobian f_x have extremely large negative real parts, which lead to extremely stable subdynamics. Exactly these extremely stable subdynamics let the explicit integrators fail; even for relatively short stepsizes h, they overshoot the true solution and exhibit unstable oscillations. These oscillations do not just lead to inaccurate solutions, but in fact they quickly exceed the range of computer representable numbers (10^{308} for double precision), such that the explicit integrator just outputs "NaN" ("not a number") most of the time.

Fortunately, there exist integration methods that remain stable even for stiff ODE. Their only drawback is that they are implicit, i.e., they require the solution of an equation system to compute the next step. The simplest of these implicit methods is called the implicit Euler method and it iterates according to

$$x^+ = x + hf(t + h, x^+)$$

Note that the desired output value x^+ appears also on the right side of the equation. For the scalar linear ODE $\dot{x} = \lambda x$, the implicit Euler step is determined by $x^+ = x + h\lambda x^+$, which can explicitly be solved to give $x^+ = x/(1 - h\lambda)$. For any negative λ, the denominator is larger than one, and the numerical approximation $\widetilde{x}(kh) = x_0/(1 - h\lambda)^k$ therefore decays exponentially, similar to the exact solution. An integration method which has the desirable property that it remains stable for the test ODE $\dot{x} = \lambda x$ whenever $\mathrm{Re}(\lambda) < 0$ is called *A-stable*. While none of the explicit Runge-Kutta methods is A-stable, the implicit Euler method is A-stable. But it has a low order. Can we devise A-stable methods that have a higher order?

8.2.3 Implicit Runge-Kutta and Collocation Methods

Once we accept that we need to solve a nonlinear equation system in order to compute an integration step, we can extend the family of Runge-Kutta methods by allowing diagonal and upper-triangular entries in the Butcher tableau. Our hope is to find integration methods that are both A-stable and have a high order. A general *implicit Runge-Kutta method* with s stages solves the following nonlinear system in each integration step

$$\begin{array}{lllllll}
k_1 & = f(t + c_1 h\,, & x + & h\,(\,a_{11}k_1 & + a_{12}k_2 & + \ldots & + a_{1,s}k_s)\,) \\
k_2 & = f(t + c_2 h\,, & x + & h\,(\,a_{21}k_1 & + a_{22}k_2 & + \ldots & + a_{2,s}k_s)\,) \\
 & \vdots & & & \vdots & & \vdots \\
k_s & = f(t + c_s h\,, & x + & h\,(\,a_{s1}k_1 & + a_{s,2}k_2 & + \ldots & + a_{s,s}k_s)\,) \\
 & & & & & & \\
\Phi & = & & h\,(\,b_1\,k_1 & + b_2\,k_2 & + \ldots & + b_s\,k_s\,)
\end{array}$$

Note that the upper s equations are implicit and form a root-finding problem with sn nonlinear equations in sn unknowns, where s is the number of RK stages and n is the state dimension of the differential equation $\dot{x} = f(t, x)$. Nonlinear root-finding problems are usually solved by Newton's method, which is treated in the next section. For Newton's method to work, one has to assume that the Jacobian of the residual function is invertible. For the RK equations above, this can be shown to always hold if the time step h is sufficiently small, depending on the right-hand-side function f. After the values $k_1, \ldots, k_s$ have been computed, the last line can be executed and yields the resulting map $\Phi(t, x, h)$. The integrator then uses the map Φ to proceed to the next integration step exactly as the other one-step methods, according to

the update equation

$$\widetilde{x}(t+h) = \widetilde{x}(t) + \Phi(t, \widetilde{x}(t), h)$$

For implicit integrators, contrary to the explicit ones, the map Φ cannot easily be written down as a series of function evaluations. Evaluation of $\Phi(t, x, h)$ includes the root-finding procedure and typically needs several evaluations of the root-finding equations and of their derivatives. Thus, an s-stage implicit Runge-Kutta method is significantly more expensive per step compared to an s-stage explicit Runge-Kutta method. Implicit integrators are usually preferable for stiff ordinary differential equations, however, due to their better stability properties.

Many different implicit Runge-Kutta methods exist, and each of them can be defined by its Butcher tableau. For an implicit RK method, at least one of the diagonal and upper-triangular entries (a_{ij} with $j \geq i$) is nonzero. Some methods try to limit the implicit part for easier computations. For example, the diagonally implicit Runge-Kutta methods have only the diagonal entries nonzero while the upper-triangular part remains zero.

Collocation methods. One particularly popular subclass of implicit Runge-Kutta methods is formed by the *collocation methods*. An s-stage collocation method first fixes the values c_i of the Butcher tableau, and chooses them so that they are all different and in the unit interval, i.e., $0 \leq c_1 < c_2 < \ldots < c_s \leq 1$. The resulting time points $(t + hc_i)$ are called the *collocation points*, and their choice uniquely determines all other entries in the Butcher tableau. The idea of collocation is to approximate the trajectory on the collocation interval by a polynomial $\widetilde{x}(\tau)$ for $\tau \in [t, t+h]$, and to require satisfaction of the ODE $\dot{x} = f(t, x)$ only on the collocation points, i.e., impose the conditions $\dot{\widetilde{x}}(t + hc_i) = f(t+hc_i, \widetilde{x}(t+hc_i))$ for $i = 1, \ldots, s$. Together with the requirement that the approximating polynomial $\widetilde{x}(\tau)$ should start at the initial value, i.e., $\widetilde{x}(t) = x$, we have $(s+1)$ conditions such that the polynomial needs to have $(s+1)$ coefficients, i.e., have the degree s, to yield a well-posed root-finding problem.

The polynomial $\widetilde{x}(\tau)$ can be represented in different ways, which are related via linear basis changes and therefore lead to numerically equivalent root-finding problems. One popular way is to parameterize $\widetilde{x}(\tau)$ as the interpolating polynomial through the initial value x and the state values at the collocation points. This only gives a unique parameterization if $c_1 \neq 0$. To have a more generally applicable derivation

of collocation, we use instead the value x together with the s derivative values $k_1, \ldots, k_s$ at the collocation time points to parameterize $\tilde{x}(\tau)$. More precisely, we use the identity $\tilde{x}(\tau) = x + \int_t^\tau \dot{\tilde{x}}(\tau_1; k_1, k_2, \ldots, k_s)\, d\tau_1$, where $\dot{\tilde{x}}(\cdot)$ is the time derivative of $\tilde{x}(\tau)$, and therefore a polynomial of degree $(s-1)$ that can be represented by s coefficients. Fortunately, due to the fact that all collocation points are different, the interpolating polynomial through the s vectors $k_1, \ldots, k_s$ is well defined and can easily be represented in a Lagrange basis, with basis functions $L_i\left(\frac{\tau - t}{h}\right)$ that are one on the i-th collocation point and zero on all others.[1] Collocation thus approximates $\dot{x}(\tau)$ by the polynomial

$$\dot{\tilde{x}}(\tau; k_1, k_2, \ldots, k_s) := k_1 L_1\left(\frac{\tau - t}{h}\right) + k_2 L_2\left(\frac{\tau - t}{h}\right) + \ldots + k_s L_s\left(\frac{\tau - t}{h}\right)$$

and $x(\tau)$ by its integral

$$\tilde{x}(\tau; x, k_1, k_2, \ldots, k_s) := x + \int_t^\tau \dot{\tilde{x}}(\tau_1; k_1, k_2, \ldots, k_s)\, d\tau_1$$

To obtain the state at the collocation point $(t + c_i h)$, we just need to evaluate $\tilde{x}(t + c_i h; x, k_1, k_2, \ldots, k_s)$, which is given by the following integral

$$x + \int_t^{t+c_i h} \dot{\tilde{x}}(\tau_1; k_1, k_2, \ldots, k_s)\, d\tau_1 = x + \sum_{j=1}^{s} k_j h \underbrace{\int_0^{c_i} L_j(\sigma)\, d\sigma}_{=:a_{ij}}$$

Note that the integrals over the Lagrange basis polynomials depend only on the relative positions of the collocation time points, and directly yield the coefficients a_{ij}. Likewise, to obtain the coefficients b_i, we evaluate $\tilde{x}(t + h; x, k_1, k_2, \ldots, k_s)$, which is given by

$$x + \int_t^{t+h} \dot{\tilde{x}}(\tau; k_1, k_2, \ldots, k_s)\, d\tau = x + \sum_{i=1}^{s} k_i h \underbrace{\int_0^{1} L_i(\sigma)\, d\sigma}_{=:b_i}$$

In Figure 8.3, the difference between the exact solution $x(\tau)$ and the collocation polynomial $\tilde{x}(\tau)$ as well as the difference between their

[1] The Lagrange basis polynomials are defined by

$$L_i(\sigma) := \prod_{1 \le j \le s,\, j \ne i} \frac{(\sigma - c_j)}{(c_i - c_j)}$$

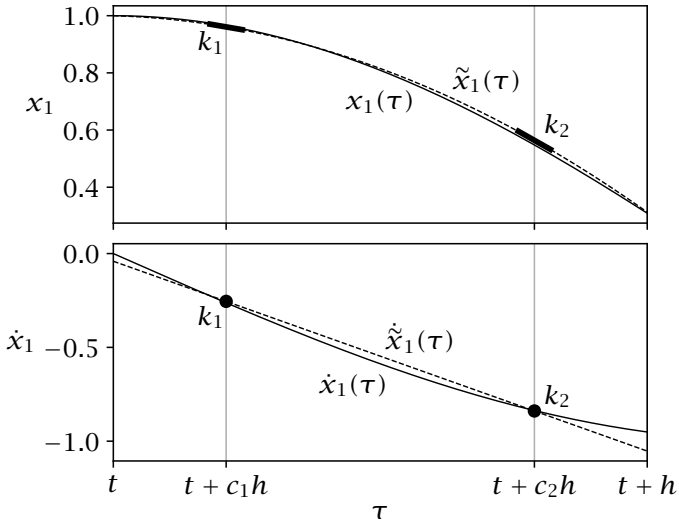


Figure 8.3: Polynomial approximation $\widetilde{x}_1(t)$ and true trajectory $x_1(t)$ of the first state and its derivative, computed at the first integration step of the GL4 collocation method applied to the stiff ODE from Example 8.4. Note that the accuracy of the polynomial at the end of the interval is significantly higher than in the interior. The result of this first GL4 step can also be seen on the right side of Figure 8.4.

time derivatives is visualized, for a collocation method with $s = 2$ collocation points (GL4) applied to the ODE from Example 8.4. Note that in this example, $\dot{\widetilde{x}}(\tau; k_1, k_2, \ldots, k_s)$ is a polynomial of order one, i.e., an affine function, and its integral, $\widetilde{x}(\tau; x, k_1, k_2, \ldots, k_s)$, is a polynomial of order two.

The Butcher tableau of three popular collocation methods is

Implicit Euler		Midpoint rule (GL2)		Gauss-Legendre of order 4 (GL4)		
1	1	$1/2$	$1/2$	$1/2-\sqrt{3}/6$	$1/4$	$1/4-\sqrt{3}/6$
	1		1	$1/2+\sqrt{3}/6$	$1/4+\sqrt{3}/6$	$1/4$
					$1/2$	$1/2$

An interesting remark is that the highest order that an s-stage implicit Runge-Kutta method can achieve is given by $2s$, and that the Gauss-Legendre collocation methods achieve this order, due to a particularly smart choice of collocation points (namely as roots of the orthogonal Legendre polynomials, following the idea of Gaussian quadrature). The midpoint rule is a Gauss-Legendre method of second order (GL2). The Gauss-Legendre methods, like many other popular collocation methods, are A-stable. Some methods, such as the Radau IIA collocation methods, have even stronger stability properties (they are also L-stable), and are often preferable for stiff problems. All collocation methods need to solve a nonlinear system of equations in ns dimensions in each step, which can become costly for large state dimensions and many stages.

Example 8.4: Implicit integrators for a stiff ODE system

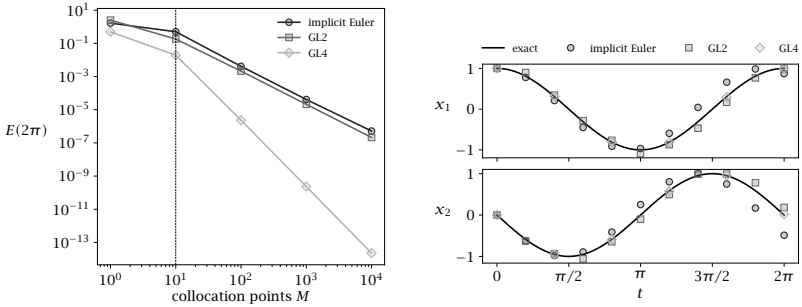
We consider the following ODE

$$\dot{x} = Ax - 500\,x\,(|x|^2 - 1)$$

with A and initial conditions as before in Example 8.3. In contrast to the previous example, this ODE is nonlinear and stiff, due to the additive nonlinear term $-500\,x\,(|x|^2 - 1)$. This term is zero only if the norm of x is one, i.e., if the state lies on the unit circle. If not, the state is strongly pushed toward the unit circle. This makes the system a stiff ODE. As we start at $[1, 0]'$, the exact solution lies again on the unit circle, and also ends at $[1, 0]'$. For comparison, we solve the initial value problem with three implicit integration methods, all of collocation type (implicit Euler, GL2, GL4). To have an approximate measure of the computational costs of the different methods, we denote by M the total number of collocation points on the time horizon. The results are shown in Figure 8.4. On the left-hand side, the different order behavior is observed. On the right-hand side, the trajectories resulting from a total of $M = 10$ collocation points are shown for the three different methods. In Figure 8.3, the first step of the GL4 method is visualized in detail, showing both the trajectory of the first state as well as its time derivative, together with their polynomial approximations. □

8.2.4 Differential Algebraic Equations

Some system models do not only contain differential, but also algebraic equations, and therefore belong to the class of *differential algebraic*



(a) Accuracy vs. collocation points. (b) Simulation result for $M = 10$ points.

Figure 8.4: Performance of implicit integration methods on a stiff ODE.

equations (DAEs). The algebraic equations might, for example, reflect conservation laws in chemical reaction models or kinematic constraints in robot models. DAE models come in many different forms, some of which are easier to treat numerically than others. One particularly favorable class of DAE are the *semiexplicit DAE of index one*, which can be written as

$$\dot{x} = f(t, x, z) \tag{8.9a}$$

$$0 = g(t, x, z) \tag{8.9b}$$

Here, the *differential states* $x \in \mathbb{R}^n$ are accompanied by *algebraic states* $z \in \mathbb{R}^{n_z}$, and the algebraic states are implicitly determined by the *algebraic equations* (8.9b). Here, the number of algebraic equations is equal to the number of algebraic states, i.e., $g : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^{n_z} \to \mathbb{R}^{n_z}$, such that for fixed t and x, the algebraic equation (8.9b) forms a nonlinear system of n_z equations for n_z unknowns.

The assumption of *index one* requires the Jacobian matrix of g with respect to z to be invertible at all points of interest. The fact that $\dot{x}$ appears alone on the left side of the differential equation (8.9a) makes the DAE *semiexplicit*. An interesting observation is that it is possible to reduce a semiexplicit DAE of index one to an ODE if one finds an explicit symbolic expression $z^*(t, x)$ for the implicit function defined by $g(t, x, z^*(t, x)) = 0$. The resulting ODE that is equivalent to the original DAE is given by $\dot{x} = f(t, x, z^*(t, x))$. Usually, this reduction from an index-one DAE to an ordinary differential equation is not possible analytically. A numerical computation of $z^*(t, x)$ is always possible in principle, but

requires the use of an underlying root-finding method. This way it is possible to solve a DAE with explicit integration methods. For implicit integration methods, however, one can simply augment the nonlinear equation system by the algebraic equations g at all evaluation points of the right-hand-side of the differential function f, and then rely on the root-finding method of the integrator. For this reason, and because they are often stiff, DAE are usually addressed with implicit integrators.

8.2.5 Integrator Adaptivity

Many practical integration methods use an *adaptive stepsize selection* to attain a good trade-off between numerical accuracy and computational effort. Instead of performing steps of equal length h, adaptive methods vary h in each step. Usually, they try to keep an estimate of the local error constant. The details are beyond our interest here, but we note that integrator adaptivity can be a crucial feature for the efficiency of nonlinear MPC implementations, in particular for the long simulation intervals which appear when one appends a prediction horizon at the end of the control horizon. On the other hand, integrator adaptivity needs to be treated with care when numerical derivatives of the simulation result are computed, as discussed in Section 8.4.6.

8.3 Solving Nonlinear Equation Systems

We have seen that an important subtask within numerical simulation—as well as in numerical optimization—is the solution of nonlinear equation systems. In this section, we therefore discuss the basic technologies that make it possible to solve implicit equation systems with thousands of unknowns within a few milliseconds. We start with linear equations, and then proceed to nonlinear equations and their solution with *Newton-type methods*.

8.3.1 Linear Systems

Solving a linear system of equations $Az = b$ with a square invertible matrix $A \in \mathbb{R}^{n_z \times n_z}$ is an easy task in the age of digital computers. The direct solution of the system requires only two computational steps: first, a factorization of the matrix A, for example, a lower-upper-factorization (LU-factorization) that yields a lower-triangular matrix L and an upper-triangular matrix U such that $LU = A$. Second, one

needs to perform a forward and a back substitution, yielding the solution as $z = U^{-1}(L^{-1}b)$. The computation of the LU-factorization, or LU-decomposition, requires $(2/3)n_z^3$ floating-point operations (FLOPs), while the forward and back substitution require together n_z^2 operations. Additional row or column permutations—in a process called *pivoting*—usually need to be employed and improve numerical stability, but only add little extra computational cost. The LU-decomposition algorithm was introduced by Alan Turing (1912–1954), and can be traced back to *Gaussian elimination*, after Carl Friedrich Gauss (1777–1855). Solving a dense linear system with $n_z = 3000$ variables needs about $18 \cdot 10^9$ FLOPs, which on a current quadcore processor (2.9 GHz Intel Core i5) need only 600 ms.

The runtime of the LU-decomposition and the substitutions can significantly be reduced if the matrix A is sparse, i.e., if it has many more zero than nonzero entries. Sparsity is particularly simple to exploit in case of banded matrices, which have their nonzero entries only in a band around the diagonal. Tailored direct methods also can exploit other structures, like block sparsity, or symmetry of the matrix A. For symmetric A, one usually performs a lower-diagonal-lower-transpose-factorization (LDLT-factorization) of the form $LDL' = A$ (with lower-triangular L and diagonal D), which reduces the computational cost by a factor of two compared to an LU-factorization. For symmetric and positive definite matrices A, one can even apply a *Cholesky decomposition* of the form $LL' = A$, with similar costs as the LDLT-factorization.

For huge linear systems that cannot be addressed by direct factorization approaches, there exist a variety of *indirect* or *iterative* solvers. Linear system solving is one of the most widely used numerical techniques in science and engineering, and the field of computational linear algebra is investigated by a vibrant and active research community. Contrary to only a century ago, when linear system solving was a tedious and error-prone task, today we rarely notice when we solve a linear equation, e.g., by using the backslash operator in MATLAB in the expression `A\b`, because computational linear algebra is such a reliable and mature technology.

8.3.2 Nonlinear Root-Finding Problems

A more difficult situation occurs when a nonlinear equation system $R(z) = 0$ needs to be solved, for example, in each step of an implicit Runge-Kutta method, or in nonlinear optimization. Depending

on the problem, one can usually not even be sure that a solution z^0 with $R(z^0) = 0$ exists. And if one has found a solution, one usually cannot be sure that it is the only one. Despite these theoretical difficulties with nonlinear root-finding problems, they are nearly as widely formulated and solved in science and engineering as linear equation systems.

In this section we therefore consider a continuously differentiable function $R : \mathbb{R}^{n_z} \to \mathbb{R}^{n_z}$, $z \mapsto R(z)$, where our aim is to solve the nonlinear equation

$$R(z) = 0$$

Nearly all algorithms to solve this system derive from an algorithm called *Newton's method* or *Newton-Raphson method* that is accredited to Isaac Newton (1643–1727) and Joseph Raphson (about 1648–1715), but which was first described in its current form by Thomas Simpson (1710–1761). The idea is to start with an initial guess z_0, and to generate a sequence of iterates $(z_k)_{k=0}^{\infty}$ by linearizing the nonlinear equation at the current iterate

$$R(z_k) + \frac{\partial R}{\partial z}(z_k)(z - z_k) = 0$$

This equation is a linear system in the variable z, and if the Jacobian $J(z_k) := \frac{\partial R}{\partial z}(z_k)$ is invertible, we can explicitly compute the next iterate as

$$z_{k+1} = z_k - J(z_k)^{-1} R(z_k)$$

Here, we use the notation $J(z_k)^{-1}R(z_k)$ as a shorthand for the algorithm that solves the linear system $J(z_k)\Delta z = R(z_k)$. In the actual computation of a Newton step, the inverse $J(z_k)^{-1}$ is never computed, but only a LU-decomposition of $J(z_k)$, and a forward and a back substitution, as described in the previous subsection.

More generally, we can use an invertible approximation M_k of the Jacobian $J(z_k)$, leading to the *Newton-type methods*. The general Newton-type method iterates according to

$$z_{k+1} = z_k - M_k^{-1} R(z_k)$$

Depending on how closely M_k approximates $J(z_k)$, the local convergence can be fast or slow, or the sequence may even not converge. The advantages of using an M_k that is different from $J(z_k)$ could be that it can be chosen to be invertible even if $J(z_k)$ is not, or that computation of M_k, or of its factorization, can be cheaper. For example, one could

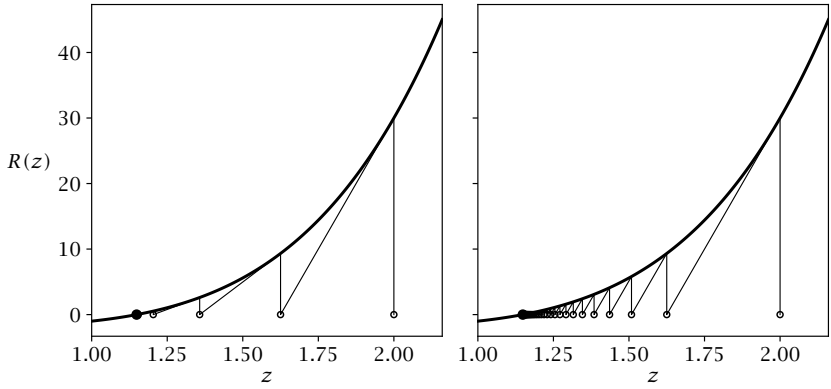


Figure 8.5: Newton-type iterations for solution of $R(z) = 0$ from Example 8.5. Left: exact Newton method. Right: constant Jacobian approximation.

reuse one matrix and its factorization throughout several Newton-type iterations.

Example 8.5: Finding a fifth root with Newton-type iterations

We find the zero of $R(z) = z^5 - 2$ for $z \in \mathbb{R}$. Here, the derivative is $\frac{\partial R}{\partial z}(z) = 5z^4$, such that the Newton method iterates

$$z_{k+1} = z_k - (5z_k^4)^{-1}(z_k^5 - 2)$$

When starting at $z_0 = 2$, the first step is given by $z_1 = 2 - (80)^{-1}(32 - 2) = 13/8$, and the following iterates quickly converge to the solution z^* with $R(z^*) = 0$, as visualized in Figure 8.5 on the left side.

Alternatively, we could use a Jacobian approximation $M_k \neq J(z_k)$, e.g., the constant value $M_k = 80$ corresponding to the true Jacobian at $z = 2$. The resulting iteration would be

$$z_{k+1} = z_k - (80)^{-1}(z_k^5 - 2)$$

When started at $z_0 = 2$ the first iteration would be the same as for Newton's method, but then the Newton-type method with constant Jacobian produces a different sequence, as can be seen on the right side of Figure 8.5. Here, the approximate method also converges; but in general,

when does a Newton-type method converge, and when it converges, how quickly? □

8.3.3 Local Convergence of Newton-Type Methods

Next we investigate the conditions on $R(z)$, z_0 and on M_k required to ensure local convergence of Newton-type iterations. In particular we discuss the speed of convergence. In fact, even if we assume that a sequence of iterates $z_k \in \mathbb{R}^n$ converges to a solution point z^*, i.e., if $z_k \to z^*$, the rate of convergence can be painstakingly slow or lightning fast. The speed of convergence can make the difference between a method being useful or useless for practical computations. Mathematically speaking, a sequence (z_k) is said to converge *q-linearly* if there exists a positive integer k_0 and a positive real number $c_{\max} < 1$, and sequence $(c_k)_{k_0}^{\infty}$ such that for all $k \geq k_0$ holds that $c_k \leq c_{\max}$ and that

$$|z_{k+1} - z^*| \leq c_k |z_k - z^*| \tag{8.10}$$

If in addition, $c_k \to 0$, the sequence is said to converge *q-superlinearly*. If in addition, $c_k = O(|z_k - z^*|)$, the sequence is said to converge *q-quadratically*.[2]

Example 8.6: Convergence rates

We discuss and visualize four examples with $z_k \in (0, \infty)$ and $z_k \to 0$, see Figure 8.6.

- $z_k = \frac{1}{2^k}$ converges q-linearly: $\frac{z_{k+1}}{z_k} = \frac{1}{2}$
- $z_k = 0.99^k$ also converges q-linearly: $\frac{z_{k+1}}{z_k} = 0.99$. This example converges very slowly. In practice we desire $c_{\max}$ to be smaller than, say, $\frac{1}{2}$
- $z_k = \frac{1}{k!}$ converges q-superlinearly, as $\frac{z_{k+1}}{z_k} = \frac{1}{k+1}$
- $z_k = \frac{1}{2^{2^k}}$ converges q-quadratically, because $\frac{z_{k+1}}{(z_k)^2} = \frac{(2^{2^k})^2}{2^{2^{k+1}}} = 1 < \infty$. For $k = 6$, $z_k = \frac{1}{2^{64}} \approx 0$. This is a typical feature of q-quadratic convergence: often, convergence up to machine precision is obtained in about six iterations. □

[2] The historical prefix "q" stands for "quotient," to distinguish it from a weaker form of convergence that is called "r-convergence," where "r" stands for "root."

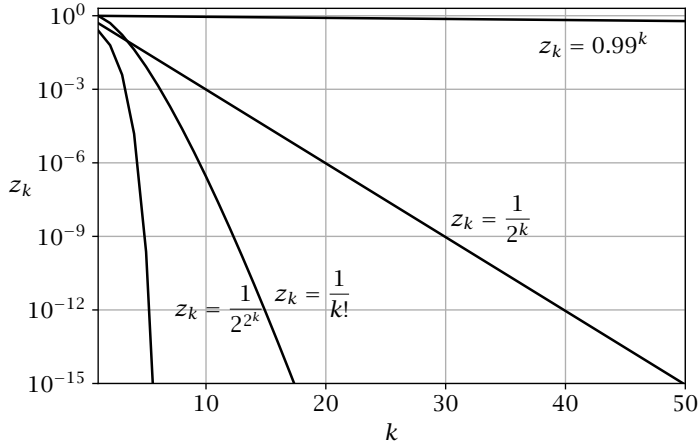


Figure 8.6: Convergence of different sequences as a function of k.

Local convergence of a Newton-type method can be guaranteed by the following classical result (see, e.g., Bock (1983) or Deuflhard (2011)), which also specifies the rate of convergence.

Theorem 8.7 (Local contraction for Newton-type methods)**.** *Regard a nonlinear continuously differentiable function $R : D \to \mathbb{R}^{n_z}$ defined on an open domain $D \subset \mathbb{R}^{n_z}$ and a solution point $z^* \in D$ with $R(z^*) = 0$. We start the Newton-type iteration with the initial guess $z_0 \in D$ and iterate according to $z_{k+1} = z_k - M_k^{-1} R(z_k)$. The sequence (z_k) converges at least q-linearly to z^* and obeys the contraction inequality*

$$|z_{k+1} - z^*| \leq \left(\kappa_k + \frac{\omega}{2} |z_k - z^*|\right) |z_k - z^*| \tag{8.11}$$

if there exist constants $\omega \in [0, \infty)$, $\kappa_{\max} \in [0, 1)$, and a sequence $(\kappa_k)_{k=0}^{\infty}$ with $\kappa_k \in [0, \kappa_{\max}]$, that satisfy for all z_k and all $z \in D$ the following two inequalities

$$\left|M_k^{-1}(J(z_k) - J(z))\right| \leq \omega |z_k - z| \qquad \textit{(Lipschitz condition)}$$

$$\left|M_k^{-1}(J(z_k) - M_k)\right| \leq \kappa_k \qquad \textit{(compatibility condition)}$$

and if the ball $B := \left\{z \in \mathbb{R}^{n_z} \mid |z - z^*| < \frac{2(1-\kappa_{\max})}{\omega}\right\}$ is completely contained in D and if $z_0 \in B$. If in addition $\kappa_k \to 0$, the sequence converges q-superlinearly. If in addition $\kappa_k = O(|z_k - z^*|)$ or even $\kappa_{\max} = 0$, the sequence converges q-quadratically.

Corollary 8.8 (Convergence of exact Newton's method)**.** *For an exact Newton's method, the convergence rate is q-quadratic, because we have* $M_k = J(z_k)$*, i.e.,* $\kappa_{\max} = 0$*.*

8.3.4 Affine Invariance

An iterative method to solve a root-finding problem $R(z) = 0$ is called *affine invariant* if affine basis transformations of the equations or variables do not change the resulting iterations. This is an important property in practice. It is not unreasonable to ask that a good numerical method should behave the same if it is applied to problems formulated in different units or coordinate systems.

The exact Newton method is affine invariant, and also some popular Newton-type optimization methods like the Gauss-Newton method for nonlinear least squares problems share this property. Their affine invariance makes them insensitive to the chosen problem scaling, and this is one reason why they are successful in practice. On the other hand, a method that is not affine invariant usually needs careful scaling of the model equations and decision variables to work well.

8.3.5 Globalization for Newton-Type Methods

The iterations of a Newton-type method can be regarded the trajectory of a nonlinear discrete time system, and the solution z^0 a fixed point. This system is autonomous if M_k is constant or a function of z, i.e., if $M_k = M(z_k)$. In this case, the discrete time system is given by $z^+ = f(z)$ with $f(z) := z - M(z)^{-1}R(z)$. When designing the Newton-type method, one usually wants the solution z^0 to be a stable fixed point with a large area of attraction. Local convergence to this fixed point usually can be guaranteed under conditions stated in Theorem 8.7, in particular if the exact Jacobian is available. On the other hand, the area of attraction for the full-step Newton-type methods described so far is unfortunately not very large in practice, and Newton-type methods usually need extra *globalization* features to make them globally convergent from arbitrary initial guesses. Some globalization techniques are based on a *merit function* that plays the role of a Lyapunov function to be reduced in each iteration; others are based on a *filter* as a measure of merit of a new iterate. To ensure progress from one iteration to the next, some form of *damping* is applied that either reduces the unmodified Newton-type step by doing a *line-search* along the proposed direction, or changes the step computation by adding a *trust-region*

constraint. For a detailed description of globalization techniques, we refer to textbooks on optimization such as Nocedal and Wright (2006).

8.4 Computing Derivatives

Whenever a Newton-type method is used for numerical simulation or optimization, we need to provide derivatives of nonlinear functions that exist as computer code. Throughout this section, we consider a differentiable function $F(u)$ with m inputs and p outputs $y = F(u)$, i.e., a function $F : \mathbb{R}^m \to \mathbb{R}^p$. The main object of interest is the Jacobian $J(u) \in \mathbb{R}^{m \times p}$ of F at the point u, or some of its elements.

Among the many ways to compute the derivatives of $F(u)$, the most obvious would be to apply the known differentiation rules on paper for each of its components, and then to write another computer code by hand that delivers the desired derivatives. This process can become tedious and error prone, but can be automated by using symbolic computer algebra systems such as Maple or Mathematica. This *symbolic differentiation* often works well, but typically suffers from two disadvantages. First, it requires the code to exist in the specific symbolic language. Second, the resulting derivative expressions can become much longer than the original function, such that the CPU time needed to evaluate the Jacobian $J(u)$ by symbolic differentiation can become significantly larger than the CPU time to evaluate $F(u)$.

In contrast, we next present three ways to evaluate the Jacobian $J(u)$ of any computer-represented function $F(u)$ by algorithms that have bounded costs: numerical differentiation, as well as the algorithmic differentiation (AD) in forward mode and in reverse mode. All three ways are based on the evaluation of directional derivatives of the form $J(u)\dot{u}$ with a vector $\dot{u} \in \mathbb{R}^m$ (forward directional derivatives used in numerical differentiation and forward AD) or of the form $\bar{y}'J(u)$ with $\bar{y} \in \mathbb{R}^p$ (reverse directional derivatives used in reverse AD). When unit vectors are used for $\dot{u}$ or $\bar{y}$, the directional derivatives correspond to columns or rows of $J(u)$, respectively. Evaluation of the full Jacobian thus needs either m forward derivatives or p reverse derivatives. Note that in this section, the use of a dot or a bar above a vector as in $\dot{u}$ and $\bar{y}$ just denotes another arbitrary vector with the same dimensions as the original one.

8.4.1 Numerical Differentiation

Numerical differentiation is based on multiple calls of the function $F(u)$ at different input values. In its simplest and cheapest form, it computes a *forward difference* approximation of $J(u)\dot{u}$ for given u and $\dot{u} \in \mathbb{R}^m$ by using a small but finite perturbation size $t_* > 0$ as follows

$$\frac{F(u + t_* \dot{u}) - F(u)}{t_*}$$

The optimal size of t_* for the forward difference approximation depends on the numerical accuracy of the evaluations of F, which we denote by $\epsilon > 0$, and on the relative size of the second derivatives of F compared to F, which we denote by $L > 0$. A detailed derivation leads to the optimal choice

$$t_* \approx \sqrt{\frac{\epsilon}{L}}$$

While ϵ is typically known and given by the machine precision, i.e., $\epsilon = 10^{-16}$ for double-precision floating-point computations, the relative size of the second derivative L is typically not known, but can be estimated. Often, L is just assumed to be of size one, resulting in the choice $t_* = \sqrt{\epsilon}$, i.e., $t_* = 10^{-8}$ for double precision. One can show that the accuracy of the forward derivative approximation is then also given by $\sqrt{\epsilon}$, i.e., one loses half of the valid digits compared to the function evaluation. To compute the full Jacobian $J(u)$, one needs to evaluate m forward differences, for the m seed vectors $\dot{u} = (1, 0, 0, \ldots)'$, $\dot{u} = (0, 1, 0 \ldots)'$, etc. Because the center point can be recovered, one needs in total $(m + 1)$ evaluations of the function F. Thus, we can summarize the cost for computation of the full Jacobian J (as well as the function F) by the statement

$$\text{cost}(F, J) = (1 + m)\,\text{cost}(F)$$

There exists a variety of more accurate, but also more expensive, forms of numerical differentiation, which can be derived from polynomial interpolation of multiple function evaluations of F. The easiest of these are central differences, which are based on a positive and a negative perturbation. Using such higher-order formulas with adaptive perturbation size selection, one can obtain high-accuracy derivatives with numerical differentiation, but at significant cost. One interesting way to actually *reduce* the cost of the numerical Jacobian calculation arises if the Jacobian is known to be sparse, and if many of its columns are structurally orthogonal, i.e., have their nonzero entries at different locations.

To efficiently generate a full Jacobian, one can, for example, use the algorithm by Curtis, Powell, and Reid (1974) that is implemented in the FORTRAN routine TD12 from the HSL Mathematical Software Library (formerly Harwell Subroutine Library). For details of sparse Jacobian evaluations, we refer to the review article by Gebremedhin, Manne, and Pothen (2005).

In summary, and despite the tricks to improve accuracy or efficiency, one has to conclude that numerical differentiation often results in quite inaccurate derivatives, and its only—but practically important—advantage is that it works for any black-box function that can be evaluated on a given computer. Fortunately, there exists a different technology, called AD, that also has tight bounds on the computational cost of the Jacobian evaluation, but avoids the numerical inaccuracies of numerical differentiation. It is often even faster than numerical differentiation, and in the case of reverse derivatives $\bar{y}'J$, it can be tremendously faster. It does so, however, by opening the black box.

8.4.2 Algorithmic Differentiation

We next consider a function $F : \mathbb{R}^m \to \mathbb{R}^p$ that is composed of a sequence of N elementary operations, where an elementary operation acts on only one or two variables. We also introduce a vector $x \in \mathbb{R}^n$ with $n = m + N$ that contains all intermediate variables including the inputs, $x_1 = u_1$, $x_2 = u_2$, $\dots$ $x_m = u_m$. While the inputs are given before the function is called, each elementary operation generates a new intermediate variable, x_{m+i}, for $i = 1, \dots, N$. Some of these intermediate variables are used as output $y \in \mathbb{R}^p$ of the code. This decomposition into elementary operations is automatically performed in each executable computer code, and best illustrated with an example.

Example 8.9: Function evaluation via elementary operations

We consider the function

$$F(u_1, u_2, u_3) = \begin{bmatrix} u_1 u_2 u_3 \\ \sin(u_1 u_2) + \exp(u_1 u_2 u_3) \end{bmatrix}$$

with $m = 3$ and $p = 2$. We can decompose this function into $N = 5$ elementary operations that are preceded by m and followed by p

renaming operations, as follows

$$\begin{array}{ll} x_1 & = u_1 \\ x_2 & = u_2 \\ x_3 & = u_3 \\ \hline x_4 & = x_1 x_2 \\ x_5 & = \sin(x_4) \\ x_6 & = x_4 x_3 \\ x_7 & = \exp(x_6) \\ x_8 & = x_5 + x_7 \\ \hline y_1 & = x_6 \\ y_2 & = x_8 \end{array} \tag{8.12}$$

Thus, if the $m = 3$ inputs u_1, u_2, u_3 are given, the $N = 5$ nontrivial elementary operations compute the intermediate quantities $x_4, \ldots, x_8$, and the sixth and eighth of the intermediate quantities are then used as the output $y = F(u)$ of our function. □

The idea of AD is to use the chain rule and differentiate each of the elementary operations separately. There exist two modes of AD, the forward mode and the reverse mode. Both can be derived in a mathematically rigorous way by interpreting the computer function $y = F(u)$ as the output of an implicit function, as explained next.

8.4.3 Implicit Function Interpretation

Let us regard all equations that recursively define the intermediate quantities $x \in \mathbb{R}^n$ for a given $u \in \mathbb{R}^m$ as one large nonlinear equation system

$$G(x, u) = 0 \tag{8.13}$$

with $G : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$. Here, the partial derivative $\frac{\partial G}{\partial x} \in \mathbb{R}^{n \times n}$ is a lower-triangular invertible matrix and $\frac{\partial G}{\partial u} \in \mathbb{R}^{n \times m}$ turns out to be an m-dimensional unit matrix augmented by zeros, which we will denote by B. The function G defines an implicit function $x^* : \mathbb{R}^m \to \mathbb{R}^n$, $u \mapsto x^*(u)$ that satisfies $G(x^*(u), u) = 0$. The output $y = F(u)$ is given by the selection of some entries of $x^*(u)$ via a selection matrix $C \in \mathbb{R}^{p \times n}$, i.e., the computer function is represented by the expression $F(u) = Cx^*(u)$. The derivative $\frac{dx^*}{du}$ of the implicit function satisfies $\frac{\partial G}{\partial x}\frac{dx^*}{du} + \frac{\partial G}{\partial u} = 0$ and is therefore given by

$$\frac{dx^*}{du} = \left(-\frac{\partial G}{\partial x}\right)^{-1} \underbrace{\frac{\partial G}{\partial u}}_{=:B} = \left(-\frac{\partial G}{\partial x}\right)^{-1} B$$

and the Jacobian of F is simply given by $J(u) = C\frac{dx^*}{du}(u)$. The forward directional derivative is given by

$$J(u)\dot{u} = C\underbrace{\left(-\frac{\partial G}{\partial x}\right)^{-1} B\dot{u}}_{=:\dot{x}} = C\dot{x}$$

Here, we have introduced the *dot quantities* $\dot{x}$ that denote the directional derivative of $x^*(u)$ into the direction $\dot{u}$, i.e., $\dot{x} = \frac{dx^*}{du}\dot{u}$. An efficient algorithm to compute $\dot{x}$ corresponds to the solution of a lower-triangular linear equation system that is given by

$$\left(-\frac{\partial G}{\partial x}\right)\dot{x} = B\dot{u} \tag{8.14}$$

Since the matrix $\frac{\partial G}{\partial x}$ is lower triangular, the linear system can be solved by a forward sweep that computes the components of $\dot{x}$ in the same order as the elementary operations, i.e., it first computes $\dot{x}_1$, then $\dot{x}_2$, etc. This leads to the forward mode of AD.

The reverse directional derivative, on the other hand, is given by

$$\bar{y}'J(u) = \underbrace{\bar{y}'\,C\left(-\frac{\partial G}{\partial x}\right)^{-1}}_{=:\bar{x}'} B = \bar{x}'B$$

where we define the *bar quantities* $\bar{x}$ that have a different meaning than the dot quantities. For computing $\bar{x}$, we need to also solve a linear system, but with the transposed system matrix

$$\left(-\frac{\partial G}{\partial x}\right)' \bar{x} = C'\bar{y} \tag{8.15}$$

Due to the transpose, this system involves an upper-triangular matrix and can thus be solved by a reverse sweep, i.e., one first computes $\bar{x}_n$, then $\bar{x}_{n-1}$, etc. This procedure leads to the reverse mode of AD.

Example 8.10: Implicit function representation

Let us regard Example 8.9 and find the corresponding function $G(x,u)$ as well as the involved matrices. The function G corresponds to the

first $n = 8$ rows of (8.12) and is given by

$$G(x,u) = \left[\begin{array}{c} u_1 - x_1 \\ u_2 - x_2 \\ u_3 - x_3 \\ \hline x_1 x_2 - x_4 \\ \sin(x_4) - x_5 \\ x_4 x_3 - x_6 \\ \exp(x_6) - x_7 \\ x_5 + x_7 - x_8 \end{array}\right]$$

It is obvious that the nonlinear equation $G(x,u) = 0$ can be solved for any given u by a simple forward elimination of the variables $x_1, x_2, \ldots$, yielding the map $x^*(u)$. This fact implies also the lower-triangular structure of the Jacobian $\frac{\partial G}{\partial x}$ which is given by

$$\frac{\partial G}{\partial x} = \left[\begin{array}{cccccccc} -1 & & & & & & & \\ 0 & -1 & & & & & & \\ 0 & 0 & -1 & & & & & \\ \hline x_2 & x_1 & 0 & -1 & & & & \\ 0 & 0 & 0 & \cos(x_4) & -1 & & & \\ 0 & 0 & x_4 & x_3 & 0 & -1 & & \\ 0 & 0 & 0 & 0 & 0 & \exp(x_6) & -1 & \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & -1 \end{array}\right]$$

The derivative of G with respect to u is given by a unit matrix to which zero rows are appended, and given by

$$B := \frac{\partial G}{\partial u} = \left[\begin{array}{ccc} 1 & & \\ & 1 & \\ & & 1 \\ \hline 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{array}\right]$$

The identity $y = Cx$ corresponds to the last $p = 2$ rows of (8.12), and the matrix $C \in \mathbb{R}^{p \times n}$ is therefore given by

$$C = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

The right-hand-side vectors in the equations (8.14) and (8.15) are given by

$$B\dot{u} = \begin{bmatrix} \dot{u}_1 \\ \dot{u}_2 \\ \dot{u}_3 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad \text{and} \quad C'\bar{y} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \bar{y}_1 \\ 0 \\ \bar{y}_2 \end{bmatrix}$$

□

8.4.4 Algorithmic Differentiation in Forward Mode

The forward mode of AD computes $\dot{x}$ by solving the lower-triangular linear system (8.14) with a forward sweep. After the trivial definition of the first m components of $\dot{x}$, it goes through all elementary operations in the same order as in the original function to compute the components of $\dot{x}$ one by one. If an original line of code reads $x_k = \phi_k(x_i, x_j)$, the corresponding line to compute $\dot{x}_k$ by forward AD is simply given by

$$\dot{x}_k = \frac{\partial \phi_k}{\partial x_i}(x_i, x_j)\,\dot{x}_i + \frac{\partial \phi_k}{\partial x_j}(x_i, x_j)\,\dot{x}_j$$

In forward AD, the function evaluation and the derivative evaluation can be performed simultaneously, if desired, eliminating the need to store any internal information. The algorithm is best explained by looking again at the example.

Example 8.11: Forward algorithmic differentiation

We differentiate the algorithm from Example 8.9. To highlight the relation to the original code, we list the original command again on the left side, and show the algorithm to compute $\dot{x}$ on the right side. For given $u = [u_1\ u_2\ u_3]'$ and $\dot{u} = [\dot{u}_1\ \dot{u}_2\ \dot{u}_3]'$, the two algorithms proceed as

follows

$$
\begin{array}{ll}
x_1 = u_1 & \dot{x}_1 = \dot{u}_1 \\
x_2 = u_2 & \dot{x}_2 = \dot{u}_2 \\
x_3 = u_3 & \dot{x}_3 = \dot{u}_3 \\
\hline
x_4 = x_1 x_2 & \dot{x}_4 = x_2 \dot{x}_1 + x_1 \dot{x}_2 \\
x_5 = \sin(x_4) & \dot{x}_5 = \cos(x_4)\dot{x}_4 \\
x_6 = x_4 x_3 & \dot{x}_6 = x_3 \dot{x}_4 + x_4 \dot{x}_3 \\
x_7 = \exp(x_6) & \dot{x}_7 = \exp(x_6)\dot{x}_6 \\
x_8 = x_5 + x_7 & \dot{x}_8 = \dot{x}_5 + \dot{x}_7 \\
\hline
y_1 = x_6 & \dot{y}_1 = \dot{x}_6 \\
y_2 = x_8 & \dot{y}_2 = \dot{x}_8
\end{array}
$$

The result of the original algorithm is $y = [y_1\ y_2]'$ and the result of the forward AD sweep is $\dot{y} = [\dot{y}_1\ \dot{y}_2]'$. If desired, one could perform both algorithms in parallel, i.e., evaluate first the left side, then the right side of each row consecutively. This procedure would allow one to delete each intermediate variable and the corresponding dot quantity after its last usage, making the memory demands of the joint evaluation just twice as big as those of the original function evaluation. □

One can see that the dot-quantity evaluations on the right-hand side—which we call a forward sweep—are never longer than about twice the original line of code. This is because each elementary operation depends on at maximum two intermediate variables. More generally, it can be proven that the computational cost of one forward sweep in AD is smaller than a small constant times the cost of a plain function evaluation. This constant depends on the chosen set of elementary operations, but is usually much less than two, so that we conclude

$$\text{cost}(J\dot{u}) \leq 2\,\text{cost}(F)$$

To obtain the full Jacobian J, we need to perform the forward sweep several times, each time with the seed vector corresponding to one of the m unit vectors in $\mathbb{R}^m$. The m forward sweeps all could be performed simultaneously with the evaluation of the function itself, so that one needs in total one function evaluation plus m forward sweeps, i.e., we have

$$\text{cost}(F, J) \leq (1 + 2m)\,\text{cost}(F)$$

This is a conservative bound, and depending on the AD tool used the cost of several combined forward sweeps can be significantly reduced,

and often become much cheaper than a finite difference approximation. Most important, the result of forward AD is exact up to machine precision.

8.4.5 Algorithmic Differentiation in Reverse Mode

The reverse mode of AD computes $\bar{x}$ by solving the upper-triangular linear system (8.15) with a reverse sweep. It does so by first computing the right-hand-side $C'\bar{y}$ vector and initializing all bar quantities with the respective values, i.e., it initially sets $\bar{x} = C'\bar{y}$. Then, the reverse AD algorithm modifies the bar quantities by going through all elementary operations in reverse order. The value of $\bar{x}_i$ is modified for each elementary operation in which x_i is involved. If two quantities x_i and x_j are used in the elementary operation $x_k = \phi_k(x_i, x_j)$, then the corresponding two update equations are given by

$$\bar{x}_i = \bar{x}_i + \bar{x}_k \frac{\partial \phi_k}{\partial x_i}(x_i, x_j) \quad \text{and}$$
$$\bar{x}_j = \bar{x}_j + \bar{x}_k \frac{\partial \phi_k}{\partial x_j}(x_i, x_j)$$

Again, the algorithm is best illustrated with the example.

Example 8.12: Algorithmic differentiation in reverse mode

We consider again the code from Example 8.9. In contrast to before in Example 8.11, now we compute the reverse directional derivative $\bar{y}'J(u)$ for given $[u_1\ u_2\ u_3]'$ and $\bar{y} = [\bar{y}_1\ \bar{y}_2]'$. After the forward evaluation of the function, which is needed to define all intermediate quantities, we need to solve the linear system (8.15) to obtain $\bar{x}$. In the example, this system is explicitly given by

$$\begin{bmatrix} 1 & & & -x_2 & & & & & \\ & 1 & & -x_1 & & & & & \\ & & 1 & 0 & & -x_4 & & & \\ & & & 1 & -\cos(x_4) & -x_3 & & & \\ & & & & 1 & 0 & & -1 & \\ & & & & & 1 & -\exp(x_6) & 0 & \\ & & & & & & 1 & -1 & \\ & & & & & & & 1 & \end{bmatrix} \begin{bmatrix} \bar{x}_1 \\ \bar{x}_2 \\ \bar{x}_3 \\ \bar{x}_4 \\ \bar{x}_5 \\ \bar{x}_6 \\ \bar{x}_7 \\ \bar{x}_8 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \bar{y}_1 \\ 0 \\ \bar{y}_2 \end{bmatrix}$$

To solve this equation without forming the matrix explicitly, we process the elementary operations in reverse order, i.e., one column after the

other, noting that the final result for each $\bar{x}_i$ will be a sum of the right-hand-side vector component $C'\bar{y}$ and a weighted sum of the values $\bar{x}_j$ for those $j > i$ which correspond to elementary operations that have x_i as an input. We therefore initialize all variables by $\bar{x} = C'\bar{y}$, which results for the example in the initialization

$$\begin{aligned} \bar{x}_1 &= 0 & \bar{x}_5 &= 0 \\ \bar{x}_2 &= 0 & \bar{x}_6 &= \bar{y}_1 \\ \bar{x}_3 &= 0 & \bar{x}_7 &= 0 \\ \bar{x}_4 &= 0 & \bar{x}_8 &= \bar{y}_2 \end{aligned}$$

In the reverse sweep, the algorithm updates the bar quantities in reverse order compared to the original algorithm, processing one column after the other.

$$\begin{aligned} &// \text{ differentiation of } x_8 = x_5 + x_7 \\ &\bar{x}_5 = \bar{x}_5 + \bar{x}_8 \\ &\bar{x}_7 = \bar{x}_7 + \bar{x}_8 \\ &// \text{ differentiation of } x_7 = \exp(x_6) \\ &\bar{x}_6 = \bar{x}_6 + \bar{x}_7 \exp(x_6) \\ &// \text{ differentiation of } x_6 = x_4 x_3 \\ &\bar{x}_4 = \bar{x}_4 + \bar{x}_6 x_3 \\ &\bar{x}_3 = \bar{x}_3 + \bar{x}_6 x_4 \\ &// \text{ differentiation of } x_5 = \sin(x_4) \\ &\bar{x}_4 = \bar{x}_4 + \bar{x}_5 \cos(x_4) \\ &// \text{ differentiation of } x_4 = x_1 x_2 \\ &\bar{x}_1 = \bar{x}_1 + \bar{x}_4 x_2 \\ &\bar{x}_2 = \bar{x}_2 + \bar{x}_4 x_1 \end{aligned}$$

At the very end, the algorithm sets

$$\begin{aligned} \bar{u}_1 &= \bar{x}_1 \\ \bar{u}_2 &= \bar{x}_2 \\ \bar{u}_3 &= \bar{x}_3 \end{aligned}$$

to read out the desired result $\bar{y}'J(x) = [\bar{u}_1\ \bar{u}_2\ \bar{u}_3]$. Note that all three of the components are returned by *only one* reverse sweep. □

It can be shown that the cost of one reverse sweep of AD is less than a small constant (which is certainly less than three) times the cost of a

function evaluation, i.e.,

$$\text{cost}(\bar{y}' J) \leq 3\,\text{cost}(F)$$

To obtain the full Jacobian of F, we need to call the reverse sweep p times, with the seed vectors corresponding to the unit vectors in $\mathbb{R}^p$, i.e., together with one forward evaluation, we have

$$\text{cost}(F, J) \leq (1 + 3p)\,\text{cost}(F)$$

Remarkably, reverse AD can compute the full Jacobian at a cost that is independent of the input dimension m. This is particularly advantageous if $p \ll m$, e.g., if we compute the gradient of a scalar function like the objective in optimization. The reverse mode can be much faster than what we can obtain by forward finite differences, where we always need $(m+1)$ function evaluations. For example, to compute the gradient of a scalar function $f : \mathbb{R}^m \to \mathbb{R}$ when $m = 1,000,000$ and each call of the function requires one second of CPU time, the finite difference approximation of the gradient would take $1,000,001$ seconds, while the computation of the same quantity with the backward mode of AD requires only four seconds (one call of the function plus one backward sweep). Thus, besides being more accurate, reverse AD can also be much faster than numerical finite differences. This astonishing fact is also known as the "cheap gradient result" in the AD community, and in the field of neural networks it is exploited in the *back propagation* algorithm. The only disadvantage of the reverse mode of AD is that we have to store all intermediate variables and partial derivatives, in contrast to finite differences or forward AD.

Backward sweep for discrete time optimal control. In numerical optimal control we often have to differentiate a function that is the result of a dynamic system simulation. If the system simulation is in discrete time, one can directly apply the principles of AD to compute the desired derivatives by the forward or the reverse mode. For evaluating the gradient of the objective, the reverse mode is most efficient. If the controls are given by $\mathbf{u} = [u(0)' \ \cdots \ u(N-1)']'$ and the states $x(k)$ are obtained by a discrete time forward simulation of the form $x(k+1) = f(x(k), u(k))$ for $k = 0, \ldots, N-1$ started at $x(0) = x_0$, and if the objective function is given by $J(\mathbf{u}) := \sum_{k=0}^{N-1} \ell(x(k), u(k)) + V(x_N)$, then the backward sweep to compute $\nabla_{\mathbf{u}} J(\mathbf{u})$ performs the following

steps

$$\begin{aligned}
&\bar{x}(N)' = V_x(x(N)) \\
&\text{for} \quad k = N-1, N-2, \ldots, 0 \\
&\qquad \bar{x}(k)' = \ell_x(x(k), u(k)) + \bar{x}(k+1)' f_x(x(k), u(k)) \\
&\qquad \bar{u}(k)' = \ell_u(x(k), u(k)) + \bar{x}(k+1)' f_u(x(k), u(k)) \\
&\text{end}
\end{aligned} \tag{8.16}$$

The output of this algorithm is the vector $\bar{\mathbf{u}} = [\bar{u}(0)' \ \cdots \ \bar{u}(N-1)']'$ which equals the gradient $\nabla_{\mathbf{u}} J(\mathbf{u})$. This method to compute the objective gradient in the sequential approach was well known in the field of optimal control even before the field of algorithmic differentiation developed. From a modern perspective, however, it is simply an application of reverse AD to the algorithm that computes the objective function.

8.4.6 Differentiation of Simulation Routines

When a continuous time system is simulated by numerical integration methods and one wants to compute the derivatives of the state trajectory with respect to initial values or controls, as needed in shooting methods, there are many different approaches and many possible pitfalls. While a complete textbook could be written on the differentiation of just numerical integrators, we present and discuss only three popular approaches here.

External numerical differentiation (END). Probably the simplest approach to differentiate an integrator is to regard the integrator call as a black box, and to compute the desired derivatives by numerical finite differences. Here one computes one nominal trajectory, and one or more perturbed trajectories, depending on the desired number of forward derivatives. This approach, called external numerical differentiation (END), is easy to implement; it is generally not recommended because it suffers from some disadvantages.

- It is typically inaccurate because integrator accuracies ϵ_{int} are well above machine precision, e.g., $\epsilon_{\text{int}} \approx 10^{-6}$, such that the perturbation size needs to be chosen rather large, in particular for adaptive integrators.
- It usually is expensive because each call of the integrator for a perturbed trajectory creates some overhead, such as error control or matrix factorizations, which can be avoided in other approaches.

- It can only compute forward derivatives.

The first disadvantage can be mitigated for explicit integrators with fixed stepsize, where one is allowed to choose smaller perturbation sizes, in the order of the square root of the machine precision. For this special case, END becomes equivalent to the approach described next.

Internal numerical differentiation (IND). The idea behind internal numerical differentiation (IND) (Bock, 1981) is to regard the numerical integrator as a differentiable computer code in the spirit of algorithmic differentiation (AD). Similar to END, it works with perturbed trajectories. What is different from END is that all perturbed trajectories are treated in one single forward sweep, and that all adaptive integrator components are switched off for the perturbed trajectories. Thus, for an adaptive explicit integrator, the stepsize selection works only on the nominal trajectory; once the stepsize is chosen, the same size also is used for all perturbed trajectories.

For implicit integrators, where one performs Newton-type iterations in each step, the philosophy of IND is to choose the sequence of iteration matrices and numbers of Newton-type iterations for only the nominal trajectory, and to regard the iteration matrices as constant for all perturbed trajectories. Because all adaptive components are switched off during the numerical differentiation process, one can regard the integrator code as a function that evaluates its output with machine precision. For this reason, the perturbation size can be chosen significantly smaller than in END. Thus, IND is both more accurate and cheaper than END.

Algorithmic differentiation of integrators. Another approach that is related to IND is to directly apply the principles of AD to the integration algorithm. In an extreme case, one could just take the integrator code and process it with an AD tool—this approach can work well for explicit integrators with fixed stepsize, as we show in Example 8.13, but otherwise needs to be applied with care to avoid the many possible pitfalls of a blind application of AD. In particular, for adaptive integrators, one needs to avoid the differentiation of the stepsize selection procedure. If this simple rule is respected, AD in both forward and reverse modes can be easily applied to adaptive explicit integrators, and is both efficient and yields highly accurate results.

For implicit integrators, one should also regard the number and type of Newton-type iterations in each step as constant. Otherwise, the AD tool also tries to differentiate the Jacobian evaluations and factoriza-

tions, which would create unnecessary overhead. When AD is implemented in this way, i.e., if it respects the same guidelines as the IND approach, its forward mode has similar costs, but yields more accurate derivatives than IND. Depending on input and output dimensions, the reverse mode can accelerate computations further.

8.4.7 Algorithmic and Symbolic Differentiation Software

A crucial property of many AD tools is that they are able to process generic code from a standard programming language like C, C++, MATLAB, or FORTRAN, with no or only minor modifications to the source code. For example, the AD tools ADOL-C and CppAD can process generic user-supplied C or C++ code. This is in contrast to computer algebra systems such as Maple, Mathematica, or MATLAB's Symbolic Math Toolbox, which require the user to define the function to be differentiated using symbolic expressions in a domain-specific language. A further advantage of AD over symbolic differentiation is that it is able to provide tight bounds on the length of the resulting derivative code, as well as its runtime and memory requirements. On the other hand, some symbolic tools—such as AMPL or CasADi—make use of AD internally, so the performance differences between algorithmic and symbolic differentiation can become blurry.

An overview of nearly all available AD tools is given at `www.autodiff.org`. Most AD tools implement both the forward and reverse mode of AD, and allow recursive application of AD to generate higher-order derivatives. Some AD tools automatically perform graph-coloring strategies to reduce the cost of Jacobian evaluations, similar to the sparse numerical differentiation algorithm by Curtis et al. (1974) mentioned before in the context of numerical differentiation. We refer to the textbook on algorithmic differentiation by Griewank and Walther (2008) for an in-depth analysis of the different concepts of AD.

8.4.8 CasADi for Optimization

Many of the computational exercises in this text use the open-source tool CasADi, which implements AD on user-defined symbolic expressions. CasADi also provides standardized interfaces to a variety of numerical routines: simulation and optimization, and solution of linear and nonlinear equations. A key feature of these interfaces is that every user-defined CasADi function passed to a numerical solver automatically provides the necessary derivatives to this solver, without

any additional user input. Often, the result of the numerical solver itself can be interpreted as a differentiable CasADi function, such that derivatives up to any order can be generated without actually differentiating the source code of the solver. Thus, concatenated and recursive calls to numerical solvers are possible and still result in differentiable CasADi functions.

CasADi is written in C++, but allows user input to be provided from either C++, Python, Octave, or MATLAB. When CasADi is used from the interpreter languages Python, Octave, or MATLAB, the user does not have any direct contact with C++; but because the internal handling of all symbolic expressions as well as the numerical computations are performed in a compiled environment, the speed of simulation or optimization computations is similar to the performance of compiled C-code. One particularly powerful optimization solver interfaced to CasADi is IPOPT, an open-source C++ code developed and described by Wächter and Biegler (2006). IPOPT is automatically provided in the standard CasADi installation. For more information on CasADi and how to install it, we refer the reader to `casadi.org`. Here, we illustrate the use of CasADi for optimal control in a simple example.

Example 8.13: Sequential optimal control using CasADi from Octave

In the following example we formulate and solve a simple nonlinear MPC problem. The problem is formulated and solved by the sequential approach in discrete time, but the discrete time dynamics are the result of one step of an integrator applied to a continuous time ordinary differential equation (ODE). We go through the example problem and the corresponding solution using CasADi from Octave, which works without changes from MATLAB. The code is available from the book website as the file `casadi-example-mpc-book-1.m` along with a Python version of the same code, `casadi-example-mpc-book-1.py`.

As a first step, we define the ODE describing the system, which is given by a nonlinear oscillator described by the following ODE with $x \in \mathbb{R}^2$ and $u \in \mathbb{R}$

$$\frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \underbrace{\begin{bmatrix} x_2 \\ -x_1 - x_1^3 + u \end{bmatrix}}_{=:f_c(x,u)}$$

with the initial condition $x(0) = [0, 1]'$. We can encode this in Octave as follows

```
% Continuous time dynamics
f_c = @(x, u) [x(2); -x(1) - x(1)^3 + u];
```

To define the discrete time dynamics $x^+ = f(x,u)$, we perform one step of the classical Runge-Kutta method of fourth order. We choose a stepsize of 0.2 seconds. Given $x^+ = f(x,u)$, we can state an MPC optimization problem with zero terminal constraint that we solve, as follows

$$\begin{aligned}
\underset{\mathbf{x},\mathbf{u}}{\text{minimize}} \quad & \sum_{k=0}^{N-1} x(k)' \begin{bmatrix} 10 & 0 \\ 0 & 5 \end{bmatrix} x(k) + u(k)^2 && && \text{(8.17a)}\\
\text{subject to} \quad & x(0) = [1,0]' && && \text{(8.17b)}\\
& x(k+1) = f(x(k),u(k)), && k = 0,1,\ldots,N-1 && \text{(8.17c)}\\
& u(k) \in [-1,1], && k = 0,1,\ldots,N-1 && \text{(8.17d)}\\
& x(N) = [0,0]' && && \text{(8.17e)}
\end{aligned}$$

For its numerical solution, we formulate this problem using the sequential approach, i.e., we regard only $\mathbf{u}$ as optimization variables and eliminate $\mathbf{x}$ by a system simulation. This elimination allows us to generate a cost function $c(\mathbf{u})$ and a constraint function $G(\mathbf{u})$ such that the above problem is equivalent to

$$\begin{aligned}
\underset{\mathbf{u}}{\text{minimize}} \quad & c(\mathbf{u}) && \text{(8.18a)}\\
\text{subject to} \quad & \mathbf{u} \in [-1,1]^N && \text{(8.18b)}\\
& G(\mathbf{u}) = 0 && \text{(8.18c)}
\end{aligned}$$

Here, $c : \mathbb{R}^N \to \mathbb{R}$ and $G : \mathbb{R}^N \to \mathbb{R}^2$, with $N = 50$.

To code this into CasADi/Octave, we begin by declaring a symbolic variable corresponding to $\mathbf{u}$ as follows

```
% Decision variable
N = 50;
U = casadi.SX.sym('U', N);
```

This symbolic variable can be used to construct expressions for c and G

```
% System simulation
xk = [1; 0];
c = 0;
for k=1:N
  % RK4 method
```

```
  dt = 0.2;
  k1 = f_c(xk, U(k));
  k2 = f_c(xk+0.5*dt*k1, U(k));
  k3 = f_c(xk+0.5*dt*k2, U(k));
  k4 = f_c(xk+dt*k3, U(k));
  xk = xk + dt/6.0*(k1 + 2*k2 + 2*k3 + k4);
  % Add contribution to objective function
  c = c + 10*xk(1)^2 + 5*xk(2)^2 + U(k)^2;
end
% Terminal constraint
G = xk - [0; 0];
```

The last remaining step is to pass the expressions for c and G to an optimization solver, more specifically, to the nonlinear programming solver IPOPT. The solver expects an optimization problem with lower and upper bounds for all variables and constraints of the form

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & f(x) \\ \text{subject to} \quad & x_{\text{lb}} \leq x \leq x_{\text{ub}} \\ & g_{\text{lb}} \leq g(x) \leq g_{\text{ub}} \end{aligned} \tag{8.19}$$

To formulate equality constraints in the CasADi syntax for NLPs, one just sets the upper and lower bounds to equal values. The solver also expects an initial guess x_0 for the optimization variables (the initial guess x_0 for the NLP solver is not to be confused with the initial value x_0 for the state trajectory). The interface to the NLP solver uses the keywords `f` and `g` for the functions f and g, `x` for the variables x, `lbx` for x_{lb} etc. The corresponding CasADi code to pass all data to the NLP solver, call it, and retrieve the solution looks as follows.

```
% Create an NLP solver object
nlp = struct('x', U, 'f', c, 'g', G);
solver = casadi.nlpsol('solver', 'ipopt', nlp);
% Solve the NLP
solution = solver('x0', 0, 'lbx', -1, 'ubx', 1,
                  'lbg', 0, 'ubg', 0);
U_opt = solution.x;
```

□

8.5 Direct Optimal Control Parameterizations

Direct optimal control methods transform a continuous time optimal control problem of the form (8.5) into a finite-dimensional optimization

problem. For convenience, we restate the OCP (8.5) in a form that replaces the constraint sets $\mathbb{Z}$ and $\mathbb{X}_f$ by equivalent inequality constraints, as follows

$$\begin{aligned}
\underset{x(\cdot),\,u(\cdot)}{\text{minimize}} \quad & \int_0^T \ell_c(x(t),u(t))\,dt + V_f(x(T)) && \text{(8.20a)}\\
\text{subject to} \quad & x(0) = x_0 && \text{(8.20b)}\\
& \dot{x}(t) = f_c(x(t),u(t)), \quad t \in [0,T] && \text{(8.20c)}\\
& h(x(t),u(t)) \le 0, \qquad t \in [0,T] && \text{(8.20d)}\\
& h_f(x(T)) \le 0 && \text{(8.20e)}
\end{aligned}$$

While the above problem has infinitely many variables and constraints, the idea of direct optimal control methods is to solve instead a related finite-dimensional problem of the general form

$$\begin{aligned}
\underset{w \in \mathbb{R}^{n_w}}{\text{minimize}} \quad & F(w)\\
\text{subject to} \quad & G(x_0,w) = 0\\
& H(w) \le 0
\end{aligned} \qquad (8.21)$$

This finite-dimensional optimization problem is solved for given initial value x_0 with any of the Newton-type optimization methods described in the following section, Section 8.6. In this section, we are concerned only with the transformation of the continuous problem (8.20) into a finite-dimensional problem of form (8.21).

First, one chooses a finite representation of the continuous functions, which is often called *discretization*. This encompasses three parts of the OCP, namely the control trajectory (which is often represented by a piecewise constant function), the state trajectory (which is often discretized using a numerical integration rule), and the path constraints (which are often only imposed on some grid points). Second, one selects the variables w that are finally passed to the optimization solver. These can be all of the discretization variables (in the fully simultaneous or direct transcription approach), but are often only a subset of the parameters that represent the control and state trajectories. The remaining discretization parameters are hidden to the optimization solver, but are implicitly computed during the optimization computations—such as the state trajectories in the sequential approach, or the intermediate quantities in a Runge-Kutta step. Next we present some of the most widely used direct optimal control parameterizations.

8.5.1 Direct Single Shooting

Like most direct methods, the single-shooting approach first parameterizes the control trajectory with a finite-dimensional vector $\mathbf{q} \in \mathbb{R}^{n_q}$ and sets $u(t) = \widetilde{u}(t;\mathbf{q})$ for $t \in [0,T]$. One sometimes calls this step "control vector parameterization." One example for such a function $\widetilde{u} : [0,T] \times \mathbb{R}^{n_q} \to \mathbb{R}^m$ is a polynomial of degree p, which requires $(p+1)$ coefficients for each component of $u(t) \in \mathbb{R}^m$. With this choice, the resulting control parameter $\mathbf{q}$ would have the dimension $n_q = (p+1)m$. A disadvantage of the polynomials—as of any other "global" parameterization—is that the inherent problem sparsity due to the dynamic system structure is inevitably lost. For this reason, and also because it better corresponds to the discrete time implementation of MPC, most often one chooses basis functions with local support, for example, a piecewise constant control parameterization. In this case, one divides the time horizon $[0,T]$ into N subintervals $[t_i, t_{i+1}]$ with $0 = t_0 < t_1 < \ldots < t_N = T$, and sets

$$\widetilde{u}(t;\mathbf{q}) \coloneqq q_i \quad \text{for} \quad t \in [t_i, t_{i+1})$$

For each interval, one needs one vector $q_i \in \mathbb{R}^m$, such that the total dimension of $\mathbf{q} = (q_0, q_1, \ldots, q_{N-1})$ is given by $n_q = Nm$. In the following, we assume this form of piecewise constant control parameterization.

Regarding the state discretization, the direct single-shooting method relies on any of the numerical simulation methods described in Section 8.2 to find an approximation $\widetilde{x}(t;x_0,\mathbf{q})$ of the state trajectory, given the initial value x_0 at $t = 0$ and the control trajectory $\widetilde{u}(t;\mathbf{q})$. Often, adaptive integrators are chosen. In case of piecewise constant controls, the integration needs to stop and restart briefly at the time points t_i to avoid integrating a nonsmooth right-hand-side function. Due to state continuity, the state $\widetilde{x}(t_i;x_0,\mathbf{q})$ is both the initial state of the interval $[t_i, t_{i+1}]$ as well as the last state of the previous interval $[t_{i-1}, t_i]$. The control values used in the numerical integrators on both sides differ, due to the jump at t_i, and are given by q_{i-1} and q_i, respectively.

Evaluating the integral in the objective (8.20a) requires an integration rule. One option is to just augment the ODE system with a *quadrature state* $x_{\text{quad}}(t)$ starting at $x_{\text{quad}}(0) = 0$, and obeying the trivial differential equation $\dot{x}_{\text{quad}}(t) = \ell_{\text{c}}(x(t),u(t))$ that can be solved with the same numerical solver as the standard ODE. Another option is to

evaluate $\ell_c(\widetilde{x}(t;x_0,\mathbf{q}),\widetilde{u}(t;\mathbf{q}))$ on some grid and to apply another integration rule that is external with respect to the integrator. For example, one can use a refinement of the grid that was used for the control discretization, where each interval $[t_i,t_{i+1}]$ is divided into M equally sized subintervals $[\tau_{i,j},\tau_{i,j+1}]$ with $\tau_{i,j} \coloneqq t_i + j/M(t_{i+1}-t_i)$ for $j=0,\dots,M$ and $i=0,\dots,N-1$, and just apply a Riemann sum on each interval to yield the objective function

$$F(x_0,\mathbf{q}) \coloneqq \sum_{i=0}^{N-1}\sum_{j=0}^{M-1} \ell_c(\widetilde{x}(\tau_{i,j};x_0,\mathbf{q}),\widetilde{u}(\tau_{i,j};\mathbf{q}))\ (\tau_{i,j+1}-\tau_{i,j}) + \quad V_f(\widetilde{x}(T;x_0,\mathbf{q}))$$

In the context of the Gauss-Newton method for least squares integrals, this second option is preferable because it allows one to easily obtain a Gauss-Newton Hessian approximation from the sensitivities which are provided by the integrator. Note that the fine grid evaluation as described here requires an integrator able to output the states at arbitrary locations; collocation methods, for example, have this ability. If not, one must select points $\tau_{i,j}$ that coincide with the intermediate steps or stages of the integrator.

The last discretization choice considers the path constraints (8.20d). These often are evaluated on the same grid as the control discretization, or, more generally, on a finer grid, e.g., the time points $\tau_{i,j}$ defined above for the objective integral. Then, only finitely many constraints $h(\widetilde{x}(\tau_{i,j};x_0,\mathbf{q}),\widetilde{u}(\tau_{i,j};\mathbf{q})) \le 0$ are imposed for $j=0,\dots,M$ and $i=0,1,\dots,N-1$. Together with the terminal constraint, one defines the inequality constraint function

$$H(x_0,\mathbf{q}) \coloneqq \begin{bmatrix} h(\widetilde{x}(\tau_{0,0};x_0,\mathbf{q}),\widetilde{u}(\tau_{0,0};\mathbf{q})) \\ h(\widetilde{x}(\tau_{0,1};x_0,\mathbf{q}),\widetilde{u}(\tau_{0,1};\mathbf{q})) \\ \vdots \\ h(\widetilde{x}(\tau_{1,0};x_0,\mathbf{q}),\widetilde{u}(\tau_{1,0};\mathbf{q})) \\ h(\widetilde{x}(\tau_{1,1};x_0,\mathbf{q}),\widetilde{u}(\tau_{1,1};\mathbf{q})) \\ \vdots \\ h(\widetilde{x}(\tau_{N-1,M-1};x_0,\mathbf{q}),\widetilde{u}(\tau_{N-1,M-1};\mathbf{q})) \\ h_f(\widetilde{x}(T;x_0,\mathbf{q})) \end{bmatrix}$$

If the function h maps to $\mathbb{R}^{n_h}$ and h_f to $\mathbb{R}^{n_{h_f}}$, the function H maps to $\mathbb{R}^{(NMn_h+n_{h_f})}$. The resulting finite-dimensional optimization problem in

single shooting is thus given by

$$\begin{aligned} \underset{s_0, \mathbf{q}}{\text{minimize}} \quad & F(s_0, \mathbf{q}) \\ \text{subject to} \quad & s_0 - x_0 = 0 \\ & H(s_0, \mathbf{q}) \leq 0 \end{aligned} \tag{8.22}$$

Of course, the trivial equality constraint $s_0 - x_0 = 0$ could easily be eliminated, and this is often done in single-shooting implementations. In the real-time optimization context, however, it is beneficial to include also the parameter x_0 as a trivially constrained variable s_0 of the single-shooting optimization problem, as we do here. This simple trick is called *initial-value embedding*, and allows one to initialize the optimization procedure with the past initial value s_0, for which an approximately optimal solution already exists; it also allows one to easily obtain a linearized feedback control for new values of x_0, as we discuss in the next section. Also, for moving horizon estimation (MHE) problems, one has to keep the (unconstrained) initial value s_0 as an optimization variable in the single-shooting optimization problem formulation.

In summary, the single-shooting method is a fully sequential approach that treats all intermediate state values computed in the numerical integration routine as hidden variables, and solves the optimization problem in the space of control parameters $\mathbf{q} \in \mathbb{R}^{n_q}$ and initial values $s_0 \in \mathbb{R}^n$ only.

There are many different ways to numerically solve the optimization problem (8.22) in the single-shooting approach using standard methods from the field of nonlinear programming. At first sight, the optimization problem in the single-shooting method is dense, and usually problem (8.22) is solved by a dense NLP solver. However, some single-shooting approaches use a piecewise control parameterization and are able to exploit the intrinsic sparsity structure of the OCP in the NLP solution, as discussed in Section 8.8.5.

8.5.2 Direct Multiple Shooting

The direct multiple-shooting method makes exactly the same discretization choices as the single-shooting method with piecewise control discretization, but it keeps the states $s_i \approx x(t_i)$ at the interval boundary time points as decision variables in the finite-dimensional optimization problem. This allows one to completely decouple the numerical integrations on the separate intervals. For simplicity, we regard

again a piecewise constant control parameterization that uses the constant control value $q_i \in \mathbb{R}^m$ on the interval $[t_i, t_{i+1}]$. On the same interval, we then define the N trajectory pieces $\widetilde{x}_i(t; s_i, q_i)$ that are the numerical solutions of the initial-value problems

$$\widetilde{x}_i(t_i; s_i, q_i) = s_i, \quad \frac{d\widetilde{x}_i}{dt}(t; s_i, q_i) = f_c(\widetilde{x}_i(t; s_i, q_i), q_i), \quad t \in [t_i, t_{i+1}]$$

for $i = 0, 1, \ldots, N-1$. Note that each trajectory piece only depends on the artificial initial value $s_i \in \mathbb{R}^n$ and the local control parameter $q_i \in \mathbb{R}^m$.

Using again a possibly refined grid on each interval, with time points $\tau_{i,j} \in [t_i, t_{i+1}]$ for $j = 0, \ldots, M$, we can formulate numerical approximations of the objective integrals $\int_{t_i}^{t_{i+1}} \ell_c(\widetilde{x}_i(t; s_i, q_i), q_i)\ dt$ on each interval by

$$\ell_i(s_i, q_i) := \sum_{j=0}^{M-1} \ell_c(\widetilde{x}_i(\tau_{i,j}; s_i, q_i), q_i)\ (\tau_{i,j+1} - \tau_{i,j})$$

The overall objective is thus given by $\sum_{i=0}^{N-1} \ell_i(s_i, q_i) + V_f(s_N)$. Note that the objective terms $\ell_i(s_i, q_i)$ each depend again only on the local initial values s_i and local controls q_i, and can thus be evaluated independently from each other. Likewise, we discretize the path constraints, for simplicity on the same refined grid, by defining the local inequality constraint functions

$$H_i(s_i, q_i) := \begin{bmatrix} h(\widetilde{x}_i(\tau_{0,0}; s_i, q_i), q_i) \\ h(\widetilde{x}_i(\tau_{0,1}; s_i, q_i), q_i) \\ \vdots \\ h(\widetilde{x}_i(\tau_{0,M-1}; s_i, q_i), q_i) \end{bmatrix}$$

for $i = 0, 1, \ldots, N-1$. These are again independent functions, with $H_i : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^{(Mn_h)}$. Using these definitions, and the concatenations $\mathbf{s} := (s_0, s_1, \ldots, s_N)$ and $\mathbf{q} := (q_0, \ldots, q_{N-1})$, one can state the finite-dimensional optimization problem that is formulated and solved in

the direct multiple-shooting method

$$\begin{aligned}
\underset{\mathbf{s},\mathbf{q}}{\text{minimize}} \quad & \sum_{i=0}^{N-1} \ell_i(s_i, q_i) \;+\; V_f(s_N) && && (8.23a)\\
\text{subject to} \quad & s_0 = x_0 && && (8.23b)\\
& s_{i+1} = \tilde{x}_i(t_{i+1}; s_i, q_i), && \text{for} \quad i = 0, \ldots, N-1 && (8.23c)\\
& H_i(s_i, q_i) \le 0, && \text{for} \quad i = 0, \ldots, N-1 && (8.23d)\\
& h_f(s_N) \le 0 && && (8.23e)
\end{aligned}$$

By a straightforward definition of problem functions F, G, and H, and optimization variables $w = [s_0' \; q_0' \; s_1' \; q_1' \; \cdots \; s_{N-1}' \; q_{N-1}' \; s_N']'$, the above problem can be brought into the form (8.21).

Note that, due to the presence of $\mathbf{s}$ as optimization variables, the problem dimension is higher than in the single-shooting method, namely $n_w = (N+1)n + Nm$ variables compared with only $(n + Nm)$ in the single-shooting method. On the other hand, the additional Nn equality constraints (8.23c) eliminate the additional Nn degrees of freedom, and the problems (8.23) and (8.22) are fully equivalent if the same integration routines are used. Also note that the multiple-shooting NLP (8.23) has exactly the same form as the discrete time optimal control problem (8.1). From this perspective, the single-shooting problem (8.22) is thus identical to the sequential formulation, compare (8.3), and the multiple-shooting problem is identical to the simultaneous formulation, compare (8.1), of the same discrete time OCP.

When comparing the continuous time problem (8.20) with the nonlinear program (NLP) (8.23) in direct multiple shooting, it is interesting to note that the terminal cost and terminal constraint function are identical, while the cost integrals, the system dynamics, and the path constraints are all numerically approximated in the multiple-shooting NLP.

Multiple versus single shooting. The advantages of multiple compared to single shooting are the facts that the evaluation of the integrator calls can be performed in parallel on the different subintervals, that the state values $\mathbf{s}$ can also be used for initialization of the optimization solver, and that the contraction rate of Newton-type optimization iterations is often observed to be faster, in particular for nonlinear and unstable systems. Its disadvantage for problems without state constraints is that globalization strategies cannot simply rely on the objective function as merit function, but have to also monitor

the residuals of the dynamic constraints (8.23c), which can become cumbersome. Some people also prefer the single-shooting method for the simple reason, that, as a sequential approach, it shows "feasible," or more exactly, "physical" state trajectories in each optimization iteration, i.e., trajectories that satisfy, up to numerical integration errors, the system's differential equation.

We argue here, however, that this reason is not valid, because if one wants to see "physical" trajectories during an optimization run, one could numerically simulate and plot the system evolution for the currently best available guess of the control trajectory $\mathbf{q}$ in any simultaneous method at comparably low additional cost. On the other hand, in the presence of state constraints, the iterates of both sequential and simultaneous methods always lead to slightly infeasible state trajectories, while simultaneous methods often converge even faster in this case. Thus, "feasibility" is not really a reason to prefer one approach over the other.

A theoretical comparison of sequential and simultaneous ("lifted") formulations in the context of Newton-type optimization (Albersmeyer and Diehl, 2010) shows that both methods can be implemented with nearly identical computational cost per iteration. Also, it can be shown—and observed in practice—that simultaneous formulations lead to faster contraction rates if the nonlinearities of the concatenated system dynamics reinforce each other, e.g., if an exponential $x_1 = \exp(x_0)$ is concatenated with an exponential $x_2 = \exp(x_1)$, leading to $x_2 = \exp(\exp(x_0))$. On the other hand, the sequential approach would lead to faster contraction if the concatenated nonlinearities mitigate each other, e.g., if a logarithm $x_2 = \log(x_1)$ follows the exponential $x_1 = \exp(x_0)$ and renders the concatenation $x_2 = \log(\exp(x_0)) = x_0$ the identity (a linear map). In optimal control, one often observes that the concatenation reinforces the nonlinearities, which renders the simultaneous approach favorable.

Exact expressions for linear systems with quadratic costs. In the special case of linear systems $f_c(x,u) = A_c x + B_c u$ with quadratic costs $\ell_c(x,u) = x' Q_c x + u' R_c u$, the exact multiple-shooting functions $\tilde{x}_i(t_{i+1}; s_i, q_i)$ and $\ell_i(s_i, q_i)$ also turn out to be linear and quadratic, and it is possible to compute them explicitly. Specifically

$$\tilde{x}_i(t_{i+1}; s_i, q_i) = A s_i + B q_i$$

with

$$A = \exp\left(A_{\mathrm{c}}\left(t_{i+1} - t_i\right)\right) \quad \text{and} \quad B = \int_0^{(t_{i+1}-t_i)} \exp\left(A_{\mathrm{c}}\tau\right) B_{\mathrm{c}}\, d\tau$$

and

$$\ell_i(s_i, q_i) = \begin{bmatrix} s_i \\ q_i \end{bmatrix}' \begin{bmatrix} Q & S \\ S' & R \end{bmatrix} \begin{bmatrix} s_i \\ q_i \end{bmatrix}$$

with more complicated formulas for Q, R, and S that can be found in Van Loan (1978) or Pannocchia, Rawlings, Mayne, and Mancuso (2015). Note that approximations of the above matrices also can be obtained from the differentiation of numerical integration routines that are applied to the linear ODE system, augmented by the quadratic cost integral. The first-order derivatives of the final states yield A and B, and the second-order derivative of the cost gives Q, R, and S. Because these numerical computations can be done before an actual MPC implementation, they can be performed offline and with high accuracy.

8.5.3 Direct Transcription and Collocation Methods

The idea of simultaneous optimal control can be extended even further by keeping all ODE discretization variables as optimization variables. This fully simultaneous approach is taken in the family of *direct transcription methods*, which directly transcribe all data of the continuous time OCP (8.20) into an NLP without making use of numerical integration routines. Instead, they directly formulate the numerical simulation equations as equalities of the optimization problem. One example of a direct transcription method was already given in the introduction of this chapter, in (8.6), where an explicit Euler integration rule was employed. Because the state equations are equality constraints of the optimization problem, direct transcription methods often use implicit integration rules; they offer higher orders for the same number of state discretization variables, and come with better stability properties for stiff systems. Probably the most popular class of direct transcription methods are the direct collocation methods.

Direct transcription by collocation. In direct collocation, the time horizon $[0, T]$ is first divided into a typically large number N of collocation intervals $[t_i, t_{i+1}]$, with $0 = t_0 < t_1 < \ldots < t_N = T$. On each of these intervals, an implicit Runge-Kutta integration rule of collocation type is applied to transcribe the ODE $\dot{x} = f_{\mathrm{c}}(x, u)$ to a finite set of nonlinear equations. For this aim, we first introduce the states $s_i \approx x(t_i)$ at

the time points t_i, and then regard the implicit Runge-Kutta equations with M stages on the interval with length $h_i := (t_{i+1} - t_i)$, which create an implicit relation between s_i and s_{i+1}. We introduce additional variables $K_i := [k'_{i,1} \ \cdots \ k'_{i,M}]' \in \mathbb{R}^{nM}$, where $k_{i,j} \in \mathbb{R}^n$ corresponds to the state derivative at the collocation time point $t_i + c_j h_i$ for $j = 1, \ldots, M$. These variables K_i are uniquely defined by the collocation equations if s_i and the control value $q_i \in \mathbb{R}^m$ are given. We summarize the collocation equations as $G_i^{\mathrm{RK}}(s_i, K_i, q_i) = 0$ with

$$G_i^{\mathrm{RK}}(s_i, K_i, q_i) := \begin{bmatrix} k_{i,1} & - & f_{\mathrm{c}}(s_i + h_i(a_{11}k_{i,1} + \ldots + a_{1,M}k_{i,M}), q_i) \\ k_{i,2} & - & f_{\mathrm{c}}(s_i + h_i(a_{21}k_{i,1} + \ldots + a_{2,M}k_{i,M}), q_i) \\ & \vdots & \\ k_{i,M} & - & f_{\mathrm{c}}(s_i + h_i(a_{M1}k_{i,1} + \ldots + a_{M,M}k_{i,M}), q_i) \end{bmatrix} \tag{8.24}$$

The transition to the next state is described by $s_{i+1} = F_i^{\mathrm{RK}}(s_i, K_i, q_i)$ with

$$F_i^{\mathrm{RK}}(s_i, K_i, q_i) := s_i + h_i(b_1 k_{i,1} + \ldots + b_M k_{i,M})$$

In contrast to shooting methods, where the controls are often held constant across several integration steps, in direct collocation one usually allows one new control value q_i per collocation interval, as we do here. Even a separate control parameter for every collocation time point within the interval is possible. This would introduce the maximum number of control degrees of freedom that is compatible with direct collocation methods and could be interpreted as a piecewise polynomial control parameterization of order $(M-1)$.

Derivative versus state representation. In most direct collocation implementations, one uses a slightly different formulation, where the intermediate stage derivative variables $K_i = [k'_{i,1} \ \cdots \ k'_{i,M}]' \in \mathbb{R}^{nM}$ are replaced by the stage state variables $S_i = [s'_{i,1} \ \cdots \ s'_{i,M}]' \in \mathbb{R}^{nM}$ that are related to s_i and K_i via the linear map

$$s_{i,j} = s_i + h_i(a_{j1}k_{i,1} \ldots + a_{j,M}k_{i,M}) \quad \text{for} \quad j = 1, \ldots, M \tag{8.25}$$

If $c_1 > 0$, then the relative time points $(0, c_1, \ldots, c_M)$ are all different, such that the interpolation polynomial through the $(M+1)$ states $(s_i, s_{i,1}, \ldots, s_{i,M})$ is uniquely defined, which renders the linear map (8.25) from (s_i, K_i) to (s_i, S_i) invertible. Concretely, the values $k_{i,j}$ can be obtained as the time derivatives of the interpolation polynomial at the collocation time points. The inverse map, for $j = 1, \ldots, M$, is given by

$$k_{i,j} = \frac{1}{h_i}\left(D_{j,1}(s_{i,1} - s_i) + \ldots + D_{j,M}(s_{i,M} - s_i)\right) \tag{8.26}$$

Interestingly, the matrix (D_{jl}) is the inverse of the matrix (a_{mj}) from the Butcher tableau, such that $\sum_{j=1}^{M} a_{mj} D_{jl} = \delta_{ml}$. Inserting this inverse map into $G_i^{\mathrm{RK}}(s_i, K_i, q_i)$ from Eq. (8.24) leads to the equivalent root-finding problem $G_i(s_i, S_i, q_i) = 0$ with

$$G_i(s_i, S_i, q_i) :=$$
$$\begin{bmatrix} \frac{1}{h_i}\left(D_{1,1}(s_{i,1} - s_i) + \ldots + D_{1,M}(s_{i,M} - s_i)\right) & - & f_{\mathrm{c}}(s_{i,1}, q_i) \\ \frac{1}{h_i}\left(D_{2,1}(s_{i,1} - s_i) + \ldots + D_{2,M}(s_{i,M} - s_i)\right) & - & f_{\mathrm{c}}(s_{i,2}, q_i) \\ & \vdots & \\ \frac{1}{h_i}\left(D_{M,1}(s_{i,1} - s_i) + \ldots + D_{M,M}(s_{i,M} - s_i)\right) & - & f_{\mathrm{c}}(s_{i,M}, q_i) \end{bmatrix} \tag{8.27}$$

Likewise, inserting the inverse map into $F_i^{\mathrm{RK}}(s_i, K_i, q_i)$ leads to the linear expression

$$F_i(s_i, S_i, q_i) := s_i + \widetilde{b}_1(s_{i,1} - s_i) + \ldots + \widetilde{b}_M(s_{i,M} - s_i)$$

where the coefficient vector $\widetilde{b} \in \mathbb{R}^M$ is obtained from the RK weight vector b by the relation $\widetilde{b} = D'b$. In the special case that $c_M = 1$, for example in Radau IIA collocation methods, the vector $\widetilde{b}$ becomes a unit vector and the simple relation $F_i(s_i, S_i, q_i) = s_{i,M}$ holds. Because the transition from (s_i, K_i) to (s_i, S_i) just amounts to a basis change, affine invariant Newton-type methods lead to identical iterates independent of the chosen parameterization. However, using either the derivative variables K_i or the state variables S_i leads to different sparsity patterns in the Jacobians and higher-order derivatives of the problem functions. In particular, the Hessian of the Lagrangian is typically sparser if the node state variables S_i are used. For this reason, the state representation is more often used than the derivative representation in direct collocation codes.

Direct collocation optimization problem. The objective integrals $\int_{t_i}^{t_{i+1}} \ell_{\mathrm{c}}(\widetilde{x}(t), q_i)\, dt$ on each interval are canonically approximated by a weighted sum of evaluations of ℓ_{c} on the collocation time points, as follows

$$\ell_i(s_i, S_i, q_i) := h_i \sum_{j=1}^{M} b_j \ell_{\mathrm{c}}(s_{i,j}, q_i)$$

Similarly, one might choose to impose the path constraints on all collocation time points, leading to the stage inequality function

$$H_i(s_i, S_i, q_i) := \begin{bmatrix} h(s_{i,1}, q_i) \\ h(s_{i,2}, q_i) \\ \vdots \\ h(s_{i,M}, q_i) \end{bmatrix}$$

The finite-dimensional optimization problem to be solved in direct collocation has as optimization variables the sequence of external states $\mathbf{s} := (s_0, s_1, \ldots, s_N)$, the sequence of the internal states $\mathbf{S} := (S_0, S_1, \ldots, S_{N-1})$ as well as the sequence of local control parameters, $\mathbf{q} := (q_0, q_1, \ldots, q_{N-1})$, and is formulated as follows

$$\begin{aligned}
\underset{\mathbf{s},\mathbf{S},\mathbf{q}}{\text{minimize}} \quad & \sum_{i=0}^{N-1} \ell_i(s_i, S_i, q_i) \; + \; V_f(s_N) && && \text{(8.28a)} \\
\text{subject to} \quad & s_0 = x_0 && && \text{(8.28b)} \\
& s_{i+1} = F_i(s_i, S_i, q_i), && \text{for} \quad i = 0, \ldots, N-1 && \text{(8.28c)} \\
& 0 = G_i(s_i, S_i, q_i), && \text{for} \quad i = 0, \ldots, N-1 && \text{(8.28d)} \\
& H_i(s_i, S_i, q_i) \leq 0, && \text{for} \quad i = 0, \ldots, N-1 && \text{(8.28e)} \\
& h_f(s_N) \leq 0 && && \text{(8.28f)}
\end{aligned}$$

One sees that the above nonlinear programming problem in direct collocation is similar to the NLP (8.23) arising in the direct multiple-shooting method, but is augmented by the intermediate state variables $\mathbf{S}$ and the corresponding algebraic constraints (8.28d). Typically, it is sparser, but has more variables than the multiple-shooting NLP, not only because of the presence of $\mathbf{S}$, but also because N is larger since it equals the total number of collocation intervals, each of which corresponds to one integration step in a shooting method. Typically, one chooses rather small stage orders M, e.g., two or three, and large numbers for N, e.g., 100 or 1000. The NLPs arising in the direct collocation method are large but sparse. If the sparsity is exploited in the optimization solver, direct collocation can be an extremely efficient optimal control method. For this reason, it is widely used.

Pseudospectral methods. The *pseudospectral optimal control method* can be regarded a special case of the direct collocation method, where only one collocation interval ($N = 1$) is chosen, but with a high-order M. By increasing the order M, one can obtain arbitrarily

high solution accuracies in case of smooth trajectories. The state trajectory is represented by one global polynomial of order M that is uniquely determined by the initial value s_0 and the M collocation node values $s_{0,1}, \ldots, s_{0,M}$. In this approach, the controls are typically parameterized by one parameter per collocation node, i.e., by M distinct values $q_{0,1}, \ldots, q_{0,M}$, such that the control trajectories can be regarded to be represented by global polynomials of order $(M-1)$. One gains a high approximation order, but at the cost that the typical sparsity of the direct collocation problem is lost.

8.6 Nonlinear Optimization

After the finite-dimensional optimization problem is formulated, it needs to be solved. From now on, we assume that a nonlinear program (NLP) of the form (8.21) is formulated, with variable $w \in \mathbb{R}^{n_w}$ and parameter $x_0 \in \mathbb{R}^n$, which we restate here for convenience.

$$\begin{aligned} \underset{w \in \mathbb{R}^{n_w}}{\text{minimize}} \quad & F(w) \\ \text{subject to} \quad & G(x_0, w) = 0 \\ & H(w) \leq 0 \end{aligned} \tag{8.29}$$

As before, we call the above optimization problem $\mathbb{P}_N(x_0)$ to indicate its dependence on the parameter x_0 and on the horizon length N. The aim of the optimization procedure is to reliably and efficiently find an approximation of the solution $w^0(x_0)$ of $\mathbb{P}_N(x_0)$ for a given value of x_0. Inside the MPC loop, the optimization solver is confronted with a sequence of related values of the parameter x_0, a fact that can be exploited in online optimization algorithms to improve speed and reliability compared to standard offline optimization algorithms.

Assumptions and definitions. In this chapter, we make only two assumptions on $\mathbb{P}_N(x_0)$: first, that all problem functions are at least twice continuously differentiable, and second, that the parameter x_0 enters the equalities G linearly, such that the Jacobian matrices G_x and G_w are independent of x_0. This second assumption is satisfied for all problem formulations from the previous sections, because the initial value enters only via the initial-value constraint $s_0 - x_0 = 0$. If one would encounter a problem where the parametric dependence is nonlinear, one could always use the same trick that we used in the single-shooting method and introduce a copy of the parameter as an additional optimization variable s_0—which becomes part of w—and constrain it by

the additional constraint $s_0 - x_0 = 0$. Throughout the section, we often make use of the linearization $H_{\mathrm{L}}(\cdot;\bar{w})$ of a function $H(\cdot)$ at a point $\bar{w}$, i.e., its first-order Taylor series, as follows

$$H_{\mathrm{L}}(w;\bar{w}) := H(\bar{w}) + H_w(\bar{w})\,(w - \bar{w})$$

Due to the linear parameter dependence of G, its Jacobian does not depend on x_0, such that we can write

$$G_{\mathrm{L}}(x_0, w;\bar{w}) = G(x_0,\bar{w}) + G_w(\bar{w})\,(w - \bar{w})$$

We also heavily use the Lagrangian function defined by

$$\mathcal{L}(x_0, w, \lambda, \mu) := F(w) + \lambda' G(x_0, w) + \mu' H(w) \tag{8.30}$$

whose gradient and Hessian matrix with respect to w are often used. Again, they do not depend on x_0, and can thus be written as $\nabla_w \mathcal{L}(w, \lambda, \mu)$ and $\nabla_w^2 \mathcal{L}(w, \lambda, \mu)$. Note that the dimensions of the *multipliers*, or *dual variables* λ and μ, equal the output dimensions of the functions G and H, which we denote by n_G and n_H. We sometimes call $w \in \mathbb{R}^{n_w}$ the *primal* variable. At a feasible point w, we say that an inequality with index $i \in \{1,\ldots,n_H\}$ is *active* if and only if $H_i(w) = 0$. The linear independence constraint qualification (LICQ) is satisfied if and only if the gradients of all active inequalities, $\nabla_w H_i(w) \in \mathbb{R}^{n_w}$, and the gradients of the equality constraints, $\nabla_w G_j(w) \in \mathbb{R}^{n_w}$ for $j \in \{1,\ldots, n_G\}$, form a linearly independent set of vectors.

8.6.1 Optimality Conditions and Perturbation Analysis

The first-order necessary conditions for optimality of the above optimization problem are known as the Karush-Kuhn-Tucker (KKT) conditions, which are formulated as follows.

Theorem 8.14 (KKT conditions)**.** *If w^0 is a local minimizer of the optimization problem $\mathbb{P}_N(x_0)$ defined in* (8.29) *and if LICQ holds at w^0, then there exist multiplier vectors λ^0 and μ^0 such that*

$$\nabla_w \mathcal{L}(w^0, \lambda^0, \mu^0) = 0 \tag{8.31a}$$

$$G(x_0, w^0) = 0 \tag{8.31b}$$

$$0 \geq H(w^0) \perp \mu^0 \geq 0 \tag{8.31c}$$

Here, the last condition, known as the *complementarity condition*, states not only that all components of $H(w^0)$ are negative and all components of μ^0 are positive, but also that the two vectors are orthogonal,

which implies that the products $\mu_i^0 H_i(w^0)$ are zero for each $i \in \{1, \ldots, n_H\}$. Thus, each pair $(H_i(w^0), \mu_i^0) \in \mathbb{R}^2$ must be an element of a nonsmooth, L-shaped subset of $\mathbb{R}^2$ that comprises only the negative x-axis, the positive y-axis, and the origin.

Any triple (w^0, λ^0, μ^0) that satisfies the KKT conditions (8.31) and LICQ is called a KKT point, independent of local optimality.

In general, the existence of multipliers such that the KKT conditions (8.31) hold is just a necessary condition for local optimality of a point w^0 at which LICQ holds. Only in the special case that the optimization problem is convex, the KKT conditions can be shown to be both a necessary and a sufficient condition for global optimality. For the general case, we need to formulate additional conditions on the second-order derivatives of the problem functions to arrive at sufficient conditions for local optimality. This is only possible after making a few definitions.

Strictly active constraints and null space basis. At a KKT point (w, λ, μ), an active constraint with index $i \in \{1, \ldots, n_H\}$ is called *weakly active* if and only if $\mu_i = 0$ and *strictly active* if $\mu_i > 0$. Note that for weakly active constraints, the pair $(H_i(w), \mu_i)$ is located at the origin, i.e., at the nonsmooth point of the L-shaped set. For KKT points without weakly active constraints, i.e., when the inequalities are either strictly active or inactive, we say that the *strict complementarity* condition is satisfied.

Based on the division into weakly and strictly active constraints, one can construct the linear space $\mathcal{Z}$ of directions in which the strictly active constraints and the equality constraints remain constant up to first order. This space $\mathcal{Z}$ plays an important role in the second-order sufficient conditions for optimality that we state below, and can be defined as the null space of the matrix that is formed by putting the transposed gradient vectors of all equality constraints and all strictly active inequality constraints on top of each other. To define this properly at a KKT point (w, λ, μ), we reorder the inequality constraints such that

$$H(w) = \begin{bmatrix} H^+(w) \\ H^0(w) \\ H^-(w) \end{bmatrix}$$

In this reordered view on the function $H(w)$, the strictly active inequality constraints $H^+(w)$ come first, then the weakly active constraints $H^0(w)$, and finally the inactive constraints $H^-(w)$. Note that the output dimensions of the three functions add to n_H. The set $\mathcal{Z} \subset \mathbb{R}^{n_w}$ is

now defined as null space of the matrix

$$A := \begin{bmatrix} G_w(w) \\ H_w^+(w) \end{bmatrix} \in \mathbb{R}^{n_A \times n_w}$$

One can regard an orthogonal basis matrix $Z \in \mathbb{R}^{n_w \times (n_w - n_A)}$ of $\mathcal{Z}$ that satisfies $AZ = 0$ and $Z'Z = I$ and whose columns span $\mathcal{Z}$. This allows us to compactly formulate the following sufficient conditions for optimality.

Theorem 8.15 (Strong second-order sufficient conditions for optimality)**.** *If (w^0, λ^0, μ^0) is a KKT point and if the Hessian of its Lagrangian is positive definite on the corresponding space $\mathcal{Z}$, i.e., if*

$$Z' \nabla_w^2 \mathcal{L}(w^0, \lambda^0, \mu^0) Z > 0 \tag{8.32}$$

then the point w^0 is a local minimizer of problem $\mathbb{P}_N(x_0)$.

We call a KKT point that satisfies the conditions of Theorem 8.15 a *strongly regular* KKT point. We should mention that there exists also a weaker form of second-order sufficient conditions. We prefer to work with the stronger variant because it does not only imply optimality but also existence of neighboring solutions $w^0(x_0)$ as a function of the parameter x_0. Moreover, the solution map $w^0(x_0)$ is directionally differentiable, and the directional derivative can be obtained by the solution of a quadratic program, as stated in the following theorem that summarizes standard results from parametric optimization (Robinson, 1980; Guddat, Vasquez, and Jongen, 1990) and is proven in the specific form below in Diehl (2001).

Theorem 8.16 (Tangential predictor by quadratic program)**.** *If $(\bar{w}, \bar{\lambda}, \bar{\mu})$ is a strongly regular KKT point for problem $\mathbb{P}_N(\bar{x}_0)$ (i.e., it satisfies the conditions of Theorem 8.15) then there is a neighborhood $\mathcal{N} \subset \mathbb{R}^n$ around $\bar{x}_0$ such that for each $x_0 \in \mathcal{N}$ the problem $\mathbb{P}_N(x_0)$ has a local minimizer and corresponding strongly regular KKT point $(w^0(x_0), \lambda^0(x_0), \mu^0(x_0))$. Moreover, the map from $x_0 \in \mathcal{N}$ to $(w^0(x_0), \lambda^0(x_0), \mu^0(x_0))$ is directionally differentiable at $\bar{x}_0$, and the directional derivative can be obtained by the solution of the following quadratic program*

$$\begin{aligned} \underset{w \in \mathbb{R}^{n_w}}{\text{minimize}} \quad & F_{\mathrm{L}}(w; \bar{w}) + \frac{1}{2}(w - \bar{w})' \nabla_w^2 \mathcal{L}(\bar{w}, \bar{\lambda}, \bar{\mu})(w - \bar{w}) \\ \text{subject to} \quad & G_{\mathrm{L}}(x_0, w; \bar{w}) = 0 \\ & H_{\mathrm{L}}(w; \bar{w}) \le 0 \end{aligned} \tag{8.33}$$

More specifically, the solution $(w^{\mathrm{QP}}(x_0), \lambda^{\mathrm{QP}}(x_0), \mu^{\mathrm{QP}}(x_0))$ *of the above QP satisfies*

$$\left\| \begin{bmatrix} w^{\mathrm{QP}}(x_0) - w^0(x_0) \\ \lambda^{\mathrm{QP}}(x_0) - \lambda^0(x_0) \\ \mu^{\mathrm{QP}}(x_0) - \mu^0(x_0) \end{bmatrix} \right\| = O(|x_0 - \bar{x}_0|^2)$$

8.6.2 Nonlinear Optimization with Equalities

When we solve an optimization problem without inequalities, the KKT conditions simplify to

$$\begin{aligned} \nabla_w \mathcal{L}(w^0, \lambda^0) &= 0 \\ G(x_0, w^0) &= 0 \end{aligned}$$

This is a smooth root-finding problem that can be summarized as $R(x_0, z) = 0$ with $z = [w' \ \lambda']'$. Interestingly, if one regards the Lagrangian $\mathcal{L}$ as a function of x_0 and z, we have $R(x_0, z) = \nabla_z \mathcal{L}(x_0, z)$. The classical *Newton-Lagrange method* addresses the above root-finding problem by a Newton iteration of the form

$$z_{k+1} = z_k + \Delta z_k \quad \text{with} \quad R_z(z_k)\Delta z_k = -R(x_0, z_k) \tag{8.35}$$

To simplify notation and avoid that the iteration index k interferes with the indices of the optimization variables, we usually use the following notation for the Newton step

$$z^+ = \bar{z} + \Delta z \quad \text{with} \quad R_z(\bar{z})\Delta z = -R(x_0, \bar{z}) \tag{8.36}$$

Here, the old iterate and linearization point is called $\bar{z}$ and the new iterate z^+. The square Jacobian matrix $R_z(z)$ that needs to be factorized in each iteration to compute Δz has a particular structure and is given by

$$R_z(z) = \begin{bmatrix} \nabla_w^2 \mathcal{L}(w, \lambda) & G_w(w)' \\ G_w(w) & 0 \end{bmatrix}$$

This matrix is called the *KKT matrix* and plays an important role in all constrained optimization algorithms. The KKT matrix is invertible at a point z if the LICQ condition holds, i.e., $G_w(w)$ has rank n_G, and if the Hessian of the Lagrangian is positive definite on the null space of $G_w(w)$, i.e., if $Z' \nabla_w^2 \mathcal{L}(w, \lambda, \mu) Z > 0$, for Z being a null space basis. The matrix $Z' \nabla_w^2 \mathcal{L}(w, \lambda, \mu) Z$ is also called the *reduced Hessian*. Note that the KKT matrix is invertible at a strongly regular point, as well

as in a neighborhood of it, such that Newton's method is locally well defined. The KKT matrix is the second derivative of the Lagrangian $\mathcal{L}$ with respect to the primal-dual variables z, and is therefore symmetric. For this reason, it has only real eigenvalues, but it is typically indefinite. At strongly regular KKT points, it has n_w positive and n_G negative eigenvalues.

Quadratic program interpretation and tangential predictors. A particularly simple optimization problem arises if the objective function is linear quadratic, $F(w) = b'w + (1/2)w'Bw$, and the constraint linear, $G(w) = a + Aw$. In this case, we speak of a quadratic program (QP), and the KKT conditions of the QP directly form a linear system in the variables $z = [w' \ \lambda']'$, namely

$$\begin{bmatrix} B & A' \\ A & 0 \end{bmatrix} \begin{bmatrix} w \\ \lambda \end{bmatrix} = - \begin{bmatrix} b \\ a \end{bmatrix}$$

Due to the equivalence of the KKT conditions of the QP with a linear system one can show that the new point $z^+ = \bar{z} + \Delta z$ in the Newton iteration for the nonlinear problem (8.34) also can be obtained as the solution of a QP

$$\begin{aligned} \underset{w \in \mathbb{R}^{n_w}}{\text{minimize}} \quad & F_{\mathrm{L}}(w; \bar{w}) + \frac{1}{2}(w - \bar{w})' B_{\mathrm{ex}}(\bar{z})(w - \bar{w}) \\ \text{subject to} \quad & G_{\mathrm{L}}(x_0, w; \bar{w}) = 0 \end{aligned} \tag{8.37}$$

with $B_{\mathrm{ex}}(\bar{z}) := \nabla^2_w \mathcal{L}(\bar{w}, \bar{\lambda}, \bar{\mu})$. If the primal-dual solution of the above QP is denoted by w^{QP} and λ^{QP}, one can easily show that setting $w^+ := w^{\mathrm{QP}}$ and $\lambda^+ := \lambda^{\mathrm{QP}}$ yields the same step as the Newton iteration. The interpretation of the Newton step as a QP is not particularly relevant for equality constrained problems, but becomes a powerful tool in the context of inequality constrained optimization. It directly leads to the family of sequential quadratic programming (SQP) methods, which are treated in Section 8.7.1. One interesting observation is that the QP (8.37) is identical to the QP (8.33) from Theorem 8.16, and thus its solution cannot only be used as a Newton step for a fixed value of x_0, but it can also deliver a tangential predictor for changing values of x_0. This property is used extensively in continuation methods for nonlinear MPC, such as the real-time iteration presented in Section 8.9.2.

8.6.3 Hessian Approximations

Even though the reduced exact Hessian is guaranteed to be positive definite at regular points, it can become indefinite at nonoptimal points.

In that case the Newton's method would fail because the KKT matrix would become singular in one iteration. Also, the evaluation of the exact Hessian can be costly. For this reason, Newton-type optimization methods approximate the exact Hessian matrix $B_{\text{ex}}(\bar{z})$ by an approximation $\bar{B}$ that is typically positive definite or at least positive semidefinite, and solve the QP

$$\begin{aligned} &\underset{w \in \mathbb{R}^{n_w}}{\text{minimize}} \quad F_{\text{L}}(w;\bar{w}) + \frac{1}{2}(w-\bar{w})'\bar{B}(w-\bar{w}) \\ &\text{subject to} \quad G_{\text{L}}(x_0,w;\bar{w}) = 0 \end{aligned} \tag{8.38}$$

in each iteration. These methods can be generalized to the case of inequality constrained optimization problems and then fall into the class of sequential quadratic programming (SQP) methods.

The local convergence rate of Newton-type optimization methods can be analyzed directly with the tools from Section 8.3.3. Since the difference between the exact KKT matrix $J(z_k)$ and the Newton-type iteration matrix M_k is due only to the difference in the Hessian approximation, Theorem 8.7 states that convergence can occur only if the difference $B_{\text{ex}}(z_k) - \bar{B}_k$ is sufficiently small, and that the linear contraction factor $\kappa_{\max}$ directly depends on this difference and becomes zero if the exact Hessian is used. Thus, the convergence rate for an exact Hessian SQP method is quadratic, and superlinear convergence occurs if the difference between exact and approximate Hessian shrinks to zero in the relevant directions. Note that the algorithms described in this and the following sections only approximate the Hessian matrix, but evaluate the exact constraint Jacobian $G_w(\bar{w})$ in each iteration.

The constrained Gauss-Newton method. One particularly useful Hessian approximation is possible if the objective function $F(w)$ is a sum of squared residuals, i.e., if

$$F(w) = (1/2)\,|M(w)|^2$$

for a differentiable function $M : \mathbb{R}^{n_w} \to \mathbb{R}^{n_M}$. In this case, the exact Hessian $B_{\text{ex}}(\bar{z})$ is given by

$$\underbrace{M_w(\bar{w})'M_w(\bar{w})}_{=:B_{\text{GN}}(\bar{w})} + \sum_{j=1}^{n_M} M_j(\bar{w})\nabla^2 M_j(\bar{w}) + \sum_{i=1}^{n_G} \bar{\lambda}_i \nabla^2 G_i(\bar{w})$$

By taking only the first part of this expression, one obtains the *Gauss-Newton Hessian approximation* $B_{\text{GN}}(\bar{w})$, which is by definition always

a positive semidefinite matrix. In the case that $M_w(\bar{w}) \in \mathbb{R}^{n_M \times n_w}$ has rank n_w, i.e., if $n_M \geq n_w$ and the n_w columns are linearly independent, the Gauss-Newton Hessian $B_{\mathrm{GN}}(\bar{w})$ is even positive definite. Note that $B_{\mathrm{GN}}(\bar{w})$ does not depend on the multipliers λ, but the error with respect to the exact Hessian does. This error would be zero if both the residuals $M_j(\bar{w})$ and the multipliers λ_i are zero. Because both can be shown to be small at a strongly regular solution with small objective function $(1/2)\,|M(w)|^2$, the Gauss-Newton Hessian $B_{\mathrm{GN}}(\bar{w})$ is a good approximation for problems with small residuals $|M(w)|$.

When the Gauss-Newton Hessian $B_{\mathrm{GN}}(\bar{w})$ is used within a constrained optimization algorithm, as we do here, the resulting algorithm is often called the *constrained* or *generalized Gauss-Newton method* (Bock, 1983). Newton-type optimization algorithms with Gauss-Newton Hessian converge only linearly, but their contraction rate can be surprisingly fast in practice, in particular for problems with small residuals. The QP subproblem that is solved in each iteration of the constrained Gauss-Newton method can be shown to be equivalent to

$$\begin{aligned} \underset{w \in \mathbb{R}^{n_w}}{\text{minimize}} \quad & (1/2)\,|M_{\mathrm{L}}(w;\bar{w})|^2 \\ \text{subject to} \quad & G_{\mathrm{L}}(x_0, w;\bar{w}) = 0 \end{aligned} \tag{8.39}$$

A particularly simple instance of the constrained Gauss-Newton method arises if the objective function is itself already a positive definite quadratic function, i.e., if $F(w) = (1/2)(w - w_{\mathrm{ref}})'B(w - w_{\mathrm{ref}})$. In this case, one could define $M(w) := B^{\frac{1}{2}}(w - w_{\mathrm{ref}})$ to see that the QP subproblem has the same objective as the NLP. Generalizing this approach to nonquadratic, but convex, objectives and convex constraint sets, leads to the class of sequential convex programming methods as discussed and analyzed in Tran-Dinh, Savorgnan, and Diehl (2012).

Hessian update methods. Another way to obtain a cheap and positive definite Hessian approximation $\bar{B}$ for Newton-type optimization is provided by Hessian update methods. In order to describe them, we recall the iteration index k to the primal-dual variables $z_k = [w_k' \; \lambda_k']'$ and the Hessian matrix B_k at the k-th iteration, such that the QP to be solved in each iteration is described by

$$\begin{aligned} \underset{w \in \mathbb{R}^{n_w}}{\text{minimize}} \quad & F_{\mathrm{L}}(w;w_k) + \frac{1}{2}(w - w_k)'B_k(w - w_k) \\ \text{subject to} \quad & G_{\mathrm{L}}(x_0, w;w_k) = 0 \end{aligned} \tag{8.40}$$

In a full step method, the primal-dual solution w_k^{QP} and λ_k^{QP} of the above QP is used as next iterate, i.e., $w_{k+1} := w_k^{\mathrm{QP}}$ and $\lambda_{k+1} := \lambda_k^{\mathrm{QP}}$. A Hessian update formula uses the previous Hessian approximation B_k and the Lagrange gradient evaluations at w_k and w_{k+1} to compute the next Hessian approximation B_{k+1}. Inspired from a directional derivative of the function $\nabla_w \mathcal{L}(\cdot, \lambda_{k+1})$ in the direction $s_k := (w_{k+1} - w_k)$, which, up-to-first order, should be equal to the finite difference approximation $y_k := \nabla_w \mathcal{L}(w_{k+1}, \lambda_{k+1}) - \nabla_w \mathcal{L}(w_k, \lambda_{k+1})$, all Hessian update formulas require the *secant condition*

$$B_{k+1} s_k = y_k$$

One particularly popular way of the many ways to obtain a matrix B_{k+1} that satisfies the secant condition is given by the Broyden-Fletcher-Goldfarb-Shanno (BFGS) formula, which sets

$$B_{k+1} := B_k - \frac{B_k s_k s_k' B_k}{s_k' B_k s_k} + \frac{y_k y_k'}{y_k' s_k}$$

One often starts the update procedure with a scaled unit matrix, i.e., sets $B_0 := \alpha I$ with some $\alpha > 0$. It can be shown that for a positive definite B_k and for $y_k' s_k > 0$, the matrix B_{k+1} resulting from the BFGS formula is also positive definite. In a practical implementation, to ensure positive definiteness of B_{k+1}, the unmodified update formula is only applied if $y_k' s_k$ is sufficiently large, say if the inequality $y_k' s_k \geq \beta s_k' B_k s_k$ is satisfied with some $\beta \in (0, 1)$, e.g., $\beta = 0.2$. If it is not satisfied, the update can either be skipped, i.e., one sets $B_{k+1} := B_k$, or the vector y_k is first modified and then the BFGS update is performed with this modified vector. An important observation is that the gradient difference y_k can be computed with knowledge of the first-order derivatives of F and G at w_k and w_{k+1}, which are needed to define the linearizations F_{L} and G_{L} in the QP (8.40) at the current and next iteration point. Thus, a Hessian update formula does not create any additional costs in terms of derivative computations compared to a fixed Hessian method (like, for example, steepest descent); but it typically improves the convergence speed significantly. One can show that Hessian update methods lead to superlinear convergence under mild conditions.

8.7 Newton-Type Optimization with Inequalities

The necessary optimality conditions for an equality constrained optimization problem form a smooth system of nonlinear equations in the

primal-dual variables, and can therefore directly be addressed by Newton's method or its variants. In contrast to this, the KKT conditions for inequality constrained problems contain the complementarity conditions (8.31c), which define an inherently nonsmooth set in the primal-dual variable space, such that Newton-type methods can be applied only after some important modifications. In this section, we present two widely used classes of methods, namely sequential quadratic programming (SQP) and nonlinear interior point (IP) methods.

8.7.1 Sequential Quadratic Programming

Sequential quadratic programming (SQP) methods solve in each iteration an inequality constrained quadratic program (QP) that is obtained by linearizing all problem functions

$$\begin{aligned} \underset{w \in \mathbb{R}^{n_w}}{\text{minimize}} \quad & F_{\mathrm{L}}(w; w_k) + \frac{1}{2}(w - w_k)' B_k (w - w_k) \\ \text{subject to} \quad & G_{\mathrm{L}}(x_0, w; w_k) = 0 \\ & H_{\mathrm{L}}(w; w_k) \leq 0 \end{aligned} \tag{8.41}$$

The above QP is a quadratic approximation of the nonlinear problem $\mathbb{P}_N(x_0)$, and is denoted by $\mathbb{P}_N^{\mathrm{QP}}(x_0; w_k, B_k)$ to express its dependence on the linearization point w_k and the choice of Hessian approximation B_k. In the full-step SQP method, the primal-dual solution $z_k^{\mathrm{QP}} = (w_k^{\mathrm{QP}}, \lambda_k^{\mathrm{QP}}, \mu_k^{\mathrm{QP}})$ of the QP $\mathbb{P}_N^{\mathrm{QP}}(x_0; w_k, B_k)$ is directly taken as the next iterate, $z_{k+1} = (w_{k+1}, \lambda_{k+1}, \mu_{k+1})$, i.e., one sets $z_{k+1} := z_k^{\mathrm{QP}}$. Note that the multipliers $(\lambda_{k+1}, \mu_{k+1})$ only have an influence on the next QP via the Hessian approximation B_{k+1}, and can be completely discarded in case a multiplier-free Hessian approximation such as a Gauss-Newton Hessian is used.

The solution of an inequality constrained QP is a nontrivial task, but for convex QP problems there exist efficient and reliable algorithms that are just treated here as a black box. To render the QP subproblem convex, one often chooses positive semidefinite Hessian approximations B_k.

Active set detection and local convergence. A crucial property of SQP methods is that the set of active inequalities (the *active set*, in short) is discovered inside the QP solver, and that the active set can change significantly from one SQP iteration to the next. However, one can show that the QP solution discovers the correct active set when

the linearization point w_k is close to a strongly regular solution of the NLP (8.29) at which strict complementarity holds. Thus, in the vicinity of the solution, the active set remains stable, and, therefore, the SQP iterates become identical to the iterates of a Newton-type method for equality constrained optimization applied to a problem where all active constraints are treated as equalities, and where all other inequalities are discarded. Therefore, the local convergence results for general Newton-type methods can be applied; and the SQP method shows quadratic convergence in case of an exact Hessian, superlinear convergence in case of Hessian updates, and linear convergence in case of a Gauss-Newton Hessian.

Generalized tangential predictors in SQP methods. An appealing property of SQP methods for problems that depend on a parameter x_0 is that they deliver a generalized tangential predictor, even at points where the active set changes, i.e., where strict complementarity does not hold. More precisely, it is easily seen that the QP $\mathbb{P}_N^{\mathrm{QP}}(x_0; \bar{w}, \bar{B})$ formulated in an SQP method, with exact Hessian $\bar{B} = \nabla^2 \mathcal{L}(\bar{z})$ at a strongly regular solution $\bar{z} = (\bar{w}, \bar{\lambda}, \bar{\mu})$ of problem $\mathbb{P}_N(\bar{x}_0)$, delivers the tangential predictor of Theorem 8.16 for neighboring problems $\mathbb{P}_N(x_0)$ with $x_0 \neq \bar{x}_0$ (Diehl, 2001). A disadvantage of SQP methods is that they require in each iteration the solution of an inequality constrained QP, which is more expensive than solution of a linear system.

8.7.2 Nonlinear Interior Point Methods

Nonlinear interior point (IP) methods remove the nonsmoothness of the KKT conditions by formulating an approximate, but smooth root-finding problem. This smooth problem corresponds to the necessary optimality conditions of an equality constrained optimization problem that is an approximation of the original problem. In a first and trivial step, the nonlinear inequalities $H(w) \leq 0$ are reformulated into equality constraints $H(w) + s = 0$ by introduction of a slack variable $s \in \mathbb{R}^{n_H}$ that is required to be positive, such that the equivalent new problem has bounds of the form $s \geq 0$ as its only inequality constraints. In the second and crucial step, these bounds are replaced by a barrier term of the form $-\tau \sum_{i=1}^{n_H} \log s_i$ with $\tau > 0$ that is added to the objective. This leads to a different and purely equality constrained optimization

problem given by

$$\begin{aligned} \underset{w,\,s}{\text{minimize}} \quad & F(w) - \tau \sum_{i=1}^{n_H} \log s_i \\ \text{subject to} \quad & G(x_0, w) = 0 \\ & H(w) + s = 0 \end{aligned} \tag{8.42}$$

For $\tau \to 0$, the barrier term $-\tau \log s_i$ becomes zero for any strictly positive $s_i > 0$ while it always grows to infinity for $s_i \to 0$, i.e., on the boundary of the feasible set. Thus, for $\tau \to 0$, the barrier function would be a perfect indicator function of the true feasible set and one can show that the solution of the modified problem (8.42) tends to the solution of the original problem (8.29) for $\tau \to 0$. For any positive $\tau > 0$, the necessary optimality conditions of problem (8.42) are a smooth set of equations, and can, if we denote the multipliers for the equalities $H(w) + s = 0$ by $\mu \in \mathbb{R}^{n_H}$ and keep the original definition of the Lagrangian from (8.30), be equivalently formulated as

$$\nabla_w \mathcal{L}(w, \lambda, \mu) = 0 \tag{8.43a}$$

$$G(x_0, w) = 0 \tag{8.43b}$$

$$H(w) + s = 0 \tag{8.43c}$$

$$\mu_i s_i = \tau \quad \text{for} \quad i = 1, \ldots, n_H \tag{8.43d}$$

Note that for $\tau > 0$, the last condition (8.43d) is a smooth version of the complementarity condition $0 \le s \perp \mu \ge 0$ that would correspond to the KKT conditions of the original problem after introduction of the slack variable s.

A nonlinear IP method proceeds as follows: it first sets τ to a rather large value, and solves the corresponding root-finding problem (8.43) with a Newton-type method for equality constrained optimization. During these iterations, the implicit constraints $s_i > 0$ and $\mu_i > 0$ are strictly enforced by shortening the steps, if necessary, to avoid being attracted by spurious solutions of $\mu_i s_i = \tau$. Then, it slowly reduces the barrier parameter τ; for each new value of τ, the Newton-type iterations are initialized with the solution of the previous problem.

Of course, with finitely many Newton-type iterations, the root-finding problems for decreasing values of τ can only be solved approximately. In practice, one often performs only one Newton-type iteration per problem, i.e., one iterates while one changes the problem. Here, we have sketched the primal-dual IP method as it is for example

implemented in the NLP solver IPOPT (Wächter and Biegler, 2006); but there exist many other variants of nonlinear interior point methods. IP methods also exist in variants that are tailored to linear or quadratic programs and IP methods also can be applied to other convex optimization problems such as second-order cone programs or semidefinite programs (SDP). For these convex IP algorithms, one can establish polynomial runtime bounds, which unfortunately cannot be established for the more general case of nonlinear IP methods described here.

Nonlinear IP methods with fixed barrier parameter. Some variants of nonlinear IP methods popular in the field of nonlinear MPC use a fixed positive barrier parameter τ throughout all iterations, and therefore solve a modified MPC problem. The advantage of this approach is that a simple and straightforward Newton-type framework for equality constrained optimization can be used out of the box. The disadvantage is that for a large value of τ, the modified MPC problem is a conservative approximation of the original MPC problem; for a small value of τ, the nonlinearity due to the condition (8.43d) is severe and slows down the convergence of the Newton-type procedure. Interestingly, these nonlinear IP variants are sometimes based on different barrier functions than the logarithmic barrier described above; they use slack formulations that make violation of the implicit constraint $s_i \geq 0$ impossible by setting, for example, $s_i = (t_i)^2$ with new slacks t_i. This last variant is successfully used for nonlinear MPC by Ohtsuka (2004), and modifies the original problem to a related problem of the form

$$\begin{aligned} \underset{w,t}{\text{minimize}} \quad & F(w) - \tau \sum_{i=1}^{n_H} t_i \\ \text{subject to} \quad & G(x_0, w) = 0 \\ & H_i(w) + (t_i)^2 = 0, \quad i = 1, \ldots, n_H \end{aligned} \tag{8.44}$$

which is then solved by a tailored Newton-type method for equality constrained optimization.

8.7.3 Comparison of SQP and Nonlinear IP Methods

While SQP methods need to solve a QP in each iteration, nonlinear IP methods only solve a linear system of similar size in each iteration, which is cheaper. Some SQP methods even solve the QP by an interior point method, and then perform about 10-30 inner iterations—each of

which is as expensive as the linear system solution in a nonlinear IP method.

On the other hand, the cost per iteration for both SQP and nonlinear IP methods also comprises the evaluation of the problem functions and their derivatives. The number of high-level iterations required to reach a desired level of accuracy is often smaller for SQP methods than for nonlinear IP methods. Also, SQP methods are better at warmstarting, which is particularly important in the context of nonlinear MPC. Roughly speaking, for an NLP with cheap function and derivative evaluations, as in direct collocation, and if no good initial guess is provided, a nonlinear IP method is preferable. An SQP method would be favorable in case of expensive function evaluations, as in direct single or multiple shooting, and when good initial guesses can be provided, for example, if a sequence of neighboring problems is solved.

8.8 Structure in Discrete Time Optimal Control

When a Newton-type optimization method is applied to an optimal control problem, the dynamic system constraints lead to a specific sparsity structure in the KKT matrix. And the quadratic program (QP) in the Newton-type iteration corresponds to a linear quadratic (LQ) optimal control problem with time-varying matrices. To discuss this structure in detail, consider an unconstrained discrete time OCP as it arises in the direct multiple-shooting method

$$\begin{aligned} \underset{w}{\text{minimize}} \quad & \sum_{i=0}^{N-1} \ell_i(x_i, u_i) \; + \; V_f(x_N) \\ \text{subject to} \quad & \bar{\bar{x}}_0 - x_0 = 0 \\ & f_i(x_i, u_i) - x_{i+1} = 0 \quad \text{for} \quad i = 0, \ldots, N-1 \end{aligned} \tag{8.45}$$

Here, the vector $w \in \mathbb{R}^{(N+1)n+Nm}$ of optimization variables is given by $w = [x_0' \; u_0' \; \cdots \; x_{N-1}' \; u_{N-1}' \; x_N']'$. The fixed vector $\bar{\bar{x}}_0$ is marked by two bars to distinguish it from the optimization variable x_0, as well as from a specific value $\bar{x}_0$ of x_0 that is used as linearization point in a Newton-type algorithm. We introduce also a partitioned vector of Lagrange multipliers, $\lambda = [\lambda_0' \; \lambda_1' \; \ldots \; \lambda_N']'$, with $\lambda \in \mathbb{R}^{(N+1)n}$, such that

the Lagrangian of the problem is given by

$$\mathcal{L}(\bar{\tilde{x}}_0, w, \lambda) = \lambda_0'(\bar{\tilde{x}}_0 - x_0) + \sum_{i=0}^{N-1} \ell_i(x_i, u_i) + \lambda_{i+1}'(f_i(x_i, u_i) - x_{i+1}) + V_f(x_N)$$

As before, we can combine w and λ to a vector $z \in \mathbb{R}^{2(N+1)n+Nm}$ of all primal-dual variables. Interestingly, the exact Hessian matrix $B_{\text{ex}}(z) = \nabla_w^2 \mathcal{L}(z)$ is block diagonal (Bock and Plitt, 1984), because the Lagrangian function $\mathcal{L}$ is a sum of independent terms that each depend only on a small subset of the variables—a property called *partial separability*. The exact Hessian is easily computed to be a matrix with the structure

$$B_{\text{ex}}(\bar{z}) = \begin{bmatrix} Q_0 & S_0' & & & & & \\ S_0 & R_0 & & & & & \\ & & \ddots & & & & \\ & & & Q_{N-1} & S_{N-1}' & & \\ & & & S_{N-1} & R_{N-1} & & \\ & & & & & P_N \end{bmatrix} \tag{8.46}$$

where the blocks with index i, only depend on the primal variables with index i and the dual variables with index $(i+1)$. More specifically, for $i = 0, \ldots, N-1$ the blocks are readily shown to be given by

$$\begin{bmatrix} Q_i & S_i' \\ S_i & R_i \end{bmatrix} = \nabla^2_{(x_i,u_i)}[\ell_i(x_i, u_i) + \lambda_{i+1}' f_i(x_i, u_i)]$$

8.8.1 Simultaneous Approach

Most simultaneous Newton-type methods for optimal control preserve the block diagonal structure of the exact Hessian $B_{\text{ex}}(\bar{z})$ and also of the Hessian approximation $\bar{B}$. Thus, the linear quadratic optimization problem (8.38) that is solved in one iteration of a Newton-type optimization method for a given linearization point $\bar{w} = [\bar{x}_0' \ \bar{u}_0' \ \cdots \ \bar{x}_{N-1}' \ \bar{u}_{N-1}' \ \bar{x}_N']'$ and a given Hessian approximation $\bar{B}$ is identical to the following time-varying LQ optimal control problem

$$\begin{aligned} \underset{w}{\text{minimize}} \quad & \sum_{i=0}^{N-1} \ell_{\text{QP},i}(x_i, u_i; \bar{w}, \bar{B}) + V_{\text{QP},f}(x_N; \bar{w}, \bar{B}) \\ \text{subject to} \quad & \bar{\tilde{x}}_0 - x_0 = 0 \\ & f_{\text{L},i}(x_i, u_i; \bar{x}_i, \bar{u}_i) - x_{i+1} = 0 \quad \text{for} \quad i = 0, \ldots, N-1 \end{aligned} \tag{8.47}$$

Here, the quadratic objective contributions $\ell_{\mathrm{QP},i}(x_i, u_i; \bar{w}, \bar{B})$ are given by

$$\ell_i(\bar{x}_i, \bar{u}_i) + \nabla_{(s,q)}\ell_i(\bar{x}_i, \bar{u}_i)' \begin{bmatrix} x_i - \bar{x}_i \\ u_i - \bar{u}_i \end{bmatrix} + \frac{1}{2} \begin{bmatrix} x_i - \bar{x}_i \\ u_i - \bar{u}_i \end{bmatrix}' \begin{bmatrix} \bar{Q}_i & \bar{S}_i' \\ \bar{S}_i & \bar{R}_i \end{bmatrix} \begin{bmatrix} x_i - \bar{x}_i \\ u_i - \bar{u}_i \end{bmatrix}$$

the terminal cost $V_{\mathrm{QP},f}(x_N; \bar{w}, \bar{B})$ is given by

$$V_f(\bar{x}_N) + \nabla V_f(\bar{x}_N)'[x_N - \bar{x}_N] + (1/2)[x_N - \bar{x}_N]'\bar{P}_N[x_N - \bar{x}_N]$$

and the linearized constraint functions $f_{\mathrm{L},i}(x_i, u_i; \bar{x}_i, \bar{u}_i)$ are simply given by

$$f_i(\bar{x}_i, \bar{u}_i) + \underbrace{\frac{\partial f_i}{\partial s}(\bar{x}_i, \bar{u}_i)}_{=:\bar{A}_i}[x_i - \bar{x}_i] + \underbrace{\frac{\partial f_i}{\partial q}(\bar{x}_i, \bar{u}_i)}_{=:\bar{B}_i}[u_i - \bar{u}_i]$$

To create a banded structure, it is advantageous to order the primal-dual variable vector as $z = [\lambda_0' \; x_0' \; u_0' \; \cdots \; \lambda_{N-1}' \; x_{N-1}' \; u_{N-1}' \; \lambda_N' \; x_N']'$; then the solution of the above LQ optimal control problem at iterate $\bar{z}$ corresponds to the solution of a block-banded linear system $\bar{M}_{\mathrm{KKT}} \cdot (z - \bar{z}) = -\nabla_z \mathcal{L}(\bar{x}_0, \bar{z})$, which we can write equivalently as

$$\bar{M}_{\mathrm{KKT}} \cdot z = -\bar{r}_{\mathrm{KKT}} \tag{8.48}$$

where the residual vector is given by $\bar{r}_{\mathrm{KKT}} := \nabla_z \mathcal{L}(\bar{x}_0, \bar{z}) - \bar{M}_{\mathrm{KKT}}\bar{z}$. The matrix $\bar{M}_{\mathrm{KKT}}$ is an approximation of the block-banded KKT matrix $\nabla_z^2 \mathcal{L}(\bar{z})$ and given by

$$\bar{M}_{\mathrm{KKT}} = \begin{bmatrix} 0 & -I & & & & & & & \\ -I & \bar{Q}_0 & \bar{S}_0' & \bar{A}_0' & & & & & \\ & \bar{S}_0 & \bar{R}_0 & \bar{B}_0' & & & & & \\ & \bar{A}_0 & \bar{B}_0 & 0 & -I & & & & \\ & & & -I & \ddots & & & & \\ & & & & & \bar{Q}_{N-1} & \bar{S}_{N-1}' & \bar{A}_{N-1}' & \\ & & & & & \bar{S}_{N-1} & \bar{R}_{N-1} & \bar{B}_{N-1}' & \\ & & & & & \bar{A}_{N-1} & \bar{B}_{N-1} & 0 & -I \\ & & & & & & & -I & \\ & & & & & & & & \bar{P}_N \end{bmatrix} \tag{8.49}$$

Ignoring the specific block structure, this is a banded symmetric matrix with bandwidth $(2n + m)$ and total size $N(2n+m) + 2n$, and the

linear system can thus in principle be solved using a banded LDLT-factorization routine at a cost that is linear in the horizon length N and cubic in $(2n + m)$. There exists a variety of even more efficient solvers for this form of KKT systems with smaller runtime and smaller memory footprint. Many of these solvers exploit the specific block-banded structure of the LQ optimal control problem. Some of these solvers are based on the backward Riccati recursion, as introduced in Section 1.3.3 and Section 6.1.1, and described in Section 8.8.3 for the time-varying case.

8.8.2 Linear Quadratic Problems (LQP)

Consider a time-varying LQ optimal control problem of the form

$$\begin{aligned} \underset{\mathbf{x},\mathbf{u}}{\text{minimize}} \quad & \sum_{i=0}^{N-1} \begin{bmatrix} \bar{q}_i \\ \bar{r}_i \end{bmatrix}' \begin{bmatrix} x_i \\ u_i \end{bmatrix} + \frac{1}{2} \begin{bmatrix} x_i \\ u_i \end{bmatrix}' \begin{bmatrix} \bar{Q}_i & \bar{S}_i' \\ \bar{S}_i & \bar{R}_i \end{bmatrix} \begin{bmatrix} x_i \\ u_i \end{bmatrix} + \bar{p}_N' x_N + \frac{1}{2} x_N' \bar{P}_N x_N \\ \text{subject to} \quad & \bar{\bar{x}}_0 - x_0 = 0 \\ & \bar{b}_i + \bar{A}_i x_i + \bar{B}_i u_i - x_{i+1} = 0 \quad \text{for} \quad i = 0, \ldots, N-1 \end{aligned} \tag{8.50}$$

Here, we use the bar above fixed quantities such as $\bar{A}_i, \bar{Q}_i$ to distinguish them from the optimization variables x_i, u_i, and the quantities that are computed during the solution of the optimization problem. This distinction makes it possible to directly interpret problem (8.50) as the LQ approximation (8.47) of a nonlinear problem (8.45) at a given linearization point $\bar{z} = [\bar{\lambda}_0' \; \bar{x}_0' \; \bar{u}_0' \; \cdots \; \bar{\lambda}_{N-1}' \; \bar{x}_{N-1}' \; \bar{u}_{N-1}' \; \bar{\lambda}_N' \; \bar{x}_N']'$ within a Newton-type optimization method. We call the above problem the linear quadratic problem (LQP), and present different solution approaches for the LQP in the following three subsections.

8.8.3 LQP Solution by Riccati Recursion

One band-structure-exploiting solution method for the above linear quadratic optimization problem is called the Riccati recursion. It can easily be derived by dynamic programming arguments. It is given by three recursions—one expensive matrix and two cheaper vector recursions.

First, and most important, we perform a backward matrix recursion which is started at $P_N := \bar{P}_N$, and goes backward through the indices

$i = N-1, \ldots, 0$ to compute $P_{N-1}, \ldots, P_0$ with the following formula

$$P_i := \bar{Q}_i + \bar{A}_i' P_{i+1} \bar{A}_i - (\bar{S}_i' + \bar{A}_i' P_{i+1} \bar{B}_i)(\bar{R}_i + \bar{B}_i' P_{i+1} \bar{B}_i)^{-1} (\bar{S}_i + \bar{B}_i' P_{i+1} \bar{A}_i) \quad (8.51)$$

The only condition for the above matrix recursion formula to be well defined is that the matrix $(\bar{R}_i + \bar{B}_i' P_{i+1} \bar{B}_i)$ is positive definite, which turns out to be equivalent to the optimization problem being well posed (otherwise, problem (8.50) would be unbounded from below). Note that the Riccati matrix recursion propagates symmetric matrices P_i, whose symmetry can and should be exploited for efficient computations.

The second recursion is a vector recursion that also goes backward in time and is based on the matrices $P_0, \ldots, P_N$ resulting from the first recursion, and can be performed concurrently. It starts with $p_N := \bar{p}_N$ and then runs through the indices $i = N-1, \ldots, 0$ to compute

$$p_i := \bar{q}_i + \bar{A}_i'(P_{i+1} \bar{b}_i + p_{i+1}) - (\bar{S}_i' + \bar{A}_i' P_{i+1} \bar{B}_i)(\bar{R}_i + \bar{B}_i' P_{i+1} \bar{B}_i)^{-1} (\bar{r}_i + \bar{B}_i'(P_{i+1} \bar{b}_i + p_{i+1})) \quad (8.52)$$

Interestingly, the result of the first and the second recursion together yield the optimal cost-to-go functions V_i^0 for the states x_i that are given by

$$V_i^0(x_i) = c_i + p_i' x_i + \frac{1}{2} x_i' P_i x_i$$

where the constants c_i are not of interest here. Also, one directly obtains the optimal feedback control laws u_i^0 that are given by

$$u_i^0(x_i) = k_i + K_i x_i$$

with

$$K_i := -(\bar{R}_i + \bar{B}_i' P_{i+1} \bar{B}_i)^{-1} (\bar{S}_i + \bar{B}_i' P_{i+1} \bar{A}_i) \quad \text{and} \quad (8.53a)$$

$$k_i := -(\bar{R}_i + \bar{B}_i' P_{i+1} \bar{B}_i)^{-1} (\bar{r}_i + \bar{B}_i'(P_{i+1} \bar{b}_i + p_{i+1})) \quad (8.53b)$$

Based on these data, the optimal solution to the optimal control problem is obtained by a forward vector recursion that is nothing other than a forward simulation of the linear dynamics using the optimal feedback control law. Thus, the third recursion starts with $x_0 := \bar{\bar{x}}_0$ and goes through $i = 0, \ldots, N-1$ computing

$$u_i := k_i + K_i x_i \quad (8.54a)$$

$$x_{i+1} := \bar{b}_i + \bar{A}_i x_i + \bar{B}_i u_i \quad (8.54b)$$

For completeness, one would simultaneously also compute the Lagrange multipliers λ_i, which are for $i = 0, \ldots, N$ given by the gradient of the optimal cost-to-go function at the solution

$$\lambda_i := p_i + P_i x_i \tag{8.54c}$$

The result of the three recursions of the Riccati algorithm is a vector $z = [\lambda_0' \; x_0' \; u_0' \; \cdots \; \lambda_{N-1}' \; x_{N-1}' \; u_{N-1}' \; \lambda_N' \; x_N']'$ that solves the linear system $\bar{M}_{\mathrm{KKT}} \cdot z = -\bar{r}_{\mathrm{KKT}}$ with a right-hand side that is given by $\bar{r}_{\mathrm{KKT}} = [\bar{\tilde{x}}_0' \; \bar{q}_0' \; \bar{r}_0' \; \bar{b}_0' \; \cdots \bar{q}_{N-1}' \; \bar{r}_{N-1}' \; \bar{b}_{N-1}' \; \bar{p}_N']'$.

The matrix recursion (8.51) can be interpreted as a factorization of the KKT matrix $\bar{M}_{\mathrm{KKT}}$, and in an efficient implementation it needs about $N(7/3n^3 + 4n^2m + 2nm^2 + 1/3m^3)$ FLOPs, which is about one-third the cost of a plain banded LDLT-factorization.

On the other hand, the two vector recursions (8.52) and (8.54a)-(8.54c) can be interpreted as a linear system solve with the already factorized matrix $\bar{M}_{\mathrm{KKT}}$. In an efficient implementation, this linear system solve needs about $N(8n^2 + 8nm + 2n^2)$ FLOPs.

If care is taken to reduce the number of memory movements and to optimize the linear algebra operations for full CPU usage, one can obtain significant speedups in the range of one order of magnitude compared to a standard implementation of the Riccati recursion—even for small- and medium-scale dynamic systems (Frison, 2015). With only minor modifications, the Riccati recursion can be used inside an interior point method for inequality constrained optimal control problems.

8.8.4 LQP Solution by Condensing

A different way to exploit the block-sparse structure of the LQ optimal control problem (8.50) is to first eliminate the state trajectory $\mathbf{x} = [x_0' \; x_1' \; \cdots \; x_N']'$ as a function of the initial value $\bar{\tilde{x}}_0$ and the control $\mathbf{u} = [u_0' \; u_1' \; \cdots \; u_{N-1}']'$. After subdivision of the variables into states and controls, the equality constraints of the QP (8.50) can be expressed in the following form, where we omit the bar above the system matrices and vectors for better readability

$$\underbrace{\begin{bmatrix} I & & & & \\ -A_0 & I & & & \\ & -A_1 & I & & \\ & & \ddots & \ddots & \\ & & & -A_{N-1} & I \end{bmatrix}}_{=:\mathcal{A}} \mathbf{x} = \underbrace{\begin{bmatrix} 0 \\ b_0 \\ b_1 \\ \vdots \\ b_{N-1} \end{bmatrix}}_{=:b} + \underbrace{\begin{bmatrix} I \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}}_{=:\mathcal{I}} \bar{\tilde{x}}_0 + \underbrace{\begin{bmatrix} 0 & & & \\ B_0 & & & \\ & B_1 & & \\ & & \ddots & \\ & & & B_{N-1} \end{bmatrix}}_{=:\mathcal{B}} \mathbf{u}$$

It can easily be shown that the inverse of $\mathcal{A}$ is given by

$$\mathcal{A}^{-1} = \begin{bmatrix} I & & & & \\ A_0 & I & & & \\ A_1 A_0 & A_1 & I & & \\ \vdots & \vdots & \vdots & \ddots & \\ (A_{N-1} \cdots A_0) & (A_{N-1} \cdots A_1) & (A_{N-1} \cdots A_2) & & I \end{bmatrix}$$

and state elimination results in the affine map

$$\mathbf{x} = \mathcal{A}^{-1} b \; + \; \mathcal{A}^{-1} \mathcal{I} \, \bar{\bar{x}}_0 \; + \; \mathcal{A}^{-1} \mathcal{B} \, \mathbf{u}$$

Using this explicit expression to eliminate all states in the objective results in a condensed, unconstrained quadratic optimization problem of the form

$$\underset{\mathbf{u}}{\text{minimize}} \quad c + \begin{bmatrix} q \\ r \end{bmatrix}' \begin{bmatrix} \bar{\bar{x}}_0 \\ \mathbf{u} \end{bmatrix} + \frac{1}{2} \begin{bmatrix} \bar{\bar{x}}_0 \\ \mathbf{u} \end{bmatrix}' \begin{bmatrix} \mathcal{Q} & S' \\ S & \mathcal{R} \end{bmatrix} \begin{bmatrix} \bar{\bar{x}}_0 \\ \mathbf{u} \end{bmatrix} \tag{8.55}$$

that is equivalent to the original optimal control problem (8.50). Condensing algorithms process the vectors and matrices of the sparse problem (8.50) to yield the data of the condensed QP (8.55)—in particular the Hessian $\mathcal{R}$—and come in different variants. One classical condensing algorithm has a cost of about $(1/3)N^3nm^2$ FLOPS; a second variant, that can be derived by applying reverse AD to the quadratic cost function, has a different complexity and costs about $N^2(2n^2m + nm^2)$ FLOPs. See Frison (2015) for a detailed overview of these and other condensing approaches.

After condensing, the condensed QP still needs to be solved, and the solution of the above unconstrained QP (8.50) is given by $\mathbf{u}^0 = -\mathcal{R}^{-1}(r + S\bar{\bar{x}}_0)$. Because the Hessian $\mathcal{R}$ is a dense symmetric and usually positive definite matrix of size (Nm), it can be factorized using a Cholesky decomposition, which costs about $(1/3)N^3m^3$ FLOPs. Interestingly, the Cholesky factorization also could be computed simultaneously with the second condensing procedure mentioned above, which results in an additional cost of only about Nm^3 FLOPs (Frison, 2015), resulting in a condensing based Cholesky factorization of quadratic complexity in N, as discovered by Axehill and Morari (2012). The condensing approach can easily be extended to the case of additional constraints, and results in a condensed QP with Nm variables and some additional equality and inequality constraints that can be addressed by a dense QP solver.

Is condensing a sequential approach? Condensing is similar in spirit to a sequential approach that is applied to the LQ subproblem. To distinguish the different algorithmic ingredients, we reserve the term "sequential" for the nonlinear OCP only, while we speak of "condensing" when we refer to an LQ optimal control problem. This distinction is useful because all four combinations of sequential or simultaneous approaches with either the Riccati recursion or the condensing algorithm are possible, and lead to different algorithms. For example, when the simultaneous approach is combined with the condensing algorithm, it leads to different Newton-type iterates than the plain sequential approach, even though the linear algebra operations in the quadratic subproblems are similar.

Comparing Riccati recursion and condensing. The Riccati recursion, or, more generally, the banded-LDLT-factorization approaches, have a runtime that is linear in the horizon length N; they are therefore always preferable to condensing for long horizons. They can easily be combined with interior point methods and result in highly competitive QP solution algorithms. On the other hand, condensing-based QP solutions become more competitive than the Riccati approach for short to moderate horizon lengths N—in particular if the state dimension n is larger than the control dimension m, and if an efficient dense active set QP solver is used for the condensed QPs. Interestingly, one can combine the advantages of condensing and band structured linear algebra to yield a *partial condensing* method (Axehill, 2015), which is even more efficient than the plain Riccati approach on long horizons.

8.8.5 Sequential Approaches and Sparsity Exploitation

So far, we have only presented the solution of the unconstrained OCP by Newton-type methods in the *simultaneous approach*, to highlight the specific sparsity structure that is inherent in the resulting LQ problem. Many Newton-type algorithms also exist which are based on the *sequential approach*, however, where the Newton-type iterations are performed in the space of control sequences $\mathbf{u} = [u_0' \ \cdots \ u_{N-1}']'$ only. We recall that one eliminates the state trajectory by a nonlinear forward simulation in the sequential approach to maintain physically feasible trajectories. The plain sequential approach does not exploit sparsity and is not applicable to strongly unstable systems. Interestingly, some sequential approaches exist that do exploit the sparsity structure of the OCP and some—notably differential dynamic programming—even

incorporate feedback into the forward simulation to better deal with unstable dynamic systems.

Plain dense sequential approach. We start by describing how the plain sequential approach—the direct single-shooting method introduced in Section 8.5.1—solves the unconstrained OCP (8.45) with a Newton-type method. Here, all states are directly eliminated as a function of the controls by a forward simulation that starts at $x_0 := \bar{\bar{x}}_0$ and recursively defines $x_{i+1} := f_i(x_i, u_i)$ for $i = 0, \dots, N-1$. The result is that the objective function $F(\bar{\bar{x}}_0, \mathbf{u}) := \sum_{i=0}^{N-1} \ell_i(x_i, u_i) + V_f(x_N)$ directly depends on all optimization variables $\mathbf{u} = [u_0' \ \cdots \ u_{N-1}']'$. The task of optimization now is to find a root of the nonlinear equation system $\nabla_{\mathbf{u}} F(\bar{\bar{x}}_0, \mathbf{u}) = 0$. At some iterate $\bar{\mathbf{u}}$, after choosing a Hessian approximation $\bar{B} \approx \nabla^2_{\mathbf{u}} F(\bar{\bar{x}}_0, \bar{\mathbf{u}})$, one has to solve linear systems of the form

$$\bar{B}(\mathbf{u} - \bar{\mathbf{u}}) = -\nabla_{\mathbf{u}} F(\bar{\bar{x}}_0, \bar{\mathbf{u}}) \tag{8.56}$$

It is important to note that the exact Hessian $\nabla^2_{\mathbf{u}} F(\bar{\bar{x}}_0, \bar{\mathbf{u}})$ is a dense matrix of size Nm (where m is the control dimension), and that one usually also chooses a dense Hessian approximation $\bar{B}$ that is ideally positive definite.

A Cholesky decomposition of a symmetric positive definite linear system of size Nm has a computational cost of $(1/3)(Nm)^3$ FLOPs, i.e., the iteration cost of the plain sequential approach grows cubically with the horizon length N. In addition to the cost of the linear system solve, one has to consider the cost of computing the gradient $\nabla_{\mathbf{u}} F(\bar{\bar{x}}_0, \bar{\mathbf{u}})$. This is ideally done by a backward sweep equivalent to the reverse mode of algorithmic differentiation (AD) as stated in (8.16), at a cost that grows linearly in N. The cost of forming the Hessian approximation depends on the chosen approximation, but is typically quadratic in N. For example, an exact Hessian could be computed by performing Nm forward derivatives of the gradient function $\nabla_{\mathbf{u}} F(\bar{\bar{x}}_0, \mathbf{u})$.

The plain dense sequential approach results in a medium-sized optimization problem without much sparsity structure but with expensive function and derivative evaluations, and can thus be addressed by a standard nonlinear programming method that does not exploit sparsity, but converges with a limited number of function evaluations. Typically, an SQP method in combination with a dense active set QP solver is used.

Sparsity-exploiting sequential approaches. Interestingly, one can form and solve the same linear system as in (8.56) by using the sparse

linear algebra techniques described in the previous section for the simultaneous approach. To implement this, it would be easiest to start with an algorithm for the simultaneous approach that computes the full iterate in the vector z that contains as subsequences the controls $\mathbf{u} = [u_0' \ \cdots \ u_{N-1}']'$, the states $\mathbf{x} = [x_0' \ \cdots \ x_N]'$, and the multipliers $\lambda = [\lambda_0' \ \cdots \ \lambda_N']'$. After the linear system solve, one would simply overwrite the states $\mathbf{x}$ by the result of a nonlinear forward simulation for the given controls $\mathbf{u}$.

The sparse sequential approach is particularly easy to implement if a Gauss-Newton Hessian approximation is used (Sideris and Bobrow, 2005). To compute the exact Hessian blocks, one performs a second reverse sweep identical to (8.16) to overwrite the values of the multipliers λ. As in the simultaneous approach, the cost for each Newton-type iteration would be linear in N with this approach, while one can show that the resulting iterates would be identical to those of the dense sequential approach for both the exact and the Gauss-Newton Hessian approximations.

8.8.6 Differential Dynamic Programming

The sequential approaches presented so far first compute the complete control trajectory $\mathbf{u}$ in each iteration, and then simulate the nonlinear system open loop with this trajectory $\mathbf{u}$ to obtain the states $\mathbf{x}$ for the next linearization point. In contrast, differential dynamic programming (DDP) (Mayne, 1966; Jacobson and Mayne, 1970) uses the time-varying affine feedback law $u_i^0(x_i) = k_i + K_i x_i$ from the Riccati recursion to simulate the nonlinear system forward in time. Like other sequential approaches, the DDP algorithm starts with an initial guess for the control trajectory—or the assumption of some feedback law—and the corresponding state trajectory. But then in each DDP iteration, starting at $x_0 := \bar{x}_0$, one recursively defines for $i = 0, 1, \ldots, N-1$

$$u_i := k_i + K_i x_i \tag{8.57a}$$

$$x_{i+1} := f_i(x_i, u_i) \tag{8.57b}$$

with K_i and k_i from (8.53a) and (8.53b), to define the next control and state trajectory. Interestingly, DDP only performs the backward recursions (8.51) and (8.52) from the Riccati algorithm. The forward simulation of the linear system (8.54b) is replaced by the forward simulation of the *nonlinear* system (8.57b). Note that both the states and the controls in DDP are different from the standard sequential approach.

DDP with Gauss-Newton Hessian. Depending on the type of Hessian approximation, different variants of DDP can be derived. Conceptually the easiest is DDP with a Gauss-Newton Hessian approximation, because it has no need of the multipliers λ_i. In case of a quadratic objective with positive semidefinite cost matrices, these matrices coincide with the Gauss-Newton Hessian blocks, and the method becomes particularly simple; one needs only to compute the system linearization matrices $\bar{A}_i, \bar{B}_i$ for $i = 0, \dots, N-1$ at the trajectory $(\mathbf{x}, \mathbf{u})$ from the previous iteration to obtain all data for the LQ optimal control problem, and then perform the backward recursions (8.51) and (8.52) to define K_i and k_i in (8.53a) and (8.53b). This DDP variant is sometimes called iterative linear quadratic regulator (LQR) (Li and Todorov, 2004) and is popular in the field of robotics. Like any method based on the Gauss-Newton Hessian, the iterative LQR algorithm has the advantage that the Hessian approximation is always positive semidefinite, but the disadvantage that its convergence rate is only linear.

DDP with exact Hessian. In contrast to the iterative LQR algorithm, the DDP algorithm from Mayne (1966) uses an exact Hessian approximation and thus offers a quadratic rate of convergence. Like all exact Hessian methods, it can encounter indefiniteness of the Hessian, which can be addressed by algorithmic modifications that are beyond our interest here. To compute the exact Hessian blocks

$$\begin{bmatrix} \bar{Q}_i & \bar{S}_i' \\ \bar{S}_i & \bar{R}_i \end{bmatrix} := \nabla^2_{(x_i,u_i)} [\ell_i(\bar{x}_i, \bar{u}_i) + \bar{\lambda}_{i+1}' f_i(\bar{x}_i, \bar{u}_i)]$$

the DDP algorithm needs not only the controls $\bar{u}_i$, but also the states $\bar{x}_i$ and the Lagrange multipliers $\bar{\lambda}_{i+1}$, which are not part of the memory of the algorithm. While the states $\bar{x}_i$ are readily obtained by the nonlinear forward simulation (8.57b), the Lagrange multipliers $\bar{\lambda}_{i+1}$ are obtained simultaneously with the combined backward recursions (8.51) and (8.52). They are chosen as the gradient of the quadratic cost-to-go function $V_i^0(x_i) = p_i' x_i + \frac{1}{2} x_i' P_i x_i$ at the corresponding state values, i.e., as

$$\bar{\lambda}_i := p_i + P_i \bar{x}_i \tag{8.58}$$

for $i = N-1, \dots, 0$. The last Hessian block (which is needed first in the backward recursion) is independent of the multipliers and just given by the second derivative of the terminal cost and defined by $\bar{P}_N := \nabla^2 V_f(\bar{x}_N)$. Because $\bar{p}_N := \nabla V_f(\bar{x}_N) - \bar{P}_N \bar{x}_N$, the last multiplier is given by $\bar{\lambda}_N := \nabla V_f(\bar{x}_N)$. Starting with these values for $\bar{P}_N, \bar{p}_N$, and $\bar{\lambda}_N$, the

backward Riccati recursions (8.51) and (8.52) can be started and the Lagrange multipliers be computed simultaneously using (8.58).

The DDP algorithm in its original form is only applicable to unconstrained problems, but can easily be adapted to deal with control constraints. In order to deal with state constraints, a variety of heuristics can be employed that include, for example, barrier methods; a similar idea was presented in the more general context of constrained OCPs under the name *feasibility perturbed sequential quadratic programming* by Tenny, Wright, and Rawlings (2004).

8.8.7 Additional Constraints in Optimal Control

Most Newton-type methods for optimal control can be generalized to problems with additional equality or inequality constraints. In nonlinear MPC, these additional constraints could be terminal equality constraints of the form $r(x_N) = 0$, as in the case of a zero terminal constraint; or terminal inequality constraints of the form $r(x_N) \leq 0$, as in the case of a terminal region. They could also be path constraints of the form $r_i(x_i, u_i) = 0$ or $r_i(x_i, u_i) \leq 0$ for $i = 0, \ldots, N-1$. The Lagrangian function then comprises additional contributions, but the block diagonal structure of the exact Hessian in (8.46) and the general sparsity of the problem is preserved.

Simultaneous approaches. If the multipliers for the extra constraints are denoted by μ_i for $i = 0, \ldots, N$, the Lagrangian in the simultaneous approaches is given by

$$\begin{aligned}\mathcal{L}(\bar{\tilde{x}}_0, w, \lambda, \mu) = \lambda_0'(\bar{\tilde{x}}_0 - x_0) + \mu_N' r_N(x_N) \; + \; V_f(x_N) \\ + \sum_{i=0}^{N-1} \ell_i(x_i, u_i) + \lambda_{i+1}'(f_i(x_i, u_i) - x_{i+1}) + \mu_i' r_i(x_i, u_i)\end{aligned}$$

We can summarize all primal-dual variables in a vector $z := [w' \; \lambda' \; \mu']'$ and write the Lagrangian as $\mathcal{L}(\bar{\tilde{x}}_0, z)$. In the purely equality-constrained case, Newton-type optimization algorithms again just try to find a root of the nonlinear equation system $\nabla_z \mathcal{L}(z) = 0$ by solving at a given iterate $\bar{z}$ the linear system $\bar{M}(z - \bar{z}) = -\nabla_z \mathcal{L}(\bar{z})$ where $\bar{M}$ is an approximation of the exact KKT matrix $\nabla_z^2 \mathcal{L}(\bar{z})$. In the presence of inequalities, one can resort to SQP or nonlinear IP methods. In all cases, the Lagrangian remains partially separable and the KKT matrix has a similar sparsity structure as for the unconstrained OCP. Therefore, the

linear algebra operations again can be performed by band-structure-exploiting algorithms that have a linear complexity in the horizon length N, if desired, or by condensing based approaches.

One major difference with unconstrained optimal control is that the overall feasibility of the optimization problem and the satisfaction of the linear independence constraint qualification (LICQ) condition is no longer guaranteed a priori, and thus, care needs to be taken in formulating well-posed constrained OCPs. For example, one immediately runs into LICQ violation problems if one adds a zero terminal constraint $x_N = 0$ to a problem with a large state dimension n, but a small control dimension m, and such a short time horizon N that the total number of control degrees of freedom, Nm, is smaller than n. In these unfortunate circumstances, the total number of equality constraints, $(N+1)n+n$, would exceed the total number of optimization variables, $(N+1)n+Nm$, making satisfaction of LICQ impossible.

Sequential approaches. Like the simultaneous approaches, most sequential approaches to optimal control—with the exception of DDP—can easily be generalized to the case of extra equality constraints, with some adaptations to the linear algebra computations in each iteration. For the treatment of inequality constraints on states and controls, one can again resort to SQP or nonlinear IP-based solution approaches. In the presence of state constraints, however, the iterates violate in general these state constraints; thus the iterates are infeasible points of the optimization problem, and the main appeal of the sequential approach is lost. On the other hand, the disadvantages of the sequential approach, i.e., the smaller region of convergence and slower contraction rate, especially for nonlinear and unstable systems, remain or become even more pronounced. For this reason, state constrained optimal control problems are most often addressed with simultaneous approaches.

8.9 Online Optimization Algorithms

Optimization algorithms for model predictive control need to solve not only one OCP, but a sequence of problems $\mathbb{P}_N(x_0)$ for a sequence of different values of x_0, and the time to work on each problem is limited by the sampling time Δt. Many different ideas can be used alone or in combination to ensure that the numerical approximation errors do not become too large and that the computation times remain bounded. In this section, we first discuss some general algorithmic considerations,

then present the important class of *continuation methods* and discuss in some detail the *real-time iteration*.

8.9.1 General Algorithmic Considerations

We next discuss one by one some general algorithmic ideas to adapt standard numerical optimal control methods to the context of online optimization for MPC.

Coarse discretization of control and state trajectories. The CPU time per Newton-type iteration strongly depends on the number of optimization variables in the nonlinear program (NLP), which itself depends on the horizon length N, the number of free control parameters, and on the state discretization method. To keep the size of the NLP small, one would classically choose a relatively small horizon length N, and employ a suitable terminal cost and constraint set that ensures recursive feasibility and nominal stability in case of exact NLP solutions. The total number of control parameters would then be Nm, and the state discretization would be equally accurate on all N control intervals.

In the presence of modeling errors and unavoidably inexact NLP solutions, however, one could also accept additional discretization errors by choosing a coarser control or state discretization, in particular in the end of the MPC horizon. Often, practitioners use *move blocking* where only the first $M \ll N$ control moves in the MPC horizon have the feedback sampling time Δt. The remaining $(N - M)$ control moves are combined into blocks of size two or larger, such that the overall number of control parameters is less than Nm. In particular if a plain dense single-shooting algorithm is employed, move blocking can significantly reduce the CPU cost per iteration. Likewise, one could argue that the state evolution need only be simulated accurately on the immediate future, while a coarser state discretization could be used toward the end of the horizon.

From the viewpoint of dynamic programming, one could argue that only the first control interval of duration Δt needs to be simulated accurately using the exact discrete time model $x_1 = f(x_0, u_0)$, while the remaining $(N - 1)$ intervals of the MPC horizon only serve the purpose of providing an approximation of the gradient of the cost-to-go function, i.e., of the gradient of $V_{N-1}(f(x_0, u_0))$. Since the discrete time dynamics usually originate from the approximation of a continuous time system, one could even decide to use a different state and control parameterization on the remaining time horizon. For example, after

the first interval of length Δt, one could use one single long collocation interval of length $(N-1)\Delta t$ with one global polynomial approximation of states and controls, as in pseudospectral collocation, in the hope of obtaining a cheaper approximation of $V_{N-1}(f(x_0, u_0))$.

Code generation and fixed matrix formats. Since MPC optimization problems differ only in the value x_0, many problem functions, and even some complete matrices in the Newton-type iterations, remain identical across different optimization problems and iterations. This allows for the code generation of optimization solvers that are tailored to the specific system model and MPC formulation. While the user interface can be in a convenient high-level language, the automatically generated code is typically in a standardized lower-level programming language such as plain C, which is supported by many embedded computing systems. The generated code has fixed matrix and vector dimensions, needs no online memory allocations, and contains no or very few switches. As an alternative to code generation, one could also just fix the matrix and vector dimensions in the most computationally intensive algorithmic components, and use a fixed specific matrix storage format that is optimized for the given computing hardware.

Delay compensations by prediction. Often, at a sampling instant t_0, one has a current state estimate x_0, but knows in advance that the MPC optimization calculations take some time, e.g., a full sampling time Δt. In the meantime, i.e., on the time interval $[t_0, t_0 + \Delta t]$, one usually has to apply some previously computed control action u_0. As all this is known before the optimization calculations start, one could first predict the expected state $x_1 := f(x_0, u_0)$ at the time $(t_0 + \Delta t)$ when the MPC computations are finished, and directly let the optimization algorithm address the problem $\mathbb{P}_N(x_1)$. Though this prediction approach cannot eliminate the feedback delay of one sampling time Δt in case of unforeseen disturbances, it can alleviate its effect in the case that model predictions and reality are close to each other.

Division into preparation and feedback phases. An additional idea is to divide the computations during each sampling interval into a long preparation phase, and a much shorter feedback phase that could, for example, consist of only a matrix vector multiplication in case of linear state feedback. We assume that the computations in the feedback phase take a computational time Δt_{fb} with $\Delta t_{\text{fb}} \ll \Delta t$, while the preparation time takes the remaining duration of one sampling interval. Thus, during the time interval $[t_0, t_0 + \Delta t - \Delta t_{\text{fb}}]$ one would perform

a preparation phase that presolves as much as possible the optimization problem that one expects at time $(t_0 + \Delta t)$, corresponding to a predicted state $\bar{x}_1$.

At time $(t_0 + \Delta t - \Delta t_{\text{fb}})$, when the preparation phase is finished, one uses the most current state estimate to predict the state at time $(t_0 + \Delta t)$ more accurately than before. Denote this new prediction x_1. During the short time interval $[t_0 + \Delta t - \Delta t_{\text{fb}}, t_0 + \Delta t]$, one performs the computations of the feedback phase to obtain an approximate feedback u_1 that is based on x_1. In case of linear state feedback, one would, for example, precompute a vector v and a matrix K in the preparation phase that are solely based on $\bar{x}_1$, and then evaluate $u_1 := v + K(x_1 - \bar{x}_1)$ in the feedback phase. Alternatively, more complex computations—such as the solution of a condensed QP—can be performed in the feedback phase. The introduction of the feedback phase reduces the delay to unforeseen disturbances from Δt to Δt_{fb}. One has to accept, however, that the feedback is not the exact MPC feedback, but only an approximation. Some online algorithms, such as the real-time iteration discussed in Section 8.9.2, achieve the division into preparation and feedback phase by reordering the computational steps of a standard optimization algorithm, without creating any additional overhead per iteration.

Tangential predictors. A particularly powerful way to obtain a cheap approximation of the exact MPC feedback is based on the tangential predictors from Theorem 8.16. In case of strict complementarity at the solution $\bar{w}$ of an expected problem $\mathbb{P}_N(\bar{x}_1)$, one can show that for sufficiently small distance $|x_1 - \bar{x}_1|$, the solution of the parametric QP (8.33) corresponds to a linear map, i.e., $w^{\text{QP}}(x_1) = \bar{w} + A(x_1 - \bar{x}_1)$. The matrix A can be precomputed based on knowledge of the exact KKT matrix at the solution $\bar{w}$, but before the state x_1 is known.

Generalized tangential predictors are based on the (approximate) solution of the full QP (8.33), which is more expensive than a matrix vector multiplication, but is also applicable to the cases where strict complementarity does not hold or where the active set changes. The aim of all tangential predictors is to achieve a second-order approximation that satisfies $\left| w^{\text{QP}}(x_1) - w^*(x_1) \right| = O(|x_1 - \bar{x}_1|^2)$, which is only possible if the exact KKT matrix is known. If the exact KKT matrix is not used in the underlying optimization algorithm, e.g., in case of a Gauss-Newton Hessian approximation, one can alternatively compute an *approximate generalized tangential predictor* $\tilde{w}^{\text{QP}}(x_1) \approx w^{\text{QP}}(x_1)$, which only approximates the exact tangential predictor, but can be obtained without creating additional overhead compared to a standard

optimization iteration.

Warmstarting and shift. Another easy way to transfer solution information from one MPC problem to the next is to use an existing solution approximation as initial guess for the next MPC optimization problem, in a procedure called *warmstarting*. In its simplest variant, one can just use the existing solution guess without any modification. In the *shift initialization*, one first shifts the current solution guess to account for the advancement of time. The shift initialization can most easily be performed if an equidistant grid is used for control and state discretization, and is particularly advantageous for systems with time-varying dynamics or objectives, e.g., if a sequence of future disturbances is known, or one is tracking a time-varying trajectory.

Iterating while the problem changes. Extending the idea of warmstarting, some MPC algorithms do not separate between one optimization problem and the next, but always iterate while the problem changes. They only perform one iteration per sampling time, and they never try to iterate the optimization procedure to convergence for any fixed problem. Instead, they continue to iterate while the optimization problem changes. When implemented with care, this approach ensures that the algorithm always works with the most current information, and never loses precious time by working on outdated information.

8.9.2 Continuation Methods and Real-Time Iterations

Several of the ideas mentioned above are related to the idea of *continuation methods*, which we now discuss in more algorithmic detail. For this aim, we first regard a parameter-dependent root-finding problem of the form

$$R(x, z) = 0$$

with variable $z \in \mathbb{R}^{n_z}$, parameter $x \in \mathbb{R}^n$, and a smooth function $R : \mathbb{R}^n \times \mathbb{R}^{n_z} \to \mathbb{R}^{n_z}$. This root-finding problem could originate from an equality constrained MPC optimization problem with fixed barrier as it arises in a nonlinear IP method. The parameter dependence on x is due to the initial state value, which varies from one MPC optimization problem to the next. In case of infinite computational resources, one could just employ one of the Newton-type methods from Section 8.3.2 to converge to an accurate approximation of the exact solution $z^*(x)$ that satisfies $R(x, z^*(x)) = 0$. In practice, however, we only have limited computing power and finite time, and need to be satisfied with an approximation of $z^*(x)$.

Fortunately, it is a realistic assumption that we have an approximate solution of a related problem available, for the previous value of x. To clarify notation, we introduce a problem index k, such that the aim of the continuation method is to solve root-finding problems $R(x_k, z) = 0$ for a sequence $(x_k)_{k\in\mathbb{I}}$. For algorithmic simplicity, we assume that the parameter x enters the function R linearly. This assumption means that the Jacobian of R with respect to z does not depend on x but only on z, and can thus be written as $R_z(z)$. As a consequence, also the linearization of R depends only on the linearization point $\bar{z}$, i.e., it can be written as $R_L(x, z; \bar{z}) := R(x, \bar{z}) + R_z(\bar{z})(z - \bar{z})$.

A simple full-step Newton iteration for a fixed parameter x would iterate $z^+ = \bar{z} - R_z(\bar{z})^{-1}R(x, \bar{z})$. If we have a sequence of values x_k, we could decide to perform only one Newton iteration for each value x_k and then proceed to the next one. Given a solution guess z_k for the parameter value x_k, a continuation method would then generate the next solution guess by the iteration formula

$$z_{k+1} := z_k - R_z(z_k)^{-1}R(x_{k+1}, z_k)$$

Another viewpoint on this iteration is that z_{k+1} solves the linear equation system $R_L(x_{k+1}, z_{k+1}; z_k) = 0$. Interestingly, assuming only regularity of R_z, one can show that if z_k equals the exact solution $z^*(x_k)$ for the previous parameter x_k, the next iterate z_{k+1} is a first-order approximation, or tangential predictor, for the exact solution $z^*(x_{k+1})$. More generally, one can show that

$$\left| z_{k+1} - z^*(x_{k+1}) \right| = O\left(\left| \begin{bmatrix} z_k - z^*(x_k) \\ x_{k+1} - x_k \end{bmatrix} \right|^2 \right) \tag{8.59}$$

From this equation it follows that one can remain in the area of convergence of the Newton method if one starts close enough to an exact solution, $z_k \approx z^*(x_k)$, and if the parameter changes $(x_{k+1} - x_k)$ are small enough. Interestingly, it also implies quadratic convergence toward the solution in case the parameter values of x_k remain constant. Roughly speaking, the continuation method delivers tangential predictors in case the parameters x_k change a lot, and nearly quadratic convergence in case they change little.

The continuation method idea can be extended to Newton-type iterations of the form

$$z_{k+1} := z_k - M_k^{-1}R(x_{k+1}, z_k)$$

with approximations $M_k \approx R_z(z_k)$. In this case, only approximate tangential predictors are obtained.

Real-time iterations. To generalize the continuation idea to a sequence of inequality constrained optimization problems $\mathbb{P}_N(x_k)$ of the general form (8.29) with primal-dual solutions $z^*(x_k)$, one performs SQP type iterations of the form (8.41), but use in each iteration a new parameter value x_{k+1}. This idea directly leads to the *real-time iteration* (Diehl, Bock, Schlöder, Findeisen, Nagy, and Allgöwer, 2002) that determines the approximate solution $z_{k+1} = (w_{k+1}, \lambda_{k+1}, \mu_{k+1})$ of problem $\mathbb{P}_N(x_{k+1})$ from the primal-dual solution of the following QP

$$\begin{array}{ll} \underset{w \in \mathbb{R}^{n_w}}{\text{minimize}} & F_{\mathrm{L}}(w; w_k) + \frac{1}{2}(w - w_k)' B_k (w - w_k) \\ \text{subject to} & G_{\mathrm{L}}(x_{k+1}, w; w_k) = 0 \\ & H_{\mathrm{L}}(w; w_k) \leq 0 \end{array} \tag{8.60}$$

which we denote by $\mathbb{P}_N^{\mathrm{QP}}(x_{k+1}; w_k, B_k)$. If one uses the exact Hessian, $B_k = \nabla_w^2 \mathcal{L}(z_k)$, Theorem 8.16 ensures that the QP solution is a generalized tangential predictor of the exact solution if z_k was equal to an exact and strongly regular solution $z^*(x_k)$. Conversely, if the values of x_k would remain constant, the exact Hessian SQP method would have quadratic convergence.

More generally, the exact Hessian real-time iteration satisfies the quadratic approximation formula (8.59), despite the fact that active set changes lead to nondifferentiability in the solution map $z^*(\cdot)$. Loosely speaking, the SQP based real-time iteration is able to easily "jump" across this nondifferentiability, and its prediction and convergence properties are not directly affected by active set changes. If the Hessian is not the exact one, the real-time iteration method delivers only approximate tangential predictors, and shows linear instead of quadratic convergence. In practice, one often uses the Gauss-Newton Hessian in conjunction with a simultaneous approach to optimal control, but also sequential approaches were suggested in a similar framework (Li and Biegler, 1989). One can generalize the SQP based real-time iteration idea further by allowing the subproblems to be more general convex optimization problems, and by approximating also the constraint Jacobians, as proposed and investigated by Tran-Dinh et al. (2012).

Shift initialization and shrinking horizon problems. If the parametric optimization problems originate from an MPC optimal control problem with time-varying dynamics or objectives, it can be beneficial to

employ a shift strategy that shifts every approximate solution by one time step backward in time before the next QP problem is solved. For notational correctness, we need to denote the MPC problem by $\mathbb{P}_N(k, x_k)$ in this case, to reflect the direct dependence on the time index k. While most of the variable shift is canonical, the addition of an extra control, state, and multiplier at the end of the prediction horizon is not trivial, and different variants exist. Some are based on an auxiliary control law and a forward simulation, but also a plain repetition of the second-to-last interval, which needs no additional computations, is a possibility.

The guiding idea of the shift initialization is that a shifted optimal solution should ideally correspond to an optimal solution of the new MPC problem, if the new initial value x_{k+1} originates from the nominal system dynamics $x_{k+1} = f(x_k, u_k)$. But while recursive feasibility can be obtained easily by a shift, recursive optimality can usually not be obtained for receding horizon problems. Thus, a shift strategy perturbs the contraction of the real-time iterations and needs to be applied with care. In the special case of time-invariant MPC problems $\mathbb{P}_N(x_k)$ with a short horizon and tight terminal constraint or cost, a shift strategy is not beneficial.

On the other hand, in the case of finite-time (batch) processes that are addressed by MPC on shrinking horizons, recursive optimality can easily be achieved by shrinking a previously optimal solution. More concretely, if the initial horizon length was N, and at time k one would have the solution to the problem $\mathbb{P}_{N-k}(k, x_k)$ on the remaining time horizon, the optimal solution to the problem $\mathbb{P}_{N-k-1}(k+1, x_{k+1})$ is easily obtained by dropping the first component of the controls, states, and multipliers. Thus, the shrinking operation is canonical and should be used if real-time iterations—or other continuation methods—are applied to shrinking horizon MPC problems.

8.10 Discrete Actuators

Optimal control problems with discrete actuators fall into the class of mixed-integer optimal control problems, which are NP-hard and known to be difficult to solve. If one is lucky and the system model and constraints are linear and the cost is linear or convex quadratic, the discrete time optimal control problem turns out to be a mixed-integer linear program (MILP) or mixed-integer quadratic program (MIQP). For both classes there exist robust and reliable solvers that can be used

as a black-box for small to moderate problem dimensions. Another lucky case arises if the sequence of switches happens to be known in advance in a continuous time system, in which case *switching-time optimization* can be used to transform the problem into a standard nonlinear program (NLP). On the other hand, if we have a nonlinear system model with unknown switching sequence, we have to confront a significantly more difficult problem after discretization, namely a mixed-integer nonlinear program (MINLP). To address this MINLP one has basically three options:

- One can use piecewise system linearizations and mixed logical dynamics (MLD) to approximate the MINLP by a MILP or MIQP.
- One can try to solve the MINLP to global optimality using techniques from the field of global optimization.
- One can use a heuristic to find an approximate solution of the MINLP.

While the first two options can lead to viable solutions for relevant applications, they often lead to excessively large runtimes, so the MPC practitioner may need to resort to the last option. Fortunately, the optimal control structure of the problem allows us to use a powerful heuristic that exploits the fact that the state of a (continuous time) system is most strongly influenced by the time average of its controls rather than their pointwise values, as illustrated in Figure 8.7. This heuristic is based on a careful MINLP formulation, which is very similar to a standard nonlinear MPC problem, but with special structure. First, divide the input vector $u = (u_c, u_b) \in \mathbb{R}^{m_c+m_b}$ into continuous inputs, u_c, and binary integer inputs, u_b, such that the system is described by $x^+ = f(x, u_c, u_b)$. Second, and without loss of generality, we restrict ourselves to binary integers $u_b \in \{0,1\}^{m_b}$ inside a convex polyhedron $P \subset [0,1]^{m_b}$, and assume that u_b enters the system *linearly*.[3] The polyhedral constraint $u_b \in P$ allows us to exclude some combinations, e.g.,

[3] If necessary, this binary representation can be achieved by a technique called *outer convexification*, which is applicable to any system $x^+ = \tilde{f}(x, u_c, u_I)$ where the integer vector u_I has dimension m_I and can take finitely many (n_I) values $u_I \in \{u_{I,1}, \ldots, u_{I,n_I}\}$. We set $m_b := n_I$ and $f(x, u_c, u_b) := \sum_{i=1}^{m_b} u_{b,i} \tilde{f}(x, u_c, u_{I,i})$ and $P := \{u_b \in [0,1]^{m_b} \mid \sum_{j=1}^{m_b} u_{b,i} = 1\}$. Due to exponential growth of n_I in the number of original integer decisions m_I, this technique should be applied with care, e.g., only partially for separate subsystems, or avoided altogether if the original system is already linear in the integer controls.

if two machines cannot be operated simultaneously. The polyhedron P can and should be chosen such that its vertices equal the admissible binary values in each time step.

We might have additional combinatorial constraints that couple different time steps with each other. Typical examples are limits on the total number of switches, or dwell-time constraints, which bound the duration that a component of u_b can be active without interruption. We introduce the binary control trajectory $\mathbf{u}_b := (u_b(0), u_b(1), \ldots, u_b(N-1)) \in [0,1]^{m_b \times N}$ and denote the set of combinatorially feasible trajectories by $\mathbf{B} \subset \{0,1\}^{m_b \times N} \cap P^N$. The MINLP arising in MPC with discrete actuators can then be formulated as follows

$$
\begin{aligned}
\underset{\mathbf{x}, \mathbf{u}_c, \mathbf{u}_b}{\text{minimize}} \quad & \sum_{k=0}^{N-1} \ell(x(k), u_c(k), u_b(k)) + V_f(x(N)) \\
\text{subject to} \quad & x(0) = x_0 \\
& x(k+1) = f(x(k), u_c(k), u_b(k)), \quad && k = 0, \ldots, N-1 \\
& h(x(k), u_c(k), u_b(k)) \le 0, && k = 0, \ldots, N-1 \\
& h_f(x(N)) \le 0 \\
& u_b(k) \in P, && k = 0, \ldots, N-1 \\
& \mathbf{u}_b \in \mathbf{B}
\end{aligned}
\tag{8.61}
$$

Without the last constraint, $\mathbf{u}_b \in \mathbf{B}$, the above problem would be a standard NLP with optimal control structure. Likewise, a standard NLP arises if the binary controls $\mathbf{u}_b$ are fixed. These two observations directly lead to the following three-step algorithm that is a heuristic to find a good feasible solution of the MINLP (8.61).

1. Solve the *relaxed NLP* (8.61) *without combinatorial constraints*, $\mathbf{u}_b \in \mathbf{B}$, leading to a relaxed solution guess $(\mathbf{x}^*, \mathbf{u}_c^*, \mathbf{u}_b^*)$, possibly with $\mathbf{u}_b^* \notin \mathbf{B}$, with objective value V_N^*.

2. Find a binary trajectory $\mathbf{u}_b^{**} \in \mathbf{B}$ that approximates $\mathbf{u}_b^*$, e.g. by minimizing the distance between $\mathbf{u}_b^*$ and $\mathbf{u}_b^{**}$ in a suitable norm.

3. Fix the binary controls to $\mathbf{u}_b^{**}$ and solve the restricted NLP (8.61) in the variables $(\mathbf{x}, \mathbf{u}_c)$ only, with solution $(\mathbf{x}^{***}, \mathbf{u}_c^{***})$ and objective value V_N^{***}.

The result of the algorithm is the triple $(\mathbf{x}^{***}, \mathbf{u}_c^{***}, \mathbf{u}_b^{**})$ which is a

feasible, but typically not an optimal point of the MINLP (8.61).[4] Note that this feasible MINLP solution has an objective value V_N^{***} that is larger than the unknown exact MINLP solution V_N^0 which in turn is larger than the relaxed NLP objective V_N^* from Step 1 (if the global NLP solution was found): $V_N^* \leq V_N^0 \leq V_N^{***}$. Thus, the objective values from Steps 1 and 3 help us to bound the optimality loss incurred by using the above three-step heuristic.

The choice of the approximation in Step 2 affects both solution quality and computational complexity. One popular choice, that is taken in the *combinatorial integral approximation (CIA)* algorithm (Sager, Jung, and Kirches, 2011) is to minimize the distance in a specially scaled maximum norm that compares integrals, and is given by

$$\|\mathbf{u}_b\|_{\mathrm{CIA}} := \max_{j \leq m_b,\, n \leq N} \left| \sum_{k=0}^{n-1} u_{b,j}(k) \right|$$

Thus, in Step 2 of the CIA algorithm, one has to find $\mathbf{u}_b^{**} = \arg\min_{\mathbf{u}_b \in \mathbf{B}} \|\mathbf{u}_b - \mathbf{u}_b^*\|_{\mathrm{CIA}}$. This problem turns out to be a MILP (see Exercise 8.11) with a special structure that can be exploited in tailored algorithms, some of which are available in the open source tool `pycombina` (Bürger, Zeile, Hahn, Altmann-Dieses, Sager, and Diehl, 2020).

For the special case of continuous time problems with trivial combinatorial constraints, $\mathbf{B} = \{0,1\}^{m_b \times N} \cap P^N$, one can show under mild conditions that the difference between the objectives V_N^* and V_N^{***} in the three-step CIA algorithm shrinks linearly with the discretization step size $h = T/N$ if the length of the continuous time horizon T is fixed while N grows (Sager, Bock, and Diehl, 2012). A more general approximation result can be established in the presence of minimum dwell-time constraints (Zeile, Robuschi, and Sager, 2020).

Example 8.17: MPC with discrete actuator

We regard a simple problem of the form (8.61) for a nonlinear and unstable system with one state $x \in \mathbb{R}$ and one binary control $u_b \in \mathbb{R}$. The continuous time system is described by $\dot{x} = x^3 - u_b$ and transformed to a discrete time system $x^+ = f(x, u_b)$ by using one RK4 step with step length $h = 0.05$. The aim is to track a reference $x_{\mathrm{ref}} = 0.7$ starting from the initial value $x_0 = 0.9$ on a horizon of length $N = 30$, resulting

[4] An important feature in practice is the relaxation of inequality constraints, e.g., by using L1-penalties, in order to ensure feasible optimization problems in Steps 1 and 3.

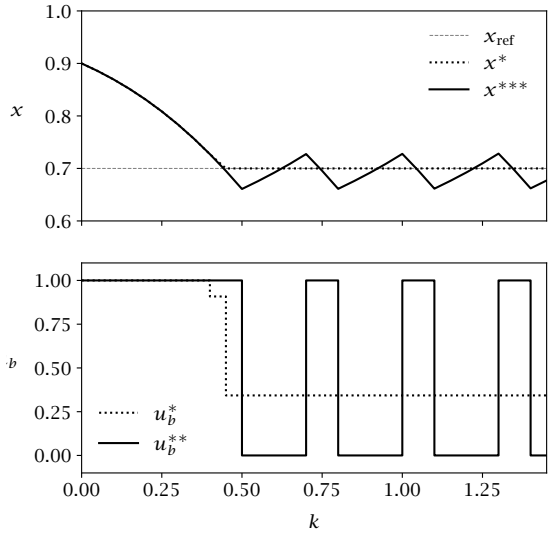


Figure 8.7: Relaxed and binary feasible solution for Example 8.17.

in the following MINLP

$$\begin{aligned}
\underset{\mathbf{x},\,\mathbf{u}_b}{\text{minimize}} \quad & \sum_{k=0}^{N} (x(k) - x_{\text{ref}})^2 \\
\text{subject to} \quad & x(0) = x_0 \\
& x(k+1) = f(x(k), u_b(k)), \quad k = 0, \ldots, N-1 \\
& u_b(k) \in [0,1], \qquad\qquad k = 0, \ldots, N-1 \\
& \mathbf{u}_b \in \mathbf{B}
\end{aligned} \tag{8.62}$$

The combinatorial constraint set $\mathbf{B}$ imposes a minimum dwell-time constraint on the uptime that requires that u_b remains active for at least two consecutive time steps, i.e., we have $\mathbf{B} = \{\mathbf{u}_b \in \{0,1\}^N \;\mid\; u_b(k) \geq u_b(k-1) - u_b(k-2), \quad k = 0, \ldots, N-1\}$. The required initial values $u_b(-1)$ and $u_b(-2)$ are both set to zero. We solve the problem using the described three-step procedure and the *combinatorial integral approximation* in Step 2. The relaxed solution $(\mathbf{x}^*, \mathbf{u}_b^*)$ after Step 1 as well as the solution $(\mathbf{x}^{***}, \mathbf{u}_b^{**})$ after Step 3 are shown in Figure 8.7. Note that due to the absence of continuous controls, Step 3 just amounts to a system simulation. The objective values are given by

$V_N^* = 0.166$ and $V_N^{***} = 0.1771$. The true optimal cost, which can for this simple example be found in a few seconds by an intelligent investigation of all $2^{30} \approx 10^9$ possibilities via branch-and-bound, is given by $V_N^0 = 0.176$. □

8.11 Notes

The description of numerical optimal control methods in this chapter is far from complete, and we have left out many details as well as many methods that are important in practice. We mention some related literature and software links that could complement this chapter.

General numerical optimal control methods are described in the textbooks by Bryson and Ho (1975); Betts (2001); Gerdts (2011); and in particular by Biegler (2010). The latter reference focuses on direct methods and also provides an in-depth treatment of nonlinear programming. The overview articles by Binder, Blank, Bock, Bulirsch, Dahmen, Diehl, Kronseder, Marquardt, Schlöder, and Stryk (2001); and Diehl, Ferreau, and Haverbeke (2009); as well a forthcoming textbook on numerical optimal control (Gros and Diehl, 2020) has a similar focus on online optimization for MPC as the current chapter.

General textbooks on numerical optimization are Bertsekas (1999); Nocedal and Wright (2006). Convex optimization is covered by Ben-Tal and Nemirovski (2001); Nesterov (2004); Boyd and Vandenberghe (2004). The last book is particularly accessible for an engineering audience, and its PDF is freely available on the home page of its first author. Newton's method for nonlinear equations and its many variants are described and analyzed in a textbook by Deuflhard (2011). An up-to-date overview of optimization tools can be found at `plato.asu.edu/guide.html`, many optimization solvers are available as source code at `www.coin-or.org`, and many optimization solvers can be accessed online via `neos-server.org`.

While the direct single-shooting method often is implemented by coupling an efficient numerical integration solver with a general nonlinear program (NLP) solver such as SNOPT (Gill, Murray, and Saunders, 2005), the direct multiple-shooting and direct collocation methods need to be implemented by using NLP solvers that fully exploit the sparsity structure, such as IPOPT[5] (Wächter and Biegler, 2006) There exist many custom implementations of the direct multiple-shooting

[5] This code is available to the public under a permissive open-source license.

method with their own structure-exploiting NLP solvers, such as, for example, HQP[5] (Franke, 1998); MUSCOD-II (Leineweber, Bauer, Schäfer, Bock, and Schlöder, 2003); ACADO[5](Houska, Ferreau, and Diehl, 2011); and FORCES-NLP (Zanelli, Domahidi, Jerez, and Morari, 2017).

Structure-exploiting QP solvers that can be used standalone for linear MPC or as subproblem solvers within SQP methods are, for example, the dense code qpOASES[5] (Ferreau, Kirches, Potschka, Bock, and Diehl, 2014), which is usually combined with condensing, or the sparse codes FORCES (Domahidi, 2013); qpDUNES[5] (Frasch, Sager, and Diehl, 2015); and HPMPC[5] (Frison, 2015). The latter is based on a CPU specific matrix storage format that by itself leads to speedups in the range of one order of magnitude, and which was made available to the public in the BLASFEO[5] library at `github.com/giaf/blasfeo`.

In Section 8.2 on numerical simulation methods, we have exclusively treated Runge-Kutta methods because they play an important role within a large variety of numerical optimal control algorithms, such as shooting, collocation, or pseudospectral methods. Another popular and important family of integration methods, however, are the *linear multistep methods*; in particular, the implicit backward differentiation formula (BDF) methods are widely used for simulation and optimization of large stiff differential algebraic equations (DAEs). For an in-depth treatment of general numerical simulation methods for ordinary differential equations (ODEs) and DAEs, we recommend the textbooks by Hairer, Nørsett, and Wanner (1993, 1996); as well as Brenan, Campbell, and Petzold (1996); Ascher and Petzold (1998).

For derivative generation of numerical simulation methods, we refer to the research articles Bauer, Bock, Körkel, and Schlöder (2000); Petzold, Li, Cao, and Serban (2006); Kristensen, Jørgensen, Thomsen, and Jørgensen (2004); Quirynen, Gros, Houska, and Diehl (2017a); Quirynen, Houska, and Diehl (2017b); and the Ph.D. theses by Albersmeyer (2010); Quirynen (2017). A collection of numerical ODE and DAE solvers with efficient derivative computations are implemented in the SUNDIALS[5] suite (Hindmarsh, Brown, Grant, Lee, Serban, Shumaker, and Woodward, 2005).

Regarding Section 8.4 on derivatives, we refer to a textbook on algorithmic differentiation (AD) by Griewank and Walther (2008), and an overview of AD tools at `www.autodiff.org`. The AD framework CasADi[5] can in its latest form be found at `casadi.org`, and is described in the article Andersson, Akesson, and Diehl (2012); and the Ph.D. theses by Andersson (2013); Gillis (2015).

8.12 Exercises

Some of the exercises in this chapter were developed for courses on numerical optimal control at the University of Freiburg, Germany. The authors gratefully acknowledge Joel Andersson, Joris Gillis, Sébastien Gros, Dimitris Kouzoupis, Jesus Lago Garcia, Rien Quirynen, Andrea Zanelli, and Mario Zanon for contributions to the formulation of these exercises; as well as Michael Risbeck, Nishith Patel, Douglas Allan, and Travis Arnold for testing and writing solution scripts.

Exercise 8.1: Newton's method for root finding

In this exercise, we experiment with a full-step Newton method and explore the dependence of the iterates on the problem formulation and the initial guess.

(a) Write a computer program that performs Newton iterations in $\mathbb{R}^n$ that takes as inputs a function $F(z)$, its Jacobian $J(z)$, and a starting point $z_{[0]} \in \mathbb{R}^n$. It shall output the first 20 full-step Newton iterations. Test your program with $R(z) = z^{32} - 2$ starting first at $z_{[0]} = 1$ and then at different positive initial guesses. How many iterations do you typically need in order to obtain a solution that is exact up to machine precision?

(b) An equivalent problem to $z^{32} - 2 = 0$ can be obtained by *lifting* it to a higher dimensional space (Albersmeyer and Diehl, 2010), as follows

$$R(z) = \begin{bmatrix} z_2 - z_1^2 \\ z_3 - z_2^2 \\ z_4 - z_3^2 \\ z_5 - z_4^2 \\ 2 - z_5^2 \end{bmatrix}$$

Use your algorithm to implement Newton's method for this lifted problem and start it at $z_{[0]} = [1\ 1\ 1\ 1\ 1]'$ (note that we use square brackets in the index to denote the Newton iteration). Compare the convergence of the iterates for the lifted problem with those of the equivalent unlifted problem from the previous task, initialized at one.

(c) Consider now the root-finding problem $R(z) = 0$ with $R : \mathbb{R} \to \mathbb{R}, R(z) := \tanh(z) - \frac{1}{2}$. Convergence of Newton's method is sensitive to the chosen initial value z_0. Plot $R(z)$ and observe the nonlinearity. Implement Newton's method with full steps for it, and test if it converges or not for different initial values $z_{[0]}$.

(d) Regard the problem of finding a solution to the nonlinear equation system $2x = e^{y/4}$ and $16x^4 + 81y^4 = 4$ in the two variables $x, y \in \mathbb{R}$. Solve it with your implementation of Newton's method using different initial guesses. Does it always converge, and, if it converges, does it always converge to the same solution?

Exercise 8.2: Newton-type methods for a boundary-value problem

Regard the scalar discrete time system

$$x(k+1) = \frac{1}{10}\left(11x(k) + x(k)^2 + u\right), \quad k = 0, \ldots, N-1$$

with boundary conditions

$$x(0) = x_0 \qquad x(N) = 0$$

We fix the initial condition to $x_0 = 0.1$ and the horizon length to $N = 30$. The aim is to find the control value $u \in \mathbb{R}$—which is kept constant over the whole horizon—in order to steer the system to the origin at the final time, i.e., to satisfy the constraint $x(N) = 0$. This is a two-point boundary-value problem (BVP). In this exercise, we formulate this BVP as a root-finding problem in two different ways: first, with the sequential approach, i.e., with only the single control as decision variable; and second, with the simultaneous approach, i.e., with all 31 states plus the control as decision variables.

(a) Formulate and solve the problem with the sequential approach, and solve it with an exact Newton's method initialized at $u = 0$. Plot the state trajectories in each iteration. Also plot the residual values $x(N)$ and the variable u as a function of the Newton iteration index.

(b) Now formulate and solve the problem with the simultaneous approach, and solve it with an exact Newton's method initialized at $u = 0$ and the corresponding state sequence that is obtained by forward simulation started at x_0. Plot the state trajectories in each iteration.

Plot again the residual values $x(N)$ and the variable u as a function of the Newton iteration index, and compare with the results that you have obtained with the sequential approach. Do you observe differences in the convergence speed?

(c) One feature of the simultaneous approach is that its states can be initialized with any trajectory, even an infeasible one. Initialize the simultaneous approach with the all-zero trajectory, and again observe the trajectories and the convergence speed.

(d) Now solve both formulations with a Newton-type method that uses a constant Jacobian. For both approaches, the constant Jacobian corresponds to the exact Jacobian at the solution of the same problem for $x_0 = 0$, where all states and the control are zero. Start with implementing the sequential approach, and initialize the iterates at $u = 0$. Again, plot the residual values $x(N)$ and the variable u as a function of iteration index.

(e) Now implement the simultaneous approach with a fixed Jacobian approximation. Again, the Jacobian approximation corresponds to the exact Jacobian at the solution of the neighboring problem with $x_0 = 0$, i.e., the all zero trajectory. Start the Newton-type iterations with all states and the control set to zero, and plot the residual values $x(N)$ and the variable u as a function of iteration index. Discuss the differences of convergence speed with the sequential approach and with the exact Newton methods from before.

(f) The performance of the sequential approach can be improved if one introduces the initial state $x(0)$ as a second decision variable. This allows more freedom for the initialization, and one can automatically profit from tangential solution

predictors. Adapt your exact Newton method, initialize the problem in the all-zero solution and again observe the results.

(g) If u^* is the exact solution that is found at the end of the iterations, plot the logarithm of $|u - u^*|$ versus the iteration number for all six numerical experiments (a)–(f), and compare.

(h) The linear system that needs to be solved in each iteration of the simultaneous approach is large and sparse. We can use condensing in order to reduce the linear system to size one. Implement a condensing-based linear system solver that only uses multiplications and additions, and one division. Compare the iterations with the full-space linear algebra approach, and discuss the differences in the iterations, if any.

Exercise 8.3: Convex functions

Determine and explain whether the following functions are convex or not on their respective domains.

(a) $f(x) = c'x + x'A'Ax$ on $\mathbb{R}^n$

(b) $f(x) = -c'x - x'A'Ax$ on $\mathbb{R}^n$

(c) $f(x) = \log(c'x) + \exp(b'x)$ on $\{x \in \mathbb{R}^n \mid c'x > 0\}$

(d) $f(x) = -\log(c'x) - \exp(b'x)$ on $\{x \in \mathbb{R}^n \mid c'x > 0\}$

(e) $f(x) = 1/(x_1x_2)$ on $\mathbb{R}^2_{++}$

(f) $f(x) = x_1/x_2$ on $\mathbb{R}^2_{++}$

Exercise 8.4: Convex sets

Determine and explain whether the following sets are convex or not.

(a) A ball, i.e., a set of the form

$$\Omega = \{x \mid |x - x_c| \leq r\}$$

(b) A sublevel set of a convex function $f : \mathbb{R}^n \to \mathbb{R}$ for a constant $c \in \mathbb{R}$

$$\Omega = \{x \in \mathbb{R}^n \mid f(x) \leq c\}$$

(c) A superlevel set of a convex function $f : \mathbb{R}^n \to \mathbb{R}$ for a constant $c \in \mathbb{R}$

$$\Omega = \{x \in \mathbb{R}^n \mid f(x) \geq c\}$$

(d) The set

$$\Omega = \{x \in \mathbb{R}^n \mid x'B'Bx \leq b'x\}$$

(e) The set

$$\Omega = \{x \in \mathbb{R}^n \mid x'B'Bx \geq b'x\}$$

(f) A cone, i.e., a set of the form

$$\Omega = \{(x, \alpha) \in \mathbb{R}^n \times \mathbb{R} \mid |x| \leq \alpha\}$$

(g) A wedge, i.e., a set of the form

$$\{x \in \mathbb{R}^n \mid a_1' x \le b_1, a_2' x \le b_2\}$$

(h) A polyhedron

$$\{x \in \mathbb{R}^n \mid Ax \le b\}$$

(i) The set of points closer to one set than another

$$\Omega = \{x \in \mathbb{R}^n \mid \operatorname{dist}(x, S) \le \operatorname{dist}(x, \mathcal{T})\}$$

where $\operatorname{dist}(x, S) := \inf\{|x - z|_2 \mid z \in S\}$.

Exercise 8.5: Finite differences: theory of optimal perturbation size

Assume we have a twice continuously differentiable function $f : \mathbb{R} \to \mathbb{R}$ and we want to evaluate its derivative $f'(x_0)$ at x_0 with finite differences. Further assume that for all $x \in [x_0 - \delta, x_0 + \delta]$ holds that

$$|f(x)| \le f_{\max} \qquad |f''(x)| \le f''_{\max} \qquad |f'''(x)| \le f'''_{\max}$$

We assume $\delta > t$ for any perturbation size t in the following finite difference approximations. Due to finite machine precision ϵ_{mach} that leads to truncation errors, the computed function $\tilde{f}(x) = f(x)(1 + \epsilon(x))$ is perturbed by noise $\epsilon(x)$ that satisfies the bound

$$|\epsilon(x)| \le \epsilon_{\text{mach}}$$

(a) Compute a bound on the error of the forward difference approximation

$$\tilde{f}'_{\text{fd},t}(x_0) := \frac{\tilde{f}(x_0 + t) - \tilde{f}(x_0)}{t}$$

namely, a function $\psi(t; f_{\max}, f''_{\max}, \epsilon_{\text{mach}})$ that satisfies

$$\left|\tilde{f}'_{\text{fd},t}(x_0) - f'(x_0)\right| \le \psi(t; f_{\max}, f''_{\max}, \epsilon_{\text{mach}})$$

(b) Which value t^* minimizes this bound and which value ψ^* has the bound at t^*?

(c) Perform a similar error analysis for the central difference quotient

$$\tilde{f}'_{\text{cd},t}(x_0) := \frac{\tilde{f}(x_0 + t) - \tilde{f}(x_0 - t)}{2t}$$

that is, compute a bound

$$\left|\tilde{f}'_{\text{fd},t}(x_0) - f'(x_0)\right| \le \psi_{\text{cd}}(t; f_{\max}, f''_{\max}, f'''_{\max}, \epsilon_{\text{mach}})$$

(d) For central differences, what is the optimal perturbation size t^*_{cd} and what is the size ψ^*_{cd} of the resulting bound on the error?

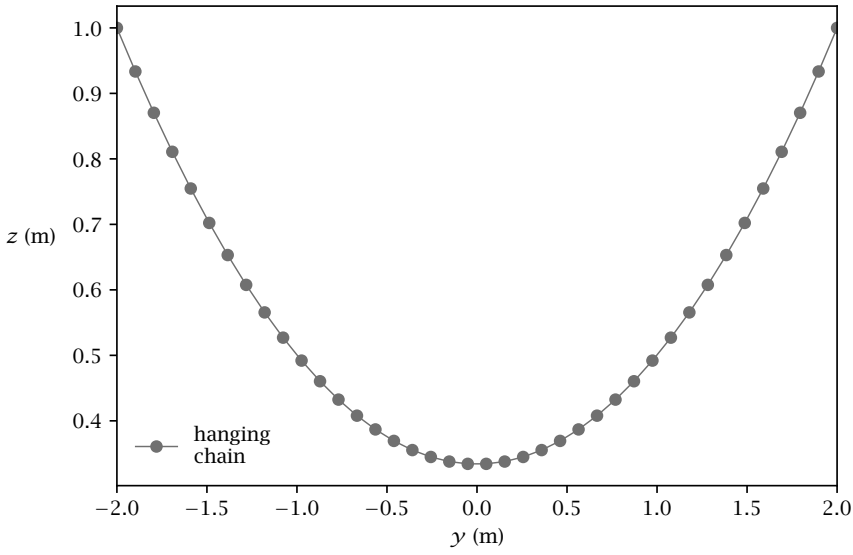


Figure 8.8: A hanging chain at rest. See Exercise 8.6(b).

Exercise 8.6: Finding the equilibrium point of a hanging chain using CasADi

Consider an elastic chain attached to two supports and hanging in-between. Let us discretize it with N mass points connected by $N-1$ springs. Each mass i has position (y_i, z_i), $i = 1, \ldots, N$.

Our task is to minimize the total potential energy, which is made up by potential energy in each string and potential energy of each mass according to

$$J(y_1, z_1, \ldots, y_n, z_n) = \underbrace{\frac{1}{2}\sum_{i=1}^{N-1} D_i\left((y_i - y_{i+1})^2 + (z_i - z_{i+1})^2\right)}_{\text{spring potential energy}} + \underbrace{g_0 \sum_{i=1}^{N} m_i z_i}_{\text{gravitational potential energy}} \tag{8.63}$$

subject to constraints modeling the ground.

(a) CasADi is an open-source software tool for solving optimization problems in general and optimal control problems (OCPs) in particular. In its most typical usage, it is used to formulate and solve constrained optimization problems of the form

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & f(x) \\ \text{subject to} \quad & \underline{x} \le x \le \overline{x} \\ & \underline{g} \le g(x) \le \overline{g} \end{aligned} \tag{8.64}$$

where $x \in \mathbb{R}^{n_x}$ is the decision variable, $f : \mathbb{R}^{n_x} \to \mathbb{R}$ is the objective function, and $g : \mathbb{R}^{n_x} \to \mathbb{R}^{n_g}$ is the constraint function. For equality constraints, the

upper and lower bounds are equal.

If you have not already done so, go to `casadi.org` and locate the installation instructions. On most platforms, installing CasADi amounts to downloading a binary installation and placing it somewhere in your path. Version 3.3 of CasADi on Octave/MATLAB was used in this edition, so make sure that you are not using a version older than this and keep an eye on the text website for incompatibilities with future versions of CasADi. Locate the CasADi user guide and, with an Octave or MATLAB interpreter in front of you, read Chapters 1 through 4. These chapters give you an overview of the scope and syntax of CasADi.

(b) We assume that f is a convex quadratic function and g is a linear function. In this case we refer to (8.64) as a convex quadratic program (QP). To solve a QP with CasADi, we construct symbolic expressions for x, f, and g, and use this to construct a solver object that can be called one or more times with different values for $\overline{x}$, $\underline{x}$, $\overline{g}$, and $\underline{g}$. An initial guess for x can also be provided, but this is less important for convex QPs, where the solution is unique.

Figure 8.8 shows the solution of the unconstrained problem using the open-source QP solver qpOASES with $N = 40$, $m_i = 40/N$ kg, $D_i = 70N$ N/m, and $g_0 = 9.81$ m/s^2. The first and last mass points are fixed to $(-2, 1)$ and $(2, 1)$, respectively. Go through the code for the figure and make sure you understand the steps.

(c) Now introduce ground constraints: $z_i \geq 0.5$ and $z_i \geq 0.5 + 0.1\, y_i$, for $i = 2, \cdots, N-2$. Resolve the QP and compare with the unconstrained solution.

(d) We now want to formulate and solve a nonlinear program (NLP). Since an NLP is a generalization of a QP, we can solve the above problem with an NLP solver. This can be done by simply changing `casadi.qpsol` in the script to `casadi.nlpsol` and the solver plugin `'qpoases'` with `'ipopt'`, corresponding to the open-source NLP solver IPOPT. Are the solutions of the NLP and QP solver the same?

(e) Now, replace the linear equalities by nonlinear ones that are given by $z_i \geq 0.5 + 0.1\, y_i^2$ for $i = 2, \cdots, N-2$. Modify the expressions from before to formulate and solve the NLP, and visualize the solution. Is the NLP convex?

(f) Now, by modifications of the expressions from before, formulate and solve an NLP where the inequality constraints are replaced by $z_i \geq 0.8 + 0.05\, y_i - 0.1\, y_i^2$ for $i = 2, \cdots, N-2$. Is this NLP convex?

Exercise 8.7: Direct single shooting versus direct multiple shooting

Consider the following OCP, corresponding to driving a Van der Pol oscillator to the origin, on a time horizon with length $T = 10$

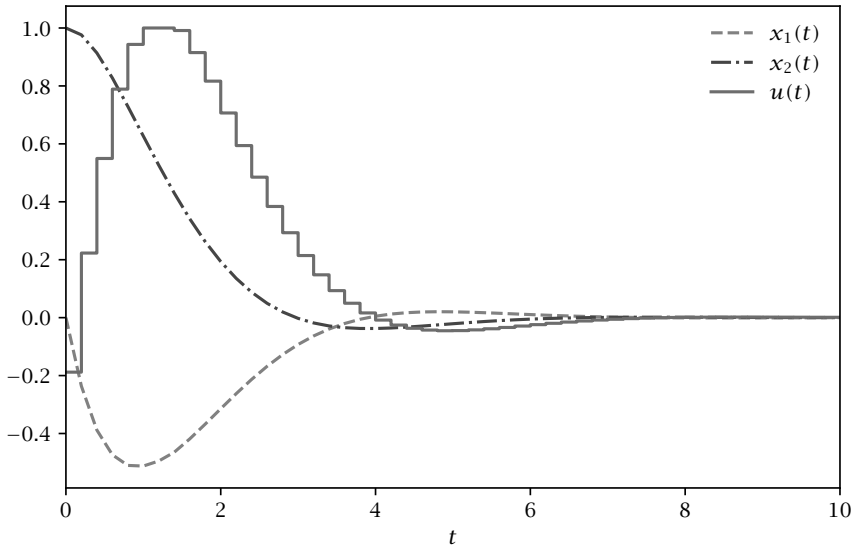


Figure 8.9: Direct single shooting solution for (8.65) without path constraints.

$$
\begin{aligned}
\underset{x(\cdot),u(\cdot)}{\text{minimize}} \quad & \int_0^T (x_1(t)^2 + x_2(t)^2 + u(t)^2)\, dt \\
\text{subject to} \quad & \dot{x}_1(t) = (1 - x_2(t)^2)\, x_1(t) - x_2(t) + u(t) \\
& \dot{x}_2(t) = x_1(t) \\
& -1 \leq u(t) \leq 1, \quad t \in [0, T] \\
& x_1(0) = 0, \quad x_1(T) = 0 \\
& x_2(0) = 1, \quad x_2(T) = 0 \\
& -0.25 \leq x_1(t), \quad t \in [0, T]
\end{aligned}
\tag{8.65}
$$

We will solve this problem using direct single shooting and direct multiple shooting using IPOPT as the NLP solver.

(a) Figure 8.9 shows the solution to the above problem using a direct single shooting approach, without enforcing the constraint $-0.25 \leq x_1(t)$. Go through the code for the figure step by step. The code begins with a modeling step, where symbolic expressions for the continuous-time model are constructed. Thereafter, the problem is transformed into discrete time by formulating an object that integrates the system forward in time using a single step of the RK4 method. This function also calculates the contribution to the objective function for the same interval using the same integrator method. In the next part of the code, a

symbolic representation of the NLP is constructed, starting with empty lists of variables and constraints. This symbolic representation of the NLP is used to define an NLP solver object using IPOPT as the underlying solver. Finally, the solver object is evaluated to obtain the optimal solution.

(b) Modify the code so that the path constraint on $x_1(t)$ is being respected. You only need to enforce this constraint at the end of each control interval. This should result in additional components to the NLP constraint function $G(w)$, which will now have upper and lower bounds similar to the decision variable w. Resolve the modified problem and compare the solution.

(c) Modify the code to implement the direct multiple-shooting method instead of direct single shooting. This means introducing decision variables corresponding to not only the control trajectory, but also the state trajectory. The added decision variables will be matched with an equal number of new equality constraints, enforcing that the NLP solution corresponds to a continuous state trajectory. The initial and terminal conditions on the state can be formulated as upper and lower bounds on the corresponding elements of w. Use $x(t) = 0$ as the initial guess for the state trajectory.

(d) Compare the IPOPT output for both transcriptions. How did the change from direct single shooting to direct multiple shooting influence

- The number of iterations?
- The number of nonzeros in the Jacobian of the constraints?
- The number of nonzeros in the Hessian of the Lagrangian?

(e) Generalize the RK4 method so that it takes $M = 4$ steps instead of just one. This corresponds to a higher-accuracy integration of the model dynamics. Approximately how much smaller discretization error can we expect from this change?

(f) Replace the RK4 integrator with the variable-order, variable-step size code CVODES from the SUNDIALS suite, available as the `'cvodes'` plugin for `casadi.integrator`. Use 10^{-8} for the relative and absolute tolerances. Consult CasADi's user guide for syntax. What are the advantages and disadvantages of using this integrator over the fixed-step RK4 method used until now?

Exercise 8.8: Direct collocation

Collocation, in its most basic sense, refers to a way of solving initial-value problems by approximating the state trajectory with piecewise polynomials. For each step of the integrator, corresponding to an interval of time, we choose the coefficients of these polynomials to ensure that the ODE becomes exactly satisfied at a given set of time points. In the following, we choose the *Gauss-Legendre* collocation integrator of sixth order, which has $d = 3$ collocation points. Together with the point 0 at the start of the interval $[0, 1]$, we have four time points to define the collocation polynomial

$$\tau_0 = 0 \qquad \tau_1 = 1/2 - \sqrt{15}/10 \qquad \tau_2 = 1/2 \qquad \tau_3 = 1/2 + \sqrt{15}/10$$

Using these time points, we define the corresponding Lagrange polynomials

$$L_j(\tau) = \prod_{r=0, r\neq j}^{d} \frac{\tau - \tau_r}{\tau_j - \tau_r}$$

Introducing a uniform time grid $t_k = k\,h$, $k = 0, \ldots, N$, with the corresponding state values $x_k := x(t_k)$, we can approximate the state trajectory inside each interval $[t_k, t_{k+1}]$ as a linear combination of these basis functions

$$\tilde{x}_k(t) = \sum_{r=0}^{d} L_r\left(\frac{t - t_k}{h}\right) x_{k,r}$$

By differentiation, we get an approximation of the time derivative at each collocation point for $j = 1, \ldots, 3$

$$\dot{\tilde{x}}_k(t_{k,j}) = \frac{1}{h} \sum_{r=0}^{d} \dot{L}_r(\tau_j)\, x_{k,r} := \frac{1}{h} \sum_{r=0}^{d} C_{r,j}\, x_{k,r}$$

We also can get an expression for the state at the end of the interval

$$\tilde{x}_{k+1,0} = \sum_{r=0}^{d} L_r(1)\, x_{k,r} := \sum_{r=0}^{d} D_r\, x_{k,r}$$

Finally, we also can integrate our approximation over the interval, giving a formula for *quadratures*

$$\int_{t_k}^{t_{k+1}} \tilde{x}_k(t)\, dt = h \sum_{r=0}^{d} \int_0^1 L_r(\tau)\, d\tau\, x_{k,r} := h \sum_{r=1}^{d} b_r\, x_{k,r}$$

(a) Figure 8.10 shows an open-loop simulation for the ODE in (8.65) using Gauss-Legendre collocation of order 2, 4, and 6. A constant control $u(t) = 0.5$ was applied and the initial conditions were given by $x(0) = [0, 1]'$. The figure on the left shows the first state $x_1(t)$ for the three methods as well as a high-accuracy solution obtained from CVODES, which uses a backward differentiation formula (BDF) method. In the figure on the right we see the discretization error, as compared with CVODES. Go through the code for the figure and make sure you understand it. Using this script as a template, replace the integrator in the direct multiple-shooting method from Exercise 8.7 with this collocation integrator. Make sure that you obtain the same solution. The structure of the NLP should remain unchanged—you are still implementing the direct multiple-shooting approach, only with a different integrator method.

(b) In the NLP transcription step, replace the embedded function call with additional degrees of freedom corresponding to the state at all the collocation points. Enforce the collocation equations at the NLP level instead of the integrator level.

Enforce upper and lower bounds on the state at all collocation points. Compare the solution time and number of nonzeros in the Jacobian and Hessian matrices with the direct multiple-shooting method.

Exercise 8.9: Gauss-Newton SQP iterations for optimal control

Consider a nonlinear pendulum defined by

$$\dot{x}(t) = f(x(t), u(t)) = \begin{bmatrix} v(t) \\ -C \sin(p(t)/C) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t)$$

with state $x = [p, v]'$ and $C := 180/\pi/10$, to solve an OCP using a direct multiple-shooting method and a self-written sequential quadratic programming (SQP) solver with a Gauss-Newton Hessian.

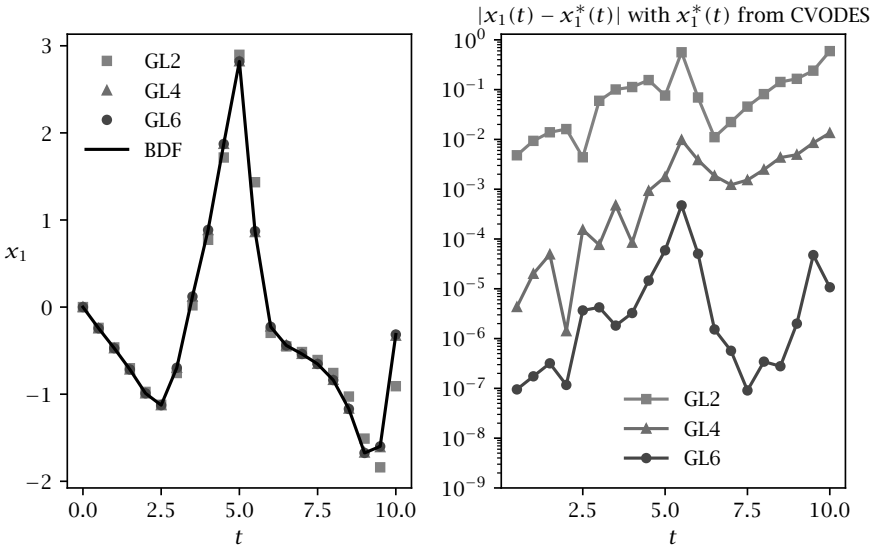


Figure 8.10: Open-loop simulation for (8.65) using collocation.

(a) Starting with the pendulum at $\bar{x}_0 = [10\ 0]'$, we aim to minimize the required controls to bring the pendulum to $x_N = [0\ 0]'$ in a time horizon $T = 10\,\text{s}$. Regarding bounds on p, v, and u, namely $p_{\max} = 10$, $v_{\max} = 10$, and $u_{\max} = 3$, the required controls can be obtained as the solution of the following OCP

$$\begin{aligned}
\underset{\substack{x_0,u_0,x_1,\ldots,\\ u_{N-1},x_N}}{\text{minimize}} \quad & \frac{1}{2}\sum_{k=0}^{N-1} |u_k|_2^2 \\
\text{subject to} \quad & \bar{x}_0 - x_0 = 0 \\
& \Phi(x_k, u_k) - x_{k+1} = 0, \quad k = 0,\ldots,N-1 \\
& x_N = 0 \\
& -x_{\max} \le x_k \le x_{\max}, \quad k = 0,\ldots,N-1 \\
& -u_{\max} \le u_k \le u_{\max}, \quad k = 0,\ldots,N-1
\end{aligned}$$

Formulate the discrete dynamics $x_{k+1} = \Phi(x_k, u_k)$ using a RK4 integrator with a time step $\Delta t = 0.2\,\text{s}$. Encapsulate the code in a single CasADi function of the form of a CasADi function object as in Exercise 8.7. Simulate the system forward in time and plot the result.

(b) Using $w = (x_0, u_0, \ldots, u_{N-1}, x_N)$ as the NLP decision variable, we can formulate the equality constraint function $G(w)$, the least squares function $M(w)$, and the

bounds vector $w_{\max}$ so that the above OCP can be written

$$\begin{aligned} \underset{w}{\text{minimize}} \quad & \frac{1}{2}|M(w)|_2^2 \\ \text{subject to} \quad & G(w) = 0 \\ & -w_{\max} \le w \le w_{\max} \end{aligned}$$

The SQP method with Gauss-Newton Hessian solves a linearized version of this problem in each iteration. More specifically, if the current iterate is $\bar{w}$, the next iterate is given by $\bar{w} + \Delta w$, where Δw is the solution of the following QP

$$\begin{aligned} \underset{\Delta w}{\text{minimize}} \quad & \frac{1}{2}\Delta w' J_M(\bar{w})' J_M(\bar{w})\Delta w + M(\bar{w})' J_M(\bar{w})\Delta w \\ \text{subject to} \quad & G(\bar{w}) + J_G(\bar{w})\Delta w = 0 \\ & -w_{\max} - \bar{w} \le \Delta w \le w_{\max} - \bar{w} \end{aligned} \tag{8.66}$$

In order to implement the Gauss-Newton method, we need the Jacobians $J_G(w) = \frac{\partial G}{\partial w}(w)$ and $J_M(w) = \frac{\partial M}{\partial w}(w)$, both of which can be efficiently obtained using CasADi's `jacobian` command. In this case the Gauss-Newton Hessian $H = J_M(\bar{w})' J_M(\bar{w})$ can readily be obtained by pen and paper. Define what H_x and H_u need to be in the Hessian

$$H = \begin{bmatrix} H_x & & & \\ & H_u & & \\ & & \ddots & \\ & & & H_x \end{bmatrix} \qquad H_x = \begin{bmatrix} & \end{bmatrix} \qquad H_u = \begin{bmatrix} & \end{bmatrix}$$

(c) Figure 8.11 shows the control trajectory after 0, 1, 2, and 6 iterations of the Gauss-Newton method applied to a direct multiple-shooting transcription of (8.65). Go through the code for the figure step by step. You should recognize much of the code from the solution to Exercise 8.7. The code represents a simplified, yet efficient way of using CasADi to solve OCPs.

Modify the code to solve the pendulum problem. Note that the sparsity patterns of the linear and quadratic terms of the QP are printed out at the beginning of the execution. $J_G(w)$ is a block sparse matrix with blocks being either identity matrices I or partial derivatives $A_k = \frac{\partial \Phi}{\partial x}(x_k, u_k)$ and $B_k = \frac{\partial \Phi}{\partial u}(x_k, u_k)$.

Initialize the Gauss-Newton procedure at $w = 0$, and stop the iterations when $|w_{k+1} - w_k|$ gets smaller than 10^{-4}. Plot the iterates as well as the vector G during the iterations. How many iterations do you need?

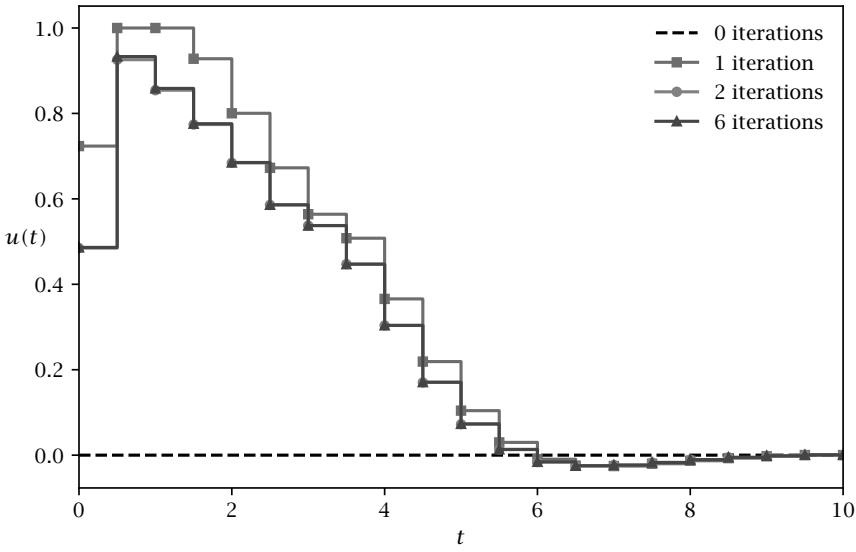


Figure 8.11: Gauss-Newton iterations for a direct multiple-shooting transcription of (8.65); $u(t)$ after 0, 1, 2, and 6 Gauss-Newton iterations.

Exercise 8.10: Real-time iterations and nonlinear MPC

We return to the OCP from Exercise 8.9

$$
\begin{aligned}
\underset{\substack{x_0,u_0,x_1,\ldots,\\ u_{N-1},x_N}}{\text{minimize}} \quad & \frac{1}{2}\sum_{k=0}^{N-1} |u_k|_2^2 \\
\text{subject to} \quad & \bar{x}_0 - x_0 = 0 \\
& \Phi(x_k, u_k) - x_{k+1} = 0, \qquad k = 0, \ldots, N-1 \\
& x_N = 0 \\
& -x_{\max} \le x_k \le x_{\max}, \quad k = 0, \ldots, N-1 \\
& -u_{\max} \le u_k \le u_{\max}, \quad k = 0, \ldots, N-1
\end{aligned}
$$

In this problem, we regard $\bar{x}_0$ as a parameter and modify the simultaneous Gauss-Newton algorithm from Exercise 8.9. In particular, we modify this algorithm to perform real-time iterations for different values of $\bar{x}_0$, so that we can use the algorithm to perform closed-loop nonlinear MPC simulations for stabilization of the nonlinear pendulum.

(a) Modify the function `sqpstep` from the solution of Exercise 8.9 so that it accepts the parameter $\bar{x}_0$. You would need to update the upper and lower bounds on w accordingly. Test it and make sure that it works.

(b) In order to visualize the generalized tangential predictor, call the `sqpstep` method with different values for $\bar{x}_0$ while resetting the variable vector $\bar{w}$ to its initial value (zero) between each call. Use a linear interpolation for $\bar{x}_0$ with 100 points between zero and the value $(10, 0)'$, i.e., set $\bar{x}_0 = \lambda[10\ 0]'$ for $\lambda \in [0, 1]$. Plot the first control u_0 as a function of λ and keep your plot.

(c) To compute the exact solution manifold with relatively high accuracy, perform now the same procedure for the same 100 increasing values of λ, but this time perform for each value of λ multiple Gauss-Newton iterations, i.e., replace each call to `sqpstep` with, e.g., 10 calls without changing $\bar{x}_0$. Plot the obtained values for u_0 and compare with the tangential predictor from the previous task by plotting them in the same plot.

(d) In order to see how the real-time iterations work in a more realistic setting, let the values of λ jump faster from 0 to 1, e.g., by doing only 10 steps, and plot the result again into the same plot.

(e) Modify the previous algorithm as follows: after each change of λ by 0.1, keep it constant for nine iterations, before you do the next jump. This results in about 100 consecutive real-time iterations. Interpret what you see.

(f) Now we do the first *closed-loop simulation*: set the value of $\bar{x}_0^{[1]}$ to $[10\ 0]'$ and initialize $w^{[0]}$ at zero, and perform the first real-time iteration by calling `sqpstep`. This iteration yields the new solution guess $w^{[1]}$ and corresponding control $u_0^{[1]}$. Use this control at the "real plant," i.e., generate the next value of $\bar{x}_0$, which we denote $\bar{x}_0^{[2]}$, by calling the one-step simulation function, $\bar{x}_0^{[2]} := \Phi(\bar{x}_0^{[1]}, u_0^{[1]})$. Close the loop by calling `sqpstep` using $w^{[1]}$ and $\bar{x}_0^{[2]}$, etc., and perform 100 iterations. For better observation, plot after each real-time iteration the control and state variables on the whole prediction horizon. (It is interesting to note that the state trajectory is not necessarily feasible).

Also observe what happens with the states $\bar{x}_0$ during the scenario, and plot them in another plot against the time index. Do they converge, and if yes, to what value?

(g) Now we make the control problem more difficult by treating the pendulum in an upright position, which is unstable. This is simply done by changing the sign in front of the sine in the differential equation, i.e., our model is now

$$f(x(t), u(t)) = \begin{bmatrix} v(t) \\ C\sin(p(t)/C) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t) \tag{8.67}$$

Start your real-time iterations again at $w^{[0]} = 0$ and set $\bar{x}_0^{[1]}$ to $[10\ 0]'$, and perform the same closed-loop simulation as before. Explain what happens.

Exercise 8.11: CIA norm and MILP

One of the heuristics discussed in Section 8.10 for approximating the solution of mixed-integer nonlinear optimal control problems is the combinatorial integral approximation

(CIA) (Sager et al., 2011). The CIA step solves the following optimization problem

$$\min_{\mathbf{u}_b} \max_{\substack{j \in \mathbb{I}_{1:n_b} \\ k \in \mathbb{I}_{0:N-1}}} \left| \sum_{i=0}^{k} u_{b,j}(i) - u^*_{b,j}(i) \right|$$

in which $\mathbf{u}_b$ is the discrete control sequence that approximates $\mathbf{u}^*_b$, the real-valued solution of a nonlinear program in the heuristic. Additional constraints can be included in this optimization such as rate-of-change constraints, dwell-time constraints, etc.

Consider the standard form of a mixed-integer linear program (MILP)

$$\min_{x,y} \; c'x + d'y$$

subject to

$$\begin{aligned} Ax + Ey &\leq b \\ y &\in \mathbb{B}^s \end{aligned}$$

with real $x \in \mathbb{R}^q$ and $b \in \mathbb{R}^r$, and binary $y \in \mathbb{B}^s$. State the CIA step in the standard form of an MILP, i.e., give the MILP variables $x, y, c, d, A, E, b, q, r, s$ for solving the CIA step.

Bibliography

J. Albersmeyer. *Adjoint-based algorithms and numerical methods for sensitivity generation and optimization of large scale dynamic systems*. PhD thesis, University of Heidelberg, 2010.

J. Albersmeyer and M. Diehl. The lifted Newton method and its application in optimization. *SIAM J. Optim.*, 20(3):1655–1684, 2010.

J. Andersson. *A General-Purpose Software Framework for Dynamic Optimization*. PhD thesis, KU Leuven, October 2013.

J. Andersson, J. Akesson, and M. Diehl. CasADi – a symbolic package for automatic differentiation and optimal control. In *Recent Advances in Algorithmic Differentiation*, volume 87 of *Lecture Notes in Computational Science and Engineering*, pages 297–307. Springer, 2012.

U. M. Ascher and L. R. Petzold. *Computer Methods for Ordinary Differential Equations and Differential–Algebraic Equations*. SIAM, Philadelphia, 1998.

D. Axehill. Controlling the level of sparsity in MPC. *Sys. Cont. Let.*, 76:1–7, 2015.

D. Axehill and M. Morari. An alternative use of the Riccati recursion for efficient optimization. *Sys. Cont. Let.*, 61(1):37–40, 2012.

I. Bauer, H. G. Bock, S. Körkel, and J. P. Schlöder. Numerical methods for optimum experimental design in DAE systems. *J. Comput. Appl. Math.*, 120 (1-2):1–15, 2000.

A. Ben-Tal and A. Nemirovski. *Lectures on Modern Convex Optimization: Analysis, Algorithms, and Engineering Applications*. MPS-SIAM Series on Optimization. MPS-SIAM, Philadelphia, 2001.

D. P. Bertsekas. *Nonlinear Programming*. Athena Scientific, Belmont, MA, second edition, 1999.

J. T. Betts. *Practical Methods for Optimal Control Using Nonlinear Programming*. SIAM, Philadelphia, 2001.

L. T. Biegler. *Nonlinear Programming*. MOS-SIAM Series on Optimization. SIAM, 2010.

T. Binder, L. Blank, H. G. Bock, R. Bulirsch, W. Dahmen, M. Diehl, T. Kronseder, W. Marquardt, J. P. Schlöder, and O. V. Stryk. Introduction to model based optimization of chemical processes on moving horizons. *Online Optimization of Large Scale Systems: State of the Art, Springer*, pages 295–340, 2001.

H. G. Bock. Numerical treatment of inverse problems in chemical reaction kinetics. In K. H. Ebert, P. Deuflhard, and W. Jäger, editors, *Modelling of Chemical Reaction Systems*, volume 18 of *Springer Series in Chemical Physics*, pages 102–125. Springer, Heidelberg, 1981.

H. G. Bock. Recent advances in parameter identification techniques for ODE. In *Numerical Treatment of Inverse Problems in Differential and Integral Equations*, pages 95–121. Birkhäuser, 1983.

H. G. Bock and K. J. Plitt. A multiple shooting algorithm for direct solution of optimal control problems. In *Proceedings of the IFAC World Congress*, pages 242–247. Pergamon Press, 1984.

S. P. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

K. E. Brenan, S. L. Campbell, and L. R. Petzold. *Numerical solution of initial-value problems in differential-algebraic equations*. Classics in Applied Mathematics 14. SIAM, Philadelphia, 1996.

A. E. Bryson and Y. Ho. *Applied Optimal Control*. Hemisphere Publishing, New York, 1975.

A. Bürger, C. Zeile, M. Hahn, A. Altmann-Dieses, S. Sager, and M. Diehl. pycombina: An open-source tool for solving combinatorial approximation problems arising in mixed-integer optimal control. In *Proceedings of the IFAC World Congress*, 2020.

A. R. Curtis, M. J. D. Powell, and J. K. Reid. On the estimation of sparse Jacobian matrices. *J. Inst. Math. Appl.*, 13:117–119, 1974.

P. Deuflhard. *Newton methods for nonlinear problems: affine invariance and adaptive algorithms*, volume 35. Springer, 2011.

M. Diehl. *Real-Time Optimization for Large Scale Nonlinear Processes*. PhD thesis, Universität Heidelberg, 2001.

M. Diehl, H. G. Bock, J. P. Schlöder, R. Findeisen, Z. Nagy, and F. Allgöwer. Real-time optimization and nonlinear model predictive control of processes governed by differential-algebraic equations. *J. Proc. Cont.*, 12(4):577–585, 2002.

M. Diehl, H. J. Ferreau, and N. Haverbeke. Efficient numerical methods for nonlinear MPC and moving horizon estimation. In L. Magni, M. D. Raimondo, and F. Allgöwer, editors, *Nonlinear model predictive control*, volume 384 of *Lecture Notes in Control and Information Sciences*, pages 391–417. Springer, 2009.

A. Domahidi. *Methods and Tools for Embedded Optimization and Control*. PhD thesis, ETH Zürich, 2013.

H. J. Ferreau, C. Kirches, A. Potschka, H. G. Bock, and M. Diehl. qpOASES: a parametric active-set algorithm for quadratic programming. *Mathematical Programming Computation*, 6(4):327–363, 2014.

R. Franke. *Integrierte dynamische Modellierung und Optimierung von Systemen mit saisonaler Wärmespeicherung*. PhD thesis, Technische Universität Ilmenau, Germany, 1998.

J. V. Frasch, S. Sager, and M. Diehl. A parallel quadratic programming method for dynamic optimization problems. *Mathematical Programming Computations*, 7(3):289–329, 2015.

G. Frison. *Algorithms and Methods for High-Performance Model Predictive Control*. PhD thesis, Technical University of Denmark (DTU), 2015.

A. H. Gebremedhin, F. Manne, and A. Pothen. What color is your Jacobian? Graph coloring for computing derivatives. *SIAM Review*, 47:629–705, 2005.

M. Gerdts. *Optimal Control of ODEs and DAEs*. Berlin, Boston: De Gruyter, 2011.

P. Gill, W. Murray, and M. Saunders. SNOPT: An SQP algorithm for large-scale constrained optimization. *SIAM Review*, 47(1):99–131, 2005.

J. Gillis. *Practical methods for approximate robust periodic optimal control of nonlinear mechanical systems*. PhD thesis, KU Leuven, 2015.

A. Griewank and A. Walther. *Evaluating Derivatives*. SIAM, 2 edition, 2008.

S. Gros and M. Diehl. *Numerical Optimal Control*. 2020. (In preparation).

J. Guddat, F. G. Vasquez, and H. T. Jongen. *Parametric Optimization: Singularities, Pathfollowing and Jumps*. Teubner, Stuttgart, 1990.

E. Hairer, S. P. Nørsett, and G. Wanner. *Solving Ordinary Differential Equations I*. Springer Series in Computational Mathematics. Springer, Berlin, 2nd edition, 1993.

E. Hairer, S. P. Nørsett, and G. Wanner. *Solving Ordinary Differential Equations II – Stiff and Differential-Algebraic Problems*. Springer Series in Computational Mathematics. Springer, Berlin, 2nd edition, 1996.

A. C. Hindmarsh, P. N. Brown, K. E. Grant, S. L. Lee, R. Serban, D. E. Shumaker, and C. S. Woodward. SUNDIALS: Suite of nonlinear and differential/algebraic equation solvers. *ACM Trans. Math. Soft.*, 31:363–396, 2005.

B. Houska, H. J. Ferreau, and M. Diehl. ACADO toolkit – an open source framework for automatic control and dynamic optimization. *Optimal Cont. Appl. Meth.*, 32(3):298–312, 2011.

D. H. Jacobson and D. Q. Mayne. *Differential dynamic programming*, volume 24 of *Modern Analytic and Computational Methods in Science and Mathematics*. American Elsevier Pub. Co., 1970.

M. R. Kristensen, J. B. Jørgensen, P. G. Thomsen, and S. B. Jørgensen. An ESDIRK method with sensitivity analysis capabilities. *Comput. Chem. Eng.*, 28:2695–2707, 2004.

D. B. Leineweber, I. Bauer, A. A. S. Schäfer, H. G. Bock, and J. P. Schlöder. An efficient multiple shooting based reduced SQP strategy for large-scale dynamic process optimization (Parts I and II). *Comput. Chem. Eng.*, 27:157–174, 2003.

W. Li and E. Todorov. Iterative linear quadratic regulator design for nonlinear biological movement systems. In *Proceedings of the 1st International Conference on Informatics in Control, Automation and Robotics*, 2004.

W. C. Li and L. T. Biegler. Multistep, Newton-type control strategies for constrained nonlinear processes. *Chem. Eng. Res. Des.*, 67:562–577, 1989.

D. Q. Mayne. A second-order gradient method for determining optimal trajectories of non-linear discrete-time systems. *Int. J. Control*, 3(1):85–96, 1966.

Y. Nesterov. *Introductory lectures on convex optimization: a basic course*, volume 87 of *Applied Optimization*. Kluwer Academic Publishers, 2004.

J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, New York, second edition, 2006.

T. Ohtsuka. A continuation/GMRES method for fast computation of nonlinear receding horizon control. *Automatica*, 40(4):563–574, 2004.

G. Pannocchia, J. B. Rawlings, D. Q. Mayne, and G. Mancuso. Whither discrete time model predictive control? *IEEE Trans. Auto. Cont.*, 60(1):246–252, January 2015.

L. R. Petzold, S. Li, Y. Cao, and R. Serban. Sensitivity analysis of differential-algebraic equations and partial differential equations. *Comput. Chem. Eng.*, 30:1553–1559, 2006.

R. Quirynen. *Numerical Simulation Methods for Embedded Optimization*. PhD thesis, KU Leuven and University of Freiburg, 2017.

R. Quirynen, S. Gros, B. Houska, and M. Diehl. Lifted collocation integrators for direct optimal control in ACADO toolkit. *Math. Prog. Comp.*, pages 1–45, 2017a.

R. Quirynen, B. Houska, and M. Diehl. Efficient symmetric Hessian propagation for direct optimal control. *J. Proc. Cont.*, 50:19–28, 2017b.

S. M. Robinson. Strongly regular generalized equations. *Mathematics of Operations Research, Vol. 5, No. 1 (Feb., 1980), pp. 43-62*, 5:43–62, 1980.

S. Sager, M. Jung, and C. Kirches. Combinatorial integral approximation. *Math. Method. Oper. Res.*, 73(3):363, 2011.

S. Sager, H. G. Bock, and M. Diehl. The integer approximation error in mixed-integer optimal control. *Math. Prog.*, 133:1–{23, 2012.

A. Sideris and J. Bobrow. An efficient sequential linear quadratic algorithm for solving unconstrained nonlinear optimal control problems. *IEEE Transactions on Automatic Control*, 50(12):2043–2047, 2005.

M. J. Tenny, S. J. Wright, and J. B. Rawlings. Nonlinear model predictive control via feasibility-perturbed sequential quadratic programming. *Comp. Optim. Appl.*, 28:87–121, 2004.

Q. Tran-Dinh, C. Savorgnan, and M. Diehl. Adjoint-based predictor-corrector sequential convex programming for parametric nonlinear optimization. *SIAM J. Optim.*, 22(4):1258–1284, 2012.

C. F. Van Loan. Computing integrals involving the matrix exponential. *IEEE Trans. Automat. Control*, 23(3):395–404, 1978.

A. Wächter and L. T. Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Math. Prog.*, 106(1):25–57, 2006.

A. Zanelli, A. Domahidi, J. Jerez, and M. Morari. FORCES NLP: An efficient implementation of interior-point methods for multistage nonlinear nonconvex programs. *Int. J. Control*, 2017.

C. Zeile, N. Robuschi, and S. Sager. Mixed-integer optimal control under minimum dwell time constraints. *Math. Prog.*, pages 1–42, 2020.

Author Index

Afonso, P. A. F. N. A., 306
Agarwal, M., 303
Aguilera, R. P., 171
Aguirre, L. A., 311
Akesson, J., 580
Alamo, T., 142, 145–147, 169, 259, 260, 359
Albersmeyer, J., 537, 580, 581
Alessandri, A., 319
Alessio, A., 477
Allan, D. A., 148, 150, 151, 169, 170, 211, 258, 273, 275–277, 279, 283, 287, 297, 298, 314–316, 320, 324, 696, 722
Allgöwer, F., 168–170, 256, 258, 260, 261, 347, 358, 573
Allwright, J. C., 259
Altmann-Dieses, A., 577
Alvarado, I., 169, 359
Amrit, R., 156, 157, 170, 184
Anderson, B. D. O., 319
Anderson, T. W., 662
Andersson, J., 580
Angeli, D., 156, 170, 260
Apostol, T. M., 638
Artstein, Z., 359
Ascher, U. M., 580
Åström, K. J., 318
Athans, M., 427
Aubin, J. P., 258, 358
Axehill, D., 561, 562

Baglietto, M., 319
Balakrishnan, V., 258
Bank, B., 477
Baotic, M., 171
Bard, Y., 682
Barić, M., 477
Bartle, R. G., 704
Başar, T., 382, 426
Bates, C. N., 148, 150, 151, 170, 211, 258, 314–316, 696
Battistelli, G., 319
Bauer, I., 580
Bayes, T., 674
Bellman, R. E., 14, 729
Bemporad, A., 170, 171, 476, 477
Ben-Tal, A., 579
Bequette, B. W., 303
Bernstein, D. S., 311
Berntorp, K., 302
Bertram, J. E., 701
Bertsekas, D. P., 14, 25, 258, 292, 358, 380, 417, 426, 579, 729, 750, 753, 755
Betts, J. T., 579
Bicchi, A., 171
Biegler, L. T., 170, 243, 257, 528, 554, 573, 579
Binder, T., 579
Blanchini, F., 109, 259, 339
Blank, L., 579
Bobrow, J., 564
Bock, H. G., 259, 512, 526, 549, 556, 573, 577, 579, 580
Bordons, C., 1
Borrelli, F., 476, 477
Boyd, S. P., 579, 624, 768, 769
Braatz, R., 259
Bravo, J. M., 259
Brenan, K. E., 580
Broustail, J. P., 167
Brown, P. N., 580
Bruyninckx, H., 305
Bryson, A. E., 366, 579
Bulirsch, R., 579

Bürger, A., 577

Cai, C., 169, 313, 722
Calafiore, G. C., 253, 254
Callier, F. M., 291, 677
Camacho, E. F., 1, 145–147, 169, 258, 259, 359
Campbell, S. L., 580
Campi, M. C., 254
Cannon, M., 256, 259–261
Cao, Y., 580
Casavola, A., 260
Castro, J. A. A. M., 306
Chatterjee, D., 248, 249, 260, 261
Chaves, M., 303
Chen, C. C., 168
Chen, H., 168, 170, 258
Cheng, Q., 260, 261
Chisci, L., 234, 250, 256, 259, 260
Chmielewski, D., 168, 169
Christofides, P. D., 170
Christophersen, F. J., 171
Chryssochoos, I., 259, 260
Clarke, D. W., 167
Clarke, F., 761, 762
Coddington, E. A., 649
Columbano, S., 476
Copp, D. A., 320, 359
Cui, H., 427
Curtis, A. R., 516, 527
Cutler, C. R., 167

Dabbene, F., 253, 254, 256, 260, 261
Dahmen, W., 579
Damborg, M. J., 168
Dantzig, G. B., 477
David, H. A., 686
Davison, E. J., 50
De Doná, J. A., 1, 476, 477
de Freitas, N., 305
De Keyser, R. M. C., 167
de la Peña, D. M., 142, 169, 259, 260
De Nicolao, G., 168, 169, 257, 258
De Schutter, J., 305
de Souza, C. E., 326
Desoer, C. A., 291, 677
Deuflhard, P., 512, 579
Deyst, J., 290, 319
Di Cairano, S., 171
Diehl, M., 157, 170, 184, 259, 320, 537, 545, 549, 552, 573, 577, 579–581
Dieudonne, J., 638
Domahidi, A., 580
Doucet, A., 305
Dreyfus, S. E., 14
Dua, V., 477
Dunbar, W. B., 427
Durand, H., 170
Durrant-Whyte, H. F., 305

Eaton, J. W., 104
Ellis, M., 170

Fagiano, L., 169
Falugi, P., 142, 169, 249, 259, 261, 359
Feller, C., 476
Ferramosca, A., 359
Ferreau, H. J., 579, 580
Filippova, T. F., 358
Findeisen, R., 259, 347, 358, 573
Fletcher, R., 66
Folkman, J., 477
Foss, B. A., 311, 358
Francis, B. A., 50
Franke, R., 580
Franże, G., 260
Frasch, J. V., 580
Frison, G., 560, 561, 580
Fukudu, K., 476

Gal, T., 477
García, C. E., 167
Gass, S. I., 477
Gebremedhin, A. H., 516
Gerdts, M., 579
Gevers, M. R., 326

Gilbert, E. G., 168, 172, 173, 224, 225, 259, 280, 339
Gill, P., 579
Gillette, R. D., 167
Gillis, J., 580
Glover, J. D., 358
Golub, G. H., 368, 629, 630
Goodwin, G. C., 1, 70, 171, 261, 326, 476, 477
Goulart, P. J., 259, 260
Grant, K. E., 580
Gray, M., 303
Griewank, A., 527, 580
Grimm, G., 169, 258, 722
Gros, S., 170, 579, 580
Grover, P., 302
Grüne, L., 169, 170
Guddat, J., 477, 545
Gudi, R., 303
Günther, S., 303

Hager, W. W., 479
Hahn, J., 311
Hahn, M., 577
Hairer, E., 580
Hale, J., 649
Harinath, E., 170
Hartman, P., 649
Hauser, J., 169
Hautus, M. L. J., 24, 42
Haverbeke, N., 579
Heemels, W. P. M. H., 170, 171, 260, 313, 314, 701, 726
Henson, M. A., 104
Herceg, M., 476
Hespanha, J. P., 320, 359
Hicks, G. A., 243
Hillar, C. J., 681
Hindmarsh, A. C., 580
Hiraishi, K., 171
Ho, Y., 366, 579
Hokayem, P., 260
Horn, R. A., 23, 64, 273, 629
Houska, B., 580
Hu, W., 320
Huang, R., 170

Imsland, L., 311, 358

Jacobsen, E. W., 427
Jacobson, D. H., 564
Jadbabaie, A., 169
Jazwinski, A. H., 38, 290, 318, 319
Jerez, J., 580
Ji, L., 278, 319, 320, 323, 696
Jia, D., 427
Jiang, Z.-P., 314, 321, 693, 699, 702, 703, 713, 717–719
Johansen, T. A., 476
Johnson, C. R., 23, 64, 273, 629, 681
Jones, C. N., 142, 169, 476, 477
Jongen, H. T., 545
Jørgensen, J. B., 580
Jørgensen, S. B., 580
Julier, S. J., 304, 305, 311
Jung, M., 577, 594

Kailath, T., 318, 319
Kalman, R. E., 21, 26, 701
Kameswaran, S., 243
Kandepu, R., 311
Keerthi, S. S., 168, 172, 280
Kellett, C. M., 208, 693
Kerrigan, E. C., 231, 259, 260, 339, 477
Khalil, H. K., 693, 695, 726
Khurzhanski, A. B., 258, 358
Kirches, C., 577, 580, 594
Klatte, D., 477
Kleinman, D. L., 167
Knüfer, S., 320
Kobayshi, K., 171
Kolås, S., 311
Kolmanovsky, I., 224, 225, 259, 339
Kolmogorov, A. N., 318
Körkel, S., 580
Kostina, E., 259
Kothare, M. V., 258

Kouramas, K. I., 231, 259, 339
Kouvaritakis, B., 256, 258–261
Krichman, M., 719
Kristensen, M. R., 580
Krogh, B. H., 427
Kronseder, T., 579
Kummer, B., 477
Kurzhanski, A. B., 358
Kvasnica, M., 476
Kwakernaak, H., 50, 319
Kwon, W. H., 1, 167

Langson, W., 250, 256, 259, 260
Larsson, T., 427
LaSalle, J. P., 693
Lazar, M., 170, 171, 260, 313, 314, 701, 726
Ledyaev, Y. S., 761, 762
Lee, E. B., 167
Lee, J. H., 258, 300, 319
Lee, S. L., 580
Lefebvre, T., 305
Leineweber, D. B., 580
Levinson, N., 649
Li, S., 580
Li, W., 565
Li, W. C., 573
Limon, D., 142, 145–147, 169, 258, 259, 359
Lorenzen, M., 256, 260, 261
Løvaas, C., 261
Lunze, J., 427
Lygeros, J., 248, 249, 260, 261

Maciejowski, J. M., 1, 259, 260
Macko, D., 427
Maestre, J. M., 428
Magni, L., 168, 169, 257, 258
Mancuso, G., 92, 538
Manne, F., 516
Manousiouthakis, V., 168, 169
Markus, L., 167
Marquardt, W., 579
Marquis, P., 167
Mayne, D. Q., 92, 142, 147, 167–169, 231, 249, 250, 256–261, 300, 302, 319, 323, 339, 347, 358, 359, 371, 462, 477, 538, 564, 565
McShane, E. J., 649
Meadows, E. S., 104, 257, 319, 479
Mehta, P. G., 302
Mesarović, M., 427
Messina, M. J., 169, 258, 722
Meyn, S. P., 302
Miani, S., 109, 259
Michalska, H., 168, 260, 319, 358
Middlebrooks, S. A., 303
Mohtadi, C., 167
Moitié, R., 358
Moore, J. B., 319
Morari, M., 167, 170, 171, 258, 476, 477, 561, 580
Morgenstern, O., 426
Morshedi, A. M., 167
Mosca, E., 260
Motee, N., 427
Müller, M. A., 169, 170, 320
Murray, R. M., 427
Murray, W., 579
Muske, K. R., 168, 172, 319

Nagy, Z., 259, 573
Narasimhan, S., 310, 311
Nash, J., 426
Nedic, A., 729
Nedoma, J., 477
Negenborn, R. R., 428
Nemirovski, A., 579
Nesterov, Y., 579
Nevistić, V., 169
Nijmeijer, H., 170, 313
Nocedal, J., 66, 514, 579, 624, 768, 769
Nørgaard, M., 305
Nørsett, S. P., 580

Odelson, B. J., 51

Ohtsuka, T., 554
Olaru, S., 476
Olsder, G. J., 382, 426
Ozdaglar, A. E., 729

Pancanti, S., 171
Pannek, J., 169
Pannocchia, G., 52, 53, 92, 147, 169, 258, 347, 352, 421, 538
Panos, C., 259
Papon, J., 167
Papoulis, A., 657, 658
Parisini, T., 168, 169
Pearson, A. E., 167
Pereira, M., 142, 169
Peressini, A. L., 769
Peterka, V., 167
Petzold, L. R., 580
Picasso, B., 171
Pistikopoulos, E. N., 477
Plitt, K. J., 556
Polak, E., 624, 757, 759, 760
Pothen, A., 516
Potschka, A., 580
Poulsen, N. K., 305
Powell, M. J. D., 516, 527
Praly, L., 357
Prasad, V., 303
Prett, D. M., 167
Price, C., 290, 319
Primbs, J. A., 169
Propoi, A. I., 167

Qiu, L., 50
Qu, C. C., 311
Quevedo, D. E., 171
Quincampoix, M., 358
Quirynen, R., 580

Raimondo, D. M., 259
Rajamani, M. R., 51
Raković, S. V., 231, 259–261, 339, 347, 358, 359, 462, 477
Ralph, D., 259
Ramaker, B. L., 167
Rao, C. V., 70, 162, 167, 169, 257, 300, 319, 323
Rault, A., 167
Ravn, O., 305
Rawlings, J. B., 51–53, 70, 92, 104, 129, 142, 147, 148, 150, 151, 156, 157, 162, 167–172, 184, 211, 257, 258, 273, 275–279, 283, 287, 297, 298, 300, 302, 303, 314–316, 319, 320, 323, 347, 352, 371, 416, 418, 421, 423, 427, 428, 479, 538, 566, 696, 707, 709, 722
Ray, W. H., 243
Reble, M., 170, 258
Reid, J. K., 516, 527
Reif, K., 303
Rengaswamy, R., 310, 311
Rhodes, I. B., 258, 358
Rice, M. J., 258
Richalet, J., 167
Rippin, D. W. T., 303
Risbeck, M. J., 129, 142, 148, 150, 151, 169–171, 211, 258, 279, 314–316, 696, 707, 709
Robertson, D. G., 319
Robinson, S. M., 545
Robuschi, N., 577
Rockafellar, R. T., 212, 439, 641, 646, 739, 740, 746, 748–750, 757, 759
Romanenko, A., 306
Roset, B. J. P., 170, 313
Rossiter, J. A., 1, 234, 250, 256, 258–260
Russo, L. P., 303

Saaty, T. L., 477
Safonov, M., 427
Sager, S., 577, 580, 594
Salas, F., 145–147
Sandell Jr., N. R., 427
Santos, L. O., 257, 306

Saunders, M., 579
Savorgnan, C., 549, 573
Sayyar-Rodsari, B., 427
Scattolini, R., 168, 169, 257, 258, 428
Schäfer, A. A. S., 580
Schei, T. S., 311
Schley, M., 303
Schlöder, J. P., 573, 580
Schur, I., 629
Schweppe, F. C., 358
Scokaert, P. O. M., 147, 167–169, 257, 258, 371, 479
Selby, S. M., 73
Sepulchre, R., 257
Serban, R., 580
Serón, M. M., 1, 259, 261, 476, 477
Shah, S., 303
Shapiro, N. Z., 477
Shaw, L., 168
Shein, W. W., 171
Sherbert, D. R., 704
Shumaker, D. E., 580
Sideris, A., 564
Šiljak, D. D., 427
Sin, K. S., 70
Sivan, R., 50, 319
Skogestad, S., 427
Smith, H. W., 50
Sontag, E. D., 23, 41, 275, 276, 303, 321, 705–707, 713, 717, 719
Stengel, R. F., 303, 319
Stern, R. J., 761, 762
Stewart, B. T., 416, 418, 421, 423, 427
Strang, G., 23, 42, 625, 626
Stryk, O. V., 579
Sullivan, F. E., 769
Sznaier, M., 168

Takahara, Y., 427
Tan, K. T., 168, 173
Tanner, K., 477
Teel, A. R., 169, 204–209, 257, 258, 275–277, 313, 314, 357, 693, 701, 709, 710, 712, 714, 722, 726
Teixeira, B. O. S., 311
Tempo, R., 253, 254, 256, 260, 261
Tenny, M. J., 300, 566
Testud, J. L., 167
Thomas, Y. A., 167
Thomsen, P. G., 580
Todorov, E., 565
Tôrres, L. A. B., 311
Tran-Dinh, Q., 549, 573
Tsitsiklis, J. N., 380, 426
Tuffs, P. S., 167
Tuna, S. E., 169, 258, 722

Uhl, Jr., J. J., 769
Uhlmann, J. K., 304, 305, 311
Unbehauen, R., 303

Vachhani, P., 310, 311
Valyi, I., 258, 358
Van Cauwenberghe, A. R., 167
van der Merwe, R., 305
Van Loan, C. F., 538
van Wyk, E. J., 259
Vandenberghe, L., 579, 624, 768, 769
Varaiya, P., 427
Vasquez, F. G., 545
Veliov, V. M., 358
Venkat, A. N., 421, 427, 428
Vidyasagar, M., 370
von Neumann, J., 426

Wächter, A., 528, 554, 579
Walther, A., 527, 580
Wan, E., 305
Wang, L., 1
Wang, Y., 275, 276, 314, 321, 693, 699, 702, 703, 713, 717–719
Wanner, G., 580
Wets, R. J.-B., 212, 646, 739, 740, 746, 748–750, 757, 759
Wiener, N., 318
Wilson, D. I., 303

Wolenski, P. R., 761, 762
Wonham, W. M., 50
Woodward, C. S., 580
Wright, S. J., 66, 147, 169, 258, 366, 416, 418, 421, 423, 427, 428, 514, 566, 579, 624, 768, 769
Wynn, A., 320

Xie, L., 320

Yang, T., 302
Yaz, E., 303
Ydstie, B. E., 167
You, K., 320
Yu, J., 169
Yu, S., 170, 258
Yu, Z., 258

Zanelli, A., 580
Zanon, M., 170
Zappa, G., 234, 250, 256, 259, 260
Zeile, C., 577
Zeilinger, M. N., 142, 169
Zoppoli, R., 168, 169

Citation Index

Aguilera and Quevedo (2013), 171, 186
Albersmeyer (2010), 580, 595
Albersmeyer and Diehl (2010), 537, 581, 595
Alessandri et al. (2008), 319, 327
Alessio and Bemporad (2009), 477, 483
Allan (2020), 279, 287, 320, 324, 327
Allan and Rawlings (2018), 169, 186, 722, 727
Allan and Rawlings (2019), 277, 279, 283, 327
Allan and Rawlings (2020), 273, 279, 287, 297, 298, 327
Allan et al. (2017), 148, 150, 151, 170, 186, 211, 258, 264, 314–316, 327, 696, 727
Allan et al. (2020), 275–277, 327
Amrit et al. (2011), 170, 186
Anderson (2003), 661, 691
Anderson and Moore (1981), 319, 327
Andersson (2013), 580, 595
Andersson et al. (2012), 580, 595
Angeli et al. (2008), 260, 264
Angeli et al. (2012), 156, 170, 186
Apostol (1974), 638, 691
Artstein and Raković (2008), 359, 361
Ascher and Petzold (1998), 580, 595
Åström (1970), 318, 327
Aubin (1991), 258, 264, 358, 361
Axehill (2015), 562, 595
Axehill and Morari (2012), 561, 595

Bank et al. (1983), 477, 483
Baotic et al. (2006), 171, 186
Bard (1974), 682, 691
Bartle and Sherbert (2000), 704, 727
Başar and Olsder (1999), 382, 426, 442
Bauer et al. (2000), 580, 595
Bayes (1763), 674, 691
Bellman (1957), 14, 86, 729, 770
Bellman and Dreyfus (1962), 14, 86
Bemporad and Morari (1999), 170, 186
Bemporad et al. (2002), 477, 483
Ben-Tal and Nemirovski (2001), 579, 595
Berntorp and Grover (2018), 302, 327
Bertsekas (1987), 14, 25, 86, 292, 327
Bertsekas (1999), 417, 442, 579, 595, 750, 753, 755, 770
Bertsekas and Rhodes (1971), 358, 361
Bertsekas and Rhodes (1971a), 258, 264
Bertsekas and Rhodes (1971b), 258, 264
Bertsekas and Tsitsiklis (1997), 380, 426, 442
Bertsekas et al. (2001), 729, 770
Betts (2001), 579, 595
Biegler (2010), 579, 595
Binder et al. (2001), 579, 595
Blanchini (1999), 259, 264, 339, 361
Blanchini and Miani (2008), 109, 186, 259, 264
Bock (1981), 526, 596
Bock (1983), 512, 549, 596
Bock and Plitt (1984), 556, 596
Borrelli (2003), 477, 483
Borrelli et al. (2017), 476, 483

Boyd and Vandenberghe (2004), 579, 596, 624, 691, 768–770
Brenan et al. (1996), 580, 596
Bryson and Ho (1975), 366, 442, 579, 596
Bürger et al. (2020), 577, 596

Cai and Teel (2008), 169, 186, 313, 327, 722, 727
Calafiore and Campi (2006), 254, 264
Callier and Desoer (1991), 291, 327, 677, 691
Camacho and Bordons (2004), 1, 86
Chatterjee and Lygeros (2015), 248, 249, 260, 261, 264
Chatterjee et al. (2011), 260, 264
Chaves and Sontag (2002), 303, 327
Chen and Allgöwer (1998), 168, 186, 258, 264
Chen and Shaw (1982), 168, 186
Chisci et al. (2001), 234, 250, 256, 259, 260, 264
Chmielewski and Manousiouthakis (1996), 168, 169, 186
Clarke et al. (1987), 167, 186
Clarke et al. (1998), 761, 762, 770
Coddington and Levinson (1955), 649, 691
Columbano et al. (2009), 476, 483
Copp and Hespanha (2014), 359, 361
Copp and Hespanha (2017), 320, 328
Cui and Jacobsen (2002), 427, 442
Curtis et al. (1974), 516, 527, 596
Cutler and Ramaker (1980), 167, 187

Dantzig et al. (1967), 477, 483
David (1981), 686, 691
Davison and Smith (1971), 50, 86
Davison and Smith (1974), 50, 86
De Keyser and Van Cauwenberghe (1985), 167, 187
De Nicolao et al. (1996), 168, 187, 257, 264
De Nicolao et al. (1998), 169, 187
de Souza et al. (1986), 326, 328
Deuflhard (2011), 512, 579, 596
Deyst and Price (1968), 290, 319, 328
Di Cairano et al. (2014), 171, 187
Diehl (2001), 545, 552, 596
Diehl et al. (2002), 573, 596
Diehl et al. (2006), 259, 264
Diehl et al. (2009), 579, 596
Diehl et al. (2011), 157, 170, 184, 187
Dieudonne (1960), 638, 691
Domahidi (2013), 580, 597
Dunbar (2007), 427, 442
Dunbar and Murray (2006), 427, 442

Ellis et al. (2014), 170, 187

Fagiano and Teel (2012), 169, 187
Falugi and Mayne (2011), 261, 265
Falugi and Mayne (2013), 359, 361
Falugi and Mayne (2013a), 169, 187
Falugi and Mayne (2013b), 142, 169, 187
Feller et al. (2013), 476, 483
Ferramosca et al. (2009), 359, 361
Ferreau et al. (2014), 580, 597
Findeisen et al. (2003), 358, 361
Fletcher (1987), 66, 86
Francis and Wonham (1976), 50, 86
Franke (1998), 580, 597
Frasch et al. (2015), 580, 597
Frison (2015), 560, 561, 580, 597

Gal and Nedoma (1972), 477, 483
García and Morshedi (1986), 167, 187
García et al. (1989), 167, 187
Gass and Saaty (1955), 477, 483
Gebremedhin et al. (2005), 516, 597
Gelb (1974), 303, 328
Gerdts (2011), 579, 597
Gilbert and Tan (1991), 168, 173, 187
Gill et al. (2005), 579, 597
Gillis (2015), 580, 597

Glover and Schweppe (1971), 358, 361
Golub and Van Loan (1996), 368, 442, 629, 630, 691
Goodwin and Sin (1984), 70, 86
Goodwin et al. (2005), 1, 86
Goulart et al. (2006), 259, 260, 265
Goulart et al. (2008), 259, 265
Griewank and Walther (2008), 527, 580, 597
Grimm et al. (2005), 169, 188, 722, 727
Grimm et al. (2007), 258, 265
Gros and Diehl (2020), 579, 597
Grüne and Pannek (2017), 169, 188
Guddat et al. (1990), 545, 597
Gudi et al. (1994), 303, 328

Hager (1979), 479, 483
Hairer et al. (1993), 580, 597
Hairer et al. (1996), 580, 597
Hale (1980), 649, 691
Hartman (1964), 649, 691
Hautus (1972), 24, 42, 86
Herceg et al. (2013), 476, 483
Hicks and Ray (1971), 243, 265
Hindmarsh et al. (2005), 580, 598
Horn and Johnson (1985), 23, 64, 86, 273, 328, 629, 691
Houska et al. (2011), 580, 598
Hu (2017), 320, 328
Hu et al. (2015), 320, 328
Huang et al. (2011), 170, 188

Imsland et al. (2003), 358, 361

Jacobson and Mayne (1970), 564, 598
Jadbabaie et al. (2001), 169, 188
Jazwinski (1970), 38, 86, 290, 318, 319, 328
Ji et al. (2016), 320, 328
Jia and Krogh (2002), 427, 442
Jiang and Wang (2001), 314, 321, 328, 693, 717–719, 727
Jiang and Wang (2002), 693, 699, 702, 703, 713, 718, 727
Johnson (1970), 681, 691
Johnson and Hillar (2002), 681, 691
Jones (2017), 477, 483
Jones et al. (2007), 477, 484
Julier and Uhlmann (1997), 311, 328
Julier and Uhlmann (2002), 305, 328
Julier and Uhlmann (2004a), 303–305, 328
Julier and Uhlmann (2004b), 304, 329
Julier et al. (2000), 305, 329

Kailath (1974), 318, 319, 329
Kalman (1960a), 26, 86
Kalman (1960b), 21, 86
Kalman and Bertram (1960), 701, 727
Kameswaran and Biegler (2006), 243, 265
Kandepu et al. (2008), 311, 329
Keerthi and Gilbert (1985), 280, 329
Keerthi and Gilbert (1987), 172, 188
Keerthi and Gilbert (1988), 168, 188
Kellett and Teel (2004), 208, 265
Kellett and Teel (2004a), 693, 727
Kellett and Teel (2004b), 693, 727
Khalil (2002), 693, 695, 726, 727
Khurzhanski and Valyi (1997), 258, 265, 358, 361
Kleinman (1970), 167, 188
Knüfer and Müller (2018), 320, 329
Kobayshi et al. (2014), 171, 188
Kolås et al. (2009), 311, 329
Kolmanovsky and Gilbert (1995), 259, 265
Kolmanovsky and Gilbert (1998), 224, 225, 265, 339, 361
Kolmogorov (1941), 318, 329
Kothare et al. (1996), 258, 265
Kouramas et al. (2005), 259, 265
Kouvaritakis and Cannon (2016), 256, 266

Kouvaritakis et al. (2010), 260, 261, 266
Krichman et al. (2001), 719, 727
Kristensen et al. (2004), 580, 598
Kurzhanski and Filippova (1993), 358, 361
Kwakernaak and Sivan (1972), 50, 87, 319, 329
Kwon (2005), 1, 87
Kwon and Pearson (1977), 167, 188

Langson et al. (2004), 259, 260, 266
Larsson and Skogestad (2000), 427, 442
LaSalle (1986), 693, 727
Lazar and Heemels (2009), 170, 188
Lazar et al. (2008), 260, 266
Lazar et al. (2009), 701, 726, 727
Lazar et al. (2013), 314, 329
Lee and Markus (1967), 167, 188
Lee and Yu (1997), 258, 266
Lefebvre et al. (2002), 305, 329
Leineweber et al. (2003), 580, 598
Li and Biegler (1989), 573, 598
Li and Todorov (2004), 565, 598
Limon et al. (2002), 258, 266
Limon et al. (2006), 145–147, 188
Limon et al. (2008), 169, 188, 259, 266, 359, 362
Limon et al. (2010), 169, 189
Limon et al. (2012), 142, 169, 189
Lorenzen et al. (2016), 256, 260, 261, 266
Løvaas et al. (2008), 261, 266
Lunze (1992), 427, 442

Maciejowski (2002), 1, 87
Maestre and Negenborn (2014), 428, 442
Magni and Sepulchre (1997), 257, 266
Magni et al. (2003), 258, 266
Marquis and Broustail (1988), 167, 189
Mayne (1966), 564, 565, 598
Mayne (1995), 258, 266
Mayne (1997), 258, 267
Mayne (2000), 169, 189
Mayne (2013), 147, 169, 189
Mayne (2016), 260, 261, 267
Mayne and Falugi (2016), 169, 189
Mayne and Falugi (2019), 249, 267
Mayne and Langson (2001), 250, 256, 259, 260, 267
Mayne and Michalska (1990), 168, 189
Mayne and Raković (2002), 477, 484
Mayne and Raković (2003), 462, 477, 484
Mayne et al. (2000), 167, 169, 189, 257, 267
Mayne et al. (2005), 259, 267
Mayne et al. (2006), 358, 362
Mayne et al. (2007), 477, 484
Mayne et al. (2009), 347, 358, 362
Mayne et al. (2011), 259, 267
McShane (1944), 649, 691
Meadows et al. (1993), 319, 329
Meadows et al. (1995), 104, 189
Mesarović et al. (1970), 427, 442
Michalska and Mayne (1993), 168, 189, 260, 267
Michalska and Mayne (1995), 319, 329, 358, 362
Middlebrooks and Rawlings (2006), 303, 329
Moitié et al. (2002), 358, 362
Motee and Sayyar-Rodsari (2003), 427, 442
Müller (2017), 320, 330
Müller and Allgöwer (2014), 169, 189
Müller and Grüne (2015), 170, 189
Müller et al. (2015), 170, 190
Muske and Rawlings (1993), 172, 190
Muske et al. (1993), 319, 330

Nagy and Braatz (2004), 259, 267
Nash (1951), 426, 443

Nesterov (2004), 579, 598
Nocedal and Wright (2006), 66, 87, 514, 579, 598, 624, 691, 768–770
Nørgaard et al. (2000), 305, 330

Odelson et al. (2006), 51, 87
Ohtsuka (2004), 554, 598

Pannocchia and Rawlings (2003), 52, 53, 87, 347, 352, 362
Pannocchia et al. (2011), 147, 169, 190, 258, 267
Pannocchia et al. (2015), 92, 190, 538, 598
Papoulis (1984), 657, 658, 692
Parisini and Zoppoli (1995), 168, 169, 190
Peressini et al. (1988), 769, 770
Peterka (1984), 167, 190
Petzold et al. (2006), 580, 598
Picasso et al. (2003), 171, 190
Polak (1997), 624, 692, 757, 759, 760, 770
Prasad et al. (2002), 303, 330
Prett and Gillette (1980), 167, 190
Primbs and Nevistić (2000), 169, 190
Propoi (1963), 167, 190

Qiu and Davison (1993), 50, 87
Qu and Hahn (2009), 311, 330
Quevedo et al. (2004), 171, 190
Quirynen (2017), 580, 599
Quirynen et al. (2017a), 580, 599
Quirynen et al. (2017b), 580, 599

Raković (2012), 259, 267
Raković et al. (2003), 259, 267
Raković et al. (2005), 339, 362
Raković et al. (2005a), 231, 259, 267
Raković et al. (2005b), 259, 268
Raković et al. (2012), 259, 268
Rao (2000), 300, 319, 330
Rao and Rawlings (1999), 70, 87, 162, 190
Rao et al. (2001), 300, 330
Rao et al. (2003), 319, 323, 330
Rawlings and Amrit (2009), 170, 190
Rawlings and Ji (2012), 278, 319, 323, 330, 696, 728
Rawlings and Mayne (2009), 300, 302, 330
Rawlings and Muske (1993), 168, 190
Rawlings and Risbeck (2015), 129, 191, 279, 330, 709, 728
Rawlings and Risbeck (2017), 142, 169–171, 191, 707, 728
Rawlings and Stewart (2008), 427, 443
Reif and Unbehauen (1999), 303, 330
Reif et al. (1999), 303, 330
Reif et al. (2000), 303, 331
Richalet et al. (1978a), 167, 191
Richalet et al. (1978b), 167, 191
Robertson and Lee (2002), 319, 331
Robinson (1980), 545, 599
Rockafellar (1970), 439, 443, 641, 692
Rockafellar and Wets (1998), 212, 268, 646, 692, 739, 740, 746, 748–750, 757, 759, 770
Romanenko and Castro (2004), 306, 331
Romanenko et al. (2004), 306, 331
Roset et al. (2008), 170, 191, 313, 331
Rossiter (2004), 1, 87
Rossiter et al. (1998), 258, 268

Sager et al. (2011), 577, 594, 599
Sager et al. (2012), 577, 599
Sandell Jr. et al. (1978), 427, 443
Santos and Biegler (1999), 257, 268
Scattolini (2009), 428, 443
Schur (1909), 629, 692
Scokaert and Mayne (1998), 258, 268
Scokaert and Rawlings (1998), 168, 169, 191

Scokaert et al. (1997), 257, 268, 479, 484
Scokaert et al. (1999), 147, 169, 191, 371, 443
Selby (1973), 73, 87
Serón et al. (2000), 476, 477, 484
Sideris and Bobrow (2005), 564, 599
Šiljak (1991), 427, 443
Sontag (1998), 23, 41, 87, 321, 331
Sontag (1998a), 706, 707, 713, 728
Sontag (1998b), 705, 728
Sontag and Wang (1995), 717, 728
Sontag and Wang (1997), 275, 276, 321, 331, 719, 728
Stengel (1994), 303, 319, 331
Stewart et al. (2010), 421, 443
Stewart et al. (2011), 416, 418, 421, 423, 443
Strang (1980), 23, 42, 87, 625, 626, 692
Sznaier and Damborg (1987), 168, 191

Teel (2004), 204–209, 257, 268, 709, 710, 712, 714, 728
Teel and Praly (1994), 357, 362
Teixeira et al. (2008), 311, 331
Tempo et al. (2013), 253, 254, 268
Tenny and Rawlings (2002), 300, 331
Tenny et al. (2004), 566, 599
Thomas (1975), 167, 191
Tran-Dinh et al. (2012), 549, 573, 599

Vachhani et al. (2006), 310, 311, 331
van der Merwe et al. (2000), 305, 331
Van Loan (1978), 538, 599
Venkat (2006), 428, 443
Venkat et al. (2006a), 428, 443
Venkat et al. (2006b), 428, 443
Venkat et al. (2007), 427, 443
Vidyasagar (1993), 370, 444
von Neumann and Morgenstern (1944), 426, 444

Wächter and Biegler (2006), 528, 554, 579, 599
Wang (2009), 1, 87
Wiener (1949), 318, 331
Wilson et al. (1998), 303, 332
Wright (1997), 366, 444

Yang et al. (2013), 302, 332
Ydstie (1984), 167, 191
Yu et al. (2011), 258, 268
Yu et al. (2014), 170, 191

Zanelli et al. (2017), 580, 599
Zanon et al. (2013), 170, 191
Zeile et al. (2020), 577, 599

Subject Index

A-stable integration methods, 501
Accumulation point, *see* Sequence
Active
  constraint, 743
  set, 552, 743
AD, 514, 516, 561, 580
  forward mode, 520
  reverse mode, 522
Adaptive stepsize, 507
Adjoint operator, 676
Admissible, 97
  control, 464, 465, 469
  control sequence, 108, 210, 732
  disconnected region, 161
  disturbance sequence, 198, 204, 215, 227
  policy, 197, 198
  set, 137, 166, 168
Affine, 446, 447, 488
  function, 447
  hull, 450, 632
  invariance, 513
  piecewise, 104, 448, 450, 452, 458, 462, 468, 763
  set, 632
Algebraic states, 506
Algorithmic (or automatic) differentiation, *see* AD
Arrival cost, 33, 70, 79, 80
  full information, 296
AS, 112, 423, 698
Asymptotically stable, *see* AS
Attraction
  domain of, 700
  global, 697
  region of, 113, 700
Attractivity, 710
Back propagation algorithm, 524
Bar quantities, 518
Bayes's theorem, 672, 674
Bellman-Gronwall lemma, 651
BFGS, 550
Bolzano-Weierstrass theorem, 111, 632
Boundary-value problem, *see* BVP
Bounded
  locally, 209, 693, 696, 710, 711
Bounded estimate error, 313
Broyden-Fletcher-Goldfarb-Shanno, *see* BFGS
Butcher tableau, 498
BVP, 493, 582

C-set, 338
Caratheodory conditions, 651
CasADi, vii, xii, xiii, 527, 580, 585, 586, 591
Cayley-Hamilton theorem, 22, 64
Central limit theorem, 657
Centralized control, 363, 376
Certainty equivalence, 194
Chain rule, 61, 638
Cholesky factorization, 508, 561
CIA, 577, 594
CLF, 131, 134, 714–716
  constrained, 716
  global, 714
Closed-loop control, 92
Code generation, 569
Collocation, 502
  direct, 540
  methods, 502
  points, 502
Combinatorial integral approximation, *see* CIA

Combining MHE and MPC, 312
  stability, 314
Complementarity condition, 543
  strict, 544
Concave function, 647
Condensing, 491, 560
Cone
  convex, 644
  normal, 439, 737, 743, 746, 748
  polar, 453, 455, 644, 740
  tangent, 737, 743, 746, 748
Constrained Gauss-Newton method, 549
Constraint qualification, 479, 750, 751
Constraints, 6
  active, 543, 743
  coupled input, 405
  hard, 7, 94
  input, 6, 94
  integrality, 8
  output, 6
  polyhedral, 743
  probabilistic, 254
  soft, 7, 132
  state, 6, 94
  terminal, 96, 144–147, 212
  tightened, 202, 223, 230, 242, 346, 357
  trust region, 514
  uncoupled input, 402
Continuation methods, 571
Continuity, 633
  lower semicontinuous, 634
  uniform, 634
  upper semicontinuous, 634
Control law, 90, 200, 445
  continuity, 104
  discontinuity, 104
  explicit, 446
  implicit, 100, 210, 446
  offline, 89, 236
  online, 89
  time-invariant, 100
Control Lyapunov function, *see* CLF
Control vector parameterization, 532
Controllability, 23
  canonical form, 68
  duality with observability, 291
  matrix, 23
  weak, 116
Controllable, 23
Converse theorem
  asymptotic stability, 705
  exponential stability, 374, 725
Convex, 646
  cone, 644
  function, 488, 583, 646
  hull, 641
  optimization problem, 487, 741
    optimality condition, 453
  set, 338, 583, 641
Cooperative control, 363, 386
  algorithm, 421
  distributed nonlinear, 419
Correlation, 668
Cost function, 11, 95, 369

DAE, 505
  semiexplicit DAE of index one, 506
Damping, 514
DARE, 25, 69, 136
DDP, 564
  exact Hessian, 565
  Gauss-Newton Hessian, 565
Decentralized control, 363, 377
Decreasing, *see* Sequence
Derivatives, 636
Detectability, 50, 120, 275, 319, 321, 322, 719
  duality with stabilizability, 291
  exponential, 285
Detectable, 26, 68, 72, 73, 325
Determinant, 27, 628, 659, 666
Deterministic problem, 91
Difference equation, 5
  linear, 5

nonlinear, 93, 237
uncertain systems, 203, 211
Difference inclusion, 203, 711
asymptotic stability, 150
discontinuous systems, 206
uncertain systems, 203
Differential algebraic equation, *see* DAE
Differential dynamic programming, *see* DDP
Differential equation, 91
Differential equations, 648
Differential states, 506
Differentiation
algorithmic, 516
numerical, 515
symbolic, 514
Direct collocation, 540, 588
Direct methods, 493
Direct multiple shooting, 534, 586, 589
Direct single shooting, 532, 586
Direct transcription methods, 538
Directional derivatives, 639
forward, 518
reverse, 518
Discrete actuators, 8, 160
Discrete algebraic Riccati equation, *see* DARE
Discretization, 531
Dissipativity, *see* Economic MPC
Distance
Hausdorff, set to set, 224, 339
point to set, 207, 208, 224
Distributed
gradient algorithm, 417
nonconvex optimization, 417
nonlinear cooperative control, 419
stability, 422
optimization, 426
state estimation, 399
target problem, 410
Distributed MPC, 363
disturbance models, 408
nonlinear, 414, 422
state estimation, 399
target problem, 410
zero offset, 411
Disturbances, 49
additive, 193, 224, 228
bounded, 336
integrating, 50
measurement, 269
process, 269
random, 198
stability, 712
Dot quantities, 518
DP, 14, 107, 195, 364, 367, 469, 729
backward, 14, 18
forward, 14, 33, 296
robust control, 214
Dual dynamic system, 677
Duality
of linear estimation and regulation, 290
strong, 184, 769
weak, 184, 769
Dynamic programming, *see* DP

Economic MPC, 153
asymptotic average performance, 155
asymptotic stability, 156
comparison with tracking MPC, 158
dissipativity, 156
strict dissipativity, 157, 160
EKF, 302–304
END, 525
Epigraph, 647
Equilibrium point, 694
Estimation, 26, 269, 349
convergence, 43
distributed, 399
duality with regulation, 290
full information, *see* FIE
least squares, 33

linear optimal, 29
moving horizon, *see* MHE
stability, 288
Euler integration method, 494, 497
Expectation, 655
Explicit MPC, 445
Exponential stability, *see* Stability
Extended Kalman filter, *see* EKF
External numerical differentiation, *see* END

Farkas's lemma, 453
Feasibility
recursive, 112, 132, 356
Feasible set, 487
Feedback control, 49, 195, 340
Feedback MPC, 200
Feedback particle filtering, 302
Feedforward control, 341
FIE, 269
Final-state observability, *see* FSO
Finite horizon, 21, 89
Floating point operation, *see* FLOP
FLOP, 367, 508, 560, 561
Forward mode, *see* AD
Fritz-John necessary conditions, 753
FSO, 294
Full information estimation, *see* FIE
Fundamental theorem of linear algebra, 23, 42, 625
existence, 23, 625
uniqueness, 42, 625

Game
M-player game, 412
constrained two-player, 400
cooperative, 386
noncooperative, 378
theory, 426
two-player nonconvex, 418
unconstrained two-player, 374
GAS, 112, 408, 433, 698
Gauss divergence theorem, 61
Gauss-Jacobi iteration, 380
Gauss-Legendre methods, *see* GL
Gauss-Newton Hessian, 548, 589
Gaussian distribution, *see* Normal density
Gaussian elimination, 508
Generalized Gauss-Newton method, 549
Generalized predictive control, *see* GPC
Generalized tangential predictors, 552, 570
GES, 698
GL, 505
Global error, 496
Global solutions, 741
Globalization techniques, 514
Globally asymptotically stable, *see* GAS
Globally exponentially stable, *see* GES
GPC, 167
Gramian
observability, 684
reachability, 683

Hamilton-Jacobi-Bellman equation, *see* HJB
Hausdorff metric, *see* Distance Hausdorff
Hautus lemma
controllability, 24
detectability, 72, 437, 441
observability, 42
stabilizability, 68
Hessian approximations, 547
BFGS, 550
Gauss-Newton, 548
secant condition, 550
update methods, 549
HJB, 493
Hurwitz matrix, 220, 706
Hyperplane, 472, 642, 643
support, 644
Hyperstate, 194, 333, 334

i-IOSS, 275, 285, 321, 323, 722
Implicit integrators, 500
Increasing, *see* Sequence
Incrementally, uniformly input/output-to-state-stable, *see* i-UIOSS
IND, 526
Independent, *see* Random variable
Indirect methods, 493
Infinite horizon, 21, 89
Initial-value embedding, 534
Initial-value problem, 495
Innovation, 194, 305, 334
Input-to-state-stability, *see* ISS
Input/output-to-state-stability, *see* IOSS
Integral control, *see* Offset-free control
Interior point methods, *see* IP
Internal model principle, 49
Internal numerical differentiation, *see* IND
Invariance
  control, 110
  positive, 110, 339, 694, 712
  robust control, 217
  robust positive, 212, 217, 313, 339, 350
  sequential control, 125
  sequential positive, 125, 707
IOSS, 121, 322, 323, 721
IP, 552, 580
IPOPT, 528, 554, 580
ISS, 718
i-UIOSS, 312, 325

$\mathcal{K}$ functions, 112, 275, 285, 694
  upper bounding, 709
$\mathcal{K}_\infty$ functions, 112, 694
$\mathcal{KL}$ functions, 112, 275, 285, 694
Kalman filter, *see* KF
Karush-Khun-Tucker conditions, *see* KKT
KF, 26, 33, 43, 51, 78, 79, 334
  extended, 306–311
  unscented, 304–311
KKT, 543, 755
  matrix, 546
  strongly regular, 545

L-stable integration methods, 505
Lagrange basis polynomials, 503
Lagrange multipliers, 66, 67, 365, 369, 430
Laplace transform, 3
LAR, 179, 475–476
LDLT-factorization, 508
  plain banded, 560
Least squares estimation, *see* Estimation
Leibniz formula, 61
Level set, 16, 137, 648
LICQ, 543, 755
Limit, *see* Sequence
Line search, 417, 514
Linear
  MPC, 131–139, 488
  quadratic MPC, 11, 99, 461–470
  space, 624
  subspace, 624
  system, 27, 131–139
Linear absolute regulator, *see* LAR
Linear independence constraint qualification, *see* LICQ
Linear multistep methods, 580
Linear optimal state estimation, *see* KF
Linear program, *see* LP
Linear quadratic Gaussian, *see* LQG
Linear quadratic problems, *see* LQP
Linear quadratic regulator, *see* LQR
Lipschitz continuous, 374, 406, 461, 495, 637, 761, 766
Local error, 496
Local solutions, 489, 741
Look-up table, 90
LP, 448, 451
  parametric, 470

LQG, 194, 335
LQP, 429, 430, 558
  condensing, 560
  Riccati recursion, 558
LQR, 11, 24, 364, 429, 430, 565, 736
  constrained, 461–470
  convergence, 24
  DP solution for constrained, 469
  infinite horizon, 21
  unconstrained, 132
LU-factorization, 508
Luenberger observer, 338
Lyapunov equation, 137, 706
Lyapunov function, 113, 701
  control, *see* CLF
  global, 208
  IOSS, 721
  ISS, 314, 718
  local, 239
  OSS, 720
Lyapunov stability, 370, 432
  uniform, 371
Lyapunov stability constraint, 404
Lyapunov stability theorem, 113, 700
  KL version, 703

M-player game
  constrained, 412
MATLAB, 22, 64, 65, 68, 508, 528
Mean value theorem, 638
Merit function, 514
MHE, 39, 292
  as conditional density, 40
  as least squares, 40
  combining with MPC, 312
  comparison with EKF and UKF, 306
  convergence, 296
  existence, 293
  nonzero prior weighting, 296
  zero prior weighting, 293
MILP, 575, 594
Min-max optimal control, 214
Minimum theorem, 760
Minkowski set subtraction, *see* Set algebra
MINLP, 575
MIQP, 575
Mixed continuous/discrete actuators, 162
Mixed-integer optimization, 161
Models, 1
  continuous time, 492
  deterministic, 2, 9
  discrete time, 5, 486
  distributed, 4
  disturbance, 49, 408
  input-output, 3
  linear dynamic, 2
  stochastic, 9
  time-invariant, 2, 10
  time-varying, 2
Monotonicity, 118, 435
Monte Carlo optimization, 223
Move blocking, 568
Moving horizon estimation, *see* MHE
MPCTools, vii, xii
Multipliers, 543
Multistage optimization, 12

Nash equilibrium, 382–386
Newton-Lagrange method, 546
Newton-Raphson method, 509
Newton-type methods, 507, 510
  local convergence, 511
Newton-type optimization with inequalities, 550
NLP, 534, 542
Noise, 10
  Gaussian, 287
  measurement, 10, 26, 269
  process, 26, 269
Nominal stability, *see* Stability
Nonconvex
  optimization problem, 487
Nonconvex optimization problem, 745

Nonconvexity, 166, 415
Noncooperative control, 363, 378
Nonlinear
  MPC, 139–144, 488
Nonlinear interior point methods, *see* IP
Nonlinear optimization, 542
Nonlinear program, *see* NLP
Nonlinear root-finding problems, 508
Norm, 631, 690, 696, 717
Normal cone, *see* Cone
Normal density, 27, 656
  conditional, 28, 674, 675
  degenerate, 661
  Fourier transform of, 658
  linear transformation, 28, 75
  multivariate, 659
  singular, 661
Normal distribution, *see* Normal density
Nullspace, 53, 624
Numerical differentiation, 515
  forward difference, 515
Numerical integration, 495
Numerical optimal control, 485

Observability, 41, 293, 722
  canonical form, 72
  duality with controllability, 291
  Gramian, 684
  matrix, 42
Observable, 41, 293
OCP, 490, 585, 586, 589, 592, 731
  continuous time, 492
  discrete time, 486, 555
Octave, 22, 64, 65, 68, 528
ODE, 495–507, 528
Offset-free control, 48–59
Offset-free MPC, 347
One-step integration methods, 497
Online optimization algorithms, 567
Open-loop control, 195
Optimal control problem, *see* OCP
Optimality conditions, 543, 737
  convex program, 453
  KKT, 543
  linear inequalities, 744
  nonconvex problems, 752
  normal cone, 742
  parametric LP, 472
  tangent cone, 743
Ordinary differential equation, *see* ODE
OSS, 321, 323, 719
Outer-bounding tube, *see* Tube
Output MPC, 312–318, 333
  stability, 314, 345
Output-to-state-stability, *see* OSS

Parameter, 97, 446
Parametric optimization, 97
Parametric programming, 97, 446
  computation, 476
  continuity of $V^0(\cdot)$ and $u^0(\cdot)$, 460
  linear, 470, 472, 473
  piecewise quadratic, 463
  quadratic, 451, 456, 458
Partial condensing, 562
Partial separability, 556
Particle filtering, 302
  feedback, 302
Partitioned matrix inversion theorem, 16, 65, 628
Peano's existence theorem, 651
Picard-Lindelöf theorem, 495
PID control, 49, 84
Pivoting, 508
Plantwide control, 363, 409, 418
  optimal, 376, 421
  subsystems, 364, 374, 414
Polyhedral, 446, 447, 450, 743, 761
Polytope, 203, 450, 461, 462, 464–466, 468, 761, 765, 766
Pontryagin set subtraction, *see* Set algebra
Positive definite, 121, 629, 695
Positive semidefinite, 121, 629

Principle of optimality, 734
Probability
  conditional density, 27, 672
  density, 27, 654
  distribution, 27, 654
  marginal density, 27, 659
  moments, 655
  multivariate density, 27, 659
  noninvertible transformations, 666
Projection, 97, 111, 447, 731, 756, 763, 765, 767
Proportional-integral-derivative, *see* PID control
Pseudo-inverse, 625
Pseudospectral method, 541
Python, 528

Q-convergence
  q-linearly, 511
  q-quadratically, 511
  q-superlinearly, 511
Q-function, 279, 281–283
QP, 100, 364, 437, 449, 451, 547
  parametric, 451
  parametric piecewise, 463
Quadratic
  piecewise, 104, 450, 452, 458, 463, 464, 468, 761
Quadratic program, *see* QP
Quadrature state, 532

Radau IIA collocation methods, 505, 540
Random variable, 654
  independent, 27
Range, 624
RAS, 229, 313
Reachability Gramian, 683
Real-time iterations, 573
Receding horizon control, *see* RHC
Recursive feasibility, *see* Feasibility
Recursive least squares, 38, 75
Reduced Hessian, 547
Region of attraction, *see* Attraction
Regularization, 206–209
Regulation, 89, 350
  combining with MHE, 312
  duality with estimation, 290
Relative gain array, *see* RGA
Reverse mode, *see* AD
RGA, 385
RGAS, 207, 272, 710
  convolution maximization form, 273
RGES, 285
RHC, 108, 109, 135, 163, 217
Riccati equation, 20, 68, 69, 71, 136, 291, 369
Riccati recursion, 558
RK, 498
  classical (RK4), 497
  explicit, 496
  implicit, 501
Robust min-max MPC, 220
Robust MPC, 193, 200
  min-max, 220
  tube-based, 223
Robustly asymptotically stable, *see* RAS
Robustly globally asymptotically stable, *see* RGAS
Robustly globally exponentially stable, *see* RGES
Robustness
  inherent, 204
  nominal, 204, 709
  of nominal MPC, 209
Runge-Kutta method, *see* RK

Scenario optimization, 254
Schur decomposition, 629
  real, 401, 630
Semicontinuity
  inner, 757
  outer, 757
Sequence, 632
  accumulation point, 632, 759

convergence, 632
limit, 632, 679, 680, 759
monotone, 632
nondecreasing, 44
nonincreasing, 25
subsequence, 632
Sequential optimal control, 491, 562
plain dense, 563
sparsity-exploiting, 563
Sequential quadratic programming, *see* SQP
Set
affine, 632
algebra, 224
boundary, 631
bounded, 631
closed, 631
compact, 631
complement, 631
interior, 631
level, 16, 137
open, 631
quasiregular, 751
regular, 749
relative interior, 632
sublevel, 137
Set-valued function, 99, 472, 755–757
Setpoint
nonzero, 46, 349
Shift initialization, 571
Short horizon syndrome, 311
Sigma points, 305
Simultaneous optimal control, 490
Singular-value decomposition, *see* SVD
Space
linear, 624
vector, 624
Sparsity, 491
SQP, 551, 589
feasibility perturbed, 566
local convergence, 552
Stability, 112
asymptotic, 112, 423, 698
constrained, 699
exponential, 120, 698
global, 112, 698
global asymptotic, 112, 126, 408, 433, 698
global asymptotic (KL version), 699
global attractive, 112
global exponential, 120, 698
inherent, 91
local, 112, 698
nominal, 91
robust asymptotic, *see* RAS
robust exponential, 235
robust global asymptotic, *see* RGAS
time-varying systems, 125
with disturbances, 712
Stabilizability, 68, 120
duality with detectability, 291
Stabilizable, 26, 46, 68, 73, 136, 140, 714
Stage cost, 18, 153
economic, 153
State estimation, *see* Estimation
Statistical independence, 668
Steady-state target, 48, 352
distributed, 410
Stiff equations, 500
Stochastic MPC, 193, 200, 246
stabilizing conditions, 248
tightened constraints, 253
tube-based, 250
Storage function, 156
Strong duality, *see* Duality
Subgradient, 640
convex function, 762
Sublevel set, 137, 648
Suboptimal MPC, 147, 369
asymptotic stability, 151
distributed, 369
exponential stability, 372
Subspace

linear, 624
Supply rate, 156
Support function, 648
SVD, 627
System
composite, 343, 345
deterministic, 9, 196, 333
discontinuous, 206
linear, 2, 133, 224, 228, 338
noisy, 269
nominal, 238
nonlinear, 2, 93, 123, 139, 236
periodic, 133
time-invariant, 3, 5, 10, 93, 338
time-varying, 2, 123, 141, 347, 437
uncertain, 193, 195, 196, 333, 334, 338

Tangent cone, *see* Cone
Taylor series, 3, 64
Taylor's theorem, 143
Terminal constraint, *see* Constraints
Terminal region, 93
Time to go, 108, 109, 196, 217, 469
Trace, 73, 681
Tracking, 46
periodic target, 142
Transfer function, 4, 6, 179, 383
Trust region, 514
Tube, 202, 335
bounding, 226
outer-bounding, 224
Tube-based robust MPC, 223
feedback controller, 228
improved, 234
linear systems, 228
model predictive controller, 238
nominal controller, 228
nominal trajectory, 238
nonlinear systems, 236
tightened constraints, 230, 242
Two-player game
constrained, 400
coupled input constraints, 405
unconstrained, 374
uncoupled input constraints, 402

UKF, 304–306
Uncertainty, 193
parametric, 194
Uncontrollable, 22
Unit ball, 631
Unscented Kalman filter, *see* UKF

Value function, 13, 92, 204, 240
continuity, 104, 208, 759
discontinuity, 104
Lipschitz continuity, 760
Variable
controlled, 47
disturbance, 49
dual, 543
input, 2
output, 2
primal, 543
random, 27, 654
independent, 27, 654
state, 2
Vertex, 471

Warm start, 148, 183, 221, 370, 391, 404, 413, 433, 555
shift initialization, 571
Weak controllability, *see* Controllability
Weak duality, *see* Duality
Weierstrass theorem, 97, 99, 294, 372, 636

Z-transform, 5

Note: Appendices A, B, and C can be found at www.chemengr.ucsb.edu/~jbraw/mpc

A
Mathematical Background

Version: date: April 27, 2026
Copyright © 2026 by Nob Hill Publishing, LLC

A.1 Introduction

In this appendix we give a brief review of some concepts that we need. It is assumed that the reader has had at least a first course on linear systems and has some familiarity with linear algebra and analysis. The appendices of Polak (1997); Nocedal and Wright (2006); Boyd and Vandenberghe (2004) provide useful summaries of the results we require. The material presented in Sections A.2–A.14 follows closely Polak (1997) and earlier lecture notes of Professor Polak.

A.2 Vector Spaces

The Euclidean space $\mathbb{R}^n$ is an example of a vector space that satisfies a set of axioms the most significant being: if x and z are two elements of a vector space $\mathcal{V}$, then $\alpha x + \beta z$ is also an element of $\mathcal{V}$ for all α, $\beta \in \mathbb{R}$. This definition presumes addition of two elements of $\mathcal{V}$ and multiplication of any element of $\mathcal{V}$ by a scalar are defined. Similarly, $S \subset \mathcal{V}$ is a linear subspace[1] of $\mathcal{V}$ if any two elements of x and z of S satisfy $\alpha x + \beta z \in S$ for all $\alpha, \beta \in \mathbb{R}$. Thus, in $\mathbb{R}^3$, the origin, a line or a plane passing through the origin, the whole set $\mathbb{R}^3$, and even the empty set are all subspaces.

A.3 Range and Nullspace of Matrices

Suppose $A \in \mathbb{R}^{m \times n}$. Then $\mathcal{R}(A)$, the *range* of A, is the set $\{Ax \mid x \in \mathbb{R}^n\}$; $\mathcal{R}(A)$ is a subspace of $\mathbb{R}^m$ and its dimension, i.e., the number of linearly independent vectors that span $\mathcal{R}(A)$, is the rank of A. For

[1] All of the subspaces used in this text are linear subspaces, so we often omit the adjective linear.

example, if A is the column vector $\begin{bmatrix} 1 \\ 1 \end{bmatrix}$, then $\mathcal{R}(A)$ is the subspace spanned by the vector $\begin{bmatrix} 1 \\ 1 \end{bmatrix}$ and the rank of A is 1. The *nullspace* $\mathcal{N}(A)$ is the set of vectors in $\mathbb{R}^n$ that are mapped to zero by A so that $\mathcal{N}(A) = \{x \mid Ax = 0\}$. The nullspace $\mathcal{N}(A)$ is a subspace of $\mathbb{R}^n$. For the example above, $\mathcal{N}(A)$ is the subspace spanned by the vector $\begin{bmatrix} 1 \\ -1 \end{bmatrix}$. It is an important fact that $\mathcal{R}(A') \oplus \mathcal{N}(A) = \mathbb{R}^n$ or, equivalently, that $\mathcal{N}(A) = (\mathcal{R}(A'))^\perp$ where $A' \in \mathbb{R}^{n \times m}$ is the transpose of A and $S^\perp$ denotes the orthogonal complement of any subspace S; a consequence is that the sum of the dimensions $\mathcal{R}(A)$ and $\mathcal{N}(A)$ is n. If A is square and invertible, then $n = m$ and the dimension of $\mathcal{R}(A)$ is n so that the dimension of $\mathcal{N}(A)$ is 0, i.e., the nullspace contains only the zero vector, $\mathcal{N}(A) = \{0\}$.

A.4 Linear Equations — Existence and Uniqueness

Let $A \in \mathbb{R}^{m \times n}$ be a real-valued matrix with m rows and n columns. We are often interested in solving linear equations of the type

$$Ax = b$$

in which $b \in \mathbb{R}^m$ is given, and $x \in \mathbb{R}^n$ is the unknown. The fundamental theorem of linear algebra gives a complete characterization of the existence and uniqueness of solutions to $Ax = b$ (Strang, 1980, pp.87–88). Every matrix A decomposes the spaces $\mathbb{R}^n$ and $\mathbb{R}^m$ into the four fundamental subspaces depicted in Figure A.1. A solution to $Ax = b$ *exists* for every b if and only if the *rows* of A are linearly independent. A solution to $Ax = b$ is *unique* if and only if the *columns* of A are linearly independent.

A.5 Pseudo-Inverse

The solution of $Ax = y$ when A is invertible is $x = A^{-1}y$ where A^{-1} is the inverse of A. Often an approximate inverse of $y = Ax$ is required when A is *not* invertible. This is yielded by the pseudo-inverse $A^\dagger$ of A; if $A \in \mathbb{R}^{m \times n}$, then $A^\dagger \in \mathbb{R}^{n \times m}$. The properties of the pseudo-inverse are illustrated in Figure A.2 for the case when $A \in \mathbb{R}^{2 \times 2}$ where both $\mathcal{R}(A)$ and $\mathcal{N}(A)$ have dimension 1. Suppose we require a solution to the equation $Ax = y$. Since every $x \in \mathbb{R}^2$ is mapped into $\mathcal{R}(A)$, we see that a solution may only be obtained if $y \in \mathcal{R}(A)$. Suppose this is not the case, as in Figure A.2. Then the closest point, in the Euclidean sense, to y in $\mathcal{R}(A)$ is the point y^* which is the orthogonal projection

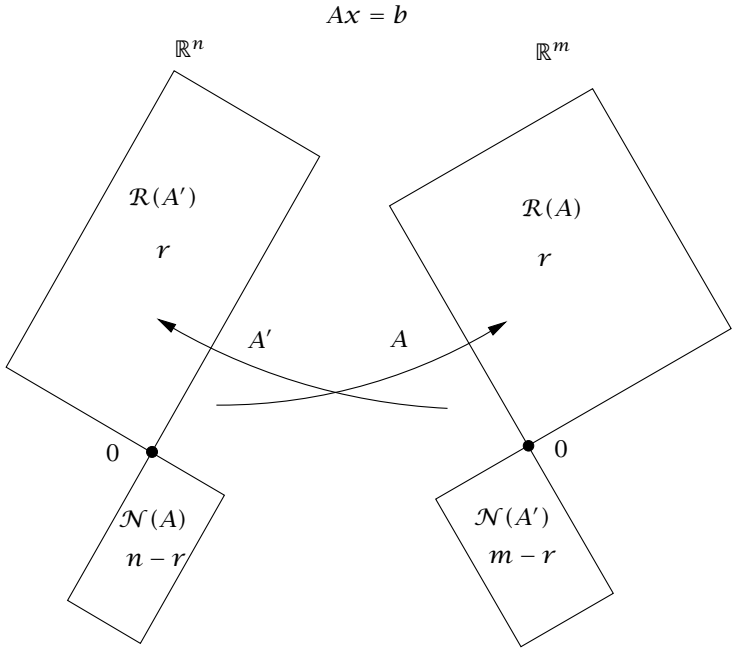


Figure A.1: The four fundamental subspaces of matrix A (after (Strang, 1980, p.88)). The dimension of the range of A and A' is r, the rank of matrix A. The nullspace of A and range of A' are orthogonal as are the nullspace of A' and range of A. Solutions to $Ax = b$ exist for all b if and only if $m = r$ (rows independent). A solution to $Ax = b$ is unique if and only if $n = r$ (columns independent).

of y onto $\mathcal{R}(A)$, i.e., $y - y^*$ is orthogonal to $\mathcal{R}(A)$. Since $y^* \in \mathcal{R}(A)$, there exists a point in $\mathbb{R}^2$ that A maps into y^*. Now A maps any point of the form $x + h$ where $h \in \mathcal{N}(A)$ into $A(x + h) = Ax + Ah = Ax$ so that there must exist a point $x^* \in (\mathcal{N}(A))^\perp = \mathcal{R}(A')$ such that $Ax^* = y^*$, as shown in Figure A.2. All points of the form $x = x^* + h$ where $h \in \mathcal{N}(A)$ are also mapped into y^*; x^* is the point of least norm that satisfies $Ax^* = y^*$ where y^* is that point in $\mathcal{R}(A)$ closest, in the Euclidean sense, to y.

The pseudo-inverse $A^\dagger$ of a matrix $A \in \mathbb{R}^{m \times n}$ is a matrix in $\mathbb{R}^{n \times m}$ that maps every $y \in \mathbb{R}^m$ to that point $x \in \mathcal{R}(A')$ of least Euclidean norm that minimizes $|y - Ax|_2$. The operation of $A^\dagger$ is illustrated in

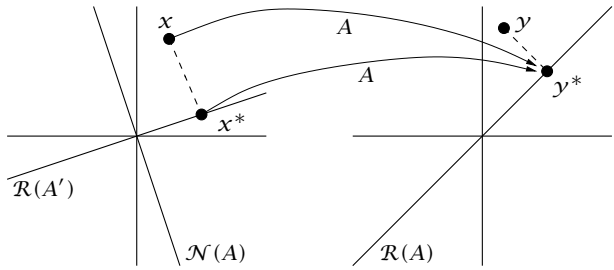


Figure A.2: Matrix A maps into $\mathcal{R}(A)$.

Figure A.3. Hence $AA^\dagger$ projects any point $y \in \mathbb{R}^m$ orthogonally onto $\mathcal{R}(A)$, i.e., $AA^\dagger y = y^*$, and $A^\dagger A$ projects any $x \in \mathbb{R}^n$ orthogonally onto $\mathcal{R}(A')$, i.e., $A^\dagger A x = x^*$.

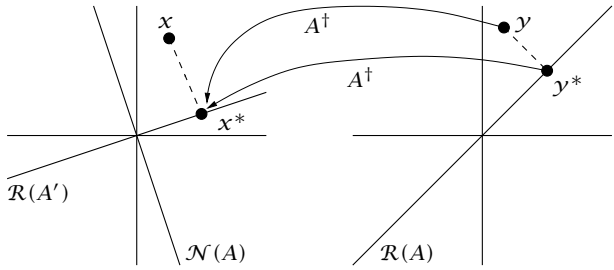


Figure A.3: Pseudo-inverse of A maps into $\mathcal{R}(A')$.

If $A \in \mathbb{R}^{m\times n}$ where $m < n$ has maximal rank m, then $AA' \in \mathbb{R}^{m\times m}$ is invertible and $A^\dagger = A'(AA')^{-1}$; in this case, $\mathcal{R}(A) = \mathbb{R}^m$ and every $y \in \mathbb{R}^m$ lies in $\mathcal{R}(A)$. Similarly, if $n < m$ and A has maximal rank n, then $A'A \in \mathbb{R}^{n\times n}$ is invertible and $A^\dagger = (A'A)^{-1}A'$; in this case, $\mathcal{R}(A') = \mathbb{R}^n$ and every $x \in \mathbb{R}^n$ lies in $\mathcal{R}(A')$. More generally, if $A \in \mathbb{R}^{m\times n}$ has rank r, then A has the *singular-value decomposition* $A = U\Sigma V'$ where $U \in \mathbb{R}^{m\times r}$ and $V \in \mathbb{R}^{r\times n}$ are orthogonal matrices, i.e., $U'U = I_r$ and $V'V = I_r$, and $\Sigma = \mathrm{diag}(\sigma_1, \sigma_2, \ldots, \sigma_r) \in \mathbb{R}^{r\times r}$ where $\sigma_1 > \sigma_2 > \cdots > \sigma_r > 0$. The pseudo-inverse of A is then

$$A^\dagger = V\Sigma^{-1}U'$$

A.6 Partitioned Matrix Inversion Theorem

Let matrix Z be partitioned into

$$Z = \begin{bmatrix} B & C \\ D & E \end{bmatrix}$$

and assume Z^{-1}, B^{-1} and E^{-1} exist. Performing row elimination gives

$$Z^{-1} = \begin{bmatrix} B^{-1} + B^{-1}C(E - DB^{-1}C)^{-1}DB^{-1} & -B^{-1}C(E - DB^{-1}C)^{-1} \\ -(E - DB^{-1}C)^{-1}DB^{-1} & (E - DB^{-1}C)^{-1} \end{bmatrix}$$

Note that this result is still valid if E is singular. Performing column elimination gives

$$Z^{-1} = \begin{bmatrix} (B - CE^{-1}D)^{-1} & -(B - CE^{-1}D)^{-1}CE^{-1} \\ -E^{-1}D(B - CE^{-1}D)^{-1} & E^{-1} + E^{-1}D(B - CE^{-1}D)^{-1}CE^{-1} \end{bmatrix}$$

Note that this result is still valid if B is singular. A host of other useful control-related inversion formulas follow from these results. Equating the (1,1) or (2,2) entries of Z^{-1} gives the identity

$$(A + BCD)^{-1} = A^{-1} - A^{-1}B(DA^{-1}B + C^{-1})^{-1}DA^{-1}$$

A useful special case of this result is

$$(I + X^{-1})^{-1} = I - (I + X)^{-1}$$

Equating the (1,2) or (2,1) entries of Z^{-1} gives the identity

$$(A + BCD)^{-1}BC = A^{-1}B(DA^{-1}B + C^{-1})^{-1}$$

Determinants. We require some results on determinants of partitioned matrices when using normal distributions in the discussion of probability. If E is nonsingular

$$\det(A) = \det(E)\det(B - CE^{-1}D)$$

If B is nonsingular

$$\det(A) = \det(B)\det(E - DB^{-1}C)$$

A.7 Quadratic Forms

Positive definite and positive semidefinite matrices show up often in LQ problems. Here are some basic facts about them. In the following Q is real and symmetric and R is real.

The matrix Q is positive definite ($Q > 0$), if

$$x'Qx > 0, \quad \forall \text{ nonzero } x \in \mathbb{R}^n$$

The matrix Q is positive semidefinite ($Q \geq 0$), if

$$x'Qx \geq 0, \quad \forall x \in \mathbb{R}^n$$

You should be able to prove the following facts.

1. $Q > 0$ if and only if $\lambda > 0, \quad \lambda \in \text{eig}(Q)$.
2. $Q \geq 0$ if and only if $\lambda \geq 0, \quad \lambda \in \text{eig}(Q)$.
3. $Q \geq 0 \Rightarrow R'QR \geq 0 \quad \forall R$.
4. $Q > 0$ and R nonsingular $\Rightarrow R'QR > 0$.
5. $Q > 0$ and R full column rank $\Rightarrow R'QR > 0$.
6. $Q_1 > 0, Q_2 \geq 0 \Rightarrow Q = Q_1 + Q_2 > 0$.
7. $Q > 0 \Rightarrow z^*Qz > 0 \quad \forall$ nonzero $z \in \mathbb{C}^n$.
8. Given $Q \geq 0$, $x'Qx = 0$ if and only if $Qx = 0$.

You may want to use the Schur decomposition (Schur, 1909) of a matrix in establishing some of these eigenvalue results. Golub and Van Loan (1996, p.313) provide the following theorem

Theorem A.1 (Schur decomposition)**.** *If $A \in \mathbb{C}^{n\times n}$ then there exists a unitary $Q \in \mathbb{C}^{n\times n}$ such that*

$$Q^*AQ = T$$

in which T is upper triangular.

Note that because T is upper triangular, its diagonal elements are the eigenvalues of A. Even if A is a real matrix, T can be complex because the eigenvalues of a real matrix may come in complex conjugate pairs. Recall a matrix Q is unitary if $Q^*Q = I$. You should also be able to prove the following facts (Horn and Johnson, 1985).

1. If $A \in \mathbb{C}^{n\times n}$ and $BA = I$ for some $B \in \mathbb{C}^{n\times n}$, then
   (a) A is nonsingular
   (b) B is unique
   (c) $AB = I$
2. The matrix Q is unitary if and only if
   (a) Q is nonsingular and $Q^* = Q^{-1}$
   (b) $QQ^* = I$
   (c) Q^* is unitary
   (d) The rows of Q form an orthonormal set
   (e) The columns of Q form an orthonormal set
3. If A is real and symmetric, then T is real and diagonal and Q can be chosen real and orthogonal. It does not matter if the eigenvalues of A are repeated.

For real, but not necessarily symmetric, A you can restrict yourself to real matrices, by using the real Schur decomposition (Golub and Van Loan, 1996, p.341), but the price you pay is that you can achieve only block upper triangular T, rather than strictly upper triangular T.

Theorem A.2 (Real Schur decomposition)**.** *If $A \in \mathbb{R}^{n\times n}$ then there exists an orthogonal $Q \in \mathbb{R}^{n\times n}$ such that*

$$Q'AQ = \begin{bmatrix} R_{11} & R_{12} & \cdots & R_{1m} \\ 0 & R_{22} & \cdots & R_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & R_{mm} \end{bmatrix}$$

in which each R_{ii} is either a real scalar or a 2×2 real matrix having complex conjugate eigenvalues; the eigenvalues of R_{ii} are the eigenvalues of A.

If the eigenvalues of R_{ii} are disjoint (i.e., the eigenvalues are not repeated), then R can be taken block diagonal instead of block triangular (Golub and Van Loan, 1996, p.366).

A.8 Norms in $\mathbb{R}^n$

A norm in $\mathbb{R}^n$ is a function $|\cdot| : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ such that

(a) $|x| = 0$ if and only if $x = 0$;

(b) $|\lambda x| = |\lambda| \ |x|$, for all $\lambda \in \mathbb{R}, x \in \mathbb{R}^n$;

(c) $|x + y| \leq |x| + |y|$, for all $x, y \in \mathbb{R}^n$.

Let $\mathcal{B} := \{x \mid |x| \leq 1\}$ denote the *closed* ball of radius 1 centered at the origin. For any $x \in \mathbb{R}^n$ and $\rho > 0$, we denote by $x \oplus \rho\mathcal{B}$ or $B(x, \rho)$ the *closed* ball $\{z \mid |z - x| \leq \rho\}$ of radius ρ centered at x. Similarly $\{x \mid |x| < 1\}$ denotes the *open* ball of radius 1 centered at the origin and $\{z \mid |z - x| < \rho\}$ the *open* ball of radius ρ centered at x; closed and open sets are defined below.

A.9 Sets in $\mathbb{R}^n$

The complement of $S \subset \mathbb{R}^n$ in $\mathbb{R}^n$, is the set $S^c := \{x \in \mathbb{R}^n \mid x \notin S\}$. A set $X \subset \mathbb{R}^n$ is said to be *open*, if for every $x \in X$, there exists a $\rho > 0$ such that $B(x, \rho) \subseteq X$. A set $X \subset \mathbb{R}^n$ is said to be *closed* if X^c, its complement in $\mathbb{R}^n$, is open.

A set $X \subset \mathbb{R}^n$ is said to be *bounded* if there exists an $M < \infty$ such that $|x| \leq M$ for all $x \in X$. A set $X \subset \mathbb{R}^n$ is said to be *compact* if X is closed and bounded. An element $x \in S \subseteq \mathbb{R}^n$ is an *interior* point of the set S if there exists a $\rho > 0$ such that $z \in S$, for all $|z - x| < \rho$. The interior of a set $S \subset \mathbb{R}^n$, $\text{int}(S)$, is the set of all interior points of S; $\text{int}(S)$ is an open set, the *largest*[2] open subset of S. For example, if $S = [a, b] \subset \mathbb{R}$, then $\text{int}(S) = (a, b)$; as another example, $\text{int}(B(x, \rho)) = \{z \mid |z - x| < \rho\}$. The closure of a set $S \subset \mathbb{R}^n$, denoted $\bar{S}$, is the *smallest*[3] closed set containing S. For example, if $S = (a, b] \subset \mathbb{R}$, then $\bar{S} = [a, b]$. The boundary of $S \subset \mathbb{R}^n$, is the set $\delta S := \bar{S} \setminus \text{int}(S) = \{s \in \bar{S} \mid s \notin \text{int}(S)\}$. For example, if $S = (a, b] \subset \mathbb{R}$, then $\text{int}(S) = (a, b), \bar{S} = [a, b], \partial S = \{a, b\}$.

An *affine* set $S \subset \mathbb{R}^n$ is a set that can be expressed in the form $S = \{x\} \oplus \mathcal{V} := \{x + v \mid v \in \mathcal{V}\}$ for some $x \in \mathbb{R}^n$ and some subspace $\mathcal{V}$ of $\mathbb{R}^n$. An example is a line in $\mathbb{R}^n$ not passing through the origin. The *affine hull* of a set $S \subset \mathbb{R}^n$, denoted $\text{aff}(S)$, is the smallest[4] affine

[2] Largest in the sense that every open subset of S is a subset of $\text{int}(S)$.

[3] Smallest in the sense that $\bar{S}$ is a subset of any closed set containing S.

[4] In the sense that $\text{aff}(S)$ is a subset of any other affine set containing S.

set that contains S. That is equivalent to the intersection of all affine sets containing S.

Some sets S, such as a line in $\mathbb{R}^n, n \geq 2$, do not have an interior, but do have an interior *relative* to the smallest affine set in which S lies, which is $\text{aff}(S)$ defined above. The *relative interior* of S is the set $\{x \in S \mid \exists \rho > 0 \text{ such that } \text{int}(B(x,\rho)) \cap \text{aff}(S) \subset S\}$. Thus the line segment, $S := \{x \subset \mathbb{R}^2 \mid x = \lambda \begin{bmatrix} 1 \\ 0 \end{bmatrix} + (1-\lambda) \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \lambda \in [0,1]\}$ does not have an interior, but does have an interior relative to the line containing it, $\text{aff}(S)$. The relative interior of S is the open line segment $\{x \in \mathbb{R}^2 \mid x = \lambda \begin{bmatrix} 1 \\ 0 \end{bmatrix} + (1-\lambda) \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \lambda \in (0,1)\}$.

A.10 Sequences

Let the set of nonnegative integers be denoted by $\mathbb{I}_{\geq 0}$. A *sequence* is a function from $\mathbb{I}_{\geq 0}$ into $\mathbb{R}^n$. We denote a sequence by its values, $(x_i)_{i \in \mathbb{I}_{\geq 0}}$. A *subsequence* of $(x_i)_{i \in \mathbb{I}_{\geq 0}}$ is a sequence of the form $(x_i)_{i \in K}$, where K is an infinite subset of $\mathbb{I}_{\geq 0}$.

A sequence $(x_i)_{i \in \mathbb{I}_{\geq 0}}$ in $\mathbb{R}^n$ is said to *converge* to a point $\hat{x}$ if $\lim_{i \to \infty} |x_i - \hat{x}| = 0$, i.e., if, for all $\delta > 0$, there exists an integer k such that $|x_i - \hat{x}| \leq \delta$ for all $i \geq k$; we write $x_i \to \hat{x}$ as $i \to \infty$ to denote the fact that the sequence (x_i) converges to $\hat{x}$. The point $\hat{x}$ is called a *limit* of the sequence (x_i). A point x^* is said to be an *accumulation* point of a sequence $(x_i)_{i \in \mathbb{I}_{\geq 0}}$ in $\mathbb{R}^n$, if there exists an infinite subset $K \subset \mathbb{I}_{\geq 0}$ such that $x_i \to x^*$ as $i \to \infty, i \in K$ in which case we say $x_i \overset{K}{\to} x^*$.[5]

Let (x_i) be a bounded infinite sequence in $\mathbb{R}$ and let the S be the set of all accumulation points of (x_i). Then S is compact and $\limsup x_i$ is the largest and $\liminf x_i$ the smallest accumulation point of (x_i):

$$\limsup_{i \to \infty} x_i := \max\{x \mid x \in S\}, \text{ and}$$
$$\liminf_{i \to \infty} x_i := \min\{x \mid x \in S\}$$

Theorem A.3 (Bolzano-Weierstrass)**.** *Suppose $X \subset \mathbb{R}^n$ is compact and $(x_i)_{i \in \mathbb{I}_{\geq 0}}$ takes its values in X. Then $(x_i)_{i \in \mathbb{I}_{\geq 0}}$ must have at least one accumulation point.*

From Exercise A.7, it follows that the accumulation point postulated by Theorem A.3 lies in X. In proving asymptotic stability we need the following property of monotone sequences.

[5] Be aware of inconsistent usage of the term *limit point.* Some authors use limit point as synonymous with limit. Others use limit point as synonymous with accumulation point. For this reason we avoid the term limit point.

Proposition A.4 (Convergence of monotone sequences)**.** *Suppose that* $(x_i)_{i\in\mathbb{I}_{\geq 0}}$ *is a sequence in* $\mathbb{R}$ *such that* $x_0 \geq x_1 \geq x_2 \geq \ldots$*, i.e., suppose the sequence is monotone nonincreasing. If* (x_i) *has an accumulation point* x^**, then* $x_i \to x^*$ *as* $i \to \infty$*, i.e.,* x^* *is a limit.*

Proof. For the sake of contradiction, suppose that $(x_i)_{i\in\mathbb{I}_{\geq 0}}$ does not converge to x^*. Then, for some $\rho > 0$, there exists a subsequence $(x_i)_{i\in K}$ such that $x_i \notin B(x^*, \rho)$ for all $i \in K$, i.e., $|x_i - x^*| > \rho$ for all $i \in K$. Since x^* is an accumulation point, there exists a subsequence $(x_i)_{i\in K^*}$ such that $x_i \overset{K^*}{\to} x^*$. Hence there is an $i_1 \in K^*$ such that $|x_i - x^*| \leq \rho/2$, for all $i \geq i_1, i \in K^*$. Let $i_2 \in K$ be such that $i_2 > i_1$. Then we must have that $x_{i_2} \leq x_{i_1}$ and $|x_{i_2} - x^*| > \rho$, which leads to the conclusion that $x_{i_2} < x^* - \rho$. Now let $i_3 \in K^*$ be such that $i_3 > i_2$. Then we must have that $x_{i_3} \leq x_{i_2}$ and hence that $x_{i_3} < x^* - \rho$ which implies that $|x_{i_3} - x^*| > \rho$. But this contradicts the fact that $|x_{i_3} - x^*| \leq \rho/2$, and hence we conclude that $x_i \to x^*$ as $i \to \infty$. ■

It follows from Proposition A.4 that if $(x_i)_{i\in\mathbb{I}_{\geq 0}}$ is a monotone decreasing sequence in $\mathbb{R}$ bounded below by b, then the sequence $(x_i)_{i\in\mathbb{I}_{\geq 0}}$ converges to some $x^* \in \mathbb{R}$ where $x^* \geq b$.

A.11 Continuity

We now summarize some essential properties of continuous functions.

1. A function $f : \mathbb{R}^n \to \mathbb{R}^m$ is said to be *continuous at a point* $x \in \mathbb{R}^n$, if for every $\delta > 0$ there exists a $\rho > 0$ such that

$$|f(x') - f(x)| < \delta \ \forall x' \in \text{int}(B(x, \rho))$$

   A function $f : \mathbb{R}^n \to \mathbb{R}^m$ is said to be *continuous* if it is continuous at all $x \in \mathbb{R}^n$.

2. Let X be a closed subset of $\mathbb{R}^n$. A function $f : X \to \mathbb{R}^m$ is said to be *continuous at a point* x in X if for every $\delta > 0$ there exists a $\rho > 0$ such that

$$|f(x') - f(x)| < \delta \ \forall x' \in \text{int}(B(x, \rho)) \cap X$$

   A function $f : \mathbb{R}^n \to \mathbb{R}^m$ is said to be *continuous* on X if it is continuous at all x in X.

3. A function $f : \mathbb{R}^n \to \mathbb{R}^m$ is said to be *upper semicontinuous at a point* $x \in \mathbb{R}^n$, if for every $\delta > 0$ there exists a $\rho > 0$ such that

$$f(x') - f(x) < \delta \ \forall x' \in \text{int}(B(x, \rho))$$

A function $f : \mathbb{R}^n \to \mathbb{R}^m$ is said to be *upper semicontinuous* if it is upper semicontinuous at all $x \in \mathbb{R}^n$.

4. A function $f : \mathbb{R}^n \to \mathbb{R}^m$ is said to be *lower semicontinuous at a point* $x \in \mathbb{R}^n$, if for every $\delta > 0$ there exists a $\rho > 0$ such that

$$f(x') - f(x) > -\delta \ \forall x' \in \text{int}(B(x, \rho))$$

A function $f : \mathbb{R}^n \to \mathbb{R}^m$ is said to be *lower semicontinuous* if it is lower semicontinuous at all $x \in \mathbb{R}^n$.

5. A function $f : \mathbb{R}^n \to \mathbb{R}^m$ is said to be *uniformly continuous* on a subset $X \subset \mathbb{R}^n$ if for any $\delta > 0$ there exists a $\rho > 0$ such that for any $x', x'' \in X$ satisfying $|x' - x''| < \rho$,

$$\left| f(x') - f(x'') \right| < \delta$$

Proposition A.5 (Uniform continuity)**.** *Suppose that $f : \mathbb{R}^n \to \mathbb{R}^m$ is continuous and that $X \subset \mathbb{R}^n$ is compact. Then f is uniformly continuous on X.*

Proof. For the sake of contradiction, suppose that f is *not* uniformly continuous on X. Then, for some $\delta > 0$, there exist sequences $\left(x_i'\right)$, $\left(x_i''\right)$ in X such that

$$\left| x_i' - x_i'' \right| < (1/i), \text{ for all } i \in \mathbb{I}_{\geq 0}$$

but

$$\left| f(x_i') - f(x_i'') \right| > \delta, \text{ for all } i \in \mathbb{I}_{\geq 0} \tag{A.1}$$

Since X is compact, there must exist a subsequence $\left(x_i'\right)_{i \in K}$ such that $x_i' \overset{K}{\to} x^* \in X$ as $i \to \infty$. Furthermore, because of (A.1), $x_i'' \overset{K}{\to} x^*$ also holds. Hence, since $f(\cdot)$ is continuous, we must have $f(x_i') \overset{K}{\to} f(x^*)$ and $f(x_i'') \overset{K}{\to} f(x^*)$. Therefore, there exists a $i_0 \in K$ such that for all $i \in K, i \geq i_0$

$$\left| f(x_i') - f(x_i'') \right| \leq \left| f(x_i') - f(x^*) \right| + \left| f(x^*) - f(x_i'') \right| < \delta/2$$

contradicting (A.1). This completes our proof. ∎

Proposition A.6 (Compactness of continuous functions of compact sets)**.** *Suppose that $X \subset \mathbb{R}^n$ is compact and that $f : \mathbb{R}^n \to \mathbb{R}^m$ is continuous. Then the set*

$$f(X) \coloneqq \{f(x) \mid x \in X\}$$

is compact.

Proof.

(a) First we show that $f(X)$ is closed. Thus, let $(f(x_i) \mid i \in \mathbb{I}_{\geq 0})$, with $x_i \in X$, be any sequence in $f(X)$ such that $f(x_i) \to y$ as $i \to \infty$. Since (x_i) is in a compact set X, there exists a subsequence $(x_i)_{i \in K}$ such that $x_i \overset{K}{\to} x^* \in X$ as $i \to \infty$. Since $f(\cdot)$ is continuous, $f(x_i) \overset{K}{\to} f(x^*)$ as $i \to \infty$. But y is the limit of $(f(x_i))_{i \in \mathbb{I}_{\geq 0}}$ and hence it is the limit of any subsequence of $(f(x_i))$. We conclude that $y = f(x^*)$ and hence that $y \in f(X)$, i.e., $f(X)$ is closed.

(b) Next, we prove that $f(X)$ is bounded. Suppose $f(X)$ is not bounded. Then there exists a sequence (x_i) such that $|f(x_i)| \geq i$ for all $i \in \mathbb{I}_{\geq 0}$. Now, since (x_i) is in a compact set, there exists a subsequence $(x_i)_{i \in K}$ such that $x_i \overset{K}{\to} x^*$ with $x^* \in X$, and $f(x_i) \overset{K}{\to} f(x^*)$ by continuity of $f(\cdot)$. Hence there exists an i_0 such that for any $j > i > i_0$, $j, i \in K$

$$\left|f(x_j) - f(x_i)\right| \leq \left|f(x_j) - f(x^*)\right| + \left|f(x_i) - f(x^*)\right| < 1/2 \quad \text{(A.2)}$$

Let $i \geq i_0$ be given. By hypothesis there exists a $j \in K$, $j \geq i$ such that $\left|f(x_j)\right| \geq j \geq |f(x_i)| + 1$. Hence

$$\left|f(x_j) - f(x_i)\right| \geq \left|\, \left|f(x_j)\right| - |f(x_i)| \,\right| \geq 1$$

which contradicts (A.2). Thus $f(X)$ must be bounded, which completes the proof. ∎

Let $Y \subset \mathbb{R}$. Then $\inf(Y)$, the *infimum* of Y, is defined to be the greatest lower bound[6] of Y. If $\inf(Y) \in Y$, then $\min(Y) \coloneqq \min\{y \mid y \in Y\}$, the minimum of the set Y, exists and is equal to $\inf(Y)$. The infimum of a set Y always exists if Y is not empty and is bounded from below, in which case there always exist sequences $(y_i) \in Y$ such that $y_i \searrow \beta \coloneqq \inf(Y)$ as $i \to \infty$. Note that $\beta \coloneqq \inf(Y)$ does not necessarily lie in the set Y.

[6]The value $\alpha \in \mathbb{R}$ is the greatest lower bound of Y if $y \geq \alpha$ for all $y \in Y$, and $\beta > \alpha$ implies that β is *not* a lower bound for Y.

Proposition A.7 (Weierstrass)**.** *Suppose that $f : \mathbb{R}^n \to \mathbb{R}$ is continuous and that $X \subset \mathbb{R}^n$ is compact. Then there exists an $\hat{x} \in X$ such that*

$$f(\hat{x}) = \inf_{x \in X} f(x)$$

i.e., $\min_{x \in X} f(x)$ *is well defined.*

Proof. Since X is compact, $f(X)$ is bounded. Hence $\inf_{x \in X} f(x) = \alpha$ is finite. Let (x_i) be an infinite sequence in X such that $f(x_i) \searrow \alpha$ as $i \to \infty$. Since X is compact, there exists a converging subsequence $(x_i)_{i \in K}$ such that $x_i \overset{K}{\to} \hat{x} \in X$. By continuity, $f(x_i) \overset{K}{\to} f(\hat{x})$ as $i \to \infty$. Because $(f(x_i))$ is a monotone nonincreasing sequence that has an accumulation point $f(\hat{x})$, it follows from Proposition A.4 that $f(x_i) \to f(\hat{x})$ as $i \to \infty$. Since the limit of the sequence $(f(x_i))$ is unique, we conclude that $f(\hat{x}) = \alpha$. ∎

A.12 Derivatives

We first define some notation. If $f : \mathbb{R}^n \to \mathbb{R}$, then $(\partial/\partial x)f(x)$ is a *row* vector defined by

$$(\partial/\partial x)f(x) := [(\partial/\partial x_1)f(x), \ldots, (\partial/\partial x_n)f(x)]$$

provided the partial derivatives $(\partial/\partial x_i)f(x)$, $i = 1, 2, \ldots, n$ exist. Similarly, if $f : \mathbb{R}^n \to \mathbb{R}^m$, $(\partial/\partial x)f(x)$ is defined to be the matrix

$$(\partial/\partial x)f(x) := \begin{bmatrix} (\partial/\partial x_1)f_1(x) & (\partial/\partial x_2)f_1(x) & \ldots & (\partial/\partial x_n)f_1(x) \\ (\partial/\partial x_1)f_2(x) & (\partial/\partial x_2)f_2(x) & \ldots & (\partial/\partial x_n)f_2(x) \\ \vdots & \vdots & \vdots & \vdots \\ (\partial/\partial x_1)f_m(x) & (\partial/\partial x_2)f_m(x) & \ldots & (\partial/\partial x_n)f_m(x) \end{bmatrix}$$

where x_i and f_i denote, respectively, the ith component of the vectors x and f. We sometimes use $f_x(x)$ in place of $(\partial/\partial x)f(x)$. If $f : \mathbb{R}^n \to \mathbb{R}$, then its *gradient* $\nabla f(x)$ is a *column* vector defined by

$$\nabla f(x) := \begin{bmatrix} (\partial/\partial x_1)f(x) \\ (\partial/\partial x_2)f(x) \\ \vdots \\ (\partial/\partial x_n)f(x) \end{bmatrix}$$

and its *Hessian* is $\nabla^2 f(x) = (\partial^2/\partial x^2) f(x) = f_{xx}(x)$ defined by

$$\nabla^2 f(x) \coloneqq \begin{bmatrix} (\partial^2/\partial x_1^2) f(x) & (\partial^2/\partial x_1 \partial x_2) f(x) & \dots & (\partial^2/\partial x_1 \partial x_n) f(x) \\ (\partial^2/\partial x_2 \partial x_1) f(x) & (\partial x_2^2) f(x) & \dots & (\partial^2/\partial x_2 \partial x_n) f(x) \\ \vdots & \vdots & \ddots & \vdots \\ (\partial^2/\partial x_n \partial x_1) f(x) & (\partial^2/\partial x_n \partial x_2) f(x) & \dots & (\partial^2/\partial x_n^2) f(x) \end{bmatrix}$$

We note that $\nabla f(x) = [(\partial/\partial x) f(x)]' = f_x'(x)$.

We now define what we mean by the derivative of $f(\cdot)$. Let $f : \mathbb{R}^n \to \mathbb{R}^m$ be a continuous function with domain $\mathbb{R}^n$. We say that $f(\cdot)$ is differentiable at $\hat{x}$ if there exists a matrix $Df(\hat{x}) \in \mathbb{R}^{m \times n}$ (the Jacobian) such that

$$\lim_{h \to 0} \frac{|f(\hat{x} + h) - f(\hat{x}) - Df(\hat{x})h|}{|h|} = 0$$

in which case $Df(\cdot)$ is called the derivative of $f(\cdot)$ at $\hat{x}$. When $f(\cdot)$ is differentiable at all $x \in \mathbb{R}^n$, we say that f is *differentiable*.

We note that the affine function $h \mapsto f(\hat{x}) + Df(\hat{x})h$ is a first order approximation of $f(\hat{x} + h)$. The Jacobian can be expressed in terms of the partial derivatives of $f(\cdot)$.

Proposition A.8 (Derivative and partial derivative)**.** *Suppose that the function $f : \mathbb{R}^n \to \mathbb{R}^m$ is differentiable at $\hat{x}$. Then its derivative $Df(\hat{x})$ satisfies*

$$Df(\hat{x}) = f_x(\hat{x}) \coloneqq \partial f(\hat{x})/\partial x$$

Proof. From the definition of $Df(\hat{x})$ we deduce that for each $i \in \{1, 2, \dots, m\}$

$$\lim_{h \to 0} \frac{|f_i(\hat{x} + h) - f_i(\hat{x}) - Df_i(\hat{x})h|}{|h|} = 0$$

where f_i is the ith element of f and $(Df)_i$ the ith row of Df. Set $h = te_j$, where e_j is the j-th unit vector in $\mathbb{R}^n$ so that $|h| = t$. Then $(Df)_i(\hat{x})h = t(Df)_i(\hat{x})e_j = (Df)_{ij}(\hat{x})$, the ijth element of the matrix $Df(\hat{x})$. It then follows that

$$\lim_{t \searrow 0} \frac{\left| f^i(\hat{x} + te_j) - f(\hat{x}) - t(Df)_{ij}(\hat{x}) \right|}{t} = 0$$

which shows that $(Df)_{ij}(\hat{x}) = \partial f_i(\hat{x})/\partial x_j$. ∎

A function $f : \mathbb{R}^n \to \mathbb{R}^m$ is *locally Lipschitz continuous* at $\hat{x}$ if there exist $L \in [0, \infty)$, $\hat{\rho} > 0$ such that

$$|f(x) - f(x')| \leq L\,|x - x'|\,, \text{ for all } x, x' \in B(\hat{x}, \hat{\rho})$$

The function f is globally Lipschitz continuous if the inequality holds for all $x, x' \in \mathbb{R}^n$. The constant L is called the *Lipschitz constant* of f. It should be noted that the existence of partial derivatives of $f(\cdot)$ does not ensure the existence of the derivative $Df(\cdot)$ of $f(\cdot)$; see e.g. Apostol (1974, p.103). Thus consider the function

$$f(x, y) = x + y \text{ if } x = 0 \text{ or } y = 0$$

$$f(x, y) = 1 \text{ otherwise}$$

In this case

$$\frac{\partial f(0,0)}{\partial x} = \lim_{t \to 0} \frac{f(t,0) - f(0,0)}{t} = 1$$
$$\frac{\partial f(0,0)}{\partial y} = \lim_{t \to 0} \frac{f(0,t) - f(0,0)}{t} = 1$$

but the function is not even continuous at $(0,0)$. In view of this, the following result is relevant.

Proposition A.9 (Continuous partial derivatives)**.** *Consider a function $f : \mathbb{R}^n \to \mathbb{R}^m$ such that the partial derivatives $\partial f^i(x)/dx^j$ exist in a neighborhood of $\hat{x}$, for $i = 1, 2, \ldots, n, j = 1, 2, \ldots, m$. If these partial derivatives are continuous at $\hat{x}$, then the derivative $Df(\hat{x})$ exists and is equal to $f_x(\hat{x})$.*

The following *chain rule* holds.

Proposition A.10 (Chain rule)**.** *Suppose that $f : \mathbb{R}^n \to \mathbb{R}^m$ is defined by $f(x) = h(g(x))$ with both $h : \mathbb{R}^l \to \mathbb{R}^m$ and $g : \mathbb{R}^n \to \mathbb{R}^l$ differentiable. Then*

$$\frac{\partial f(\hat{x})}{\partial x} = \frac{\partial h(g(\hat{x}))}{\partial y} \frac{\partial g(\hat{x})}{\partial x}$$

The following result Dieudonne (1960), replaces, *inter alia*, the mean value theorem for functions $f : \mathbb{R}^n \to \mathbb{R}^m$ when $m > 1$.

Proposition A.11 (Mean value theorem for vector functions)**.**

(a) Suppose that $f : \mathbb{R}^n \to \mathbb{R}^m$ has continuous partial derivatives at each point x of $\mathbb{R}^n$. Then for any $x, y \in \mathbb{R}^n$,

$$f(y) = f(x) + \int_0^1 f_x(x + s(y - x))(y - x) ds$$

(b) Suppose that $f : \mathbb{R}^n \to \mathbb{R}^m$ has continuous partial derivatives of order two at each point x of $\mathbb{R}^n$. Then for any $x, y \in \mathbb{R}^n$,

$$f(y) = f(x)+f_x(x)(y-x)+\int_0^1 (1-s)(y-x)' f_{xx}(x+s(y-x))(y-x)ds$$

Proof.

(a) Consider the function $g(s) = f(x + s(y - x))$ where $f : \mathbb{R}^n \to \mathbb{R}^m$. Then $g(1) = f(y), g(0) = f(x)$ and

$$\begin{aligned} g(1) - g(0) &= \int_0^1 g'(s)ds \\ &= \int_0^1 Df(x + s(y - x))(y - x)ds \end{aligned}$$

which completes the proof for $p = 1$.

(b) Consider the function $g(s) = f(x + s(y - x))$ where $f : \mathbb{R}^n \to \mathbb{R}$. Then

$$\frac{d}{ds}[g'(s)(1 - s) + g(s)] = g''(s)(1 - s)$$

Integrating from 0 to 1 yields

$$g(1) - g(0) - g'(0) = \int_0^1 (1 - s)g''(s)ds$$

But $g''(s) = (y-x)' f_{xx}(x+s(y-x))(y-x)$ so that the last equation yields

$$f(y)-f(x) = f_x(x)(y-x)+\int_0^1 (1-s)(y-x)' f_{xx}(x+s(y-x))(y-x)ds$$

when $g(s)$ is replaced by $f(x + s(y - x))$.

■

Finally, we define directional derivatives which may exist even when a function fails to have a derivative. Let $f : \mathbb{R}^n \to \mathbb{R}^m$. We define the *directional derivative* of f at a point $\hat{x} \in \mathbb{R}^n$ in the direction $h \in \mathbb{R}^n (h \neq 0)$ by

$$df(\hat{x}; h) := \lim_{t \searrow 0} \frac{f(\hat{x} + th) - f(\hat{x})}{t}$$

if this limit exists (note that $t > 0$ is required). The directional derivative is positively homogeneous, i.e., $df(x; \lambda h) = \lambda df(x; h)$ for all $\lambda > 0$.

Not all the functions we discuss are differentiable everywhere. Examples include the max function $\psi(\cdot)$ defined by $\psi(x) := \max_i\{f^i(x) \mid i \in I\}$ where each function $f^i : \mathbb{R}^n \to \mathbb{R}$ is continuously differentiable everywhere. The function $\psi(\cdot)$ is not differentiable at those x for which the active set $I^0(x) := \{i \in I \mid f^i(x) = \psi(x)\}$ has more than one element. The directional derivative $d(x;h)$ exists for all x, h in $\mathbb{R}^n$, however, and is given by

$$d\psi(x;h) = \max_i\{df_i(x;h) \mid i \in I^0(x)\} = \max_i\{\langle \nabla f_i(x), h\rangle \mid i \in I^0(x)\}$$

When, as in this example, the directional derivative exists for all x, h in $\mathbb{R}^n$ we can define a generalization, called the *subgradient*, of the conventional gradient. Suppose that $f : \mathbb{R}^n \to \mathbb{R}$ has a directional derivative for all x, h in $\mathbb{R}^n$. The $f(\cdot)$ has a subgradient $\partial f(\cdot)$ defined by

$$\partial\psi(x) := \{g \in \mathbb{R}^n \mid df(x;h) \geq \langle g, h\rangle \ \forall h \in \mathbb{R}^n\}$$

The subgradient at a point x is, unlike the ordinary gradient, a set. For our max example ($f(x) = \psi(x) = \max_i\{f_i(x) \mid i \in I\}$) we have $d\psi(x;h) = \max_i\{\langle \nabla f^i(x), h\rangle \mid i \in I^0(x)\}$. In this case, it can be shown that

$$\partial\psi(x) = \text{co}\{\nabla f^i(x) \mid i \in I^0(x)\}$$

If the directional derivative $h \mapsto df(x;h)$ is convex, then the subgradient $\partial f(x)$ is nonempty and the directional derivative $df(x;h)$ may be expressed as

$$df(x;h) = \max_g\{\langle g, h\rangle \mid g \in \partial f(x)\}$$

Figure A.4 illustrates this for the case when $\psi(x) := \max\{f_1(x), f_2(x)\}$ and $I^0(x) = \{1, 2\}$.

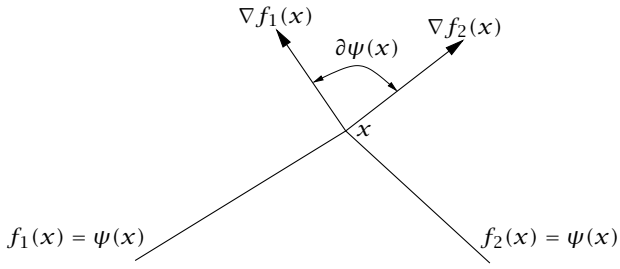


Figure A.4: Subgradient.

A.13 Convex Sets and Functions

Convexity is an enormous subject. We collect here only a few essential results that we will need in our study of optimization; for further details see Rockafellar (1970). We begin with convex sets.

A.13.1 Convex Sets

Definition A.12 (Convex set)**.** A set $S \in \mathbb{R}^n$ is said to be *convex* if, for any $x', x'' \in S$ and $\lambda \in [0,1]$, $(\lambda x' + (1-\lambda)x'') \in S$.

Let S be a subset of $\mathbb{R}^n$. We say that $\text{co}(S)$ is the *convex hull* of S if it is the smallest[7] convex set containing S.

Theorem A.13 (Caratheodory)**.** *Let S be a subset of $\mathbb{R}^n$. If $\bar{x} \in co(S)$, then it may be expressed as a convex combination of no more than $n+1$ points in S, i.e., there exist $m \leq n+1$ distinct points, $\{x_i\}_{i=1}^m$, in S such that $\bar{x} = \sum_{i=1}^m \mu^i x_i, \mu^i > 0, \sum_{i=1}^m \mu^i = 1$.*

Proof. Consider the set

$$C_s := \{x \mid x = \sum_{i=1}^{k_x} \mu^i x_i, x_i \in S, \mu^i \geq 0, \sum_{i=1}^{k_x} \mu^i = 1, k_x \in \mathbb{I}_{\geq 0}\}$$

First, it is clear that $S \subset C_s$. Next, since for any $x', x'' \in C_s, \lambda x' + (1-\lambda x'') \in C_s$, for $\lambda \in [0,1]$, it follows that C_s is convex. Hence we must have that $\text{co}(S) \subset C_s$. Because C_s consists of all the convex combinations of points in S, however, we must also have that $C_s \subset \text{co}(S)$. Hence $C_s = \text{co}(S)$. Now suppose that

$$\bar{x} = \sum_{i=1}^{\bar{k}} \bar{\mu}^i x_i$$

with $\bar{\mu}^i \geq 0, i = 1, 2, \ldots, \bar{k}, \sum_{i=1}^{\bar{k}} \bar{\mu}^i = 1$. Then the following system of equations is satisfied

$$\sum_{i=1}^{\bar{k}} \bar{\mu}^i \begin{bmatrix} x_i \\ 1 \end{bmatrix} = \begin{bmatrix} \bar{x} \\ 1 \end{bmatrix} \tag{A.3}$$

with $\bar{\mu}^i \geq 0$. Suppose that $\bar{k} > n+1$. Then there exist coefficients α^j, $j = 1, 2, \ldots, \bar{k}$, not all zero, such that

$$\sum_{i=1}^{\bar{k}} \alpha^i \begin{bmatrix} x_i \\ 1 \end{bmatrix} = 0 \tag{A.4}$$

[7] Smallest in the sense that any other convex set containing S also contains $\text{co}(S)$.

Adding (A.4) multiplied by θ to (A.3) we get

$$\sum_{i=1}^{\bar{k}} (\bar{\mu}^i + \theta\alpha^i) \begin{bmatrix} x_i \\ 1 \end{bmatrix} = \begin{bmatrix} \bar{x} \\ 1 \end{bmatrix}$$

Suppose, without loss of generality, that at least one $\alpha^i < 0$. Then there exists a $\bar{\theta} > 0$ such that $\bar{\mu}^j + \bar{\theta}\alpha^j = 0$ for some j while $\bar{\mu}^i + \bar{\theta}\alpha^i \geq 0$ for all other i. Thus we have succeeded in expressing $\bar{x}$ as a convex combination of $\bar{k} - 1$ vectors in S. Clearly, these reductions can go on as long as $\bar{x}$ is expressed in terms of more than $(n + 1)$ vectors in S. This completes the proof. ■

Let S_1, S_2 be any two sets in $\mathbb{R}^n$. We say that the hyperplane

$$H = \{x \in \mathbb{R}^n \mid \langle x, v \rangle = \alpha\}$$

separates S_1 and S_2 if

$$\langle x, v \rangle \geq \alpha \text{ for all } x \in S_1$$
$$\langle y, v \rangle \leq \alpha \text{ for all } y \in S_2$$

The separation is said to be *strong* if there exists an $\varepsilon > 0$ such that

$$\langle x, v \rangle \geq \alpha + \varepsilon \text{ for all } x \in S_1$$
$$\langle y, v \rangle \leq \alpha - \varepsilon \text{ for all } y \in S_2$$

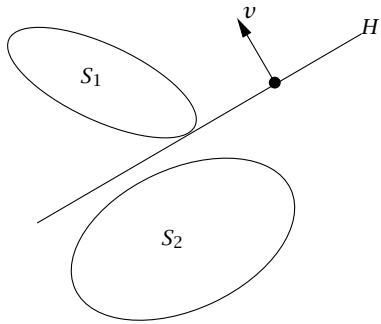


Figure A.5: Separating hyperplane.

Theorem A.14 (Separation of convex sets)**.** *Let S_1, S_2 be two convex sets in $\mathbb{R}^n$ such that $S_1 \cap S_2 = \emptyset$. Then there exists a hyperplane which separates S_1 and S_2. Furthermore, if S_1 and S_2 are closed and either S_1 or S_2 is compact, then the separation can be made strict.*

Theorem A.15 (Separation of convex set from zero)**.** *Suppose that* $S \subset \mathbb{R}^n$ *is closed and convex and* $0 \notin S$*. Let*

$$\hat{x} = \arg\min\{|x|^2 \mid x \in S\}$$

Then

$$H = \{x \mid \langle \hat{x}, x \rangle = |\hat{x}|^2\}$$

separates S *from 0, i.e.,* $\langle \hat{x}, x \rangle \geq |\hat{x}|^2$ *for all* $x \in S$*.*

Proof. Let $x \in S$ be arbitrary. Then, since S is convex, $[\hat{x} + \lambda(x - \hat{x})] \in S$ for all $\lambda \in [0, 1]$. By definition of $\hat{x}$, we must have

$$\begin{aligned} 0 < |\hat{x}|^2 &\leq |\hat{x} + \lambda(x - \hat{x})|^2 \\ &= |\hat{x}|^2 + 2\lambda\langle \hat{x}, x - \hat{x} \rangle + \lambda^2 |x - \hat{x}|^2 \end{aligned}$$

Hence, for all $\lambda \in (0, 1]$,

$$0 \leq 2\langle \hat{x}, x - \hat{x} \rangle + \lambda |x - \hat{x}|^2$$

Letting $\lambda \to 0$ we get the desired result. ∎

Theorem A.15 can be used to prove the following special case of Theorem A.14:

Corollary A.16 (Existence of separating hyperplane)**.** *Let* S_1, S_2 *be two compact convex sets in* $\mathbb{R}^n$ *such that* $S_1 \cap S_2 = \emptyset$*. Then there exists a hyperplane which separates* S_1 *and* S_2*.*

Proof. Let $C = S_1 - S_2 := \{x_1 - x_2 \mid x_1 \in S_1, x_2 \in S_2\}$. Then C is convex and compact and $0 \notin C$. Let $\hat{x} = (\hat{x}_1 - \hat{x}_2) = \arg\min\{|x|^2 \mid x \in C\}$, where $\hat{x}_1 \in S_1$ and $\hat{x}_2 \in S_2$. Then, by Theorem A.15

$$\langle x - \hat{x}, \hat{x} \rangle \geq 0, \text{ for all } x \in C \tag{A.5}$$

Let $x = x_1 - \hat{x}_2$, with $x_1 \in S_1$. Then (A.5) leads to

$$\langle x_1 - \hat{x}_2, \hat{x} \rangle \geq |\hat{x}|^2 \tag{A.6}$$

for all $x_1 \in S_1$. Similarly, letting $x = \hat{x}_1 - x_2$, in (A.5) yields

$$\langle \hat{x}_1 - x_2, \hat{x} \rangle \geq |\hat{x}|^2 \tag{A.7}$$

for all $x_2 \in S_2$. The inequality in (A.7) implies that

$$\langle \hat{x}_1 - \hat{x}_2 + \hat{x}_2 - x_2, \hat{x} \rangle \geq |\hat{x}|^2$$

Since $\hat{x}_1 - \hat{x}_2 = \hat{x}$, we obtain

$$\langle x_2 - \hat{x}_2, \hat{x} \rangle \leq 0 \tag{A.8}$$

for all $x_2 \in S_2$. The desired result follows from (A.6) and (A.8), the separating hyperplane H being $\{x \in \mathbb{R}^n \mid \langle \hat{x}, x - \hat{x}_2 \rangle = 0\}$. ■

Definition A.17 (Support hyperplane)**.** Suppose $S \subset \mathbb{R}^n$ is convex. We say that $H = \{x \mid \langle x - \bar{x}, v \rangle = 0\}$ is a *support hyperplane* to S through $\bar{x}$ with *inward (outward) normal* v if $\bar{x} \in \overline{S}$ and

$$\langle x - \bar{x}, v \rangle \geq 0 \ (\leq 0) \text{ for all } x \in S$$

Theorem A.18 (Convex set and halfspaces)**.** *A closed convex set is equal to the intersection of the halfspaces which contain it.*

Proof. Let C be a closed convex set and A the intersection of halfspaces containing C. Then clearly $C \subset A$. Now suppose $\bar{x} \notin C$. Then there exists a support hyperplane H which separates strictly $\bar{x}$ and C so that $\bar{x}$ does not belong to one halfspace containing C. It follows that $\bar{x} \notin A$. Hence $C^c \subset A^c$ which leads to the conclusion that $A \subset C$. ■

An important example of a convex set is a convex *cone*.

Definition A.19 (Convex cone)**.** A subset C of $\mathbb{R}^n$, $C \neq \emptyset$, is called a *cone* if $x \in C$ implies $\lambda x \in C$ for all $\lambda \geq 0$. A cone C is *pointed* if $C \cap -C = \{0\}$. A *convex* cone is a cone that is convex.

An example of a cone is a halfspaces with a boundary that is a hyperplane passing through the origin; an example of a pointed cone is the positive orthant. A polyhedron C defined by $C := \{x \mid \langle a_i, x \rangle \leq 0, i \in \mathcal{I}\}$ is a convex cone that is pointed

Definition A.20 (Polar cone)**.** Given a cone $C \subset \mathbb{R}^n$, the cone C^* defined by

$$C^* := \{h \mid \langle h, x \rangle \leq 0 \ \forall x \in C\}$$

is called the *polar cone* of C.

An illustration of this definition when C is a polyhedron containing the origin is given in Figure A.6. In this figure, H is the hyperplane with normal h passing through the origin.

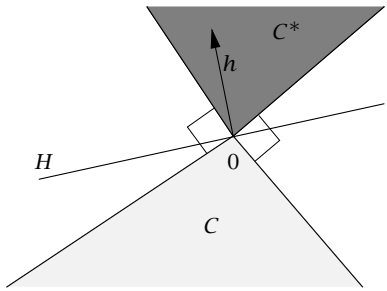


Figure A.6: Polar cone.

Definition A.21 (Cone generator)**.** A cone K is said to be *generated* by a set $\{a_i \mid i \in \mathcal{I}\}$ where $\mathcal{I}$ is an index set if

$$K = \left\{ \sum_{i \in \mathcal{I}} \mu_i a_i \mid \mu_i \geq 0, i \in \mathcal{I} \right\}$$

in which case we write $K = \text{cone}\{a_i \mid i \in \mathcal{I}\}$.

We make use of the following result:

Proposition A.22 (Cone and polar cone generator)**.**

(a) Suppose C is a convex cone containing the origin and defined by

$$C := \{x \in \mathbb{R}^n \mid \langle a_i, x \rangle \leq 0,\ i \in \mathcal{I}\}$$

Then

$$C^* = \text{cone}\{a_i \mid i \in \mathcal{I}\}$$

(b) If C is a closed convex cone, then $(C^*)^* = C$.

(c) If $C_1 \subset C_2$, then $C_2^* \subset C_1^*$.

Proof.

(a) Let the convex set K be defined by

$$K := \text{cone}\{a_i \mid i \in \mathcal{I}\}$$

We wish to prove $C^* = K$. To prove $K \subset C^*$, suppose h is an arbitrary point in $K := \text{cone}\{a_i \mid i \in \mathcal{I}\}$. Then $h = \sum_{i \in \mathcal{I}} \mu_i a_i$ where $\mu_i \geq 0$ for all $i \in \mathcal{I}$. Let x be an arbitrary point in C so that $\langle a_i, x \rangle \leq 0$ for all $i \in \mathcal{I}$. Hence

$$\langle h, x \rangle = \langle \sum_{i \in \mathcal{I}} \mu_i a_i, x \rangle = \sum_{i \in \mathcal{I}} \mu_i \langle a_i, x \rangle \leq 0$$

so that $h \in C^*$. This proves that $K \subset C^*$. To prove that $C^* \subset K$, assume that $h \in C^*$ but that, contrary to what we wish to prove, $h \notin K$. Hence $h = \sum_{i \in \mathcal{I}} \mu_i a_i + \widetilde{h}$ where either $\mu_j > 0$ for at least one $j \in \mathcal{I}$, or $\widetilde{h}$, which is orthogonal to $a_i, i \in \mathcal{I}$, is not zero, or both. If $\mu_j < 0$, let $x \in C$ be such that $\langle a_i, x \rangle = 0$ for all $i \in \mathcal{I}, i \neq j$ and $\langle a_j, x \rangle < 0$; if $\widetilde{h} \neq 0$, let $x \in C$ be such that $\langle \widetilde{h}, x \rangle > 0$ (both conditions can be satisfied). Then

$$\langle h, x \rangle = \langle \mu_j a_j, x \rangle + \langle \widetilde{h}, x \rangle = \mu_j \langle a_j, x \rangle + \langle \widetilde{h}, x \rangle > 0$$

since either both μ_j and $\langle a_j, x \rangle$ are strictly negative or $\widetilde{h} \neq 0$ or both. This contradicts the fact that $x \in C$ and $h \in C^*$ (so that $\langle h, x \rangle \leq 0$). Hence $h \in K$ so that $C^* \subset K$. It follows that $C^* = \text{cone}\{a_i \mid i \in \mathcal{I}\}$.

(b) That $(C^*)^* = C$ when C is a closed convex cone is given in Rockafellar and Wets (1998), Corollary 6.21.

(c) This result follows directly from the definition of a polar cone.

■

A.13.2 Convex Functions

Next we turn to convex functions. For an example see Figure A.7.

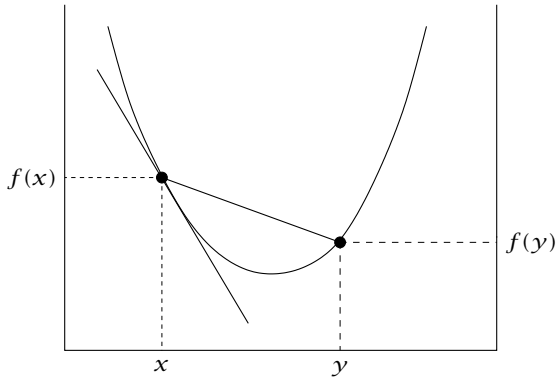


Figure A.7: A convex function.

A function $f : \mathbb{R}^n \to \mathbb{R}$ is said to be *convex* if for any $x', x'' \in \mathbb{R}^n$ and $\lambda \in [0, 1]$,

$$f(\lambda x' + (1 - \lambda)x'') \leq \lambda f(x') + (1 - \lambda) f(x'')$$

A function $f : \mathbb{R}^n \to \mathbb{R}$ is said to be *concave* if $-f$ is convex.

The *epigraph* of a function $f : \mathbb{R}^n \to \mathbb{R}$ is defined by

$$\text{epi}(f) := \{(x, y) \in \mathbb{R}^n \times \mathbb{R} \mid y \geq f(x)\}$$

Theorem A.23 (Convexity implies continuity)**.** *Suppose $f : \mathbb{R}^n \to \mathbb{R}$ is convex. Then f is continuous in the interior of it domain.*

The following property is illustrated in Figure A.7.

Theorem A.24 (Differentiability and convexity)**.** *Suppose $f : \mathbb{R}^n \to \mathbb{R}$ is differentiable. Then f is convex if and only if*

$$f(y) - f(x) \geq \langle \nabla f(x), y - x \rangle \text{ for all } x, y \in \mathbb{R}^n \tag{A.9}$$

Proof. $\Rightarrow$ Suppose f is convex. Then for any $x, y \in \mathbb{R}^n$, and $\lambda \in [0, 1]$

$$f(x + \lambda(y - x)) \leq (1 - \lambda) f(x) + \lambda f(y) \tag{A.10}$$

Rearranging (A.10) we get

$$\frac{f(x + \lambda(y - x)) - f(x)}{\lambda} \leq f(y) - f(x) \text{ for all } \lambda \in [0, 1]$$

Taking the limit as $\lambda \to 0$ we get (A.9).

$\Leftarrow$ Suppose (A.9) holds. Let x and y be arbitrary points in $\mathbb{R}^n$ and let λ be an arbitrary point in $[0, 1]$. Let $z = \lambda x + (1 - \lambda) y$. Then

$$\begin{aligned} f(x) &\geq f(z) + f'(z)(x - z), \text{ and} \\ f(y) &\geq f(z) + f'(z)(y - z) \end{aligned}$$

Multiplying the first equation by λ and the second by $(1 - \lambda)$, adding the resultant equations, and using the fact that $z = \lambda x + (1 - \lambda) y$ yields

$$\lambda f(x) + (1 - \lambda) f(y) \geq f(z) = f(\lambda x + (1 - \lambda) y)$$

Since x and y in $\mathbb{R}^n$ and λ in $[0, 1]$ are all arbitrary, the convexity of $f(\cdot)$ is established. ∎

Theorem A.25 (Second derivative and convexity)**.** *Suppose that $f : \mathbb{R}^n \to \mathbb{R}$ is twice continuously differentiable. Then f is convex if and only if the Hessian (second derivative) matrix $\partial^2 f(x)/\partial x^2$ is positive semidefinite for all $x \in \mathbb{R}^n$, i.e., $\langle y, \partial^2 f(x)/\partial x^2 y \rangle \geq 0$ for all $x, y \in \mathbb{R}^n$.*

Proof. $\Rightarrow$ Suppose f is convex. Then for any $x, y \in \mathbb{R}^n$, because of Theorem A.24 and Proposition A.11

$$\begin{aligned} 0 &\le f(y) - f(x) - \langle \nabla f(x), y - x \rangle \\ &= \int_0^1 (1-s) \left\langle y - x, \frac{\partial^2 f(x + s(y-x))}{\partial x^2}(y-x) \right\rangle ds \end{aligned} \tag{A.11}$$

Hence, dividing by $|y-x|^2$ and letting $y \to x$, we obtain that $\partial^2 f(x)/\partial x^2$ is positive semidefinite.

$\Leftarrow$ Suppose that $\partial^2 f(x)/\partial x^2$ is positive semidefinite for all $x \in \mathbb{R}$. Then it follows directly from the equality in (A.11) and Theorem A.24 that f is convex. ■

Definition A.26 (Level set)**.** Suppose $f : \mathbb{R}^n \to \mathbb{R}$. A *level set* of f is a set of the form $\{x \mid f(x) = \alpha\}, \alpha \in \mathbb{R}$.

Definition A.27 (Sublevel set)**.** Suppose $f : \mathbb{R}^n \to \mathbb{R}$. A *sublevel set* $\mathbb{X}$ of f is a set of the form $\mathbb{X} = \{x \mid f(x) \le \alpha\}, \alpha \in \mathbb{R}$. We also write the sublevel set as $\mathbb{X} = \text{lev}_\alpha f$.

Definition A.28 (Support function)**.** Suppose $Q \subset \mathbb{R}^n$. The support function $\sigma_Q : \mathbb{R}^n \to \mathbb{R}_e = \mathbb{R} \cup \{+\infty\}$ is defined by:

$$\sigma_Q(p) = \sup_x \{\langle p, x \rangle \mid x \in Q\}$$

$\sigma_Q(p)$ measures how far Q extends in direction p.

Proposition A.29 (Set membership and support function)**.** *Suppose $Q \subset \mathbb{R}^n$ is a closed and convex set. Then $x \in Q$ if and only if $\sigma_Q(p) \ge \langle p, x \rangle$ for all $p \in \mathbb{R}^n$*

Proposition A.30 (Lipschitz continuity of support function)**.** *Suppose $Q \subset \mathbb{R}^n$ is bounded. Then σ_Q is bounded and Lipschitz continuous $|\sigma_Q(p) - \sigma_Q(q)| \le K |p - q|$ for all $p, q \in \mathbb{R}^n$, where $K := \sup\{|x| \mid x \in Q\} < \infty$.*

A.14 Differential Equations

Although difference equation models are employed extensively in this book, the systems being controlled are most often described by differential equations. Thus, if the system being controlled is described by

the differential equation $\dot{x} = f_c(x,u)$, as is often the case, and if it is decided to control the system using piecewise constant control with period Δ, then, at sampling instants $k\Delta$ where $k \in \mathbb{I}$, the system is described by the difference equation

$$x^+ = f(x,u)$$

then $f(\cdot)$ may be derived from $f_c(\cdot)$ as follows

$$f(x,u) = x + \int_0^\Delta f_c(\phi_c(s;x,u),u)ds$$

where $\phi_c(s;x,u)$ is the solution of $\dot{x} = f_c(x,u)$ at time s if its initial state at time 0 is x and the control has a constant value u in the interval $[0,\Delta]$. Thus x in the difference equation is the state at time k, say, u is the control in the interval $[0,\Delta]$, and x^+ is the state at time $k+1$.

Because the discrete time system is most often obtained by a continuous time system, we must be concerned with conditions which guarantee the existence and uniqueness of solutions of the differential equation describing the continuous time system. For excellent expositions of the theory of ordinary differential equations see the books by Hale (1980), McShane (1944), Hartman (1964), and Coddington and Levinson (1955).

Consider, first, the unforced system described by

$$(d/dt)x(t) = f(x(t),t) \text{ or } \dot{x} = f(x,t) \tag{A.12}$$

with initial condition

$$x(t_0) = x_0 \tag{A.13}$$

Suppose $f : D \to \mathbb{R}^n$, where D is an open set in $\mathbb{R}^n \times \mathbb{R}$, is continuous. A function $x : T \to \mathbb{R}^n$, where T is an interval in $\mathbb{R}$, is said to be a (conventional) solution of (A.12) with initial condition (A.13) (or passing through (x_0,t_0)) if:

(a) x is continuously differentiable and x satisfies (A.12) on T,

(b) $x(t_0) = x_0$,

and $(x(t),t) \in D$ for all t in T. It is easily shown, when f is continuous, that x satisfies (A.12) and (A.13) if and only if:

$$x(t) = x_0 + \int_{t_0}^{t} f(x(s),s)ds \tag{A.14}$$

Peano's existence theorem states that if f is continuous on D, then, for all $(x_0, t_0) \in D$ there exists at least one solution of (A.12)) passing through (x_0, t_0). The solution is not necessarily unique - a counter example being $\dot{x} = \sqrt{x}$ for $x \geq 0$. To proceed we need to be able to deal with systems for which $f(\cdot)$ is not necessarily continuous for the following reason. If the system is described by $\dot{x} = f(x, u, t)$ where $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ is continuous, and the control $u : \mathbb{R} \to \mathbb{R}^m$ is continuous, then, for given $u(\cdot)$, the function $f^u : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^n$ defined by:

$$f^u(x, t) := f(x, u(t), t)$$

is continuous in t. We often encounter controls that are not continuous, however, in which case $f^u(\cdot)$ is also not continuous. We need a richer class of controls. A suitable class is the class of *measurable* functions which, for the purpose of this book, we may take to be a class rich enough to include all controls, such as those that are merely piecewise continuous, that we may encounter. If the control $u(\cdot)$ is measurable and $f(\cdot)$ is continuous, then $f^u(\cdot)$, defined above, is continuous in x but measurable in t, so we are forced to study such functions. Suppose, as above, D is an open set in $\mathbb{R}^n \times \mathbb{R}$. The function $f : D \to \mathbb{R}^n$ is said to satisfy the *Caratheodory* conditions in D if:

(a) f is measurable in t for each fixed x,

(b) f is continuous in x for each fixed t,

(c) for each compact set F in D there exists a measurable function $t \mapsto m_F(t)$ such that

$$|f(x, t)| \leq m_F(t)$$

for all $(x, t) \in F$. We now make use of the fact that if $t \mapsto h(t)$ is measurable, its integral $t \mapsto H(t) \stackrel{\Delta}{=} \int_{t_0}^{t} h(s)ds$ is absolutely continuous and, therefore, has a derivative almost everywhere. Where $H(\cdot)$ is differentiable, its derivative is equal to $h(\cdot)$. Consequently, if $f(\cdot)$ satisfies the Caratheodory conditions, then the solution of (A.14), i.e., a function $\phi(\cdot)$ satisfying (A.14) everywhere does not satisfy (A.12) everywhere but only almost everywhere, at the points where $\phi(\cdot)$ is differentiable. In view of this, we may speak *either* of a solution of (A.14) *or* of a solution of (A.12) provided we interpret the latter as an absolutely continuous function which satisfies (A.12)) almost everywhere. The appropriate generalization of Peano's existence theorem is the following result due to Caratheodory:

Theorem A.31 (Existence of solution to differential equations)**.** *If D is an open set in $\mathbb{R}^n \times \mathbb{R}$ and $f(\cdot)$ satisfies the Caratheodory conditions on D, then, for any (x_0, t_0) in D, there exists a solution of* (A.14) *or* (A.12) *passing through (x_0, t_0).*

Two other classical theorems on ordinary differential equations that are relevant are:

Theorem A.32 (Maximal interval of existence)**.** *If D is an open set in $\mathbb{R}^n \times \mathbb{R}$, $f(\cdot)$ satisfies the Caratheodory conditions on D, and $\phi(\cdot)$ is a solution of* (A.10) *on some interval, then there is a continuation $\phi'(\cdot)$ of $\phi(\cdot)$ to a maximal interval (t_a, t_b) of existence. The solution $\phi'(\cdot)$, the continuation of $\phi(\cdot)$, tends to the boundary of D as $t \searrow t_a$ and $t \nearrow t_b$.*

Theorem A.33 (Continuity of solution to differential equation)**.** *Suppose D is an open set in $\mathbb{R}^n \times \mathbb{R}$, f satisfies the Caratheodory condition and, for each compact set U in D, there exists an integrable function $t \mapsto k_u(t)$ such that*

$$|f(x,t) - f(y,t)| \leq k_u(t)\,|x - y|$$

for all $(x,t), (y,t)$ in U. Then, for any (x_0, t_0) in U there exists a unique solution $\phi(\cdot; x_0, t_0)$ passing through (x_0, t_0). The function $(t, x_0, t_0) \mapsto \phi(t; x_0, t_0) : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^n$ is continuous in its domain E which is open.

Note that D is often $\mathbb{R}^n \times \mathbb{R}$, in which case Theorem A.32 states that a solution $x(\cdot)$ of (A.14) escapes, i.e., $|x(t)| \to \infty$ as $t \searrow t_a$ or $t \nearrow t_b$ if t_a and t_b are finite; t_a and t_b are the escape times. An example of a differential equation with finite escape time is $\dot{x} = x^2$ which has, if $x_0 > 0, t_0 = 0$, a solution $x(t) = x_0[1 - (t - t_0)x_0]^{-1}$ and the maximal interval of existence is $(t_a, t_b) = (-\infty, t_0 + 1/x_0)$.

These results, apart from absence of a control u which is trivially corrected, do not go far enough. We require solutions on an interval $[t_0, t_f]$ given a priori. Further assumptions are needed for this. A useful tool in developing the required results is the Bellman-Gronwall lemma:

Theorem A.34 (Bellman-Gronwall)**.** *Suppose that $c \in (0, \infty)$ and that $\alpha : [0,1] \to \mathbb{R}_+$ is a bounded, integrable function, and that the integrable function $y : [0,1] \to \mathbb{R}$ satisfies the inequality*

$$y(t) \leq c + \int_0^t \alpha(s) y(s) ds \tag{A.15}$$

for all $t \in [0,1]$. *Then*

$$y(t) \leq ce^{\int_0^t \alpha(s)ds} \tag{A.16}$$

for all $t \in [0,1]$.

Note that, if the inequality in (A.15) were replaced by an equality, (A.15) could be integrated to yield (A.16).

Proof. Let the function $Y : [0,1] \to \mathbb{R}$ be defined by

$$Y(t) = \int_0^t \alpha(s)y(s)ds \tag{A.17}$$

so that $\dot{Y}(t) = \alpha(t)y(t)$ almost everywhere on $[0,1]$. It follows from (A.15) and (A.17) that:

$$y(t) \leq c + Y(t) \ \forall t \in [0,1]$$

Hence

$$\begin{aligned}(d/dt)[e^{-\int_0^t \alpha(s)ds}Y(t)] &= e^{-\int_0^t \alpha(s)ds}(\dot{Y}(t) - \alpha(t)Y(t)) \\ &= (e^{-\int_0^t \alpha(s)ds})\alpha(t)(y(t) - Y(t)) \\ &\leq c(e^{-\int_0^t \alpha(s)ds})\alpha(t) \end{aligned} \tag{A.18}$$

almost everywhere on $[0,1]$. Integrating both sides of (A.18) from 0 to t yields

$$e^{-\int_0^t \alpha(s)ds}Y(t) \leq c[1 - e^{-\int_0^t \alpha(s)ds}]$$

for all $t \in [0,1]$. Hence

$$Y(t) \leq c[e^{\int_0^t \alpha(s)ds} - 1]$$

and

$$y(t) \leq ce^{\int_0^t \alpha(s)ds}$$

for all $t \in [0,1]$. ∎

The interval $[0,1]$ may, of course, be replaced by $[t_0, t_f]$ for arbitrary $t_0, t_f \in (-\infty, \infty)$. Consider now the forced system described by

$$\dot{x}(t) = f(x(t), u(t), t) \text{ a.e} \tag{A.19}$$

with initial condition

$$x(0) = 0$$

The period of interest is now $T := [0,1]$ and "a.e." denotes "almost everywhere on T." Admissible controls $u(\cdot)$ are measurable and satisfy the control constraint

$$u(t) \in \Omega \text{ a.e.}$$

where $\Omega \subset \mathbb{R}^m$ is compact. For convenience, we denote the set of admissible controls by

$$\mathcal{U} := \{u : T \to \mathbb{R}^m \mid u(\cdot) \text{ is measurable}, u(t) \in \Omega \text{ a.e.}\}$$

Clearly $\mathcal{U}$ is a subset of L_∞. For simplicity we assume, in the sequel, that f is continuous; this is not restrictive. For each u in $\mathcal{U}$, x in $\mathbb{R}^n$, the function $t \mapsto f^u(x,t) := f(x,u(t),t)$ is measurable so that f^u satisfies the Caratheodory conditions and our previous results, Theorems A.31–A.33, apply. Our concern now is to show that, with additional assumptions, for each u in $\mathcal{U}$, a solution to (A.12) or (A.13) exists on T, rather than on some maximal interval that may be a subset of T, and that this solution is unique and bounded.

Theorem A.35 (Existence of solutions to forced systems)**.** *Suppose:*

(a) f is continuous and

(b) there exists a positive constant c such that

$$|f(x',u,t) - f(x,u,t)| \le c\,|x' - x|$$

for all $(x,u,t) \in \mathbb{R}^n \times \Omega \times T$. Then, for each u in $\mathcal{U}$, there exists a unique, absolutely continuous solution $x^u : T \to \mathbb{R}^n$ of (A.19) *on the interval T passing through $(x_0,0)$. Moreover, there exists a constant K such that*

$$|x^u(t)| \le K$$

for all $t \in T$, all $u \in \mathcal{U}$.

Proof. A direct consequence of (b) is the existence of a constant which, without loss of generality, we take to be c, satisfying

(c) $|f(x,u,t)| \le c(1 + |x|)$ for all $(x,u,t) \in \mathbb{R}^n \times \Omega \times T$.

Assumptions (a) and (b) and their corollary (c), a growth condition on $f(\cdot)$, ensure that $f^u(\cdot)$ satisfies the Caratheodory conditions stated earlier. Hence, our previous results apply, and there exists an interval $[0,t_b]$ on which a unique solution $x^u(\cdot)$ exists; moreover $|x^u(t)| \to \infty$ as $t \nearrow t_b$. Since $x^u(\cdot)$ satisfies

$$x^u(t) = x_0 + \int_0^t f(x^u(s),u(s),s)\,ds$$

it follows from the growth condition that

$$\begin{aligned} |x^u(t)| &\le |x_0| + \int_0^t |f(x^u(s), u(s), s)|\, ds \\ &\le |x_0| + c\int_0^t (1 + |x^u(s)|) ds \\ &\le (|x_0| + c) + c\int_0^t |x^u(s)|\, ds \end{aligned}$$

Applying the Bellman-Gronwall lemma yields

$$|x^u(t)| \le (c + |x_0|)e^{ct}$$

for all $t \in [0, t_b), u \in \mathcal{U}$. If follows that the escape time t_b cannot be finite, so that, for all u in $\mathcal{U}$, there exists a unique absolutely continuous solution $x^u(\cdot)$ on T passing through $(x_0, (0))$. Moreover, for all u in $\mathcal{U}$, all $t \in T$

$$|x^u(t)| \le K$$

where $K := (c + |x_0|)e^c$. ■

A.15 Random Variables and the Probability Density

Let ξ be a random variable taking values in the field of real numbers and the function $F_\xi(x)$ denote the **probability distribution function** of the random variable so that

$$F_\xi(x) = \Pr(\xi \le x)$$

i.e., $F_\xi(x)$ is the probability that the random variable ξ takes on a value less than or equal to x. F_ξ is obviously a nonnegative, nondecreasing function and has the following properties due to the axioms of probability

$$F_\xi(x_1) \le F_\xi(x_2) \quad \text{if } x_1 < x_2$$

$$\lim_{x\to-\infty} F_\xi(x) = 0$$
$$\lim_{x\to\infty} F_\xi(x) = 1$$

We next define the **probability density function**, denoted $p_\xi(x)$, such that

$$F_\xi(x) = \int_{-\infty}^{x} p_\xi(s) ds, \qquad -\infty < x < \infty \tag{A.20}$$

We can allow discontinuous F_ξ if we are willing to accept generalized functions (delta functions and the like) for p_ξ. Also, we can define the density function for discrete as well as continuous random variables if we allow delta functions. Alternatively, we can replace the integral in (A.20) with a sum over a discrete density function. The random variable may be a coin toss or a dice game, which takes on values from a discrete set contrasted to a temperature or concentration measurement, which takes on a values from a continuous set. The density function has the following properties

$$p_\xi(x) \geq 0$$

$$\int_{-\infty}^{\infty} p_\xi(x)dx = 1$$

and the interpretation in terms of probability

$$\Pr(x_1 \leq \xi \leq x_2) = \int_{x_1}^{x_2} p_\xi(x)dx$$

The **mean** or **expectation** of a random variable ξ is defined as

$$\mathcal{E}(\xi) = \int_{-\infty}^{\infty} x p_\xi(x)dx$$

The moments of a random variable are defined by

$$\mathcal{E}(\xi^n) = \int_{-\infty}^{\infty} x^n p_\xi(x)dx$$

and it is clear that the mean is the first moment. Moments of ξ about the mean are defined by

$$\mathcal{E}((\xi - \mathcal{E}(\xi))^n) = \int_{-\infty}^{\infty} (x - \mathcal{E}(\xi))^n p_\xi(x)dx$$

and the variance is defined as the second moment about the mean

$$\text{var}(\xi) = \mathcal{E}((\xi - \mathcal{E}(\xi))^2) = \mathcal{E}(\xi^2) - \mathcal{E}^2(\xi)$$

The standard deviation is the square root of the variance

$$\sigma(\xi) = (\text{var}(\xi))^{1/2}$$

Normal distribution. The normal or Gaussian distribution is ubiquitous in applications. It is characterized by its mean, m and variance, σ^2, and is given by

$$p_\xi(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2}\frac{(x-m)^2}{\sigma^2}\right) \tag{A.21}$$

We proceed to check that the mean of this distribution is indeed m and the variance is σ^2 as claimed and that the density is normalized so that its integral is one. We require the definite integral formulas

$$\int_{-\infty}^{\infty} e^{-x^2} dx = \sqrt{\pi}$$
$$\int_0^{\infty} x^{1/2} e^{-x} dx = \Gamma(3/2) = \frac{\sqrt{\pi}}{2}$$

The first formula may also be familiar from the error function in transport phenomena

$$\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-u^2} du$$
$$\mathrm{erf}(\infty) = 1$$

We calculate the integral of the normal density as follows

$$\int_{-\infty}^{\infty} p_\xi(x) dx = \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{\infty} \exp\left(-\frac{1}{2}\frac{(x-m)^2}{\sigma^2}\right) dx$$

Define the change of variable

$$u = \frac{1}{\sqrt{2}}\left(\frac{x-m}{\sigma}\right)$$

which gives

$$\int_{-\infty}^{\infty} p_\xi(x) dx = \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\infty} \exp\left(-u^2\right) du = 1$$

and (A.21) does have unit area. Computing the mean gives

$$\mathcal{E}(\xi) = \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{\infty} x \exp\left(-\frac{1}{2}\frac{(x-m)^2}{\sigma^2}\right) dx$$

using the same change of variables as before yields

$$\mathcal{E}(\xi) = \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\infty} (\sqrt{2}u\sigma + m) e^{-u^2} du$$

The first term in the integral is zero because u is an odd function, and the second term produces

$$\mathcal{E}(\xi) = m$$

as claimed. Finally the definition of the variance of ξ gives

$$\text{var}(\xi) = \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{\infty} (x-m)^2 \exp\left(-\frac{1}{2}\frac{(x-m)^2}{\sigma^2}\right) dx$$

Changing the variable of integration as before gives

$$\text{var}(\xi) = \frac{2}{\sqrt{\pi}}\sigma^2 \int_{-\infty}^{\infty} u^2 e^{-u^2} du$$

and because the integrand is an even function,

$$\text{var}(\xi) = \frac{4}{\sqrt{\pi}}\sigma^2 \int_{0}^{\infty} u^2 e^{-u^2} du$$

Now changing the variable of integration again using $s = u^2$ gives

$$\text{var}(\xi) = \frac{2}{\sqrt{\pi}\sigma^2} \int_{0}^{\infty} s^{1/2} e^{-s} ds$$

The second integral formula then gives

$$\text{var}(\xi) = \sigma^2$$

Shorthand notation for the random variable ξ having a normal distribution with mean m and variance σ^2 is

$$\xi \sim N(m, \sigma^2)$$

Figure A.8 shows the normal distribution with a mean of one and variances of 1/2, 1 and 2. Notice that a large variance implies that the random variable is likely to take on large values. As the variance shrinks to zero, the probability density becomes a delta function and the random variable approaches a deterministic value.

Central limit theorem.

> The central limit theorem states that if a set of n random variables $x_i, i = 1, 2, \ldots, n$ are independent, then under general conditions the density p_y of their sum
>
> $$y = x_1 + x_2 + \cdots + x_n$$
>
> properly normalized, tends to a normal density as $n \to \infty$. (Papoulis, 1984, p. 194).

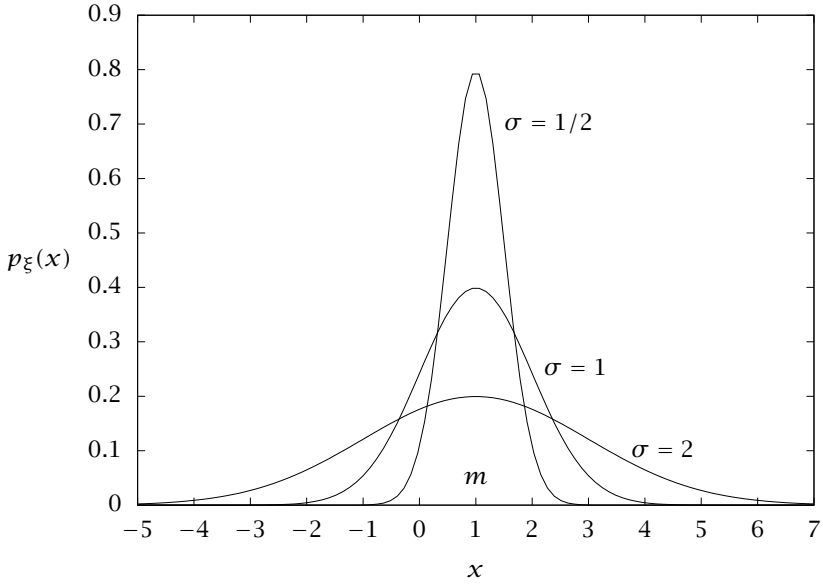


Figure A.8: Normal distribution, $p_\xi(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2}\frac{(x-m)^2}{\sigma^2}\right)$. Mean is one and standard deviations are 1/2, 1 and 2.

Notice that we require only mild restrictions on how the x_i themselves are distributed for the sum y to tend to a normal. See Papoulis (1984, p. 198) for one set of sufficient conditions and a proof of this theorem.

Fourier transform of the density function. It is often convenient to handle the algebra of density functions, particularly normal densities, by using the Fourier transform of the density function rather than the density itself. The transform, which we denote as $\varphi_\xi(u)$, is often called the characteristic function or generating function in the statistics literature. From the definition of the Fourier transform

$$\varphi_\xi(u) = \int_{-\infty}^{\infty} e^{iux} p_\xi(x) dx$$

The transform has a one-to-one correspondence with the density function, which can be seen from the inverse transform formula

$$p_\xi(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{-iux} \varphi_\xi(u) du$$

Example A.36: Fourier transform of the normal density.

Show the Fourier transform of the normal density is $\varphi_\xi(u) = \exp\left(ium - \frac{1}{2}u^2\sigma^2\right)$. □

A.16 Multivariate Density Functions

In applications we normally do not have a single random variable but a collection of random variables. We group these variables together in a vector and let random variable ξ now take on values in $\mathbb{R}^n$. The probability density function is still a nonnegative scalar function

$$p_\xi(x) : \mathbb{R}^n \to \mathbb{R}^+$$

which is sometimes called the **joint density function**. As in the scalar case, the probability that the n-dimensional random variable ξ takes on values between a and b is given by

$$\Pr(a \le \xi \le b) = \int_{a_n}^{b_n} \cdots \int_{a_1}^{b_1} p_\xi(x) dx_1 \cdots dx_n$$

Marginal density functions. We are often interested in only some subset of the random variables in a problem. Consider two vectors of random variables, $\xi \in \mathbb{R}^n$ and $\eta \in \mathbb{R}^m$. We can consider the joint distribution of both of these random variables $p_{\xi,\eta}(x, y)$ or we may only be interested in the ξ variables, in which case we can integrate out the m η variables to obtain the marginal density of ξ

$$p_\xi(x) = \int_{-\infty}^{\infty} \cdots \int p_{\xi,\eta}(x, y) dy_1 \cdots dy_m$$

Analogously to produce the marginal density of η we use

$$p_\eta(y) = \int_{-\infty}^{\infty} \cdots \int p_{\xi,\eta}(x, y) dx_1 \cdots dx_n$$

Multivariate normal density. We define the multivariate normal density of the random variable $\xi \in \mathbb{R}^n$ as

$$p_\xi(x) = \frac{1}{(2\pi)^{n/2}(\det P)^{1/2}} \exp\left[-\frac{1}{2}(x - m)' P^{-1}(x - m)\right] \quad \text{(A.22)}$$

in which $m \in \mathbb{R}^n$ is the mean and $P \in \mathbb{R}^{n \times n}$ is the covariance matrix. The notation $\det P$ denotes determinant of P. As noted before, P is a

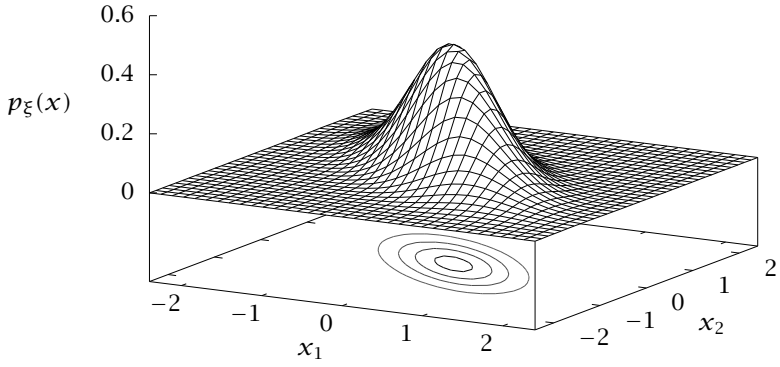


Figure A.9: Multivariate normal in two dimensions.

real, symmetric matrix. The multivariate normal density is well-defined only for $P > 0$. The singular, or degenerate, case $P \geq 0$ is discussed subsequently. Shorthand notation for the random variable ξ having a normal distribution with mean m and covariance P is

$$\xi \sim N(m, P)$$

The matrix P is a real, symmetric matrix. Figure A.9 displays a multivariate normal for

$$P^{-1} = \begin{bmatrix} 3.5 & 2.5 \\ 2.5 & 4.0 \end{bmatrix} \qquad m = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

As displayed in Figure A.9, lines of constant probability in the multivariate normal are lines of constant

$$(x - m)' P^{-1} (x - m)$$

To understand the geometry of lines of constant probability (ellipses in two dimensions, ellipsoids or hyperellipsoids in three or more dimensions) we examine the eigenvalues and eigenvectors of the P^{-1} matrix.

$$x'Ax = b$$
$$Av_i = \lambda_i v_i$$

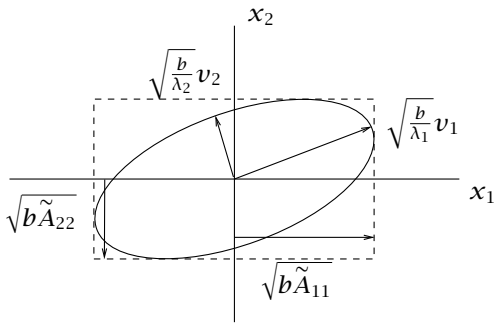


Figure A.10: The geometry of quadratic form $x'Ax = b$.

Consider the quadratic function $x'Ax$ depicted in Figure A.10. Each eigenvector of A points along one of the axes of the ellipse $x'Ax = b$. The eigenvalues show us how stretched the ellipse is in each eigenvector direction. If we want to put simple bounds on the ellipse, then we draw a box around it as shown in Figure A.10. Notice the box contains much more area than the corresponding ellipse and we have lost the correlation between the elements of x. This loss of information means we can put different tangent ellipses of quite different shapes inside the same box. The size of the bounding box is given by

$$\text{length of } i\text{th side} = \sqrt{b\widetilde{A}_{ii}}$$

in which

$$\widetilde{A}_{ii} = (i,i) \text{ element of } A^{-1}$$

See Exercise A.45 for a derivation of the size of the bounding box. Figure A.10 displays these results: the eigenvectors are aligned with the ellipse axes and the eigenvalues scale the lengths. The lengths of the sides of the box that are tangent to the ellipse are proportional to the square root of the diagonal elements of A^{-1}.

Singular or degenerate normal distributions. It is often convenient to extend the definition of the normal distribution to admit positive *semidefinite* covariance matrices. The distribution with a semidefinite covariance is known as a singular or degnerate normal distribution (An-

derson, 2003, p. 30). Figure A.11 shows a nearly singular normal distribution.

To see how the singular normal arises, let the scalar random variable ξ be distributed normally with zero mean and positive definite covariance, $\xi \sim N(0, P_x)$, and consider the simple linear transformation

$$\eta = A\xi \qquad A = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

in which we have created two identical copies of ξ for the two components η_1 and η_2 of η. Now consider the density of η. If we try to use the standard formulas for transformation of a normal, we would have

$$\eta \sim N(0, P_y) \qquad P_y = AP_xA' = \begin{bmatrix} P_x & P_x \\ P_x & P_x \end{bmatrix}$$

and P_y is singular since its rows are linearly dependent. Therefore one of the eigenvalues of P_y is zero and P_y is positive semidefinite and not positive definite. Obviously we cannot use (A.22) for the density in this case because the inverse of P_y does not exist. To handle these cases, we first provide an interpretation that remains valid when the covariance matrix is singular and semidefinite.

Definition A.37 (Density of a singular normal)**.** A singular joint normal density of random variables (ξ_1, ξ_2), $\xi_1 \in \mathbb{R}^{n_1}$, $\xi_2 \in \mathbb{R}^{n_2}$, is denoted

$$\begin{bmatrix} \xi_1 \\ \xi_2 \end{bmatrix} \sim N\left[\begin{bmatrix} m_1 \\ m_2 \end{bmatrix}, \begin{bmatrix} \Lambda_1 & 0 \\ 0 & 0 \end{bmatrix} \right]$$

with $\Lambda_1 > 0$. The density is defined by

$$p_\xi(x_1, x_2) = \frac{1}{(2\pi)^{\frac{n_1}{2}} (\det \Lambda_1)^{\frac{1}{2}}} \exp\left[-\frac{1}{2} |x_1 - m_1|^2_{\Lambda_1^{-1}} \right] \delta(x_2 - m_2) \tag{A.23}$$

In this limit, the "random" variable ξ_2 becomes deterministic and equal to its mean m_2. For the case $n_1 = 0$, we have the completely degenerate case in which $p_{\xi_2}(x_2) = \delta(x_2 - m_2)$, which describes the completely deterministic case $\xi_2 = m_2$ and there is no random component ξ_1. This expanded definition enables us to generalize the important result that the linear transformation of a normal is normal, so that it holds for *any* linear transformation, including rank deficient transformations such as the A matrix given above in which the rows

are not independent (see Exercise 1.40). Starting with the definition of a singular normal, we can obtain the density for $\xi \sim N(m_x, P_x)$ for any positive semidefinite $P_x \geq 0$. The result is

$$p_\xi(x) = \frac{1}{(2\pi)^{\frac{r}{2}}(\det \Lambda_1)^{\frac{1}{2}}} \exp\left[-\frac{1}{2} |(x - m_x)|^2_{Q_1} \right] \delta(Q_2'(x - m_x)) \tag{A.24}$$

in which matrices $\Lambda \in \mathbb{R}^{r \times r}$ and orthonormal $Q \in \mathbb{R}^{n \times n}$ are obtained from the eigenvalue decomposition of P_x

$$P_x = Q\Lambda Q' = \begin{bmatrix} Q_1 & Q_2 \end{bmatrix} \begin{bmatrix} \Lambda_1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} Q_1' \\ Q_2' \end{bmatrix}$$

and $\Lambda_1 > 0 \in \mathbb{R}^{r \times r}$, $Q_1 \in \mathbb{R}^{n \times r}$, $Q_2 \in \mathbb{R}^{n \times (n-r)}$. This density is nonzero for x satisfying $Q_2'(x - m_x) = 0$. If we let $N(Q_2')$ denote the r dimensional nullspace of Q_2', we have that the density is nonzero for $x \in N(Q_2') \oplus \{m_x\}$ in which $\oplus$ denotes set addition.

Example A.38: Marginal normal density

Given that ξ and η are jointly, normally distributed with mean

$$m = \begin{bmatrix} m_x \\ m_y \end{bmatrix}$$

and covariance matrix

$$P = \begin{bmatrix} P_x & P_{xy} \\ P_{yx} & P_y \end{bmatrix}$$

show that the marginal density of ξ is normal with the following parameters

$$\xi \sim N(m_x, P_x) \tag{A.25}$$

Solution

As a first approach to establish (A.25), we directly integrate the y variables. Let $\bar{x} = x - m_x$ and $\bar{y} = y - m_y$, and n_x and n_y be the dimension of the ξ and η variables, respectively, and $n = n_x + n_y$. Then the definition of the marginal density gives

$$p_\xi(x) = \frac{1}{(2\pi)^{n/2}(\det P)^{1/2}} \int_{-\infty}^{\infty} \exp\left[-\frac{1}{2} \begin{bmatrix} \bar{x} \\ \bar{y} \end{bmatrix}' \begin{bmatrix} P_x & P_{xy} \\ P_{yx} & P_y \end{bmatrix}^{-1} \begin{bmatrix} \bar{x} \\ \bar{y} \end{bmatrix} \right] d\bar{y}$$

Let the inverse of P be denoted as $\widetilde{P}$ and partition $\widetilde{P}$ as follows

$$\begin{bmatrix} P_x & P_{xy} \\ P_{yx} & P_y \end{bmatrix}^{-1} = \begin{bmatrix} \widetilde{P}_x & \widetilde{P}_{xy} \\ \widetilde{P}_{yx} & \widetilde{P}_y \end{bmatrix} \tag{A.26}$$

Substituting (A.26) into the definition of the marginal density and expanding the quadratic form in the exponential yields

$$(2\pi)^{n/2}(\det P)^{1/2} p_\xi(x) = \exp\left(-(1/2)\bar{x}'\widetilde{P}_x\bar{x}\right)\int_{-\infty}^{\infty}\exp\left(-(1/2)(2\bar{y}'\widetilde{P}_{yx}\bar{x} + \bar{y}'\widetilde{P}_y\bar{y})\right)d\bar{y}$$

We complete the square on the term in the integral by noting that

$$(\bar{y}+\widetilde{P}_y^{-1}\widetilde{P}_{yx}\bar{x})'\widetilde{P}_y(\bar{y}+\widetilde{P}_y^{-1}\widetilde{P}_{yx}\bar{x}) = \bar{y}'\widetilde{P}_y\bar{y}+2\bar{y}'\widetilde{P}_{yx}\bar{x}+\bar{x}'\widetilde{P}_{yx}'\widetilde{P}_y^{-1}\widetilde{P}_{yx}\bar{x}$$

Substituting this relation into the previous equation gives

$$(2\pi)^{n/2}(\det P)^{1/2} p_\xi(x) = \exp\left(-(1/2)\bar{x}'(\widetilde{P}_x - \widetilde{P}_{yx}'\widetilde{P}_y^{-1}\widetilde{P}_{yx})\bar{x}\right)\int_{-\infty}^{\infty}\exp\left(-(1/2)(\bar{y}+a)'\widetilde{P}_y(\bar{y}+a)\right)d\bar{y}$$

in which $a = \widetilde{P}_y^{-1}\widetilde{P}_{yx}\bar{x}$. Using (A.22) to evaluate the integral gives

$$p_\xi(x) = \frac{1}{(2\pi)^{n_x/2}\left(\det(P)\det(\widetilde{P}_y)\right)^{1/2}}\exp\left(-(1/2)\bar{x}'(\widetilde{P}_x - \widetilde{P}_{yx}'\widetilde{P}_y^{-1}\widetilde{P}_{yx})\bar{x}\right)$$

From the matrix inversion formula we conclude

$$\widetilde{P}_x - \widetilde{P}_{xy}'\widetilde{P}_y^{-1}\widetilde{P}_{yx} = P_x^{-1}$$

and

$$\det(P) = \det(P_x)\det(P_y - P_{yx}P_x^{-1}P_{xy}) = \det P_x \det\widetilde{P}_y^{-1} = \frac{\det P_x}{\det\widetilde{P}_y}$$

Substituting these results into the previous equation gives

$$p_\xi(x) = \frac{1}{(2\pi)^{n_x/2}(\det P_x)^{1/2}}\exp\left(-(1/2)\bar{x}'P_x^{-1}\bar{x}\right)$$

Therefore

$$\xi \sim N(m_x, P_x)$$

□

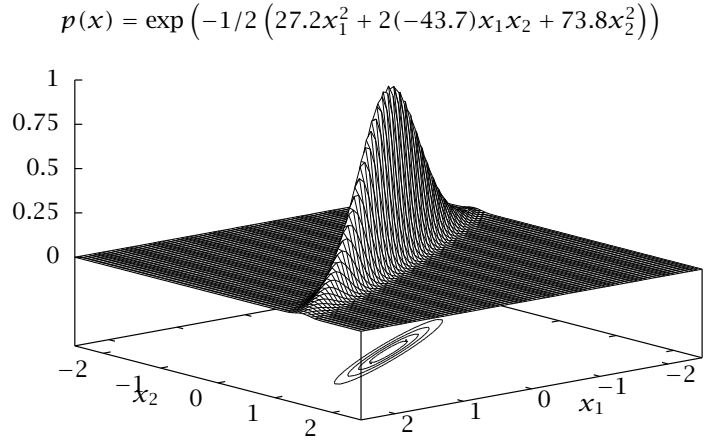


Figure A.11: A nearly singular normal density in two dimensions.

Functions of random variables. In stochastic dynamical systems we need to know how the density of a random variable is related to the density of a function of that random variable. Let $f : \mathbb{R}^n \to \mathbb{R}^n$ be a mapping of the random variable ξ into the random variable η and assume that the inverse mapping also exits

$$\eta = f(\xi), \qquad \xi = f^{-1}(\eta)$$

Given the density of $\xi, p_\xi(x)$, we wish to compute the density of η, $p_\eta(y)$, induced by the function f. Let S denote an arbitrary region of the field of the random variable ξ and define the set S' as the transform of this set under the function f

$$S' = \{y \mid y = f(x), x \in S\}$$

Then we seek a function $p_\eta(y)$ such that

$$\int_S p_\xi(x)dx = \int_{S'} p_\eta(y)dy \tag{A.27}$$

for every admissible set S. Using the rules of calculus for transforming a variable of integration we can write

$$\int_S p_\xi(x)dx = \int_{S'} p_\xi(f^{-1}(y)) \left| \det\left(\frac{\partial f^{-1}(y)}{\partial y}\right) \right| dy \tag{A.28}$$

in which $\left|\det(\partial f^{-1}(y)/\partial y)\right|$ is the absolute value of the determinant of the Jacobian matrix of the transformation from η to ξ. Subtracting (A.28) from (A.27) gives

$$\int_{S'} \left(p_\eta(y) - p_\xi(f^{-1}(y)) \left|\det(\partial f^{-1}(y)/\partial y)\right|\right) dy = 0 \tag{A.29}$$

Because (A.29) must be true for any set S', we conclude (a proof by contradiction is immediate)[8]

$$\boxed{p_\eta(y) = p_\xi(f^{-1}(y)) \left|\det(\partial f^{-1}(y)/\partial y)\right|} \tag{A.30}$$

Example A.39: Nonlinear transformation

Show that

$$p_\eta(y) = \frac{1}{3\sqrt{2\pi}\sigma y^{2/3}} \exp\left[-\frac{1}{2}\left(\frac{y^{1/3} - m}{\sigma}\right)^2\right]$$

is the density function of the random variable η under the transformation

$$\eta = \xi^3$$

for $\xi \sim N(m, \sigma^2)$. Notice that the density p_η is singular at $y = 0$. □

Noninvertible transformations. Given n random variables $\xi = (\xi_1, \xi_2, \ldots, \xi_n)$ with joint density p_ξ and k random variables $\eta = (\eta_1, \eta_2, \ldots, \eta_k)$ defined by the transformation $\eta = f(\xi)$

$$\eta_1 = f_1(\xi) \quad \eta_2 = f_2(\xi) \quad \cdots \quad \eta_k = f_k(\xi)$$

We wish to find p_η in terms of p_ξ. Consider the region generated in $\mathbb{R}^n$ by the vector inequality

$$f(x) \leq c$$

[8]Some care should be exercised if one has generalized functions in mind for the conditional density.

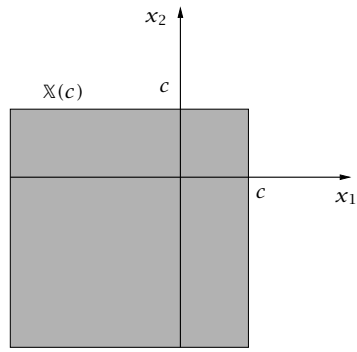


Figure A.12: The region $\mathbb{X}(c)$ for $y = \max(x_1, x_2) \leq c$.

Call this region $\mathbb{X}(c)$, which is by definition

$$\mathbb{X}(c) = \{x \mid f(x) \leq c\}$$

Note $\mathbb{X}$ is not necessarily simply connected. The probability distribution (not density) for η then satisfies

$$P_\eta(y) = \int_{\mathbb{X}(y)} p_\xi(x)dx \tag{A.31}$$

If the density p_η is of interest, it can be obtained by differentiating P_η.

Example A.40: Maximum of two random variables

Given two independent random variables, ξ_1, ξ_2 and the new random variable defined by the noninvertible, nonlinear transformation

$$\eta = \max(\xi_1, \xi_2)$$

Show that η's density is given by

$$p_\eta(y) = p_{\xi_1}(y) \int_{-\infty}^{y} p_{\xi_2}(x)dx + p_{\xi_2}(y) \int_{-\infty}^{y} p_{\xi_1}(x)dx$$

Solution

The region $\mathbb{X}(c)$ generated by the inequality $y = \max(x_1, x_2) \leq c$ is sketched in Figure A.12. Applying (A.31) then gives

$$\begin{aligned} P_\eta(y) &= \int_{-\infty}^{y} \int_{-\infty}^{y} p_\xi(x_1, x_2)dx_1dx_2 \\ &= P_\xi(y, y) \\ &= P_{\xi_1}(y)P_{\xi_2}(y) \end{aligned}$$

which has a clear physical interpretation. It says the probability that the *maximum* of two independent random variables is less than some value is equal to the probability that *both* random variables are less than that value. To obtain the density, we differentiate

$$\begin{aligned} p_\eta(y) &= p_{\xi_1}(y)P_{\xi_2}(y) + P_{\xi_1}(y)p_{\xi_2}(y) \\ &= p_{\xi_1}(y)\int_{-\infty}^{y} p_{\xi_2}(x)dx + p_{\xi_2}(y)\int_{-\infty}^{y} p_{\xi_1}(x)dx \end{aligned}$$

□

A.16.1 Statistical Independence and Correlation

We say two random variables ξ, η are **statistically independent** or simply independent if

$$p_{\xi,\eta}(x,y) = p_\xi(x)p_\eta(y), \qquad \text{all } x, y$$

The covariance of two random variables ξ, η is defined as

$$\text{cov}(\xi,\eta) = \mathcal{E}\left((\xi - \mathcal{E}(\xi))\,(\eta - \mathcal{E}(\eta))\right)$$

The covariance of the vector-valued random variable ξ with components $\xi_i, i = 1,\ldots n$ can be written as

$$P_{ij} = \text{cov}(\xi_i,\xi_j)$$

$$P = \begin{bmatrix} \text{var}(\xi_1) & \text{cov}(\xi_1,\xi_2) & \cdots & \text{cov}(\xi_1,\xi_n) \\ \text{cov}(\xi_2,\xi_1) & \text{var}(\xi_2) & \cdots & \text{cov}(\xi_2,\xi_n) \\ \vdots & \vdots & \ddots & \vdots \\ \text{cov}(\xi_n,\xi_1) & \text{cov}(\xi_n,\xi_2) & \cdots & \text{var}(\xi_n) \end{bmatrix}$$

We say two random variables, ξ and η, are **uncorrelated** if

$$\text{cov}(\xi,\eta) = 0$$

Example A.41: Independent implies uncorrelated

Prove that if ξ and η are statistically independent, then they are uncorrelated.

Solution

The definition of covariance gives

$$\begin{aligned}\operatorname{cov}(\xi,\eta) &= \mathcal{E}((\xi-\mathcal{E}(\xi))(\eta-\mathcal{E}(\eta)))\\ &= \mathcal{E}(\xi\eta-\xi\mathcal{E}(\eta)-\eta\mathcal{E}(\xi)+\mathcal{E}(\xi)\mathcal{E}(\eta))\\ &= \mathcal{E}(\xi\eta)-\mathcal{E}(\xi)\mathcal{E}(\eta)\end{aligned}$$

Taking the expectation of the product $\xi\eta$ and using the fact that ξ and η are independent gives

$$\begin{aligned}\mathcal{E}(\xi\eta) &= \int_{-\infty}^{\infty}\!\!\int xyp_{\xi,\eta}(x,y)dxdy\\ &= \int_{-\infty}^{\infty}\!\!\int xyp_{\xi}(x)p_{\eta}(y)dxdy\\ &= \int_{-\infty}^{\infty} xp_{\xi}(x)dx\int_{-\infty}^{\infty} yp_{\eta}(y)dy\\ &= \mathcal{E}(\xi)\mathcal{E}(\eta)\end{aligned}$$

Substituting this fact into the covariance equation gives

$$\operatorname{cov}(\xi,\eta)=0$$

□

Example A.42: Does uncorrelated imply independent?

Let ξ and η be jointly distributed random variables with probability density function

$$p_{\xi,\eta}(x,y)=\begin{cases}\frac{1}{4}[1+xy(x^2-y^2)], & |x|<1, \quad |y|<1\\ 0, & \text{otherwise}\end{cases}$$

(a) Compute the marginals $p_{\xi}(x)$ and $p_{\eta}(y)$. Are ξ and η independent?

(b) Compute $\operatorname{cov}(\xi,\eta)$. Are ξ and η uncorrelated?

(c) What is the relationship between independent and uncorrelated? Are your results on this example consistent with this relationship? Why or why not?

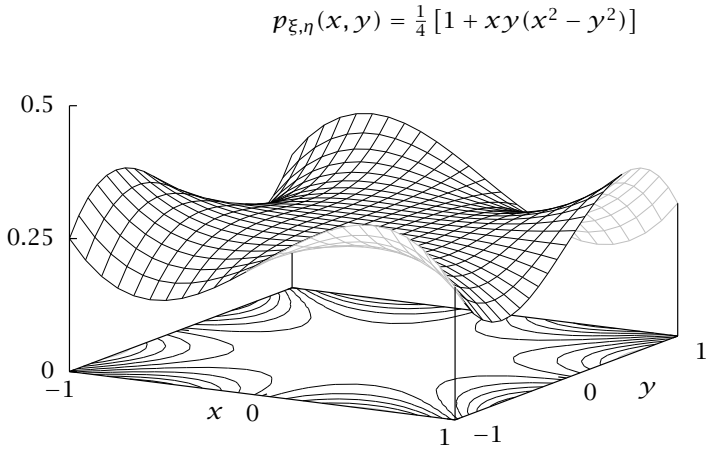


Figure A.13: A joint density function for the two uncorrelated random variables in Example A.42.

Solution

The joint density is shown in Figure A.13.

(a) Direct integration of the joint density produces

$$p_\xi(x) = (1/2), \quad |x| < 1 \qquad \mathcal{E}(\xi) = 0$$
$$p_\eta(y) = (1/2), \quad |y| < 1 \qquad \mathcal{E}(\eta) = 0$$

and we see that both marginals are zero mean, uniform densities. Obviously ξ and η are not independent because the joint density is not the product of the marginals.

(b) Performing the double integral for the expectation of the product term gives

$$\mathcal{E}(\xi\eta) = \int_{-1}^{1}\int xy + (xy)^2(x^2 - y^2)dxdy$$
$$= 0$$

and the covariance of ξ and η is therefore

$$\begin{aligned}\operatorname{cov}(\xi,\eta) &= \mathcal{E}(\xi\eta) - \mathcal{E}(\xi)\mathcal{E}(\eta)\\ &= 0\end{aligned}$$

and ξ and η are uncorrelated.

(c) We know independent implies uncorrelated. This example does not contradict that relationship. This example shows uncorrelated does not imply independent, in general, but see the next example for normals.

□

Example A.43: Independent and uncorrelated are equivalent for normals

If two random variables are jointly normally distributed,

$$\begin{bmatrix}\xi\\ \eta\end{bmatrix} \sim N\left(\begin{bmatrix}m_x\\ m_y\end{bmatrix}, \begin{bmatrix}P_x & P_{xy}\\ P_{yx} & P_y\end{bmatrix}\right)$$

Prove ξ and η are statistically independent if and only if ξ and η are uncorrelated, or, equivalently, P is block diagonal.

Solution

We have already shown that independent implies uncorrelated for any density, so we now show that, *for normals*, uncorrelated implies independent. Given $\operatorname{cov}(\xi,\eta) = 0$, we have

$$P_{xy} = P'_{yx} = 0 \qquad \det P = \det P_x \det P_y$$

so the density can be written

$$p_{\xi,\eta}(x,y) = \frac{\exp\left(-\frac{1}{2}\begin{bmatrix}\bar{x}\\ \bar{y}\end{bmatrix}'\begin{bmatrix}P_x & 0\\ 0 & P_y\end{bmatrix}^{-1}\begin{bmatrix}\bar{x}\\ \bar{y}\end{bmatrix}\right)}{(2\pi)^{(n_x+n_y)/2}\left(\det P_x \det P_y\right)^{1/2}} \tag{A.32}$$

For any joint normal, we know the marginals are simply

$$\xi \sim N(m_x, P_x) \qquad \eta \sim N(m_y, P_y)$$

so we have

$$p_\xi(x) = \frac{1}{(2\pi)^{n_x/2}(\det P_x)^{1/2}} \exp\left(-(1/2)\bar{x}' P_x^{-1}\bar{x}\right)$$
$$p_\eta(y) = \frac{1}{(2\pi)^{n_y/2}(\det P_y)^{1/2}} \exp\left(-(1/2)\bar{y}' P_y^{-1}\bar{y}\right)$$

Forming the product and combining terms gives

$$p_\xi(x)p_\eta(y) = \frac{\exp\left(-\frac{1}{2}\begin{bmatrix}\bar{x}\\ \bar{y}\end{bmatrix}'\begin{bmatrix}P_x^{-1} & 0\\ 0 & P_y^{-1}\end{bmatrix}\begin{bmatrix}\bar{x}\\ \bar{y}\end{bmatrix}\right)}{(2\pi)^{(n_x+n_y)/2}\left(\det P_x \det P_y\right)^{1/2}}$$

Comparing this equation to (A.32), and using the inverse of a block-diagonal matrix, we have shown that ξ and η are statistically independent. □

A.17 Conditional Probability and Bayes's Theorem

Let ξ and η be jointly distributed random variables with density $p_{\xi,\eta}(x, y)$. We seek the density function of ξ given a specific realization y of η has been observed. We define the conditional density function as

$$p_{\xi|\eta}(x|y) = \frac{p_{\xi,\eta}(x,y)}{p_\eta(y)}$$

Consider a roll of a single die in which η takes on values E or O to denote whether the outcome is even or odd and ξ is the integer value of the die. The twelve values of the joint density function are simply computed

$$\begin{array}{rclrcl}
p_{\xi,\eta}(1,\mathrm{E}) &=& 0 & p_{\xi,\eta}(1,\mathrm{O}) &=& 1/6\\
p_{\xi,\eta}(2,\mathrm{E}) &=& 1/6 & p_{\xi,\eta}(2,\mathrm{O}) &=& 0\\
p_{\xi,\eta}(3,\mathrm{E}) &=& 0 & p_{\xi,\eta}(3,\mathrm{O}) &=& 1/6\\
p_{\xi,\eta}(4,\mathrm{E}) &=& 1/6 & p_{\xi,\eta}(4,\mathrm{O}) &=& 0\\
p_{\xi,\eta}(5,\mathrm{E}) &=& 0 & p_{\xi,\eta}(5,\mathrm{O}) &=& 1/6\\
p_{\xi,\eta}(6,\mathrm{E}) &=& 1/6 & p_{\xi,\eta}(6,\mathrm{O}) &=& 0
\end{array} \tag{A.33}$$

The marginal densities are then easily computed; we have for ξ

$$p_\xi(x) = \sum_{y=\mathrm{O}}^{\mathrm{E}} p_{\xi,\eta}(x,y)$$

which gives by summing across rows of (A.33)

$$p_\xi(x) = 1/6, \qquad x = 1, 2, \ldots 6$$

Similarly, we have for η

$$p_\eta(y) = \sum_{x=1}^{6} p_{\xi,\eta}(x, y)$$

which gives by summing down the columns of (A.33)

$$p_\eta(y) = 1/2, \qquad y = \mathrm{E}, \mathrm{O}$$

These are both in accordance of our intuition on the rolling of the die: uniform probability for each value 1 to 6 and equal probability for an even or an odd outcome. Now the conditional density is a different concept. The conditional density $p_{\xi|\eta}(x, y)$ tells us the density of x given that $\eta = y$ has been observed. So consider the value of this function

$$p_{\xi|\eta}(1|\mathrm{O})$$

which tells us the probability that the die has a 1 given that we know that it is odd. We expect that the additional information on the die being odd causes us to revise our probability that it is 1 from 1/6 to 1/3. Applying the defining formula for conditional density indeed gives

$$p_{\xi|\eta}(1|\mathrm{O}) = p_{\xi,\eta}(1, \mathrm{O})/p_\eta(\mathrm{O}) = \frac{1/6}{1/2} = 1/3$$

Consider the reverse question, the probability that we have an odd given that we observe a 1. The definition of conditional density gives

$$p_{\eta,\xi}(\mathrm{O}|1) = p_{\eta,\xi}(\mathrm{O}, 1)/p_\xi(1) = \frac{1/6}{1/6} = 1$$

i.e., we are sure the die is odd if it is 1. Notice that the arguments to the conditional density do not commute as they do in the joint density.

This fact leads to a famous result. Consider the definition of conditional density, which can be expressed as

$$p_{\xi,\eta}(x, y) = p_{\xi|\eta}(x|y)p_\eta(y)$$

or

$$p_{\eta,\xi}(y, x) = p_{\eta|\xi}(y|x)p_\xi(x)$$

Because $p_{\xi,\eta}(x,y) = p_{\eta,\xi}(y,x)$, we can equate the right-hand sides and deduce

$$p_{\xi|\eta}(x|y) = \frac{p_{\eta|\xi}(y|x)p_{\xi}(x)}{p_{\eta}(y)}$$

which is known as Bayes's theorem (Bayes, 1763). Notice that this result comes in handy whenever we wish to switch the variable that is known in the conditional density, which we will see is a key step in state estimation problems.

Example A.44: Conditional normal density

Show that if ξ and η are jointly normally distributed as

$$\begin{bmatrix} \xi \\ \eta \end{bmatrix} \sim N\left(\begin{bmatrix} m_x \\ m_y \end{bmatrix}, \begin{bmatrix} P_x & P_{xy} \\ P_{yx} & P_y \end{bmatrix}\right)$$

then the conditional density of ξ given η is also normal

$$(\xi|\eta) \sim N(m,P)$$

in which the mean is

$$m = m_x + P_{xy}P_y^{-1}(y - m_y) \tag{A.34}$$

and the covariance is

$$P = P_x - P_{xy}P_y^{-1}P_{yx} \tag{A.35}$$

Solution

The definition of conditional density gives

$$p_{\xi|\eta}(x|y) = \frac{p_{\xi,\eta}(x,y)}{p_{\eta}(y)}$$

Because (ξ,η) is jointly normal, we know from Example A.38

$$p_{\eta}(y) = \frac{1}{(2\pi)^{n_\eta/2}(\det P_y)^{1/2}} \exp\left(-(1/2)(y - m_y)'P_y^{-1}(y - m_y)\right)$$

and therefore

$$p_{\xi|\eta}(x|y) = \frac{(\det P_y)^{1/2}}{(2\pi)^{n_\xi/2}\left(\det\begin{bmatrix} P_x & P_{xy} \\ P_{yx} & P_y \end{bmatrix}\right)^{1/2}} \exp(-1/2a) \tag{A.36}$$

in which the argument of the exponent is

$$a = \begin{bmatrix} x - m_x \\ y - m_y \end{bmatrix}' \begin{bmatrix} P_x & P_{xy} \\ P_{yx} & P_y \end{bmatrix}^{-1} \begin{bmatrix} x - m_x \\ y - m_y \end{bmatrix} - (y - m_y)' P_y^{-1} (y - m_y)$$

If we use $P = P_x - P_{xy} P_y^{-1} P_{yx}$ as defined in (A.35) then we can use the partitioned matrix inversion formula to express the matrix inverse in the previous equation as

$$\begin{bmatrix} P_x & P_{xy} \\ P_{yx} & P_y \end{bmatrix}^{-1} = \begin{bmatrix} P^{-1} & -P^{-1} P_{xy} P_y^{-1} \\ -P_y^{-1} P_{yx} P^{-1} & P_y^{-1} + P_y^{-1} P_{yx} P^{-1} P_{xy} P_y^{-1} \end{bmatrix}$$

Substituting this expression and multiplying out terms yields

$$\begin{aligned} a = {} & (x - m_x)' P^{-1} (x - m_x) - 2(y - m_y)' (P_y^{-1} P_{yx} P^{-1}) (x - m_x) \\ & + (y - m_y)' (P_y^{-1} P_{yx} P^{-1} P_{xy} P_y^{-1}) (y - m_y) \end{aligned}$$

which is the expansion of the following quadratic term

$$a = \left[(x - m_x) - P_{xy} P_y^{-1} (y - m_y) \right]' P^{-1} \left[(x - m_x) - P_{xy} P_y^{-1} (y - m_y) \right]$$

in which we use the fact that $P_{xy} = P_{yx}'$. Substituting (A.34) into this expression yields

$$a = (x - m)' P^{-1} (x - m) \tag{A.37}$$

Finally noting that for the partitioned matrix

$$\det \begin{bmatrix} P_x & P_{xy} \\ P_{yx} & P_y \end{bmatrix} = \det P_y \det P \tag{A.38}$$

and substitution of equations (A.38) and (A.37) into (A.36) yields

$$p_{\xi|\eta}(x|y) = \frac{1}{(2\pi)^{n_\xi/2} (\det P)^{1/2}} \exp\left(-\frac{1}{2} (x - m)' P^{-1} (x - m) \right)$$

which is the desired result. □

Example A.45: More normal conditional densities

Let the joint conditional of random variables a and b given c be a normal distribution with

$$p(a, b|c) \sim N\left(\begin{bmatrix} m_a \\ m_b \end{bmatrix}, \begin{bmatrix} P_a & P_{ab} \\ P_{ba} & P_b \end{bmatrix} \right) \tag{A.39}$$

Then the conditional density of a given b and c is also normal

$$p(a|b,c) \sim N(m,P)$$

in which the mean is

$$m = m_a + P_{ab}P_b^{-1}(b - m_b)$$

and the covariance is

$$P = P_a - P_{ab}P_b^{-1}P_{ba}$$

Solution

From the definition of joint density we have

$$p(a|b,c) = \frac{p(a,b,c)}{p(b,c)}$$

Multiplying the top and bottom of the fraction by $p(c)$ yields

$$p(a|b,c) = \frac{p(a,b,c)}{p(c)}\frac{p(c)}{p(b,c)}$$

or

$$p(a|b,c) = \frac{p(a,b|c)}{p(b|c)}$$

Substituting the distribution given in (A.39) and using the result in Example A.38 to evaluate $p(b|c)$ yields

$$p(a|b,c) = \frac{N\left(\begin{bmatrix} m_a \\ m_b \end{bmatrix}, \begin{bmatrix} P_a & P_{ab} \\ P_{ba} & P_b \end{bmatrix}\right)}{N(m_b, P_b)}$$

And now applying the methods of Example A.44 this ratio of normal distributions reduces to the desired expression. □

Adjoint operator. Given a linear operator $\mathcal{G} : \mathbb{U} \to \mathbb{V}$ and inner products for the spaces $\mathbb{U}$ and $\mathbb{V}$, the adjoint of $\mathcal{G}$, denoted by $\mathcal{G}^*$ is the linear operator $\mathcal{G}^* : \mathbb{V} \to \mathbb{U}$ such that

$$\langle u, \mathcal{G}^* v\rangle = \langle \mathcal{G}u, v\rangle, \qquad \forall u \in \mathbb{U}, v \in \mathbb{V} \tag{A.40}$$

Dual dynamic system (Callier and Desoer, 1991). The dynamic system

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k), \qquad k = 0, \ldots, N-1 \\ y(k) &= Cx(k) + Du(k) \end{aligned}$$

maps an initial condition and input sequence $(x(0), u(0), \ldots, u(N-1))$ into a final condition and an output sequence $(x(N), y(0), \ldots, y(N-1))$. Call this linear operator $\mathcal{G}$

$$\begin{bmatrix} x(N) \\ y(0) \\ \vdots \\ y(N-1) \end{bmatrix} = \mathcal{G} \begin{bmatrix} x(0) \\ u(0) \\ \vdots \\ u(N-1) \end{bmatrix}$$

The dual dynamic system represents the adjoint operator $\mathcal{G}^*$

$$\begin{bmatrix} \overline{x}(0) \\ \overline{y}(1) \\ \vdots \\ \overline{y}(N) \end{bmatrix} = \mathcal{G}^* \begin{bmatrix} \overline{x}(N) \\ \overline{u}(1) \\ \vdots \\ \overline{u}(N) \end{bmatrix}$$

We define the usual inner product, $\langle a, b \rangle = a'b$, and substitute into (A.40) to obtain

$$\underbrace{x(0)'\overline{x}(0) + u(0)'\overline{y}(1) + \cdots + u(N-1)'\overline{y}(N)}_{\langle u, G^* v \rangle} - \underbrace{x(N)'\overline{x}(N) + y(0)'\overline{u}(1) + \cdots + y(N-1)'\overline{u}(N)}_{\langle Gu, v \rangle} = 0$$

If we express the $y(k)$ in terms of $x(0)$ and $u(k)$ and collect terms we obtain

$$\begin{aligned} 0 = {} & x(0)' \left[\overline{x}(0) - C'\overline{u}(1) - A'C'\overline{u}(2) - \cdots - A'^N \overline{x}(N) \right] \\ & + u(0)' \left[\overline{y}(1) - D'\overline{u}(1) - B'C'\overline{u}(2) - \cdots - B'A'^{(N-2)}C'\overline{u}(N) - B'A'^{(N-1)}\overline{x}(N) \right] \\ & + \cdots \\ & + u(N-2)' \left[\overline{y}(N-1) - D'\overline{u}(N-1) - B'C'\overline{u}(N) - B'A'\overline{x}(N) \right] \\ & + u(N-1)' \left[\overline{y}(N) - D'\overline{u}(N) - B'\overline{x}(N) \right] \end{aligned}$$

Since this equation must hold for all $(x(0), u(0), \ldots, u(N-1))$, each term in brackets must vanish. From the $u(N-1)$ term we conclude

$$\overline{y}(N) = B'\overline{x}(N) + D'\overline{u}(N)$$

Using this result, the $u(N-2)$ term gives

$$B'\left(\overline{x}(N-1) - \left(A'\overline{x}(N) + C'\overline{u}(N)\right)\right) = 0$$

From which we find the state recursion for the dual system

$$\overline{x}(N-1) = A'\overline{x}(N) + C'\overline{u}(N)$$

Passing through each term then yields the dual state space description of the adjoint operator $\mathcal{G}^*$

$$\begin{aligned}\overline{x}(k-1) &= A'\overline{x}(k) + C'\overline{u}(k), \qquad k = N,\ldots,1\\ \overline{y}(k) &= B'\overline{x}(k) + D'\overline{u}(k)\end{aligned}$$

So the primal and dual dynamic systems change matrices in the following way

$$(A,B,C,D) \longrightarrow (A',C',B',D')$$

Notice this result produces the duality variables listed in Table A.1 if we first note that we have also renamed the regulator's input matrix B to G in the estimation problem. We also note that time runs in the opposite directions in the dynamic system and the dual dynamic system, which corresponds to the fact that the Riccati equation iterations run in opposite directions in the regulation and estimation problems.

A.18 Exercises

Exercise A.1: Norms in $\mathbb{R}^n$

Show that the following three functions are all norms in $\mathbb{R}^n$

$$\begin{aligned}|x|_2 &:= \left(\sum_{i=1}^{n}(x^i)^2\right)^{1/2}\\ |x|_\infty &:= \max\{\left|x^1\right|, \left|x^2\right|, \ldots, |x^n|\}\\ |x|_1 &:= \sum_{i=1}^{n}\left|x^j\right|\end{aligned}$$

where x^j denotes the jth component of the vector x.

Exercise A.2: Equivalent norms

Show that there are finite constants $K_{ij}, i,j = 1,2,\infty$ such that

$$|x|_i \le K_{ij}\,|x|_j\,, \text{ for all } i,j \in \{1,2,\infty\}.$$

This result shows that the norms are *equivalent* and may be used interchangeably for establishing that sequences are convergent, sets are open or closed, etc.

Regulator	Estimator
A	A'
B	C'
C	G'
k	$l = N - k$
$\Pi(k)$	$P^-(l)$
$\Pi(k-1)$	$P^-(l+1)$
Π	P^-
Q	Q
R	R
$Q(N)$	$Q(0)$
K	$-\widetilde{L}'$
$A + BK$	$(A - \widetilde{L}C)'$
x	ε

Regulator	Estimator
$R > 0, \quad Q > 0$	$R > 0, \quad Q > 0$
(A, B) stabilizable	(A, C) detectable
(A, C) detectable	(A, G) stabilizable

Table A.1: Duality variables and stability conditions for linear quadratic regulation and linear estimation.

Exercise A.3: Open and closed balls

Let $x \in \mathbb{R}^n$ and $\rho > 0$ be given. Show that $\{z \mid |z - x| < \rho\}$ is open and that $B(x, \rho)$ is closed.

Exercise A.4: Condition for closed set

Show that $X \subset \mathbb{R}^n$ is closed if and only if $\text{int}(B(x, \rho)) \cap X \neq \emptyset$ for all $\rho > 0$ implies $x \in X$.

Exercise A.5: Convergence

Suppose that $x_i \to \hat{x}$ as $i \to \infty$; show that for every $\rho > 0$ there exists an $i_p \in \mathbb{I}_{\geq 0}$ such that $x_i \in B(\hat{x}, \rho)$ for all $i \geq i_p$.

Exercise A.6: Limit is unique

Suppose that $\hat{x}, \hat{x}'$ are limits of a sequence $(x_i)_{i \in \mathbb{I}_{\geq 0}}$. Show that $\hat{x} = \hat{x}'$.

Exercise A.7: Open and closed sets

(a) Show that a set $X \subset \mathbb{R}^n$ is open if and only if, for any $\hat{x} \in X$ and any sequence $(x_i) \subset \mathbb{R}^n$ such that $x_i \to \hat{x}$ as $i \to \infty$, there exists a $q \in \mathbb{I}_{\geq 0}$ such that $x_i \in X$ for all $i \geq q$.

(b) Show that a set $X \subset \mathbb{R}^n$ is closed if and only if for all $(x_i) \subset X$, if $x_i \to \hat{x}$ as $i \to \infty$, then $\hat{x} \in X$, i.e., a set X is closed if and only if it contains the limit of every convergent sequences lying in X.

Exercise A.8: Decreasing and bounded below

Prove the observation at the end of Section A.10 that a monotone decreasing sequence that is bounded below converges.

Exercise A.9: Continuous function

Show that $f : \mathbb{R}^n \to \mathbb{R}^m$ is continuous at $\hat{x}$ implies $f(x_i) \to f(\hat{x})$ for any sequence (x_i) satisfying $x_i \to \hat{x}$ as $i \to \infty$.

Exercise A.10: Alternative proof of existence of minimum of continuous function on compact set

Prove Proposition A.7 by making use of the fact that $f(X)$ is compact.

Exercise A.11: Differentiable implies Lipschitz

Suppose that $f : \mathbb{R}^n \to \mathbb{R}^m$ has a continuous derivative $f_x(\cdot)$ in a neighborhood of $\hat{x}$. Show that f is locally Lipschitz continuous at $\hat{x}$.

Exercise A.12: Continuous, Lipschitz continuous, and differentiable

Provide examples of functions meeting the following conditions.

1. Continuous but not Lipschitz continuous.
2. Lipschitz continuous but not differentiable.

Exercise A.13: Differentiating quadratic functions and time-varying matrix inverses

(a) Show that $\nabla f(x) = Qx$ if $f(x) = (1/2)x'Qx$ and Q is symmetric.

(b) Show that $(d/dt)A^{-1}(t) = -A^{-1}(t)\dot{A}(t)A^{-1}(t)$ if $A : \mathbb{R} \to \mathbb{R}^{n\times n}$, $A(t)$ is invertible for all $t \in \mathbb{R}$, and $\dot{A}(t) := (d/dt)A(t)$.

Exercise A.14: Directional derivative

Suppose that $f : \mathbb{R}^n \to \mathbb{R}^m$ has a derivative $f_x(\hat{x})$ at $\hat{x}$. Show that for any h, the directional derivative $df(\hat{x}; h)$ exists and is given by

$$df(\hat{x}; h) = f_x(\hat{x})h = (\partial f(x)/\partial x)h.$$

Exercise A.15: Convex combination

Suppose $S \subset \mathbb{R}^n$ is convex. Let $\{x_i\}_{i=1}^k$ be points in S and let $\{\mu^i\}_{i=1}^k$ be scalars such that $\mu^i \geq 0$ for $i = 1, 2, \ldots, k$ and $\sum_{i=1}^k \mu^i = 1$. Show that

$$\left(\sum_{i=1}^{k} \mu^i x_i\right) \in S.$$

Exercise A.16: Convex epigraph

Show that $f : \mathbb{R}^n \to \mathbb{R}$ is convex if and only if its epigraph is convex.

Exercise A.17: Bounded second derivative and minimum

Suppose that $f : \mathbb{R}^n \to \mathbb{R}$ is twice continuously differentiable and that for some $\infty > M \geq m > 0$, $M \left|y\right|^2 \geq \langle y, \partial^2 f / \partial x^2(x) y \rangle \geq m \left|y\right|^2$ for all $x, y \in \mathbb{R}^n$. Show that the sublevel sets of f are convex and compact and that $f(\cdot)$ attains its infimum.

Exercise A.18: Sum and max of convex functions are convex

Suppose that $f_i : \mathbb{R}^n \to \mathbb{R}, i = 1, 2, \ldots, m$ are convex. Show that

$$\psi^1(x) \coloneqq \max_i \{f_i(x) \mid i \in \{1, 2, \ldots, m\}\},$$

$$\psi^2(x) \coloneqq \sum_{i=1}^{m} f_i(x)$$

are both convex.

Exercise A.19: Einige kleine Mathprobleme

(a) Prove that if λ is an eigenvalue and v is an eigenvector of A ($Av = \lambda v$), then λ is also an eigenvalue of T in which T is upper triangular and given by the Schur decomposition of A

$$Q^* A Q = T$$

What is the corresponding eigenvector?

(b) Prove statement 1 on positive definite matrices (from Section A.7). Where is this fact needed?

(c) Prove statement 6 on positive definite matrices. Where is this fact needed?

(d) Prove statement 5 on positive definite matrices.

(e) Prove statement 8 on positive semidefinite matrices.

(f) Derive the two expressions for the partitioned A^{-1}.

Exercise A.20: Positive definite but not symmetric matrices

Consider redefining the notation $A > 0$ for $A \in \mathbb{R}^{n \times n}$ to mean $x'Ax > 0$ for all $x \in \mathbb{R}^n \neq 0$. In other words, the restriction that A is symmetric in the usual definition of positive definiteness is removed. Consider also $B \coloneqq (A + A')/2$. Show the following hold for all A. (a) $A > 0$ if and only if B is positive definite. (b) $\mathrm{tr}(A) = \mathrm{tr}(B)$. (Johnson, 1970; Johnson and Hillar, 2002)

Exercise A.21: Trace of a matrix function

Derive the following formula for differentiating the trace of a function of a square matrix

$$\frac{d \,\mathrm{tr}(f(A))}{dA} = g(A') \qquad g(x) = \frac{df(x)}{dx}$$

in which g is the usual scalar derivative of the scalar function f. This result proves useful in evaluating the change in the expectation of the stage cost in stochastic control problems.

Exercise A.22: Some matrix differentiation

Derive the following formulas (Bard, 1974). $A, B \in \mathbb{R}^{n\times n}, a, x \in \mathbb{R}^n$.

(a)
$$\frac{\partial x' Ax}{\partial x} = Ax + A'x$$

(b)
$$\frac{\partial Axa'Bx}{\partial x'} = (a'Bx)A + Axa'B$$

(c)
$$\frac{\partial a'Ab}{\partial A} = ab'$$

Exercise A.23: Partitioned matrix inversion formula

In deriving the partitioned matrix inversion formula we assumed A is partitioned into

$$A = \begin{bmatrix} B & C \\ D & E \end{bmatrix}$$

and that A^{-1}, B^{-1} and E^{-1} exist. In the final formula, the term

$$(E - DB^{-1}C)^{-1}$$

appears, but we did not assume this matrix is invertible. Did we leave out an assumption or can the existence of this matrix inverse be proven given the other assumptions? If we left out an assumption, provide an example in which this matrix is not invertible. If it follows from the other assumptions, prove this inverse exists.

Exercise A.24: Partitioned positive definite matrices

Consider the partitioned positive definite, symmetric matrix

$$H = \begin{bmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{bmatrix}$$

Prove that the following matrices are also positive definite

1. H_{11}
2. H_{22}
3. $\overline{H}$ in which
$$\overline{H} = \begin{bmatrix} H_{11} & -H_{12} \\ -H_{21} & H_{22} \end{bmatrix}$$
4. $H_{11} - H_{12}H_{22}^{-1}H_{21}$ and $H_{22} - H_{21}H_{11}^{-1}H_{12}$

Exercise A.25: Properties of the matrix exponential

Prove that the following properties of the matrix exponential, which are useful for dealing with continuous time linear systems. The matrix A is a real-valued $n \times n$ matrix, and t is real.

(a)
$$\text{rank}\left(e^{At}\right) = n \quad \forall t$$

(b)

$$\text{rank}\left(\int_0^t e^{A\tau}d\tau\right) = n \quad \forall t > 0$$

Exercise A.26: Controllability in continuous time

A linear, time-invariant, continuous time system

$$\begin{aligned}\frac{dx}{dt} &= Ax + Bu \\ x(0) &= x_0\end{aligned} \tag{A.41}$$

is **controllable** if there exists an input $u(t), 0 \le t \le t_1, t_1 > 0$ that takes the system from any x_0 at time zero to any x_1 at some finite time t_1.

(a) Prove that the system in (A.41) is controllable if and only if

$$\text{rank}\,(C) = n$$

in which C is, remarkably, the same controllability matrix that was defined for discrete time systems 1.16

$$C = \begin{bmatrix} B & AB & \cdots & A^{n-1}B \end{bmatrix}$$

(b) Describe a calculational procedure for finding this required input.

Exercise A.27: Reachability Gramian in continuous time

Consider the symmetric, $n \times n$ matrix W defined by

$$W(t) = \int_0^t e^{(t-\tau)A} BB' e^{(t-\tau)A'} d\tau$$

The matrix W is known as the reachability Gramian of the linear, time-invariant system. The reachability Gramian proves useful in analyzing controllability and reachability. Prove the following important properties of the reachability Gramian.

(a) The reachability Gramian satisfies the following matrix differential equation

$$\begin{aligned}\frac{dW}{dt} &= BB' + AW + WA' \\ W(0) &= 0\end{aligned}$$

which provides one useful way to calculate its values.

(b) The reachability Gramian $W(t)$ is full rank for all $t > 0$ if and only if the system is controllable.

Exercise A.28: Differences in continuous time and discrete time systems

Consider the definition that a system is controllable if there exists an input that takes the system from any x_0 at time zero to any x_1 at some finite time t_1.

(a) Show that x_1 can be taken as zero without changing the meaning of controllability for a linear continuous time system.

(b) In linear discrete time systems, x_1 cannot be taken as zero without changing the meaning of controllability. Why not? Which A require a distinction in discrete time. What are the eigenvalues of the corresponding A in continuous time?

Exercise A.29: Observability in continuous time

Consider the linear time-invariant continuous time system

$$\begin{aligned} \frac{dx}{dt} &= Ax \\ x(0) &= x_0 \\ y &= Cx \end{aligned} \tag{A.42}$$

and let $y(t;x_0)$ represent the solution to (A.42) as a function of time t given starting state value x_0 at time zero. Consider the output from two different initial conditions $y(t;w), \quad y(t;z)$ on the time interval $0 \le t \le t_1$ with $t_1 > 0$.

The system in (A.42) is **observable** if

$$y(t;w) = y(t;z), \quad 0 \le t \le t_1 \Longrightarrow w = z$$

In other words, if two output measurement trajectories agree, the initial conditions that generated the output trajectories must agree, and hence, the initial condition is unique. This uniqueness of the initial condition allows us to consider building a state estimator to reconstruct $x(0)$ from $y(t;x_0)$. After we have found the unique $x(0)$, solving the model provides the rest of the state trajectory $x(t)$. We will see later that this procedure is not the preferred way to build a state estimator; it simply shows that if the system is observable, the goal of state estimation is reasonable.

Show that the system in (A.42) is observable if and only if

$$\text{rank}\,(\mathcal{O}) = n$$

in which $\mathcal{O}$ is, again, the same observability matrix that was defined for discrete time systems 1.36

$$\mathcal{O} = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix}$$

Hint: what happens if you differentiate $y(t;w) - y(t;z)$ with respect to time? How many times is this function differentiable?

Exercise A.30: Observability Gramian in continuous time

Consider the symmetric, $n \times n$ matrix W_o defined by

$$W_o(t) = \int_0^t e^{A'\tau} C' C e^{A\tau} d\tau$$

The matrix W_o is known as the observability Gramian of the linear, time-invariant system. Prove the following important properties of the observability Gramian.

(a) The observability Gramian $W_o(t)$ is full rank for all $t > 0$ if and only if the system is observable.

(b) Consider an observable linear time invariant system with $u(t) = 0$ so that $y(t) = Ce^{At}x_0$. Use the observability Gramian to solve this equation for x_0 as a function of $y(t), 0 \le t \le t_1$.

(c) Extend your result from the previous part to find x_0 for an arbitrary $u(t)$.

Exercise A.31: Detectability of (A, C) and output penalty

Given a system

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k) \end{aligned}$$

Suppose (A, C) is detectable and an input sequence has been found such that

$$u(k) \to 0 \qquad y(k) \to 0$$

Show that $x(k) \to 0$.

Exercise A.32: Prove your favorite Hautus lemma

Prove the Hautus lemma for controllability, Lemma 1.2, or observability, Lemma 1.4.

Exercise A.33: Positive semidefinite Q penalty and its square root

Consider the linear quadratic problem with system

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Q^{1/2}x(k) \end{aligned}$$

and infinite horizon cost function

$$\begin{aligned} \Phi &= \sum_{k=0}^{\infty} x(k)'Qx(k) + u(k)'Ru(k) \\ &= \sum_{k=0}^{\infty} y(k)'y(k) + u(k)'Ru(k) \end{aligned}$$

with $Q \geq 0$, $R > 0$, and (A, B) stabilizable. In Exercise A.31 we showed that if $(A, Q^{1/2})$ is detectable and an input sequence has been found such that

$$u(k) \to 0 \qquad y(k) \to 0$$

then $x(k) \to 0$.

(a) Show that if $Q \geq 0$, then $Q^{1/2}$ is a well defined, real, symmetric matrix and $Q^{1/2} \geq 0$.

Hint: apply Theorem A.1 to Q, using the subsequent fact 3.

(b) Show that $(A, Q^{1/2})$ is detectable (observable) if and only if (A, Q) is detectable (observable). So we can express one of the LQ existence, uniqueness, and stability conditions using detectability of (A, Q) instead of $(A, Q^{1/2})$.

Exercise A.34: Probability density of the inverse function

Consider a scalar random variable $\xi \in \mathbb{R}$ and let the random variable η be defined by the inverse function

$$\eta = \xi^{-1}$$

(a) If ξ is distributed uniformly on $[a, 1]$ with $0 < a < 1$, what is the density of η?

(b) Is η's density well defined if we allow $a = 0$? Explain your answer.

Exercise A.35: Expectation as a linear operator

(a) Consider the random variable x to be defined as a linear combination of the random variables a and b

$$x = a + b$$

Show that

$$\mathcal{E}(x) = \mathcal{E}(a) + \mathcal{E}(b)$$

Do a and b need to be statistically independent for this statement to be true?

(b) Next consider the random variable x to be defined as a scalar multiple of the random variable a

$$x = \alpha a$$

Show that

$$\mathcal{E}(x) = \alpha \mathcal{E}(a)$$

(c) What can you conclude about $\mathcal{E}(x)$ if x is given by the linear combination

$$x = \sum_i \alpha_i v_i$$

in which v_i are random variables and α_i are scalars.

Exercise A.36: Minimum of two random variables

Given two independent random variables, ξ_1, ξ_2 and the random variable defined by the minimum operator

$$\eta = \min(\xi_1, \xi_2)$$

(a) Sketch the region $\mathbb{X}(c)$ for the inequality $\min(x_1, x_2) \leq c$.

(b) Find η's probability density in terms of the probability densities of ξ_1, ξ_2.

Exercise A.37: Maximum of n normally distributed random variables

Given n independent, identically distributed normal random variables, $\xi_1, \xi_2, \ldots, \xi_n$ and the random variable defined by the maximum operator

$$\eta = \max(\xi_1, \xi_2, \ldots \xi_n)$$

(a) Derive a formula for η's density.

(b) Plot p_η for $\xi_i \sim N(0,1)$ and $n = 1, 2, \ldots 5$. Describe the trend in p_η as n increases.

Exercise A.38: Another picture of mean

Consider a scalar random variable ξ with probability distribution P_ξ shown in Figure A.14. Consider the inverse probability distribution, P_ξ^{-1}, also shown in Figure A.14.

(a) Show that the expectation of ξ is equal to the following integral of the probability distribution (David, 1981, p. 38)

$$\mathcal{E}(\xi) = -\int_{-\infty}^{0} P_\xi(x)dx + \int_0^{\infty} (1 - P_\xi(x))dx \tag{A.43}$$

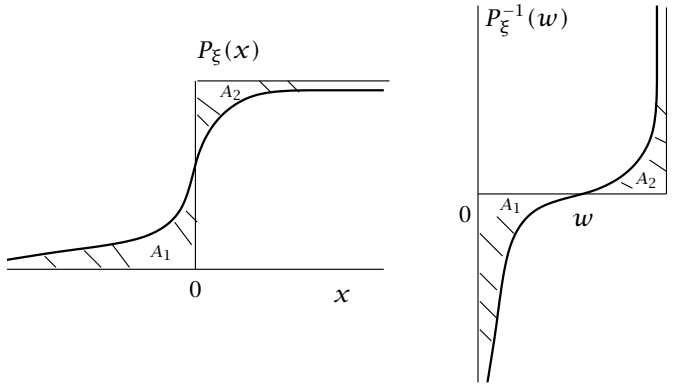


Figure A.14: The probability distribution and inverse distribution for random variable ξ. The mean of ξ is given by the difference in the hatched areas, $\mathcal{E}(\xi) = A_2 - A_1$.

(b) Show that the expectation of ξ is equal to the following integral of the inverse probability distribution

$$\mathcal{E}(\xi) = \int_0^1 P_\xi^{-1}(w)dw \tag{A.44}$$

These interpretations of mean are shown as the hatched areas in Figure A.14, $\mathcal{E}(\xi) = A_2 - A_1$.

Exercise A.39: Ordering random variables

We can order two random variables A and B if they obey an inequality such as $A \geq B$. The frequency interpretation of the probability distribution, $P_A(c) = \Pr(A \leq c)$, then implies that $P_A(c) \leq P_B(c)$ for all c.

If $A \geq B$, show that

$$\mathcal{E}(A) \geq \mathcal{E}(B)$$

Exercise A.40: Max of the mean and mean of the max

Given two random variables A and B, establish the following inequality

$$\max(\mathcal{E}(A), \mathcal{E}(B)) \leq \mathcal{E}(\max(A, B))$$

In other words, the max of the mean is an underbound for the mean of the max.

Exercise A.41: Observability

Consider the linear system with zero input

$$\begin{aligned} x(k+1) &= Ax(k) \\ y(k) &= Cx(k) \end{aligned}$$

with

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 2 & 1 & 1 \end{bmatrix}, \qquad C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

(a) What is the observability matrix for this system? What is its rank?

(b) Consider a string of data measurements

$$y(0) = y(1) = \cdots = y(n-1) = 0$$

Now $x(0) = 0$ is clearly consistent with these data. Is this $x(0)$ unique? If yes, prove it. If no, characterize the set of all $x(0)$ that are consistent with these data.

Exercise A.42: Nothing is revealed

An agitated graduate student shows up at your office. He begins, "I am afraid I have discovered a deep contradiction in the foundations of systems theory." You ask him to calm down and tell you about it. He continues, "Well, we have the pole placement theorem that says if (A, C) is observable, then there exists a matrix L such that the eigenvalues of an observer

$$A - ALC$$

can be assigned arbitrarily."

You reply, "Well, they do have to be conjugate pairs because the matrices A, L, C are real-valued, but yeah, sure, so what?"

He continues, "Well we also have the Hautus lemma that says (A, C) is observable if and only if

$$\operatorname{rank} \begin{bmatrix} \lambda I - A \\ C \end{bmatrix} = n \qquad \forall \lambda \in \mathbb{C}$$

"You know, the Hautus lemma has always been one of my favorite lemmas; I don't see a problem," you reply.

"Well," he continues, "isn't the innovations form of the system, $(A - ALC, C)$, observable if and only if the original system, (A, C), is observable?"

"Yeah ...I seem to recall something like that," you reply, starting to feel a little uncomfortable.

"OK, how about if I decide to put all the observer poles at zero?" he asks, innocently.

You object, "Wait a minute, I guess you can do that, but that's not going to be a very good observer, so I don't think it matters if ...."

"Well," he interrupts, "how about we put all the eigenvalues of $A - ALC$ at zero, like I said, and then we check the Hautus condition at $\lambda = 0$? I get

$$\operatorname{rank} \begin{bmatrix} \lambda I - (A - ALC) \\ C \end{bmatrix} = \operatorname{rank} \begin{bmatrix} 0 \\ C \end{bmatrix} \qquad \lambda = 0$$

"So tell me, how is that matrix on the right ever going to have rank n with that big, fat zero sitting there?" At this point, you start feeling a little dizzy.

What's causing the contradiction here: the pole placement theorem, the Hautus lemma, the statement about equivalence of observability in innovations form, something else? How do you respond to this student?

Exercise A.43: The sum of throwing two dice

Using (A.30), what is the probability density for the sum of throwing two dice? On what number do you want to place your bet? How often do you expect to win if you bet on this outcome?

Make the standard assumptions: the probability density for each die is uniform over the integer values from one to six, and the outcome of each die is independent of the other die.

Exercise A.44: The product of throwing two dice

Using (A.30), what is the probability density for the product of throwing two dice? On what number do you want to place your bet? How often do you expect to win if you bet on this outcome?

Make the standard assumptions: the probability density for each die is uniform over the integer values from one to six, and the outcome of each die is independent of the other die.

Exercise A.45: The size of an ellipse's bounding box

Here we derive the size of the bounding box depicted in Figure A.10. Consider a real, positive definite, symmetric matrix $A \in \mathbb{R}^{n \times n}$ and a real vector $x \in \mathbb{R}^n$. The set of x for which the scalar $x'Ax$ is constant are n-dimensional ellipsoids. Find the length of the sides of the smallest box that contains the ellipsoid defined by

$$x'Ax = b$$

Hint: Consider the equivalent optimization problem to minimize the value of $x'Ax$ such that the ith component of x is given by $x_i = c$. This problem defines the ellipsoid that is tangent to the plane $x_i = c$, and can be used to answer the original question.

Exercise A.46: The tangent points of an ellipse's bounding box

Find the tangent points of an ellipsoid defined by $x'Ax = b$, and its bounding box as depicted in Figure A.10 for $n = 2$. For $n = 2$, draw the ellipse, bounding box and compute the tangent points for the following parameters taken from Figure A.10

$$A = \begin{bmatrix} 3.5 & 2.5 \\ 2.5 & 4.0 \end{bmatrix} \qquad b = 1$$

Exercise A.47: Let's make a deal!

Consider the following contest of the American television game show of the 1960s, Let's Make a Deal. In the show's grand finale, a contestant is presented with three doors. Behind one of the doors is a valuable prize such as an all-expenses-paid vacation to Hawaii or a new car. Behind the other two doors are goats and donkeys. The contestant selects a door, say door number one. The game show host, Monty Hall, then says,

"Before I show you what is behind your door, let's reveal what is behind door number three!" Monty always chooses a door that has one of the booby prizes behind it. As the goat or donkey is revealed, the audience howls with laughter. Then Monty asks innocently,

"Before I show you what is behind your door, I will allow you one chance to change your mind. Do you want to change doors?" While the contestant considers this option, the audience starts screaming out things like,

"Stay with your door! No, switch, switch!" Finally the contestant chooses again, and then Monty shows them what is behind their chosen door.

Let's analyze this contest to see how to *maximize* the chance of winning. Define

$$p(i, j, y), \qquad i, j, y = 1, 2, 3$$

to be the probability that you chose door i, the prize is behind door j and Monty showed you door y (named after the data!) after your initial guess. Then you would want to

$$\max_j p(j|i, y)$$

for your optimal choice after Monty shows you a door.

(a) Calculate this conditional density and give the probability that the prize is behind door i, your original choice, and door $j \neq i$.

(b) You will need to specify a model of Monty's behavior. Please state the one that is appropriate to Let's Make a Deal.

(c) For what other model of Monty's behavior is the answer that it doesn't matter if you switch doors. Why is this a poor model for the game show?

Exercise A.48: Norm of an extended state

Consider $x \in \mathbb{R}^n$ with a norm denoted $|\cdot|_\alpha$, and $u \in \mathbb{R}^m$ with a norm denoted $|\cdot|_\beta$. Now consider a proposed norm for the extended state (x, u)

$$|(x, u)|_\gamma := |x|_\alpha + |u|_\beta$$

Show that this proposal satisfies the definition of a norm given in Section A.8.

If the α and β norms are chosen to be p-norms, is the γ norm also a p-norm? Show why or why not.

Exercise A.49: Distance of an extended state to an extended set

Let $x \in \mathbb{R}^n$ and $\mathbb{X}$ a set of elements in $\mathbb{R}^n$, and $u \in \mathbb{R}^m$ and $\mathbb{U}$ a set of elements in $\mathbb{R}^m$. Denote distances from elements to their respective sets as

$$|x|_\mathbb{X} := \inf_{y \in \mathbb{X}} |x - y|_\alpha \qquad |u|_\mathbb{U} := \inf_{v \in \mathbb{U}} |u - v|_\beta$$

$$|(x, u)|_{\mathbb{X}\times\mathbb{U}} := \inf_{(y,v)\in\mathbb{X}\times\mathbb{U}} |(x, u) - (y, v)|_\gamma$$

Use the norm of the extended state defined in Exercise A.48 to show that

$$|(x, u)|_{\mathbb{X}\times\mathbb{U}} = |x|_\mathbb{X} + |u|_\mathbb{U}$$

Bibliography

T. W. Anderson. *An Introduction to Multivariate Statistical Analysis*. John Wiley & Sons, New York, third edition, 2003.

T. M. Apostol. *Mathematical analysis*. Addison-Wesley, 1974.

Y. Bard. *Nonlinear Parameter Estimation*. Academic Press, New York, 1974.

T. Bayes. An essay towards solving a problem in the doctrine of chances. *Phil. Trans. Roy. Soc.*, 53:370–418, 1763. Reprinted in *Biometrika*, 35:293–315, 1958.

S. P. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

F. M. Callier and C. A. Desoer. *Linear System Theory*. Springer-Verlag, New York, 1991.

E. A. Coddington and N. Levinson. *Theory of Ordinary Differential Equations*. McGraw Hill, 1955.

H. A. David. *Order Statistics*. John Wiley & Sons, Inc., New York, second edition, 1981.

J. Dieudonne. *Foundations of modern analysis*. Academic Press, 1960.

G. H. Golub and C. F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, Baltimore, Maryland, third edition, 1996.

J. Hale. *Ordinary Differential Equations*. Robert E. Krieger Publishing Company, second edition, 1980.

P. Hartman. *Ordinary Differential Equations*. John Wiley and Sons, 1964.

R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press, 1985.

C. R. Johnson. Positive definite matrices. *Amer. Math. Monthly*, 77(3):259–264, March 1970.

C. R. Johnson and C. J. Hillar. Eigenvalues of words in two positive definite letters. *SIAM J. Matrix Anal. and Appl.*, 23(4):916–928, 2002.

E. J. McShane. *Integration*. Princeton University Press, 1944.

J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, New York, second edition, 2006.

A. Papoulis. *Probability, Random Variables, and Stochastic Processes*. McGraw-Hill, Inc., second edition, 1984.

E. Polak. *Optimization: Algorithms and Consistent Approximations*. Springer Verlag, New York, 1997. ISBN 0-387-94971-2.

R. T. Rockafellar. *Convex Analysis*. Princeton University Press, Princeton, N.J., 1970.

R. T. Rockafellar and R. J.-B. Wets. *Variational Analysis*. Springer-Verlag, 1998.

I. Schur. On the characteristic roots of a linear substitution with an application to the theory of integral equations (German). *Math Ann.*, 66:488–510, 1909.

G. Strang. *Linear Algebra and its Applications*. Academic Press, New York, second edition, 1980.

B
Stability Theory

Version: date: April 27, 2026
Copyright © 2026 by Nob Hill Publishing, LLC

B.1 Introduction

In this appendix we consider stability properties of discrete time systems. A good general reference for stability theory of continuous time systems is Khalil (2002). There are not many texts for stability theory of discrete time systems; a useful reference is LaSalle (1986). Recently stability theory for discrete time systems has received more attention in the literature. In the notes below we draw on Jiang and Wang (2001, 2002); Kellett and Teel (2004a,b).

We consider systems of the form

$$x^+ = f(x,u)$$

where the state x lies in $\mathbb{R}^n$ and the control (input) u lies in $\mathbb{R}^m$; in this formulation x and u denote, respectively, the current state and control, and x^+ the successor state. We assume in the sequel that the function $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ is continuous. Let $\phi(k;x,\mathbf{u})$ denote the solution of $x^+ = f(x,u)$ at time k if the initial state is $x(0) = x$ and the control sequence is $\mathbf{u} = (u(0),u(1),u(2),\ldots)$; the solution exists and is unique. If a state-feedback control law $u = \kappa(x)$ has been chosen, the closed-loop system is described by $x^+ = f(x,\kappa(x))$, which has the same form $x^+ = f_c(x)$ where $f_c(\cdot)$ is defined by $f_c(x) := f(x, \kappa(x))$. Let $\phi(k;x,\kappa(\cdot))$ denote the solution of this difference equation at time k if the initial state at time 0 is $x(0) = x$; the solution exists and is unique (even if $\kappa(\cdot)$ is discontinuous). If $\kappa(\cdot)$ is not continuous, as may be the case when $\kappa(\cdot)$ is an implicit model predictive control (MPC) law, then $f_c(\cdot)$ may not be continuous. In this case we assume that $f_c(\cdot)$ is *locally bounded*.[1]

[1] A function $f : X \to X$ is locally bounded if, for any $x \in X$, there exists a neighborhood $\mathcal{N}$ of x such that $f(\mathcal{N})$ is a bounded set, i.e., if there exists a $M > 0$ such that $|f(x)| \leq M$ for all $x \in \mathcal{N}$.

We would like to be sure that the controlled system is "stable", i.e., that small perturbations of the initial state do not cause large variations in the subsequent behavior of the system, and that the state converges to a desired state or, if this is impossible due to disturbances, to a desired set of states. These objectives are made precise in Lyapunov stability theory; in this theory, the system $x^+ = f(x)$ is assumed given and conditions ensuring the stability, or asymptotic stability of a specified state or set are sought; the terms *stability* and *asymptotic stability* are defined below. If convergence to a specified state, x^* say, is sought, it is desirable for this state to be an *equilibrium* point:

Definition B.1 (Equilibrium point)**.** A point x^* is an equilibrium point of $x^+ = f(x)$ if $x(0) = x^*$ implies $x(k) = \phi(k; x^*) = x^*$ for all $k \geq 0$. Hence x^* is an equilibrium point if it satisfies

$$x^* = f(x^*)$$

An equilibrium point x^* is isolated if there are no other equilibrium points in a sufficiently small neighborhood of x^*. A linear system $x^+ = Ax + b$ has a single equilibrium point $x^* = (I - A)^{-1}b$ if $I - A$ is invertible; if not, the linear system has a continuum $\{x \mid (I - A)x = b\}$ of equilibrium points. A nonlinear system, unlike a linear system, may have several isolated equilibrium points.

In other situations, for example when studying the stability properties of an oscillator, convergence to a specified closed set $\mathcal{A} \subset \mathbb{R}^n$ is sought. In the case of a linear oscillator with state dimension 2, this set is an ellipse. If convergence to a set $\mathcal{A}$ is sought, it is desirable for the set $\mathcal{A}$ to be *positive invariant*:

Definition B.2 (Positive invariant set)**.** A closed set $\mathcal{A}$ is positive invariant for the system $x^+ = f(x)$ if $x \in \mathcal{A}$ implies $f(x) \in \mathcal{A}$.

Clearly, any solution of $x^+ = f(x)$ with initial state in $\mathcal{A}$, remains in $\mathcal{A}$. The closed set $\mathcal{A} = \{x^*\}$ consisting of a (single) equilibrium point is a special case; $x \in \mathcal{A}$ $(x = x^*)$ implies $f(x) \in \mathcal{A}$ $(f(x) = x^*)$. Define $|x|_{\mathcal{A}} := \inf_{z \in \mathcal{A}} |x - z|$ to be the distance of a point x from the set $\mathcal{A}$; if $\mathcal{A} = \{x^*\}$, then $|x|_{\mathcal{A}} = |x - x^*|$ which reduces to $|x|$ when $x^* = 0$.

Before introducing the concepts of stability and asymptotic stability and their characterization by Lyapunov functions, it is convenient to make a few definitions.

Definition B.3 ($\mathcal{K}$, $\mathcal{K}_\infty$, $\mathcal{KL}$, and $\mathcal{PD}$ functions)**.** A function $\sigma : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ belongs to class $\mathcal{K}$ if it is continuous, zero at zero, and strictly increasing; $\sigma : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ belongs to class $\mathcal{K}_\infty$ if it is a class $\mathcal{K}$ and unbounded ($\sigma(s) \to \infty$ as $s \to \infty$). A function $\beta : \mathbb{R}_{\geq 0} \times \mathbb{I}_{\geq 0} \to \mathbb{R}_{\geq 0}$ belongs to class $\mathcal{KL}$ if it is continuous and if, for each $t \geq 0$, $\beta(\cdot, t)$ is a class $\mathcal{K}$ function and for each $s \geq 0$, $\beta(s, \cdot)$ is nonincreasing and satisfies $\lim_{t \to \infty} \beta(s, t) = 0$. A function $\gamma : \mathbb{R} \to \mathbb{R}_{\geq 0}$ belongs to class $\mathcal{PD}$ (is positive definite) if it is zero at zero and positive everywhere else.[2]

The following useful properties of these functions are established in Khalil (2002, Lemma 4.2): if $\alpha_1(\cdot)$ and $\alpha_2(\cdot)$ are $\mathcal{K}$ functions ($\mathcal{K}_\infty$ functions), then $\alpha_1^{-1}(\cdot)$ and $(\alpha_1 \circ \alpha_2)(\cdot)$[3] are $\mathcal{K}$ functions[4] ($\mathcal{K}_\infty$ functions). Moreover, if $\alpha_1(\cdot)$ and $\alpha_2(\cdot)$ are $\mathcal{K}$ functions and $\beta(\cdot)$ is a $\mathcal{KL}$ function, then $\sigma(r, s) = \alpha_1(\beta(\alpha_2(r), s))$ is a $\mathcal{KL}$ function.

The following properties prove useful when analyzing the robustness of perturbed systems.

1. For $\gamma(\cdot) \in \mathcal{K}$, the following holds for all $a_i \in \mathbb{R}_{\geq 0}$, $i \in \mathbb{I}_{1:n}$

$$\frac{1}{n}(\gamma(a_1) + \cdots + \gamma(a_n)) \leq \gamma(a_1 + \cdots + a_n) \leq \gamma(na_1) + \cdots + \gamma(na_n) \quad \text{(B.1)}$$

2. Similarly, for $\beta(\cdot) \in \mathcal{KL}$, the following holds for all $a_i \in \mathbb{R}_{\geq 0}$, $i \in \mathbb{I}_{1:n}$, and $t \in \mathbb{R}_{\geq 0}$

$$\frac{1}{n}(\beta(a_1, t) + \cdots + \beta(a_n, t)) \leq \beta((a_1 + \cdots + a_n), t) \leq \beta(na_1, t) + \beta(na_2, t) + \cdots + \beta(na_n, t) \quad \text{(B.2)}$$

3. If $\alpha_i(\cdot) \in \mathcal{K}(\mathcal{K}_\infty)$, for $i \in \mathbb{I}_{1:n}$ then

$$\min_i \{\alpha_i(\cdot)\} := \underline{\alpha}(\cdot) \in \mathcal{K}(\mathcal{K}_\infty) \quad \text{(B.3)}$$

$$\max_i \{\alpha_i(\cdot)\} := \overline{\alpha}(\cdot) \in \mathcal{K}(\mathcal{K}_\infty) \quad \text{(B.4)}$$

[2] Be aware that the existing stability literature sometimes includes continuity in the definition of a positive definite function. We used such a definition in the first edition of this text, for example. But in the second edition, we remove continuity and retain only the requirement of positivity in the definition of positive definite function.

[3] $(\alpha_1 \circ \alpha_2)(\cdot)$ is the composition of the two functions $\alpha_1(\cdot)$ and $\alpha_2(\cdot)$ and is defined by $(\alpha_1 \circ \alpha_2)(s) := \alpha_1(\alpha_2(s))$.

[4] Note, however, that the domain of $\alpha^{-1}(\cdot)$ may be restricted from $\mathbb{R}_{\geq 0}$ to $[0, a)$ for some $a > 0$.

4. Let $v_i \in \mathbb{R}^{n_i}$ for $i \in \mathbb{I}_{1:n}$, and $v \coloneqq (v_1, \dots, v_n) \in \mathbb{R}^{\sum n_i}$. If $\alpha_i(\cdot) \in \mathcal{K}(\mathcal{K}_\infty)$ and $\beta_i(\cdot) \in \mathcal{KL}$ for $i \in \mathbb{I}_{1:n}$, then there exist $\underline{\alpha}(\cdot), \overline{\alpha}(\cdot) \in \mathcal{K}(\mathcal{K}_\infty)$ and $\underline{\beta}(\cdot), \overline{\beta}(\cdot) \in \mathcal{KL}$ such that

$$\underline{\alpha}(|v|) \le \alpha_1(|v_1|) + \cdots + \alpha_n(|v_n|) \le \overline{\alpha}(|v|) \tag{B.5}$$

and, for all $t \in \mathbb{R}_{\ge 0}$

$$\underline{\beta}(|v|, t) \le \beta_1(|v_1|, t) + \cdots + \beta_n(|v_n|, t) \le \overline{\beta}(|v|, t) \tag{B.6}$$

5. Let $v_i, v, \alpha_i(\cdot), \beta_i(\cdot)$ be defined as in 4. Then there exist $\underline{\alpha}(\cdot)$, $\overline{\alpha}(\cdot) \in \mathcal{K}(\mathcal{K}_\infty)$ and $\underline{\beta}(\cdot), \overline{\beta}(\cdot) \in \mathcal{KL}$ such that

$$\underline{\alpha}(|v|) \le \alpha_1(|v_1|) \oplus \cdots \oplus \alpha_n(|v_n|) \le \overline{\alpha}(|v|) \tag{B.7}$$

and, for all $t \in \mathbb{R}_{\ge 0}$

$$\underline{\beta}(|v|, t) \le \beta_1(|v_1|, t) \oplus \cdots \oplus \beta_n(|v_n|, t) \le \overline{\beta}(|v|, t) \tag{B.8}$$

See (Rawlings and Ji, 2012) for short proofs of (B.1) and (B.2), and (Allan, Bates, Risbeck, and Rawlings, 2017, Proposition 23) for a short proof of (B.3). The result (B.4) follows similarly to (B.3). Result (B.5) and (B.7) follow from (B.1) and (B.3)–(B.4), and (B.6) and (B.8) follow from (B.5) and (B.7), respectively. See also Exercises B.9 and B.10.

B.2 Stability and Asymptotic Stability

In this section we consider the stability properties of the autonomous system $x^+ = f(x)$; we assume that $f(\cdot)$ is locally bounded, and that the set $\mathcal{A}$ is closed and positive invariant for $x^+ = f(x)$ unless otherwise stated.

Definition B.4 (Local stability)**.** The (closed, positive invariant) set $\mathcal{A}$ is *locally stable* for $x^+ = f(x)$ if, for all $\varepsilon > 0$, there exists a $\delta > 0$ such that $|x|_{\mathcal{A}} < \delta$ implies $|\phi(i; x)|_{\mathcal{A}} < \varepsilon$ for all $i \in \mathbb{I}_{\ge 0}$.

See Figure B.1 for an illustration of this definition when $\mathcal{A} = \{0\}$; in this case we speak of stability of the origin.

Remark. Stability of the origin, as defined above, is equivalent to continuity of the map $x \mapsto \mathbf{x} \coloneqq (x, \phi(1; x), \phi(2; x), \dots)$, $\mathbb{R} \to \ell_\infty$ at the origin so that $\|\mathbf{x}\| \to 0$ as $x \to 0$ (a small perturbation in the initial state causes a small perturbation in the subsequent motion).

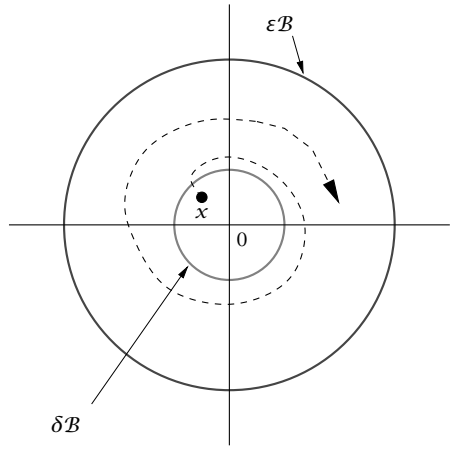


Figure B.1: Stability of the origin. $\mathcal{B}$ denotes the unit ball.

Definition B.5 (Global attraction)**.** The (closed, positive invariant) set $\mathcal{A}$ is *globally attractive* for the system $x^+ = f(x)$ if $|\phi(i;x)|_{\mathcal{A}} \to 0$ as $i \to \infty$ for all $x \in \mathbb{R}^n$.

Definition B.6 (Global asymptotic stability)**.** The (closed, positive invariant) set $\mathcal{A}$ is *globally asymptotically stable* (GAS) for $x^+ = f(x)$ if it is locally stable and globally attractive.

It is possible for the origin to be globally attractive but *not* locally stable. Consider a second order system

$$x^+ = Ax + \phi(x)$$

where A has eigenvalues $\lambda_1 = 0.5$ and $\lambda_2 = 2$ with associated eigenvectors w_1 and w_2, shown in Figure B.2; w_1 is the "stable" and w_2 the "unstable" eigenvector; the smooth function $\phi(\cdot)$ satisfies $\phi(0) = 0$ and $(\partial/\partial x)\phi(0) = 0$ so that $x^+ = Ax + \phi(x)$ behaves like $x^+ = Ax$ near the origin. If $\phi(x) \equiv 0$, the motion corresponding to an initial state αw_1, $\alpha \neq 0$, converges to the origin, whereas the motion corresponding to an initial state αw_2 diverges. If $\phi(\cdot)$ is such that it steers nonzero states toward the horizontal axis, we get trajectories of the form shown in Figure B.2. All trajectories converge to the origin but the motion corresponding to an initial state αw_2, *no matter how small*, is similar to that shown in Figure B.2 and cannot satisfy the ε, δ definition of local stability. The origin is globally attractive but not stable. A

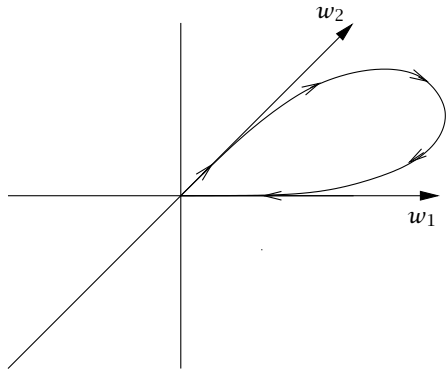


Figure B.2: An attractive but unstable origin.

trajectory that joins an equilibrium point to itself, as in Figure B.2, is called a homoclinic orbit.

We collect below a set of useful definitions:

Definition B.7 (Various forms of stability)**.** The (closed, positive invariant) set $\mathcal{A}$ is

(a) locally stable if, for each $\varepsilon > 0$, there exists a $\delta = \delta(\varepsilon) > 0$ such that $|x|_{\mathcal{A}} < \delta$ implies $|\phi(i;x)|_{\mathcal{A}} < \varepsilon$ for all $i \in \mathbb{I}_{\geq 0}$.

(b) unstable, if it is not locally stable.

(c) locally attractive if there exists $\eta > 0$ such that $|x|_{\mathcal{A}} < \eta$ implies $|\phi(i;x)|_{\mathcal{A}} \to 0$ as $i \to \infty$.

(d) globally attractive if $|\phi(i;x)|_{\mathcal{A}} \to 0$ as $i \to \infty$ for all $x \in \mathbb{R}^n$.

(e) locally asymptotically stable if it is locally stable and locally attractive.

(f) globally asymptotically stable if it is locally stable and globally attractive.

(g) locally exponentially stable if there exist $\eta > 0$, $c > 0$, and $\gamma \in (0,1)$ such that $|x|_{\mathcal{A}} < \eta$ implies $|\phi(i;x)|_{\mathcal{A}} \leq c\,|x|_{\mathcal{A}}\,\gamma^i$ for all $i \in \mathbb{I}_{\geq 0}$.

(h) globally exponentially stable if there exists a $c > 0$ and a $\gamma \in (0,1)$ such that $|\phi(i;x)|_{\mathcal{A}} \leq c\,|x|_{\mathcal{A}}\,\gamma^i$ for all $x \in \mathbb{R}^n$, all $i \in \mathbb{I}_{\geq 0}$.

The following stronger definition of GAS has recently started to become popular.

Definition B.8 (Global asymptotic stability (KL version))**.** The (closed, positive invariant) set $\mathcal{A}$ is *globally asymptotically stable* (GAS) for $x^+ = f(x)$ if there exists a $\mathcal{KL}$ function $\beta(\cdot)$ such that, for each $x \in \mathbb{R}^n$

$$|\phi(i;x)|_{\mathcal{A}} \leq \beta(|x|_{\mathcal{A}}, i) \qquad \forall i \in \mathbb{I}_{\geq 0} \tag{B.9}$$

Proposition B.9 (Connection of classical and KL global asymptotic stability)**.** *Suppose $\mathcal{A}$ is compact (and positive invariant) and that $f(\cdot)$ is continuous. Then the classical and KL definitions of global asymptotic stability of $\mathcal{A}$ for $x^+ = f(x)$ are equivalent.*

The KL version of global asymptotic stability implies the classical version from (B.9) and the definition of a $\mathcal{KL}$ function. The converse is harder to prove but is established in Jiang and Wang (2002) where Proposition 2.2 establishes the equivalence of the existence of a $\mathcal{KL}$ function satisfying (2) with UGAS (uniform global asymptotic stability), and Corollary 3.3 which establishes the equivalence, *when* $\mathcal{A}$ is compact, of uniform global asymptotic stability and global asymptotic stability. Note that $f(\cdot)$ must be continuous for the two definitions to be equivalent. See Exercise B.8 for an example with discontinuous $f(\cdot)$ where the system is GAS in the classical sense but does not satisfy (B.9), i.e., is not GAS in the KL sense.

For a KL version of exponential stability, one simply restricts the form of the KL function $\beta(\cdot)$ of asymptotic stability to $\beta(|x|_{\mathcal{A}}, i) = c\,|x|_{\mathcal{A}}\,\lambda^i$ with $c > 0$ and $\lambda \in (0,1)$, but, as we see, that is exactly the classical definition so there is no distinction between the two forms for exponential stability.

In practice, global asymptotic stability of $\mathcal{A}$ often cannot be achieved because of state constraints. Hence we have to extend slightly the definitions given above. In the following, let $\mathcal{B}$ denote a unit ball in $\mathbb{R}^n$ with center at the origin.

Definition B.10 (Various forms of stability (constrained))**.** Suppose $X \subset \mathbb{R}^n$ is positive invariant for $x^+ = f(x)$, that $\mathcal{A} \subseteq X$ is closed and positive invariant for $x^+ = f(x)$. Then $\mathcal{A}$ is

(a) locally stable in X if, for each $\varepsilon > 0$, there exists a $\delta = \delta(\varepsilon) > 0$ such that $x \in X \cap (\mathcal{A} \oplus \delta\mathcal{B})$, implies $|\phi(i;x)|_{\mathcal{A}} < \varepsilon$ for all $i \in \mathbb{I}_{\geq 0}$.

(b) locally attractive in X if there exists a $\eta > 0$ such that $x \in X \cap (\mathcal{A} \oplus \eta\mathcal{B})$ implies $|\phi(i;x)|_{\mathcal{A}} \to 0$ as $i \to \infty$.

(c) attractive in X if $|\phi(i;x)|_{\mathcal{A}} \to 0$ as $i \to \infty$ for all $x \in X$.

(d) locally asymptotically stable in X if it is locally stable in X and locally attractive in X.

(e) asymptotically stable in X if it is locally stable in X and attractive in X.

(f) locally exponentially stable in X if there exist $\eta > 0$, $c > 0$, and $\gamma \in (0,1)$ such that $x \in X \cap (\mathcal{A} \oplus \eta\mathcal{B})$ implies $|\phi(i;x)|_{\mathcal{A}} \leq c\,|x|_{\mathcal{A}}\,\gamma^i$ for all $i \in \mathbb{I}_{\geq 0}$.

(g) exponentially stable in X if there exists a $c > 0$ and a $\gamma \in (0,1)$ such that $|\phi(i;x)|_{\mathcal{A}} \leq c\,|x|_{\mathcal{A}}\,\gamma^i$ for all $x \in X$, all $i \in \mathbb{I}_{\geq 0}$.

The assumption that X is positive invariant for $x^+ = f(x)$ ensures that $\phi(i;x) \in X$ for all $x \in X$, all $i \in \mathbb{I}_{\geq 0}$. The KL version of asymptotic stability in X is the following.

Definition B.11 (Asymptotic stability (constrained, KL version))**.** Suppose that X is positive invariant and the set $\mathcal{A} \subseteq X$ is closed and positive invariant for $x^+ = f(x)$. The set $\mathcal{A}$ is *asymptotically stable in* X for $x^+ = f(x)$ if there exists a $\mathcal{KL}$ function $\beta(\cdot)$ such that, for each $x \in X$

$$|\phi(i;x)|_{\mathcal{A}} \leq \beta(|x|_{\mathcal{A}}, i) \qquad \forall i \in \mathbb{I}_{\geq 0} \tag{B.10}$$

Finally, we define the *domain of attraction* of an asymptotically stable set $\mathcal{A}$ for the system $x^+ = f(x)$ to be the set of all initial states x such that $|\phi(i;x)|_{\mathcal{A}} \to 0$ as $i \to \infty$. We use the term *region of attraction* to denote any set of initial states x such that $|\phi(i;x)|_{\mathcal{A}} \to 0$ as $i \to \infty$. From these definitions, if $\mathcal{A}$ is attractive *in* X, then X is a region of attraction of set $\mathcal{A}$ for the system $x^+ = f(x)$.

B.3 Lyapunov Stability Theory

Energy in a passive electrical or mechanical system provides a useful analogy to Lyapunov stability theory. In a lumped mechanical system, the total mechanical energy is the sum of the potential and kinetic energies. As time proceeds, this energy is dissipated by friction into heat and the total mechanical energy decays to zero at which point the system is in equilibrium. To establish stability or asymptotic stability, Lyapunov theory follows a similar path. If a real-valued function can be found that is positive and decreasing if the state does not lie in the set $\mathcal{A}$, then the state converges to this set as time tends to infinity. We now make this intuitive idea more precise.

B.3.1 Time-Invariant Systems

First we consider the time-invariant (autonomous) model $x^+ = f(x)$.

Definition B.12 (Lyapunov function (unconstrained and constrained))**.** Suppose that X is positive invariant and the set $\mathcal{A} \subseteq X$ is closed and positive invariant for $x^+ = f(x)$, and $f(\cdot)$ is locally bounded. A function $V : X \to \mathbb{R}_{\geq 0}$ is said to be a Lyapunov function in X for the system $x^+ = f(x)$ and set $\mathcal{A}$ if there exist functions $\alpha_1, \alpha_2 \in \mathcal{K}_\infty$, and *continuous* function $\alpha_3 \in \mathcal{PD}$ such that for any $x \in X$

$$V(x) \geq \alpha_1(|x|_{\mathcal{A}}) \tag{B.11}$$

$$V(x) \leq \alpha_2(|x|_{\mathcal{A}}) \tag{B.12}$$

$$V(f(x)) - V(x) \leq -\alpha_3(|x|_{\mathcal{A}}) \tag{B.13}$$

If $X = \mathbb{R}^n$, then we drop the restrictive phrase "in X."

Remark (Discontinuous f and V)**.** In MPC, the value function for the optimal control problem solved online is often employed as a Lyapunov function. The reader should be aware that many similar but different definitions of Lyapunov functions are in use in many different branches of the science and engineering literature. To be of the most use in MPC analysis, we do not assume here that $f(\cdot)$ or $V(\cdot)$ is continuous. We assume only that $f(\cdot)$ is locally bounded; $V(\cdot)$ is also locally bounded due to (B.12), and continuous on the set $\mathcal{A}$ (but not necessarily on a neighborhood of $\mathcal{A}$) due to (B.11)–(B.12).

Remark (Continuous (and positive definite) α_3)**.** One may wonder why $\alpha_3(\cdot)$ is assumed continuous in addition to positive definite in the definition of the Lyapunov function, when much of the classical literature leaves out continuity; see for example the autonomous case given in Kalman and Bertram (1960). Again, most of this classical literature assumes instead that $f(\cdot)$ is continuous, which we do not assume here. See Exercise B.7 for an example from Lazar, Heemels, and Teel (2009) with discontinuous $f(\cdot)$ for which removing continuity of $\alpha_3(\cdot)$ in Definition B.12 would give a Lyapunov function that fails to imply asymptotic stability.

For making connections to the wide body of existing stability literature, which mainly uses the classical definition of asymptotic stability, and because the proof is instructive, we first state and prove the classical version of the Lyapunov stability theorem.

Theorem B.13 (Lyapunov function and GAS (classical definition))**.** *Suppose that X is positive invariant and the set $\mathcal{A} \subseteq X$ is closed and positive invariant for $x^+ = f(x)$, and $f(\cdot)$ is locally bounded. Suppose $V(\cdot)$ is a Lyapunov function for $x^+ = f(x)$ and set $\mathcal{A}$. Then $\mathcal{A}$ is globally asymptotically stable (classical definition).*

Proof.

(a) Stability. Let $\varepsilon > 0$ be arbitrary and let $\delta := \alpha_2^{-1}(\alpha_1(\varepsilon))$. Suppose $|x|_{\mathcal{A}} < \delta$ so that, by (B.12), $V(x) \leq \alpha_2(\delta) = \alpha_1(\varepsilon)$. From (B.13), $(V(x(i)))_{i \in \mathbb{I}_{\geq 0}}$, $x(i) := \phi(i; x)$, is a nonincreasing sequence so that, for all $i \in \mathbb{I}_{\geq 0}$, $V(x(i)) \leq V(x)$. From (B.11), $|x(i)|_{\mathcal{A}} \leq \alpha_1^{-1}(V(x)) \leq \alpha_1^{-1}(\alpha_1(\varepsilon)) = \varepsilon$ for all $i \in \mathbb{I}_{\geq 0}$.

(b) Attractivity. Let $x \in \mathbb{R}^n$ be arbitrary. From (B.12) $V(x)$ is finite, and from (B.11) and (B.13), the sequence $(V(x(i)))_{i \in \mathbb{I}_{\geq 0}}$ is nonincreasing and bounded below by zero and therefore converges to $\bar{V} \geq 0$ as $i \to \infty$. We next show that $\bar{V} = 0$. From (B.11) and (B.12) and the properties of K_∞ functions, we have that for all $i \geq 0$,

$$\alpha_2^{-1}(V(x(i))) \leq |x(i)|_{\mathcal{A}} \leq \alpha_1^{-1}(V(x(i))) \tag{B.14}$$

Assume for contradiction that $\bar{V} > 0$. Since $\alpha_3(\cdot)$ is continuous and positive definite and interval $\mathcal{I} := [\alpha_2^{-1}(\bar{V}), \alpha_1^{-1}(\bar{V})]$ is compact, the following optimization has a positive solution

$$\rho := \min_{|x|_{\mathcal{A}} \in \mathcal{I}} \alpha_3(|x|_{\mathcal{A}}) > 0$$

From repeated use of (B.13), we have that for all $i \geq 0$

$$V(x(i)) \leq V(x) - \sum_{j=0}^{i-1} \alpha_3(\,|x(j)|_{\mathcal{A}})$$

Since $|x(i)|_{\mathcal{A}}$ converges to interval $\mathcal{I}$ where $\alpha_3(|x(i)|_{\mathcal{A}})$ is underbounded by $\rho > 0$, $\alpha_3(\cdot)$ is continuous, and $V(x)$ is finite, the inequality above implies that $V(x(i)) \to -\infty$ as $i \to \infty$, which is a contradiction. Therefore $V(x(i))$ converges to $\bar{V} = 0$ and (B.14) implies $x(i)$ converges to $\mathcal{A}$ as $i \to \infty$. ■

Next we establish the analogous Lyapunov stability theorem using the stronger KL definition of GAS, Definition B.8. Before establishing the Lyapunov stability theorem, it is helpful to present the following lemma established by Jiang and Wang (2002, Lemma 2.8) that enables us to assume when convenient that $\alpha_3(\cdot)$ in (B.13) is a $\mathcal{K}_\infty$ function rather than just a continuous $\mathcal{PD}$ function.

Lemma B.14 (From $\mathcal{PD}$ to $\mathcal{K}_\infty$ function (Jiang and Wang (2002)))**.** *Assume $V(\cdot)$ is a Lyapunov function for system $x^+ = f(x)$ and set $\mathcal{A}$, and $f(\cdot)$ is locally bounded. Then there exists a smooth function[5] $\rho(\cdot) \in \mathcal{K}_\infty$ such that $W(\cdot) := \rho \circ V(\cdot)$ is also a Lyapunov function for system $x^+ = f(x)$ and set $\mathcal{A}$ that satisfies for all $x \in \mathbb{R}^n$*

$$W(f(x)) - W(x) \leq -\alpha(|x|_{\mathcal{A}})$$

with $\alpha(\cdot) \in \mathcal{K}_\infty$.

Note that Jiang and Wang (2002) prove this lemma under the assumption that both $f(\cdot)$ and $V(\cdot)$ are continuous, but their proof remains valid if both $f(\cdot)$ and $V(\cdot)$ are only locally bounded.

We next establish the Lyapunov stability theorem in which we add the parenthetical (KL definition) purely for emphasis and to distinguish this result from the previous classical result, but we discontinue this emphasis after this theorem, and use exclusively the KL definition.

Theorem B.15 (Lyapunov function and global asymptotic stability (KL definition))**.** *Suppose that X is positive invariant and the set $\mathcal{A} \subseteq X$ is closed and positive invariant for $x^+ = f(x)$, and $f(\cdot)$ is locally bounded. Suppose $V(\cdot)$ is a Lyapunov function for $x^+ = f(x)$ and set $\mathcal{A}$. Then $\mathcal{A}$ is globally asymptotically stable (KL definition).*

Proof. Due to Lemma B.14 we assume without loss of generality that $\alpha_3 \in \mathcal{K}_\infty$. From (B.13) we have that

$$V(\phi(i+1;x)) \leq V(\phi(i;x)) - \alpha_3(|\phi(i;x)|_{\mathcal{A}}) \qquad \forall x \in \mathbb{R}^n \quad i \in \mathbb{I}_{\geq 0}$$

Using (B.12) we have that

$$\alpha_3(|x|_{\mathcal{A}}) \geq \alpha_3 \circ \alpha_2^{-1}(V(x)) \qquad \forall x \in \mathbb{R}^n$$

Combining these we have that

$$V(\phi(i+1;x)) \leq \sigma_1(V(\phi(i;x))) \qquad \forall x \in \mathbb{R}^n \quad i \in \mathbb{I}_{\geq 0}$$

in which

$$\sigma_1(s) := s - \alpha_3 \circ \alpha_2^{-1}(s)$$

We have that $\sigma_1(\cdot)$ is continuous on $\mathbb{R}_{\geq 0}$, $\sigma_1(0) = 0$, and $\sigma_1(s) < s$ for $s > 0$. But $\sigma_1(\cdot)$ may not be increasing. We modify σ_1 to achieve this property in two steps. First define

$$\sigma_2(s) := \max_{s' \in [0,s]} \sigma_1(s') \qquad s \in \mathbb{R}_{\geq 0}$$

[5] A smooth function has derivatives of all orders.

in which the maximum exists for each $s \in \mathbb{R}_{\geq 0}$ because $\sigma_1(\cdot)$ is continuous. By its definition, $\sigma_2(\cdot)$ is nondecreasing, $\sigma_2(0) = 0$, and $0 \leq \sigma_2(s) < s$ for $s > 0$, and we next show that $\sigma_2(\cdot)$ is continuous on $\mathbb{R}_{\geq 0}$. Assume that $\sigma_2(\cdot)$ is discontinuous at a point $c \in \mathbb{R}_{\geq 0}$. Because it is a nondecreasing function, there is a positive jump in the function $\sigma_2(\cdot)$ at c (Bartle and Sherbert, 2000, p. 150). Define [6]

$$a_1 := \lim_{s \nearrow c} \sigma_2(s) \qquad a_2 := \lim_{s \searrow c} \sigma_2(s)$$

We have that $\sigma_1(c) \leq a_1 < a_2$ or we violate the limit of σ_2 from below. Since $\sigma_1(c) < a_2$, $\sigma_1(s)$ must achieve value a_2 for some $s < c$ or we violate the limit from above. But $\sigma_1(s) = a_2$ for $s < c$ also violates the limit from below, and we have a contradiction and $\sigma_2(\cdot)$ is continuous. Finally, define

$$\sigma(s) := (1/2)(s + \sigma_2(s)) \qquad s \in \mathbb{R}_{\geq 0}$$

and we have that $\sigma(\cdot)$ is a continuous, strictly increasing, and unbounded function satisfying $\sigma(0) = 0$. Therefore, $\sigma(\cdot) \in \mathcal{K}_\infty$, $\sigma_1(s) < \sigma(s) < s$ for $s > 0$ and therefore

$$V(\phi(i+1;x)) \leq \sigma(V(\phi(i;x))) \qquad \forall x \in \mathbb{R}^n \quad i \in \mathbb{I}_{\geq 0} \tag{B.15}$$

Repeated use of (B.15) and then (B.12) gives

$$V(\phi(i;x)) \leq \sigma^i \circ \alpha_2(|x|_{\mathcal{A}}) \qquad \forall x \in \mathbb{R}^n \quad i \in \mathbb{I}_{\geq 0}$$

in which σ^i represents the composition of σ with itself i times. Using (B.11) we have that

$$|\phi(i;x)|_{\mathcal{A}} \leq \beta(|x|_{\mathcal{A}}, i) \qquad \forall x \in \mathbb{R}^n \quad i \in \mathbb{I}_{\geq 0}$$

in which

$$\beta(s,i) := \alpha_1^{-1} \circ \sigma^i \circ \alpha_2(s) \qquad \forall s \in \mathbb{R}_{\geq 0} \quad i \in \mathbb{I}_{\geq 0}$$

For all $s \geq 0$, the sequence $w_i := \sigma^i(\alpha_2(s))$ is nonincreasing with i, bounded below (by zero), and therefore converges to a, say, as $i \to \infty$. Therefore, both $w_i \to a$ and $\sigma(w_i) \to a$ as $i \to \infty$. Since $\sigma(\cdot)$ is continuous we also have that $\sigma(w_i) \to \sigma(a)$ so $\sigma(a) = a$, which implies that $a = 0$, and we have shown that for all $s \geq 0$, $\alpha_1^{-1} \circ \sigma^i \circ \alpha_2(s) \to 0$ as

[6] The limits from above and below exist because $\sigma_2(\cdot)$ is nondecreasing (Bartle and Sherbert, 2000, p. 149). If the point $c = 0$, replace the limit from below by $\sigma_2(0)$.

$i \to \infty$. Since $\alpha_1^{-1}(\cdot)$ also is a $\mathcal{K}$ function, we also have that for all $s \geq 0$, $\alpha_1^{-1} \circ \sigma^i \circ \alpha_2(s)$ is nonincreasing with i. We have from the properties of $\mathcal{K}$ functions that for all $i \geq 0$, $\alpha_1^{-1} \circ \sigma^i \circ \alpha_2(s)$ is a $\mathcal{K}$ function, and can therefore conclude that $\beta(\cdot)$ is a $\mathcal{KL}$ function and the proof is complete. ■

Theorem B.15 provides merely a sufficient condition for global asymptotic stability that might be thought to be conservative. Next we establish a *converse* stability theorem that demonstrates necessity. In this endeavor we require a useful preliminary result on $\mathcal{KL}$ functions (Sontag, 1998b, Proposition 7)

Proposition B.16 (Improving convergence (Sontag (1998b))). *Assume that $\beta(\cdot) \in \mathcal{KL}$. Then there exists $\theta_1(\cdot), \theta_2(\cdot) \in \mathcal{K}_\infty$ so that*

$$\beta(s,t) \leq \theta_1(\theta_2(s)e^{-t}) \qquad \forall s \geq 0, \quad \forall t \geq 0 \tag{B.16}$$

Theorem B.17 (Converse theorem for global asymptotic stability). *Suppose that the (closed, positive invariant) set $\mathcal{A}$ is globally asymptotically stable for the system $x^+ = f(x)$. Then there exists a Lyapunov function for the system $x^+ = f(x)$ and set $\mathcal{A}$.*

Proof. Since the set $\mathcal{A}$ is GAS we have that for each $x \in \mathbb{R}^n$ and $i \in \mathbb{I}_{\geq 0}$

$$\left| \phi(i;x)_{\mathcal{A}} \right| \leq \beta(|x|_{\mathcal{A}}, i)$$

in which $\beta(\cdot) \in \mathcal{KL}$. Using (B.16) then gives for each $x \in \mathbb{R}^n$ and $i \in \mathbb{I}_{\geq 0}$

$$\theta_1^{-1}\left(\left| \phi(i;x) \right|_{\mathcal{A}} \right) \leq \theta_2(|x|_{\mathcal{A}})e^{-i}$$

in which $\theta_1^{-1}(\cdot) \in \mathcal{K}_\infty$. Propose as Lyapunov function

$$V(x) = \sum_{i=0}^{\infty} \theta_1^{-1}\left(\left| \phi(i;x) \right|_{\mathcal{A}} \right)$$

Since $\phi(0;x) = x$, we have that $V(x) \geq \theta_1^{-1}(|x|_{\mathcal{A}})$ and we choose $\alpha_1(\cdot) = \theta_1^{-1}(\cdot) \in \mathcal{K}_\infty$. Performing the sum gives

$$V(x) = \sum_{i=0}^{\infty} \theta_1^{-1}\left(\left| \phi(i;x)_{\mathcal{A}} \right| \right) \leq \theta_2(|x|_{\mathcal{A}}) \sum_{i=0}^{\infty} e^{-i} = \theta_2(|x|_{\mathcal{A}}) \frac{e}{e-1}$$

and we choose $\alpha_2(\cdot) = (e/(e-1))\theta_2(\cdot) \in \mathcal{K}_\infty$. Finally, noting that $f(\phi(i;x)) = \phi(i+1;x)$ for each $x \in \mathbb{R}^n$, $i \in \mathbb{I}_{\geq 0}$, we have that

$$\begin{aligned} V(f(x)) - V(x) &= \sum_{i=0}^{\infty} \theta_1^{-1}\left(\left|f(\phi(i;x))\right|_{\mathcal{A}}\right) - \theta_1^{-1}\left(\left|\phi(i;x)\right|_{\mathcal{A}}\right) \\ &= -\theta_1^{-1}\left(\left|\phi(0;x)\right|_{\mathcal{A}}\right) \\ &= -\theta_1^{-1}(|x|_{\mathcal{A}}) \end{aligned}$$

and we choose $\alpha_3(\cdot) = \theta_1^{-1}(\cdot) \in \mathcal{K}_\infty$, and the result is established. ■

The appropriate generalization of Theorem B.15 for the constrained case is:

Theorem B.18 (Lyapunov function for asymptotic stability (constrained))**.** *If there exists a Lyapunov function in X for the system $x^+ = f(x)$ and set $\mathcal{A}$, then $\mathcal{A}$ is asymptotically stable in X for $x^+ = f(x)$.*

The proof of this result is similar to that of Theorem B.15 and is left as an exercise.

Theorem B.19 (Lyapunov function for exponential stability)**.** *If there exists $V : X \to \mathbb{R}_{\geq 0}$ satisfying the following properties for all $x \in X$*

$$\begin{aligned} a_1 |x|_{\mathcal{A}}^{\sigma} \leq V(x) &\leq a_2 |x|_{\mathcal{A}}^{\sigma} \\ V(f(x)) - V(x) &\leq -a_3 |x|_{\mathcal{A}}^{\sigma} \end{aligned}$$

in which $a_1, a_2, a_3, \sigma > 0$, then $\mathcal{A}$ is exponentially stable in X for $x^+ = f(x)$.

Linear time-invariant systems. We review some facts involving the discrete matrix Lyapunov equation and stability of the linear system

$$x^+ = Ax$$

in which $x \in \mathbb{R}^n$. The discrete time system is asymptotically stable if and only if the magnitudes of the eigenvalues of A are strictly less than unity. Such an A matrix is called stable, convergent, or discrete time Hurwitz.

In the following, $A, S, Q \in \mathbb{R}^{n \times n}$. The following matrix equation is known as a discrete matrix Lyapunov equation,

$$A'SA - S = -Q$$

The properties of solutions to this equation allow one to draw conclusions about the stability of A without computing its eigenvalues. Sontag (1998a, p. 231) provides the following lemma

Lemma B.20 (Lyapunov function for linear systems)**.** *The following statements are equivalent (Sontag, 1998a).*

(a) A is stable.

(b) For each $Q \in \mathbb{R}^{n\times n}$, there is a unique solution S of the discrete matrix Lyapunov equation

$$A'SA - S = -Q$$

and if $Q > 0$ then $S > 0$.

(c) There is some $S > 0$ such that $A'SA - S < 0$.

(d) There is some $S > 0$ such that $V(x) = x'Sx$ is a Lyapunov function for the system $x^+ = Ax$.

Exercise B.1 asks you to establish the equivalence of (a) and (b).

B.3.2 Time-Varying, Constrained Systems

Following the discussion in Rawlings and Risbeck (2017), we consider the nonempty sets $X(i) \subseteq \mathbb{R}^n$ indexed by $i \in \mathbb{I}_{\geq 0}$. We define the time-varying system

$$x^+ = f(x, i)$$

with $f(\cdot, i) : X(i) \to X(i+1)$. We assume that $f(\cdot, i)$ is locally bounded for all $i \in \mathbb{I}_{\geq 0}$. Note from the definition of f that the sets $X(i)$ satisfy positive invariance in the following sense: $x \in X(i)$ for any $i \geq 0$ implies $x(i+1) := f(x, i) \in X(i+1)$. We say that the set sequence $(X(i))_{i\geq 0}$ is *sequentially* positive invariant to denote this form of invariance.

Definition B.21 (Sequential positive invariance)**.** A sequence of sets $(X(i))_{i\geq 0}$ is sequentially positive invariant for the system $x^+ = f(x, i)$ if for any $i \geq 0$, $x \in X(i)$ implies $f(x, i) \in X(i+1)$.

We again assume that $\mathcal{A}$ is closed and positive invariant for the time-varying system, i.e., $x \in \mathcal{A}$ at any time $i \geq 0$ implies $f(x, i) \in \mathcal{A}$. We also assume that $\mathcal{A} \subseteq X(i)$ for all $i \geq 0$. We next define asymptotic stability of $\mathcal{A}$.

Definition B.22 (Asymptotic stability (time-varying, constrained))**.** Suppose that the sequence $(X(i))_{i\geq 0}$ is sequentially positive invariant and the set $\mathcal{A} \subseteq X(i)$ for all $i \geq 0$ is closed and positive invariant for $x^+ = f(x, i)$. The set $\mathcal{A}$ is *asymptotically stable* in $X(i)$ at each time

$i \geq 0$ for $x^+ = f(x,i)$ if the following holds for all $i \geq i_0 \geq 0$, and $x \in X(i_0)$

$$\left| \phi(i;x,i_0) \right|_{\mathcal{A}} \leq \beta(|x|_{\mathcal{A}}, i - i_0) \tag{B.17}$$

in which $\beta \in \mathcal{KL}$ and $\phi(i;x,i_0)$ is the solution to $x^+ = f(x,i)$ at time $i \geq i_0$ with initial condition x at time $i_0 \geq 0$.

This stability definition is somewhat restrictive because $\phi(i;x,i_0)$ is bounded by a function depending on $i - i_0$ rather than on i. For example, to be more general we could define a time-dependent set of $\mathcal{KL}$ functions, $\beta_j(\cdot)$, $j \geq 0$, and replace (B.17) with $\left| \phi(i;x,i_0) \right|_{\mathcal{A}} \leq \beta_{i_0}(|x|_{\mathcal{A}}, i)$ for all $i \geq i_0 \geq 0$.

We define a time-varying Lyapunov function for this system as follows.

Definition B.23 (Lyapunov function: time-varying, constrained case)**.** Let the sequence $(X(i))_{i \geq 0}$ be sequentially positive invariant, and the set $\mathcal{A} \subseteq X(i)$ for all $i \geq 0$ be closed and positive invariant. Let $V(\cdot, i) : X(i) \to \mathbb{R}_{\geq 0}$ satisfy for all $x \in X(i), i \in \mathbb{I}_{\geq 0}$

$$\begin{aligned} \alpha_1(|x|_{\mathcal{A}}) \leq V(x,i) \leq \quad & \alpha_2(|x|_{\mathcal{A}}) \\ V(f(x,i), i+1) - V(x,i) \leq & -\alpha_3(|x|_{\mathcal{A}}) \end{aligned}$$

with $\alpha_1, \alpha_2, \alpha_3 \in \mathcal{K}_\infty$. Then $V(\cdot,\cdot)$ is a time-varying Lyapunov function in the sequence $(X(i))_{i \geq 0}$ for $x^+ = f(x,i)$ and set $\mathcal{A}$.

Note that $f(x,i) \in X(i+1)$ since $x \in X(i)$ which verifies that $V(f(x,i), i+1)$ is well defined for all $x \in X(i), i \geq 0$. We then have the following asymptotic stability result for the time-varying, constrained case.

Theorem B.24 (Lyapunov theorem for asymptotic stability (time-varying, constrained))**.** *Let the sequence $(X(i))_{i \geq 0}$ be sequentially positive invariant, and the set $\mathcal{A} \subseteq X(i)$ for all $i \geq 0$ be closed and positive invariant, and $V(\cdot,\cdot)$ be a time-varying Lyapunov function in the sequence $(X(i))_{i \geq 0}$ for $x^+ = f(x,i)$ and set $\mathcal{A}$. Then $\mathcal{A}$ is asymptotically stable in $X(i)$ at each time $i \geq 0$ for $x^+ = f(x,i)$.*

Proof. For $x \in X(i_0)$, we have that $(\phi(i;x,i_0), i) \in X(i)$ for all $i \geq i_0$. From the first and second inequalities we have that for all $i \geq i_0$ and $x \in X(i_0)$

$$\begin{aligned} V(\phi(i+1;x,i_0), i+1) &\leq V(\phi(i;x,i_0), i) - \alpha_3(\left| \phi(i;x,i_0) \right|_{\mathcal{A}}) \\ &\leq \sigma_1(V(\phi(i;x,i_0), i)) \end{aligned}$$

with $\sigma_1(s) \coloneqq s - \alpha_3 \circ \alpha_2^{-1}(s)$. Note that $\sigma_1(\cdot)$ may not be $\mathcal{K}_\infty$ because it may not be increasing. But given this result we can find, as in the proof of Theorem B.15, $\sigma(\cdot) \in \mathcal{K}_\infty$ satisfying $\sigma_1(s) < \sigma(s) < s$ for all $s \in \mathbb{R}_{>0}$ such that $V(\phi(i+1;x,i_0),i+1) \leq \sigma(V(\phi(i;x,i_0),i))$. We then have that

$$\left|\phi(i;x,i_0)\right|_{\mathcal{A}} \leq \beta\left(|x|_{\mathcal{A}}, i-i_0\right) \qquad \forall x \in X(i_0), \quad i \geq i_0$$

in which $\beta(s,i) \coloneqq \alpha_1^{-1} \circ \sigma^i \circ \alpha_2(s)$ for $s \in \mathbb{R}_{\geq 0}, i \geq 0$ is a $\mathcal{KL}$ function, and the result is established. ■

B.3.3 Upper bounding $\mathcal{K}$ functions

In using Lyapunov functions for stability analysis, we often have to establish that the upper bound inequality holds on some closed set. The following result proves useful in such situations.

Proposition B.25 (Global K function overbound)**.** *Let $X \subseteq \mathbb{R}^n$ be closed and suppose that a function $V : X \to \mathbb{R}_{\geq 0}$ is continuous at $x_0 \in X$ and locally bounded on X, i.e., bounded on every compact subset of X. Then, there exists a K function α such that*

$$|V(x) - V(x_0)| \leq \alpha(|x - x_0|) \quad \textit{for all } x \in X$$

A proof is given in Rawlings and Risbeck (2015).

B.4 Robust Stability

We now turn to the task of obtaining stability conditions for discrete time systems subject to disturbances. There are two separate questions that should be addressed. The first is *nominal* robustness; is asymptotic stability of a set $\mathcal{A}$ for a (nominal) system $x^+ = f(x)$ maintained in the presence of arbitrarily small disturbances? The second question is the determination of conditions for asymptotic stability of a set $\mathcal{A}$ for a system perturbed by disturbances lying in a given compact set.

B.4.1 Nominal Robustness

Here we follow Teel (2004). The nominal system is $x^+ = f(x)$. Consider the perturbed system

$$x^+ = f(x+e) + w \tag{B.18}$$

where e is the state error and w the additive disturbance. Let $\mathbf{e} := (e(0), e(1), \ldots)$ and $\mathbf{w} := (w(0), w(1), \ldots)$ denote the disturbance sequences with norms $\|\mathbf{e}\| := \sup_{i\geq 0} |e(i)|$ and $\|\mathbf{w}\| := \sup_{i\geq 0} |w(i)|$. Let $M_\delta := \{(\mathbf{e}, \mathbf{w}) \mid \|\mathbf{e}\| \leq \delta, \|\mathbf{w}\| \leq \delta\}$ and, for each $x \in \mathbb{R}^n$, let S_δ denote the set of solutions $\phi(\cdot; x, \mathbf{e}, \mathbf{w})$ of (B.18) with initial state x (at time 0) and perturbation sequences $(\mathbf{e}, \mathbf{w}) \in M_\delta$. A closed, compact set $\mathcal{A}$ is *nominally* robustly asymptotically stable for the (nominal) system $x^+ = f(x)$ if a small neighborhood of $\mathcal{A}$ is locally stable and attractive for all sufficiently small perturbation sequences. We use the adjective *nominal* to indicate that we are examining how a system $x^+ = f(x)$ for which $\mathcal{A}$ is known to be asymptotically stable behaves when subjected to small disturbances. More precisely Teel (2004):

Definition B.26 (Nominal robust global asymptotic stability)**.** The closed, compact set $\mathcal{A}$ is said to be nominally robustly globally asymptotically stable (nominally RGAS) for the system $x^+ = f(x)$ if there exists a $\mathcal{KL}$ function $\beta(\cdot)$ and, for each $\varepsilon > 0$ and each compact set X, there exists a $\delta > 0$ such that, for each $x \in X$ and each solution $\phi(\cdot)$ of the perturbed system lying in S_δ, $|\phi(i)|_{\mathcal{A}} \leq \beta(|x|_{\mathcal{A}}, i) + \varepsilon$ for all $i \in \mathbb{I}_{\geq 0}$.

Thus, for each $\varepsilon > 0$, there exists a $\delta > 0$ such that each solution $\phi(\cdot)$ of $x^+ = f(x+e)+w$ starting in a δ neighborhood of $\mathcal{A}$ remains in a $\beta(\delta, 0) + \varepsilon$ neighborhood of $\mathcal{A}$, and each solution starting anywhere in $\mathbb{R}^n$ converges to a ε neighborhood of $\mathcal{A}$. These properties are a necessary relaxation (because of the perturbations) of local stability and global attractivity.

Remark. What we call "nominally robustly globally asymptotically stable" in the above definition is called "robustly globally asymptotically stable" in Teel (2004); we use the term "nominal" to indicate that we are concerned with the effect of perturbations e and w on the stability properties of a "nominal" system $x^+ = f(x)$ for which asymptotic stability of a set $\mathcal{A}$ has been established (in the absence of perturbations). We use the expression "$\mathcal{A}$ is globally asymptotically stable for $x^+ = f(x + e) + w$" to refer to the case when asymptotic stability of a set $\mathcal{A}$ has been established for the perturbed system $x^+ = f(x+e)+w$.

The following result, where we add the adjective "nominal", is established in (Teel, 2004, Theorem 2):

Theorem B.27 (Nominal robust global asymptotic stability and Lyapunov function)**.** *Suppose set $\mathcal{A}$ is closed and compact and $f(\cdot)$ is locally*

bounded. Then the set $\mathcal{A}$ is nominally robustly globally asymptotically stable for the system $x^+ = f(x)$ if and only if there exists a continuous (in fact, smooth) Lyapunov function for $x^+ = f(x)$ and set $\mathcal{A}$.

The significance of this result is that while a nonrobust system, for which $\mathcal{A}$ is globally asymptotically stable, has a Lyapunov function, that function is *not* continuous. For the globally asymptotically stable example $x^+ = f(x)$ discussed in Section 3.2 of Chapter 3, where $f(x) = (0, |x|)$ when $x_1 \neq 0$ and $f(x) = (0,0)$ otherwise, one Lyapunov function $V(\cdot)$ is $V(x) = 2|x|$ if $x_1 \neq 0$ and $V(x) = |x|$ if $x_1 = 0$. That $V(\cdot)$ is a Lyapunov function follows from the fact that it satisfies $V(x) \geq |x|$, $V(x) \leq 2|x|$ and $V(f(x)) - V(x) = -|x|$ for all $x \in \mathbb{R}^2$. It follows immediately from its definition that $V(\cdot)$ is not continuous; but we can also deduce from Theorem B.27 that every Lyapunov function for this system is not continuous since, as shown in Section 3.2 of Chapter 3, global asymptotic stability for this system is not robust. Theorem B.27 shows that existence of a continuous Lyapunov function guarantees nominal robustness. Also, it follows from Theorem B.17 that there exists a smooth Lyapunov function for $x^+ = f(x)$ if $f(\cdot)$ is continuous and $\mathcal{A}$ is GAS for $x^+ = f(x)$. Since $f(\cdot)$ is locally bounded if it is continuous, it then follows from Theorem B.27 that $\mathcal{A}$ is nominally robust GAS for $x^+ = f(x)$ if it is GAS and $f(\cdot)$ is continuous.

B.4.2 Robustness

We turn now to stability conditions for systems subject to bounded disturbances (not vanishingly small) and described by

$$x^+ = f(x, w) \tag{B.19}$$

where the disturbance w lies in the compact set $\mathbb{W}$. This system may equivalently be described by the difference inclusion

$$x^+ \in F(x) \tag{B.20}$$

where the set $F(x) := \{f(x,w) \mid w \in \mathbb{W}\}$. Let $S(x)$ denote the set of all solutions of (B.19) or (B.20) with initial state x. We require, in the sequel, that the closed set $\mathcal{A}$ is positive invariant for (B.19) (or for $x^+ \in F(x)$):

Definition B.28 (Positive invariance with disturbances)**.** The closed set $\mathcal{A}$ is positive invariant for $x^+ = f(x,w)$, $w \in \mathbb{W}$ if $x \in \mathcal{A}$ implies $f(x,w) \in \mathcal{A}$ for all $w \in \mathbb{W}$; it is positive invariant for $x^+ \in F(x)$ if $x \in \mathcal{A}$ implies $F(x) \subseteq \mathcal{A}$.

Clearly the two definitions are equivalent; $\mathcal{A}$ is positive invariant for $x^+ = f(x,w)$, $w \in \mathbb{W}$, if and only if it is positive invariant for $x^+ \in F(x)$.

Remark. In the MPC literature, but not necessarily elsewhere, the term robust positive invariant is often used in place of positive invariant to emphasize that positive invariance is maintained despite the presence of the disturbance w. However, since the uncertain system $x^+ = f(x, w)$, $w \in \mathbb{W}$ is specified ($x^+ = f(x,w)$, $w \in \mathbb{W}$ or $x^+ \in F(x)$) in the assertion that a closed set $\mathcal{A}$ is positive invariant, the word "robust" appears to be unnecessary. In addition, in the systems literature, the closed set $\mathcal{A}$ is said to be robust positive invariant for $x^+ \in F(x)$ if it satisfies conditions similar to those of Definition B.26 with $x^+ \in F(x)$ replacing $x^+ = f(x)$; see Teel (2004), Definition 3.

In Definitions B.29–B.31, we use "positive invariant" to denote "positive invariant for $x^+ = f(x,w)$, $w \in \mathbb{W}$" or for $x^+ \in F(x)$.

Definition B.29 (Local stability (disturbances))**.** The closed, positive invariant set $\mathcal{A}$ is *locally stable* for $x^+ = f(x,w)$, $w \in \mathbb{W}$ (or for $x^+ \in F(x)$) if, for all $\varepsilon > 0$, there exists a $\delta > 0$ such that, for each x satisfying $|x|_{\mathcal{A}} < \delta$, each solution $\phi(\cdot) \in S(x)$ satisfies $|\phi(i)|_{\mathcal{A}} < \varepsilon$ for all $i \in \mathbb{I}_{\geq 0}$.

Definition B.30 (Global attraction (disturbances))**.** The closed, positive invariant set $\mathcal{A}$ is *globally attractive* for the system $x^+ = f(x,w)$, $w \in \mathbb{W}$ (or for $x^+ \in F(x)$) if, for each $x \in \mathbb{R}^n$, each solution $\phi(\cdot) \in S(x)$ satisfies $|\phi(i)|_{\mathcal{A}} \to 0$ as $i \to \infty$.

Definition B.31 (GAS (disturbances))**.** The closed, positive invariant set $\mathcal{A}$ is *globally asymptotically stable* for $x^+ = f(x,w)$, $w \in \mathbb{W}$ (or for $x^+ \in F(x)$) if it is locally stable and globally attractive.

An alternative definition of global asymptotic stability of closed set $\mathcal{A}$ for $x^+ = f(x,w)$, $w \in \mathbb{W}$, if $\mathcal{A}$ is compact, is the existence of a $\mathcal{KL}$ function $\beta(\cdot)$ such that for each $x \in \mathbb{R}^n$, each $\phi \in S(x)$ satisfies $|\phi(i)|_{\mathcal{A}} \leq \beta(|x|_{\mathcal{A}}, i)$ for all $i \in \mathbb{I}_{\geq 0}$. To cope with disturbances we require a modified definition of a Lyapunov function.

Definition B.32 (Lyapunov function (disturbances))**.** A function $V : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ is said to be a Lyapunov function for the system $x^+ = f(x, w)$, $w \in \mathbb{W}$ (or for $x^+ \in F(x)$) and closed set $\mathcal{A}$ if there exist functions

$\alpha_i \in \mathcal{K}_\infty$, $i = 1, 2, 3$ such that for any $x \in \mathbb{R}^n$,

$$V(x) \geq \alpha_1(|x|_{\mathcal{A}}) \tag{B.21}$$

$$V(x) \leq \alpha_2(|x|_{\mathcal{A}}) \tag{B.22}$$

$$\sup_{z \in F(x)} V(z) - V(x) \leq -\alpha_3(|x|_{\mathcal{A}}) \tag{B.23}$$

Remark. Without loss of generality, we can choose the function $\alpha_3(\cdot)$ in (B.23) to be a class $\mathcal{K}_\infty$ function if $f(\cdot)$ is continuous (see Jiang and Wang (2002), Lemma 2.8).

Inequality B.23 ensures $V(f(x,w)) - V(x) \leq -\alpha_3(|x|_{\mathcal{A}})$ for all $w \in \mathbb{W}$. The existence of a Lyapunov function for the system $x^+ \in F(x)$ and closed set $\mathcal{A}$ is a sufficient condition for $\mathcal{A}$ to be globally asymptotically stable for $x^+ \in F(x)$ as shown in the next result.

Theorem B.33 (Lyapunov function for global asymptotic stability (disturbances))**.** *Suppose $V(\cdot)$ is a Lyapunov function for $x^+ = f(x,w)$, $w \in \mathbb{W}$ (or for $x^+ \in F(x)$) and closed set $\mathcal{A}$ with $\alpha_3(\cdot)$ a $\mathcal{K}_\infty$ function. Then $\mathcal{A}$ is globally asymptotically stable for $x^+ = f(x,w)$, $w \in \mathbb{W}$ (or for $x^+ \in F(x)$).*

Proof. (i) Local stability: Let $\varepsilon > 0$ be arbitrary and let $\delta := \alpha_2^{-1}(\alpha_1(\varepsilon))$. Suppose $|x|_{\mathcal{A}} < \delta$ so that, by (B.22), $V(x) \leq \alpha_2(\delta) = \alpha_1(\varepsilon)$. Let $\phi(\cdot)$ be any solution in $S(x)$ so that $\phi(0) = x$. From (B.23), $(V(\phi(i)))_{i \in \mathbb{I}_{\geq 0}}$ is a nonincreasing sequence so that, for all $i \in \mathbb{I}_{\geq 0}$, $V(\phi(i)) \leq V(x)$. From (B.21), $|\phi(i)|_{\mathcal{A}} \leq \alpha_1^{-1}(V(x)) \leq \alpha_1^{-1}(\alpha_1(\varepsilon)) = \varepsilon$ for all $i \in \mathbb{I}_{\geq 0}$. (ii) Global attractivity: Let $x \in \mathbb{R}^n$ be arbitrary. Let $\phi(\cdot)$ be any solution in $S(x)$ so that $\phi(0) = x$. From Equations B.21 and B.23, since $\phi(i+1) \in F(\phi(i))$, the sequence $(V(\phi(i)))_{i \in \mathbb{I}_{\geq 0}}$ is nonincreasing and bounded from below by zero. Hence both $V(\phi(i))$ and $V(\phi(i+1))$ converge to $\bar{V} \geq 0$ as $i \to \infty$. But $\phi(i+1) \in F(\phi(i))$ so that, from (B.23), $\alpha_3(|\phi(i)|_{\mathcal{A}}) \to 0$ as $i \to \infty$. Since $|\phi(i)|_{\mathcal{A}} = \alpha_3^{-1}(\alpha_3(|\phi(i)|_{\mathcal{A}}))$ where $\alpha_3^{-1}(\cdot)$ is a $\mathcal{K}_\infty$ function, $|\phi(i)|_{\mathcal{A}} \to 0$ as $i \to \infty$. ■

B.5 Control Lyapunov Functions

A control Lyapunov function is a useful generalization, due to Sontag (1998a, pp.218–233), of a Lyapunov function; while a Lyapunov function is relevant for a system $x^+ = f(x)$ and provides conditions for the (asymptotic) stability of a set for this system, a control Lyapunov function is relevant for a control system $x^+ = f(x,u)$ and provides condi-

tions for the existence of a controller $u = \kappa(x)$ that ensures (asymptotic) stability of a set for the controlled system $x^+ = f(x, \kappa(x))$. Consider the control system

$$x^+ = f(x, u)$$

where the control u is subject to the constraint

$$u \in \mathbb{U}$$

Our standing assumptions in this section are that $f(\cdot)$ is continuous and $\mathbb{U}$ is compact.

Definition B.34 (Global control Lyapunov function (CLF))**.** A function $V : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ is a global control Lyapunov function for the system $x^+ = f(x, u)$, $u \in \mathbb{U}$, and closed set $\mathcal{A}$ if there exist $\mathcal{K}_\infty$ functions $\alpha_1(\cdot), \alpha_2(\cdot), \alpha_3(\cdot)$ satisfying for all $x \in \mathbb{R}^n$:

$$\alpha_1(|x|_{\mathcal{A}}) \leq V(x) \leq \alpha_2(|x|_{\mathcal{A}})$$
$$\inf_{u \in \mathbb{U}} V(f(x, u)) - V(x) \leq -\alpha_3(|x|_{\mathcal{A}})$$

Definition B.35 (Global stabilizability)**.** Let set $\mathcal{A}$ be compact. The set $\mathcal{A}$ is globally stabilizable for the system $x^+ = f(x, u)$ if there exists a state-feedback function $\kappa : \mathbb{R}^n \to \mathbb{U}$ such that $\mathcal{A}$ is globally asymptotically stable for $x^+ = f(x, \kappa(x))$.

Remark. Given a global control Lyapunov function $V(\cdot)$, one can choose a control law $\kappa : \mathbb{R}^n \to \mathbb{U}$ satisfying

$$V(f(x, \kappa(x))) \leq V(x) - \alpha_3(|x|_{\mathcal{A}})/2$$

for all $x \in \mathbb{R}^n$ (seeTeel (2004)). Since $\mathbb{U}$ is compact, $\kappa(\cdot)$ is locally bounded and, hence, so is $x \mapsto f(x, \kappa(x))$. Thus we may use Theorem B.13 to deduce that $\mathcal{A}$ is globally asymptotically stable for $x^+ = f(x, \kappa(x))$. If $V(\cdot)$ is continuous, one can also establish nominal robustness properties.

In a similar fashion one can extend the concept of control Lyapunov functions to the case when the system is subject to disturbances. Consider the system

$$x^+ = f(x, u, w)$$

where the control u is constrained to lie in $\mathbb{U}$ and the disturbance takes values in the set $\mathbb{W}$. We assume that $f(\cdot)$ is continuous and that $\mathbb{U}$ and $\mathbb{W}$ are compact. The system may be equivalently defined by

$$x^+ \in F(x,u)$$

where the set-valued function $F(\cdot)$ is defined by

$$F(x,u) := \{f(x,u,w) \mid w \in \mathbb{W}\}$$

We can now make the obvious generalizations of the definitions in Section B.4.2.

Definition B.36 (Positive invariance (disturbance and control))**.** The closed set $\mathcal{A}$ is positive invariant for $x^+ = f(x,u,w)$, $w \in \mathbb{W}$ (or for $x^+ \in F(x,u)$) if for all $x \in \mathcal{A}$ there exists a $u \in \mathbb{U}$ such that $f(x,u,w) \in \mathcal{A}$ for all $w \in \mathbb{W}$ (or $F(x,u) \subseteq \mathcal{A}$).

Definition B.37 (CLF (disturbance and control))**.** A function $V : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ is said to be a control Lyapunov function for the system $x^+ = f(x,u,w)$, $u \in \mathbb{U}$, $w \in \mathbb{W}$ (or $x^+ \in F(x,u)$, $u \in \mathbb{U}$) and set $\mathcal{A}$ if there exist functions $\alpha_i \in \mathcal{K}_\infty$, $i = 1,2,3$ such that for any $x \in \mathbb{R}^n$,

$$\begin{aligned} \alpha_1(|x|_{\mathcal{A}}) \leq V(x) \leq \alpha_2(|x|_{\mathcal{A}}) \\ \inf_{u \in \mathbb{U}} \sup_{z \in F(x,u)} V(z) - V(x) \leq -\alpha_3(|x|_{\mathcal{A}}) \end{aligned} \tag{B.24}$$

Remark (CLF implies control law)**.** Given a global control Lyapunov function $V(\cdot)$, one can choose a control law $\kappa : \mathbb{R}^n \to \mathbb{U}$ satisfying

$$\sup_{z \in F(x,\kappa(x))} V(z) \leq V(x) - \alpha_3(|x|_{\mathcal{A}})/2$$

for all $x \in \mathbb{R}^n$. Since $\mathbb{U}$ is compact, $\kappa(\cdot)$ is locally bounded and, hence, so is $x \mapsto f(x,\kappa(x))$. Thus we may use Theorem B.33 to deduce that $\mathcal{A}$ is globally asymptotically stable for $x^+ = f(x,\kappa(x),w)$, $w \in \mathbb{W}$ (for $x^+ \in F(x,\kappa(x))$).

These results can be further extended to deal with the constrained case. First, we generalize the definitions of positive invariance of a set.

Definition B.38 (Control invariance (constrained))**.** The closed set $\mathcal{A}$ is control invariant for $x^+ = f(x,u)$, $u \in \mathbb{U}$ if, for all $x \in \mathcal{A}$, there exists a $u \in \mathbb{U}$ such that $f(x,u) \in \mathcal{A}$.

Suppose that the state x is required to lie in the closed set $\mathbb{X} \subset \mathbb{R}^n$. In order to show that it is possible to ensure a decrease of a Lyapunov function, as in (B.24), in the presence of the state constraint $x \in \mathbb{X}$, we assume that there exists a control invariant set $X \subseteq \mathbb{X}$ for $x^+ = f(x, u, w)$, $u \in \mathbb{U}$, $w \in \mathbb{W}$. This enables us to obtain a control law that keeps the state in X and, hence, in $\mathbb{X}$, and, under suitable conditions, to satisfy a variant of (B.24).

Definition B.39 (CLF (constrained))**.** Suppose the set X and closed set $\mathcal{A}$, $\mathcal{A} \subset X$, are control invariant for $x^+ = f(x, u)$, $u \in \mathbb{U}$. A function $V : X \to \mathbb{R}_{\geq 0}$ is said to be a control Lyapunov function in X for the system $x^+ = f(x, u)$, $u \in \mathbb{U}$, and closed set $\mathcal{A}$ in X if there exist functions $\alpha_i \in \mathcal{K}_\infty$, $i = 1, 2, 3$, defined on X, such that for any $x \in X$,

$$\alpha_1(|x|_{\mathcal{A}}) \leq V(x) \leq \alpha_2(|x|_{\mathcal{A}})$$
$$\inf_{u \in \mathbb{U}} \{V(f(x, u)) \mid f(x, u) \in X\} - V(x) \leq -\alpha_3(|x|_{\mathcal{A}})$$

Remark. Again, if $V(\cdot)$ is a control Lyapunov function in X for $x^+ = f(x, u)$, $u \in \mathbb{U}$ and closed set $\mathcal{A}$ in X, one can choose a control law $\kappa : \mathbb{R}^n \to \mathbb{U}$ satisfying

$$V(f(x, \kappa(x))) - V(x) \leq -\alpha_3(|x|_{\mathcal{A}})/2$$

for all $x \in X$. Since $\mathbb{U}$ is compact, $\kappa(\cdot)$ is locally bounded and, hence, so is $x \mapsto f(x, \kappa(x))$. Thus, when $\alpha_3(\cdot)$ is a $\mathcal{K}_\infty$ function, we may use Theorem B.18 to deduce that $\mathcal{A}$ is asymptotically stable for $x^+ = f(x, \kappa(x))$, $u \in \mathbb{U}$ in X; also $\phi(i; x) \in X \subset \mathbb{X}$ for all $x \in X$, all $i \in \mathbb{I}_{\geq 0}$.

Finally we consider the constrained case in the presence of disturbances. First we define control invariance in the presence of disturbances.

Definition B.40 (Control invariance (disturbances, constrained))**.** The closed set $\mathcal{A}$ is control invariant for $x^+ = f(x, u, w)$, $u \in \mathbb{U}$, $w \in \mathbb{W}$ if, for all $x \in \mathcal{A}$, there exists a $u \in \mathbb{U}$ such that $f(x, u, w) \in \mathcal{A}$ for all $w \in \mathbb{W}$ (or $F(x, u) \subseteq \mathcal{A}$ where $F(x, u) \coloneqq \{f(x, u, w) \mid w \in \mathbb{W}\}$).

Next, we define what we mean by a control Lyapunov function in this context.

Definition B.41 (CLF (disturbances, constrained))**.** Suppose the set X and closed set $\mathcal{A}$, $\mathcal{A} \subset X$, are control invariant for $x^+ = f(x, u, w)$, $u \in \mathbb{U}$, $w \in \mathbb{W}$. A function $V : X \to \mathbb{R}_{\geq 0}$ is said to be a control Lyapunov

function in X for the system $x^+ = f(x,u,w)$, $u \in \mathbb{U}$, $w \in \mathbb{W}$ and set $\mathcal{A}$ if there exist functions $\alpha_i \in \mathcal{K}_\infty$, $i = 1,2,3$, defined on X, such that for any $x \in X$,

$$\alpha_1(|x|_{\mathcal{A}}) \le V(x) \le \alpha_2(|x|_{\mathcal{A}})$$
$$\inf_{u\in\mathbb{U}} \sup_{z\in F(x,u)\cap X} V(z) - V(x) \le -\alpha_3(|x|_{\mathcal{A}})$$

Suppose now that the state x is required to lie in the closed set $\mathbb{X} \subset \mathbb{R}^n$. Again, in order to show that there exists a condition similar to (B.24), we assume that there exists a control invariant set $X \subseteq \mathbb{X}$ for $x^+ = f(x,u,w)$, $u \in \mathbb{U}$, $w \in \mathbb{W}$. This enables us to obtain a control law that keeps the state in X and, hence, in $\mathbb{X}$, and, under suitable conditions, to satisfy a variant of (B.24).

Remark. If $V(\cdot)$ is a control Lyapunov function in X for $x^+ = f(x,u)$, $u \in \mathbb{U}$, $w \in \mathbb{W}$ and set $\mathcal{A}$ in X, one can choose a control law $\kappa : X \to \mathbb{U}$ satisfying

$$\sup_{z\in F(x,\kappa(x))} V(z) - V(x) \le -\alpha_3(|x|_{\mathcal{A}})/2$$

for all $x \in X$. Since $\mathbb{U}$ is compact, $\kappa(\cdot)$ is locally bounded and, hence, so is $x \mapsto f(x,\kappa(x))$. Thus, when $\alpha_3(\cdot)$ is a $\mathcal{K}_\infty$ function, we may use Theorem B.18 to deduce that $\mathcal{A}$ is asymptotically stable in X for $x^+ = f(x,\kappa(x),w)$, $w \in \mathbb{W}$ (or, equivalently, for $x^+ \in F(x,\kappa(x))$); also $\phi(i) \in X \subset \mathbb{X}$ for all $x \in X$, all $i \in \mathbb{I}_{\ge 0}$, all $\phi \in S(x)$.

B.6 Input-to-State Stability

We consider, as in the previous section, the system

$$x^+ = f(x,w)$$

where the disturbance w takes values in $\mathbb{R}^p$. In input-to-state stability (Sontag and Wang, 1995; Jiang and Wang, 2001) we seek a bound on the state in terms of a uniform bound on the disturbance sequence $\mathbf{w} := (w(0), w(1), \ldots)$. Let $\|\cdot\|$ denote the usual ℓ_∞ norm for sequences, i.e., $\|\mathbf{w}\| := \sup_{k\ge 0} |w(k)|$.

Definition B.42 (Input-to-state stable (ISS))**.** The system $x^+ = f(x,w)$ is (globally) input-to-state stable (ISS) if there exists a $\mathcal{KL}$ function $\beta(\cdot)$ and a $\mathcal{K}$ function $\sigma(\cdot)$ such that, for each $x \in \mathbb{R}^n$, and each disturbance sequence $\mathbf{w} = (w(0), w(1), \ldots)$ in ℓ_∞

$$\left|\phi(i; x, \mathbf{w}_i)\right| \le \beta(|x|, i) + \sigma(\|\mathbf{w}_i\|)$$

for all $i \in \mathbb{I}_{\geq 0}$, where $\phi(i; x, \mathbf{w}_i)$ is the solution, at time i, if the initial state is x at time 0 and the input sequence is $\mathbf{w}_i := (w(0), w(1), \ldots, w(i-1))$.

We note that this definition implies the origin is globally asymptotically stable if the input sequence is identically zero. Also, the norm of the state is asymptotically bounded by $\sigma(\|\mathbf{w}\|)$ where $\mathbf{w} := (w(0), w(1), \ldots)$. As before, we seek a Lyapunov function that ensures input-to-state stability.

Definition B.43 (ISS-Lyapunov function)**.** A function $V : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ is an ISS-Lyapunov function for system $x^+ = f(x, w)$ if there exist $\mathcal{K}_\infty$ functions $\alpha_1(\cdot), \alpha_2(\cdot), \alpha_3(\cdot)$ and a $\mathcal{K}$ function $\sigma(\cdot)$ such that for all $x \in \mathbb{R}^n$, $w \in \mathbb{R}^p$

$$\alpha_1(|x|) \leq V(x) \leq \alpha_2(|x|)$$
$$V(f(x, w)) - V(x) \leq -\alpha_3(|x|) + \sigma(|w|)$$

The following result appears in Jiang and Wang (2001, Lemma 3.5)

Lemma B.44 (ISS-Lyapunov function implies ISS)**.** *Suppose $f(\cdot)$ is continuous and that there exists a continuous ISS-Lyapunov function for $x^+ = f(x, w)$. Then the system $x^+ = f(x, w)$ is ISS.*

The converse, i.e., input-to-state stability implies the existence of a smooth ISS-Lyapunov function for $x^+ = f(x, w)$ is also proved in Jiang and Wang (2002, Theorem 1). We now consider the case when the state satisfies the constraint $x \in \mathbb{X}$ where $\mathbb{X}$ is a closed subset of $\mathbb{R}^n$. Accordingly, we assume that the disturbance w satisfies $w \in \mathbb{W}$ where $\mathbb{W}$ is a compact set containing the origin and that $X \subset \mathbb{X}$ is a closed robust positive invariant set for $x^+ = f(x, w)$, $w \in \mathbb{W}$ or, equivalently, for $x^+ \in F(x, u)$.

Definition B.45 (ISS (constrained))**.** Suppose that $\mathbb{W}$ is a compact set containing the origin and that $X \subset \mathbb{X}$ is a closed robust positive invariant set for $x^+ = f(x, w)$, $w \in \mathbb{W}$. The system $x^+ = f(x, w)$, $w \in \mathbb{W}$ is ISS in X if there exists a class $\mathcal{KL}$ function $\beta(\cdot)$ and a class $\mathcal{K}$ function $\sigma(\cdot)$ such that, for all $x \in X$, all $\mathbf{w} \in \mathcal{W}$ where $\mathcal{W}$ is the set of infinite sequences $\mathbf{w}$ satisfying $w(i) \in \mathbb{W}$ for all $i \in \mathbb{I}_{\geq 0}$

$$\left| \phi(i; x, \mathbf{w}_i) \right| \leq \beta(|x|, i) + \sigma(\|\mathbf{w}_i\|)$$

Definition B.46 (ISS-Lyapunov function (constrained))**.** A function $V : X \to \mathbb{R}_{\geq 0}$ is an ISS-Lyapunov function in X for system $x^+ = f(x, w)$

if there exist $\mathcal{K}_\infty$ functions $\alpha_1(\cdot), \alpha_2(\cdot), \alpha_3(\cdot)$ and a $\mathcal{K}$ function $\sigma(\cdot)$ such that for all $x \in X$, all $w \in \mathbb{W}$

$$\alpha_1(|x|) \le V(x) \le \alpha_2(|x|)$$
$$V(f(x,w)) - V(x) \le -\alpha_3(|x|) + \sigma(|w|)$$

The following result is a minor generalization of Lemma 3.5 in Jiang and Wang (2001).

Lemma B.47 (ISS-Lyapunov function implies ISS (constrained))**.** *Suppose that $\mathbb{W}$ is a compact set containing the origin and that $X \subset \mathbb{X}$ is a closed robust positive invariant set for $x^+ = f(x,w)$, $w \in \mathbb{W}$. If $f(\cdot)$ is continuous and there exists a continuous ISS-Lyapunov function in X for the system $x^+ = f(x,w)$, $w \in \mathbb{W}$, then the system $x^+ = f(x,w)$, $w \in \mathbb{W}$ is ISS in X.*

B.7 Output-to-State Stability and Detectability

We present some definitions and results that are discrete time versions of results due to Sontag and Wang (1997) and Krichman, Sontag, and Wang (2001). The output-to-state (OSS) property corresponds, informally, to the statement that "no matter what the initial state is, if the observed outputs are small, then the state must eventually be small". It is therefore a natural candidate for the concept of nonlinear (zero-state) detectability. We consider first the autonomous system

$$x^+ = f(x) \qquad y = h(x) \tag{B.25}$$

where $f(\cdot) : \mathbb{X} \to \mathbb{X}$ is locally Lipschitz continuous and $h(\cdot)$ is continuously differentiable where $\mathbb{X} = \mathbb{R}^n$ for some n. We assume $x = 0$ is an equilibrium state, i.e., $f(0) = 0$. We also assume $h(0) = 0$. We use $\phi(k; x_0)$ to denote the solution of (B.25) with initial state x_0, and $y(k; x_0)$ to denote $h(\phi(k; x_0))$. The function $y_{x_0}(\cdot)$ is defined by

$$y_{x_0}(k) := y(k; x_0)$$

We use $|\cdot|$ and $\|\cdot\|$ to denote, respectively the Euclidean norm of a vector and the sup norm of a sequence; $\|\cdot\|_{0:k}$ denotes the max norm of a sequence restricted to the interval $[0,k]$. For conciseness, $\mathbf{u}$, $\mathbf{y}$ denote, respectively, the sequences $(u(j))$, $(y(j))$.

Definition B.48 (Output-to-state stable (OSS))**.** The system (B.25) is output-to-state stable (OSS) if there exist functions $\beta(\cdot) \in \mathcal{KL}$ and $\gamma(\cdot) \in \mathcal{K}$ such that for all $x_0 \in \mathbb{R}^n$ and all $k \geq 0$

$$|x(k; x_0)| \leq \max\{\beta(|x_0|, k),\ \gamma(\|\mathbf{y}\|_{0:k})\}$$

Definition B.49 (OSS-Lyapunov function)**.** An OSS-Lyapunov function for system (B.25) is any function $V(\cdot)$ with the following properties

(a) There exist $\mathcal{K}_\infty$ functions $\alpha_1(\cdot)$ and $\alpha_2(\cdot)$ such that

$$\alpha_1(|x|) \leq V(x) \leq \alpha_2(|x|)$$

for all x in $\mathbb{R}^n$.

(b) There exist $\mathcal{K}_\infty$ functions $\alpha(\cdot)$ and $\sigma(\cdot)$ such that for all $x \in \mathbb{R}^n$ either

$$V(x^+) \leq V(x) - \alpha(|x|) + \sigma(|y|)$$

or

$$V(x^+) \leq \rho V(x) + \sigma(|y|) \tag{B.26}$$

with $x^+ = f(x)$, $y = h(x)$, and $\rho \in (0, 1)$.

Inequality (B.26) corresponds to an exponential-decay OSS-Lyapunov function.

Theorem B.50 (OSS and OSS-Lyapunov function)**.** *The following properties are equivalent for system* (B.25)*:*

(a) The system is OSS.

(b) The system admits an OSS-Lyapunov function.

(c) The system admits an exponential-decay OSS-Lyapunov function.

B.8 Input/Output-to-State Stability

Consider now a system with both inputs and outputs

$$x^+ = f(x, u) \qquad y = h(x) \tag{B.27}$$

Input/output-to-state stability corresponds roughly to the statement that, no matter what the initial state is, if the input and the output converge to zero, so does the state. We assume $f(\cdot)$ and $h(\cdot)$ are continuous. We also assume $f(0, 0) = 0$ and $h(0) = 0$. Let $x(\cdot, x_0, \mathbf{u})$ denote the solution of (B.27) which results from initial state x_0 and control $\mathbf{u} = (u(j))_{j \geq 0}$ and let $y_{x_0,\mathbf{u}}(k) := y(k; x_0, \mathbf{u})$ denote $h(x(k; x_0, \mathbf{u}))$.

Definition B.51 (Input/output-to-state stable (IOSS))**.** The system (B.27) is input/output-to-state stable (IOSS) if there exist functions $\beta(\cdot) \in \mathcal{KL}$ and $\gamma_1(\cdot)$, $\gamma_2(\cdot) \in \mathcal{K}$ such that

$$|x(k;x_0)| \leq \max\left\{\beta(|x_0|,k), \gamma_1\left(\|\mathbf{u}\|_{0:k-1}\right), \gamma_2\left(\|\mathbf{y}\|_{0:k}\right)\right\}$$

for every initial state $x_0 \in \mathbb{R}^n$, every control sequence $\mathbf{u} = (u(j))$, and all $k \geq 0$.

Definition B.52 (IOSS-Lyapunov function)**.** An IOSS-Lyapunov function for system (B.27) is any function $V(\cdot)$ with the following properties:

(a) There exist $\mathcal{K}_\infty$ functions $\alpha_1(\cdot)$ and $\alpha_2(\cdot)$ such that

$$\alpha_1(|x|) \leq V(x) \leq \alpha_2(|x|)$$

for all $x \in \mathbb{R}^n$.

(b) There exist $\mathcal{K}_\infty$ functions $\alpha(\cdot)$, $\sigma_1(\cdot)$, and $\sigma_2(\cdot)$ such that for every x and u either

$$V(x^+) \leq V(x) - \alpha(|x|) + \sigma_1(|u|) + \sigma_2(|y|)$$

or

$$V(x^+) \leq \rho V(x) + \sigma_1(|u|) + \sigma_2(|y|)$$

with $x^+ = f(x,u)$, $y = h(x)$, and $\rho \in (0,1)$.

The following result proves useful when establishing that MPC employing cost functions based on the inputs and outputs rather than inputs and states is stabilizing for IOSS systems. Consider the system $x^+ = f(x,u), y = h(x)$ with stage cost $\ell(y,u)$ and constraints $(x, u) \in \mathbb{Z}$. The stage cost satisfies $\ell(0,0) = 0$ and $\ell(y,u) \geq \alpha(|(y,u)|)$ for all $(y,u) \in \mathbb{R}^p \times \mathbb{R}^m$ with α_1 a $\mathcal{K}_\infty$ function. Let $\mathbb{X} := \{x \mid \exists u \text{ with } (x,u) \in \mathbb{Z}\}$.

Theorem B.53 (Modified IOSS-Lyapunov function)**.** *Assume that there exists an IOSS-Lyapunov function $V : \mathbb{X} \to \mathbb{R}_{\geq 0}$ for the constrained system $x^+ = f(x,u)$ such that the following holds for all $(x,u) \in \mathbb{Z}$ for which $f(x,u) \in \mathbb{X}$*

$$\alpha_1(|x|) \leq V(x) \leq \alpha_2(|x|)$$
$$V(f(x,u)) - V(x) \leq -\alpha_3(|x|) + \sigma(\ell(y,u))$$

with $\alpha_1, \alpha_2, \alpha_3 \in \mathcal{K}_\infty$ and $\sigma \in \mathcal{K}$.

For any $\overline{\alpha}_4 \in \mathcal{K}_\infty$, there exists another IOSS-Lyapunov function $\Lambda : \mathbb{X} \to \mathbb{R}_{\geq 0}$ for the constrained system $x^+ = f(x,u)$ such that the following holds for all $(x,u) \in \mathbb{Z}$ for which $f(x,u) \in \mathbb{X}$

$$\overline{\alpha}_1(|x|) \leq \Lambda(x) \leq \overline{\alpha}_2(|x|)$$
$$\Lambda(f(x,u)) - \Lambda(x) \leq -\rho(|x|) + \overline{\alpha}_4(\ell(y,u))$$

with $\overline{\alpha}_1, \overline{\alpha}_2 \in \mathcal{K}_\infty$ and continuous function $\rho \in \mathcal{PD}$. Note that $\Lambda = \gamma \circ V$ for some $\gamma \in \mathcal{K}$.

Conjecture B.54 (IOSS and IOSS-Lyapunov function)**.** *The following properties are equivalent for system* (B.27)*:*

(a) The system is IOSS.

(b) The system admits a smooth IOSS-Lyapunov function.

(c) The system admits an exponential-decay IOSS-Lyapunov function.

As discussed in the Notes section of Chapter 2, Grimm, Messina, Tuna, and Teel (2005) use a storage function like $\Lambda(\cdot)$ in Theorem B.53 to treat a semidefinite stage cost. Cai and Teel (2008) provide a discrete time converse theorem for IOSS that holds for all $\mathbb{R}^n$. Allan and Rawlings (2018) provide the converse theorem on closed positive invariant sets (Theorem 36), and also provide a lemma for changing the supply rate function (Theorem 38).

B.9 Incremental-Input/Output-to-State Stability

Definition B.55 (Incremental input/output-to-state stable)**.** The system (B.27) is incrementally input/output-to-state stable (i-IOSS) if there exists some $\beta(\cdot) \in \mathcal{KL}$ and $\gamma_1(\cdot)$, $\gamma_2(\cdot) \in \mathcal{K}$ such that, for every two initial states z_1 and z_2 and any two control sequences $\mathbf{u}_1 = (u_1(j))$ and $\mathbf{u}_2 = (u_2(j))$

$$|x(k; z_1, \mathbf{u}_1) - x(k; z_2, \mathbf{u}_2)| \leq$$
$$\max\left\{\beta(|z_1 - z_2|, k),\ \gamma_1(\|\mathbf{u}_1 - \mathbf{u}_2\|_{0:k-1}),\ \gamma_2(\|\mathbf{y}_{z_1,\mathbf{u}_1} - \mathbf{y}_{z_2,\mathbf{u}_2}\|_{0:k})\right\}$$

B.10 Observability

Definition B.56 (Observability)**.** The system (B.27) is (uniformly) observable if there exists a positive integer N and an $\alpha(\cdot) \in \mathcal{K}$ such that

$$\sum_{j=0}^{k-1} |h(x(j; x, u)) - h(x(j; z, u))| \geq \alpha(|x - z|) \tag{B.28}$$

for all x, z, all $k \geq N$ and all control sequences u; here $x(j; z, u) = \phi(j; z, u)$, the solution of (B.27) when the initial state is z at time 0 and the control sequence is u.

When the system is linear, i.e., $f(x, u) = Ax + Bu$ and $h(x) = Cx$, this assumption is equivalent to assuming the observability Gramian $\sum_{j=0}^{n-1} CA^j(A^j)'C'$ is positive definite. Consider the system described by

$$z^+ = f(z, u) + w \qquad y + v = h(z) \tag{B.29}$$

with output $y_w = y + v$. Let $z(k; z, u, w)$ denote the solution, at time k of (B.29) if the state at time 0 is z, the control sequence is u and the disturbance sequence is w. We assume, in the sequel, that

Assumption B.57 (Lipschitz continuity of model)**.**

(a) The function $f(\cdot)$ is globally Lipschitz continuous in $\mathbb{R}^n \times \mathbf{U}$ with Lipschitz constant c.

(b) The function $h(\cdot)$ is globally Lipschitz continuous in $\mathbb{R}^n$ with Lipschitz constant c.

Lemma B.58 (Lipschitz continuity and state difference bound)**.** *Suppose Assumption B.57 is satisfied (with Lipschitz constant c). Then,*

$$|x(k; x, u) - z(k; z, u, w)| \leq c^k |x - z| + \sum_{i=0}^{k-1} c^{k-i-1} |w(i)|$$

Proof. Let $\delta(k) := |x(k; x, u) - z(k; z, u, w)|$. Then

$$\begin{aligned} \delta(k+1) &= \left| f(x(k; x, u), u(k)) - f(z(k; z, u, w), u(k)) - w(k) \right| \\ &\leq c\,|\delta(k)| + |w(k)| \end{aligned}$$

Iterating this equation yields the desired result. ∎

Theorem B.59 (Observability and convergence of state)**.** *Suppose* (B.27) *is (uniformly) observable and that Assumption B.57 is satisfied. Then, $w(k) \to 0$ and $v(k) \to 0$ as $k \to \infty$ imply $|x(k; x, u) - z(k; z, u, w)| \to 0$ as $k \to \infty$.*

Proof. Let $x(k)$ and $z(k)$ denote $x(k; x, u)$ and $z(k; z, u, w)$, respectively, in the sequel. Since (B.27) is observable, there exists an integer

N satisfying (B.28). Consider the sum

$$\begin{aligned} S(k) = \sum_{j=k}^{k+N} v(k) &= \sum_{j=k}^{k+N} \left| h(x(j;x,u)) - h(z(j;z,u,w)) \right| \\ &\geq \sum_{j=k}^{k+N} \left| h(x(j;x(k),u)) - h(x(j;z(k),u)) \right| \\ &\quad - \sum_{j=k}^{k+N} \left| h(x(j;z(k),u)) - h(z(j;z(k),u,w)) \right| \end{aligned} \tag{B.30}$$

where we have used the fact that $|a+b| \geq |a| - |b|$. By the assumption of observability

$$\sum_{j=k}^{k+N} \left| h(x(j;x(k),u)) - h(x(j;z(k),u)) \right| \geq \alpha(|x(k) - z(k)|)$$

for all k. From Lemma B.58 and the Lipschitz assumption on $h(\cdot)$

$$\begin{aligned} &\left| h(x(j;z(k),u)) - h(z(j;z(k),u,w)) \right| \leq \\ &\qquad c \left| x(j;z(k),u) - z(j;z(k),u,w) \right| \leq c \sum_{i=k}^{j-1} c^{j-1-i} |w(i)| \end{aligned}$$

for all j in $\{k+1, k+2, \ldots k+N\}$. Hence there exists a $d \in (0,\infty)$ such that the last term in (B.30) satisfies

$$\sum_{j=k}^{k+N} \left| h(x(j;x(k),u)) - h(x(j;z(k),u)) \right| \leq d \, \|w\|_{k-N:k}$$

Hence, (B.30) becomes

$$\alpha(|x(k) - z(k)|) \leq N \, \|v\|_{k-N:k} + d \, \|w\|_{k-N:k}$$

Since, by assumption, $w(k) \to 0$ and $v(k) \to 0$ as $k \to \infty$, and $\alpha(\cdot) \in \mathcal{K}$, it follows that $|x(k) - z(k)| \to 0$ as $k \to \infty$. ■

B.11 Exercises

Exercise B.1: Lyapunov equation and linear systems

Establish the equivalence of (a) and (b) in Lemma B.20.

Exercise B.2: Lyapunov function for exponential stability

Let $V : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ be a Lyapunov function for the system $x^+ = f(x)$ with the following properties. For all $x \in \mathbb{R}^n$

$$a_1 |x|^\sigma \leq V(x) \leq a_2 |x|^\sigma$$
$$V(f(x)) - V(x) \leq -a_3 |x|^\sigma$$

in which $a_1, a_2, a_3, \sigma > 0$. Show that the origin of the system $x^+ = f(x)$ is globally exponentially stable.

Exercise B.3: A converse theorem for exponential stability

(a) Assume that the origin is globally exponentially stable (GES) for the system

$$x^+ = f(x)$$

in which $f(\cdot)$ is continuous. Show that there exists a continuous Lyapunov function $V(\cdot)$ for the system satisfying for all $x \in \mathbb{R}^n$

$$a_1 |x|^\sigma \leq V(x) \leq a_2 |x|^\sigma$$
$$V(f(x)) - V(x) \leq -a_3 |x|^\sigma$$

in which $a_1, a_2, a_3, \sigma > 0$.

Hint: Consider summing the solution $|\phi(i;x)|^\sigma$ on i as a candidate Lyapunov function $V(x)$.

(b) Establish that in the Lyapunov function defined above, any $\sigma > 0$ is valid, and also that the constant a_3 can be chosen as large as one wishes.

Exercise B.4: Revisit Lemma 1.3 in Chapter 1

Establish Lemma 1.3 in Chapter 1 using the Lyapunov function tools established in this appendix. Strengthen the conclusion and establish that the closed-loop system is globally exponentially stable.

Exercise B.5: Continuity of Lyapunov function for asymptotic stability

Let X be a compact subset of $\mathbb{R}^n$ containing the origin in its interior that is positive invariant for the system $x^+ = f(x)$. If $f(\cdot)$ is continuous on X and the origin is asymptotically stable with a region of attraction X, show that the Lyapunov function suggested in Theorem B.17 is continuous on X.

Exercise B.6: A Lipschitz continuous converse theorem for exponential stability

Consider the system $x^+ = f(x)$, $f(0) = 0$, with function $f : D \to \mathbb{R}^n$ Lipschitz continuous on compact set $D \subset \mathbb{R}^n$ containing the origin in its interior. Choose $R > 0$ such that $B_R \subseteq D$. Assume that there exist scalars $c > 0$ and $\lambda \in (0, 1)$ such that

$$|\phi(k;x)| \leq c |x| \lambda^k \qquad \text{for all } |x| \leq r, \quad k \geq 0$$

with $r := R/c$.

Show that there exists a *Lipschitz continuous* Lyapunov function $V(\cdot)$ satisfying for all $x \in B_r$

$$a_1 |x|^2 \leq V(x) \leq a_2 |x|^2$$
$$V(f(x)) - V(x) \leq -a_3 |x|^2$$

with $a_1, a_2, a_3 > 0$.

Hint: Use the proposed Lyapunov function of Exercise B.3 with $\sigma = 2$. See also (Khalil, 2002, Exercise 4.68).

Exercise B.7: Lyapunov function requirements: continuity of α_3

Consider the following scalar system $x^+ = f(x)$ with piecewise affine and discontinuous $f(\cdot)$ (Lazar et al., 2009)

$$f(x) = \begin{cases} 0, & x \in (-\infty, 1] \\ (1/2)(x+1), & x \in (1, \infty) \end{cases}$$

Note that the origin is a steady state

(a) Consider $V(x) = |x|$ as a candidate Lyapunov function. Show that this V satisfies (B.11)–(B.13) of Definition B.12, in which $\alpha_3(x)$ is positive definite but *not* continuous.

(b) Show by direction calculation that the origin is not globally asymptotically stable. Show that for initial conditions $x_0 \in (1, \infty)$, $x(k; x_0) \to 1$ as $k \to \infty$.

The conclusion here is that one cannot leave out continuity of α_3 in the definition of a Lyapunov function when allowing discontinuous system dynamics.

Exercise B.8: Difference between classical and KL stability definitions (Teel)

Consider the *discontinuous* nonlinear scalar example $x^+ = f(x)$ with

$$f(x) = \begin{cases} \frac{1}{2}x & |x| \in [0, 1] \\ \frac{2x}{2 - |x|} & |x| \in (1, 2) \\ 0 & |x| \in [2, \infty) \end{cases}$$

Is this system GAS under the classical definition? Is this system GAS under the KL definition? Discuss why or why not.

Exercise B.9: Combining $\mathcal{K}$ functions

Establish (B.5) and (B.7) starting from (B.3) and (B.4) and then using (B.1).

Exercise B.10

Derive $\mathcal{KL}$ bounds (B.6) and (B.8) from (B.5) and (B.7), respectively.

Bibliography

D. A. Allan and J. B. Rawlings. An input/output-to-state stability converse theorem for closed positive invariant sets. Technical Report 2018–01, TWCCC Technical Report, December 2018.

D. A. Allan, C. N. Bates, M. J. Risbeck, and J. B. Rawlings. On the inherent robustness of optimal and suboptimal nonlinear MPC. *Sys. Cont. Let.*, 106: 68–78, August 2017.

R. G. Bartle and D. R. Sherbert. *Introduction to Real Analysis*. John Wiley & Sons, Inc., New York, third edition, 2000.

C. Cai and A. R. Teel. Input–output-to-state stability for discrete-time systems. *Automatica*, 44(2):326 – 336, 2008.

G. Grimm, M. J. Messina, S. E. Tuna, and A. R. Teel. Model predictive control: For want of a local control Lyapunov function, all is not lost. *IEEE Trans. Auto. Cont.*, 50(5):546–558, 2005.

Z.-P. Jiang and Y. Wang. Input-to-state stability for discrete-time nonlinear systems. *Automatica*, 37:857–869, 2001.

Z.-P. Jiang and Y. Wang. A converse Lyapunov theorem for discrete-time systems with disturbances. *Sys. Cont. Let.*, 45:49–58, 2002.

R. E. Kalman and J. E. Bertram. Control system analysis and design via the "Second method" of Lyapunov, Part II: Discrete–time systems. *ASME J. Basic Engr.*, pages 394–400, June 1960.

C. M. Kellett and A. R. Teel. Discrete-time asymptotic controllability implies smooth control-Lyapunov function. *Sys. Cont. Let.*, 52:349–359, 2004a.

C. M. Kellett and A. R. Teel. Smooth Lyapunov functions and robustness of stability for difference inclusions. *Sys. Cont. Let.*, 52:395–405, 2004b.

H. K. Khalil. *Nonlinear Systems*. Prentice-Hall, Upper Saddle River, NJ, third edition, 2002.

M. Krichman, E. D. Sontag, and Y. Wang. Input-output-to-state stability. *SIAM J. Cont. Opt.*, 39(6):1874–1928, 2001.

J. P. LaSalle. *The stability and control of discrete processes*, volume 62 of *Applied Mathematical Sciences*. Springer-Verlag, 1986.

M. Lazar, W. P. M. H. Heemels, and A. R. Teel. Lyapunov functions, stability and input-to-state stability subtleties for discrete-time discontinuous systems. *IEEE Trans. Auto. Cont.*, 54(10):2421–2425, 2009.

J. B. Rawlings and L. Ji. Optimization-based state estimation: Current status and some new results. *J. Proc. Cont.*, 22:1439–1444, 2012.

J. B. Rawlings and M. J. Risbeck. On the equivalence between statements with epsilon-delta and K-functions. Technical Report 2015–01, TWCCC Technical Report, December 2015.

J. B. Rawlings and M. J. Risbeck. Model predictive control with discrete actuators: Theory and application. *Automatica*, 78:258–265, 2017.

E. D. Sontag. *Mathematical Control Theory*. Springer-Verlag, New York, second edition, 1998a.

E. D. Sontag. Comments on integral variants of ISS. *Sys. Cont. Let.*, 34:93–100, 1998b.

E. D. Sontag and Y. Wang. On the characterization of the input to state stability property. *Sys. Cont. Let.*, 24:351–359, 1995.

E. D. Sontag and Y. Wang. Output-to-state stability and detectability of nonlinear systems. *Sys. Cont. Let.*, 29:279–290, 1997.

A. R. Teel. Discrete time receding horizon control: is the stability robust. In Marcia S. de Queiroz, Michael Malisoff, and Peter Wolenski, editors, *Optimal control, stabilization and nonsmooth analysis*, volume 301 of *Lecture notes in control and information sciences*, pages 3–28. Springer, 2004.

C
Optimization

Version: date: April 27, 2026
Copyright © 2026 by Nob Hill Publishing, LLC

C.1 Dynamic Programming

The name *dynamic programming* dates from the 1950s when it was coined by Richard Bellman for a technique for solving dynamic optimization problems, i.e., optimization problems associated with deterministic or stochastic systems whose behavior is governed by differential or difference equations. Here we review some of the basic ideas behind dynamic programming (DP) Bellman (1957); Bertsekas, Nedic, and Ozdaglar (2001).

To introduce the topic in its simplest form, consider the simple routing problem illustrated in Figure C.1. To maintain connection with optimal control, each node in the graph can be regarded as a point (x, t) in a subset S of $X \times T$ where both the state space $X = \{a, b, c, \ldots, g\}$ and the set of times $T = \{0, 1, 2, 3\}$ are discrete. The set of permissible control actions is $\mathbb{U} = \{U, D\}$, i.e., to go "up" or "down." The control problem is to choose the lowest cost path from event $(d, 0)$ (state d at $t = 0$) to any of the states at $t = 3$; the cost of going from one event to the next is indicated on the graph. This problem is equivalent to choosing an open-loop control, i.e., a sequence $(u(0), u(1), u(2))$ of admissible control actions. There are 2^N controls where N is the number of stages, 3 in this example. The cost of each control can, in this simple example, be evaluated and is given in Table C.1.

There are two different *open-loop* optimal controls, namely (U, D, U) and (D, D, D), each incurring a cost of 16. The corresponding

control	UUU	UUD	UDU	UDD	DUU	DUD	DDU	DDD
cost	20	24	16	24	24	32	20	16

Table C.1: Control Cost.

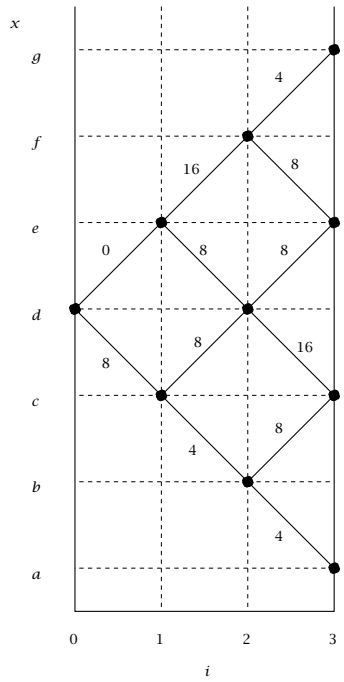


Figure C.1: Routing problem.

state trajectories are (d, e, d, e) and (d, c, b, a).

In discrete problems of this kind, DP replaces the N-stage problem by M single stage problems, where M is the total number of nodes, i.e., the number of elements in $S \subset X \times T$. The first set of optimization problems deals with the states b, d, f at time $N - 1 = 2$. The optimal decision at event $(f, 2)$, i.e., state f at time 2, is the control U and gives rise to a cost of 4. The optimal cost and control for node $(f, 2)$ are recorded; see Table C.2. The procedure is then repeated for states d and b at time $t = 2$ (nodes $(d, 2)$ and $(b, 2)$) and recorded as shown in Table C.2. Attention is next focused on the states e and c at $t = 1$ (nodes $(e, 1)$ and $(c, 1)$). The lowest cost that can be achieved at node $(e, 1)$ *if* control U is chosen, is $16 + 4$, the sum of the path cost 16 associated with the control U, and the *optimal* cost 4 associated with the node $(f, 2)$ that results from using control U at node $(e, 1)$. Similarly the lowest possible cost, if control D is chosen, is $8 + 8$. Hence

t	0	1		2		
state	d	e	c	f	d	b
control	U or D	D	D	U	U	D
optimal cost	16	16	8	4	8	4

Table C.2: Optimal Cost and Control

the optimal control and cost for node $(e, 1)$ are, respectively, D and 16. The procedure is repeated for the remaining state d at $t = 1$ (node $(d, 1)$). A similar calculation for the state d at $t = 0$ (node $(d, 0)$), where the optimal control is U or D, completes this backward recursion; this backward recursion provides the optimal cost and control for each (x, t), as recorded in Table C.2. The procedure therefore yields an optimal *feedback* control that is a function of $(x, t) \in S$. To obtain the optimal open-loop control for the initial node $(d, 0)$, the feedback law is obeyed, leading to control U or D at $t = 0$; if U is chosen, the resultant state at $t = 1$ is e. From Table C.2, the optimal control at $(e, 1)$ is D, so that the successor node is $(d, 2)$. The optimal control at node $(d, 2)$ is U. Thus the optimal open-loop control sequence (U, D, U) is re-obtained. On the other hand, if the decision at $(d, 0)$ is chosen to be D, the optimal sequence (D, D, D) is obtained. This simple example illustrates the main features of DP that we will now examine in the context of discrete time optimal control.

C.1.1 Optimal Control Problem

The discrete time system we consider is described by

$$x^+ = f(x, u) \tag{C.1}$$

where $f(\cdot)$ is continuous. The system is subject to the mixed state-control constraint

$$(x, u) \in \mathbb{Z}$$

where $\mathbb{Z}$ is a closed subset of $\mathbb{R}^n \times \mathbb{R}^m$ and $\mathcal{P}_u(\mathbb{Z})$ is compact where $\mathcal{P}_u$ is the projection operator $(x, u) \mapsto u$. Often $\mathbb{Z} = \mathbb{X} \times \mathbb{U}$ in which case the constraint $(x, u) \in \mathbb{Z}$ becomes $x \in \mathbb{X}$ and $u \in \mathbb{U}$ and $\mathcal{P}_u(\mathbb{Z}) = \mathbb{U}$ so that $\mathbb{U}$ is compact. In addition there is a constraint on the terminal state $x(N)$:

$$x(N) \in \mathbb{X}_f$$

where $\mathbb{X}_f$ is closed. In this section we find it easier to express the value function and the optimal control in terms of the current state and current time i rather than using time-to-go k. Hence we replace time-to-go k by time i where $k = N - i$, replace $V_k^0(x)$ (the optimal cost at state x when the time-to-go is k) by $V^0(x, i)$ (the optimal cost at state x, time i) and replace X_k by $X(i)$ where $X(i)$ is the domain of $V^0(\cdot, i)$).

The cost associated with an initial state x at time 0 and a control sequence $\mathbf{u} := (u(0), u(1), \ldots, u(N-1))$ is

$$V(x, 0, \mathbf{u}) = V_f(x(N)) + \sum_{i=1}^{N-1} \ell(x(i), u(i)) \tag{C.2}$$

where $\ell(\cdot)$ and $V_f(\cdot)$ are continuous and, for each i, $x(i) = \phi(i; (x, 0), \mathbf{u})$ is the solution at time i of (C.1) if the initial state is x at time 0 and the control sequence is $\mathbf{u}$. The optimal control problem $\mathbb{P}(x, 0)$ is defined by

$$V^0(x, 0) = \min_{\mathbf{u}} V(x, 0, \mathbf{u}) \tag{C.3}$$

subject to the constraints $(x(i), u(i)) \in \mathbb{Z}$, $i = 0, 1, \ldots, N-1$ and $x(N) \in \mathbb{X}_f$. Equation (C.3) may be rewritten in the form

$$V^0(x, 0) = \min_{\mathbf{u}} \{V(x, 0, \mathbf{u}) \mid \mathbf{u} \in \mathcal{U}(x, 0)\} \tag{C.4}$$

where $\mathbf{u} := (u(0), u(1), \ldots, u(N-1))$,

$$\mathcal{U}(x, 0) := \{\mathbf{u} \in \mathbb{R}^{Nm} \mid (x(i), u(i)) \in \mathbb{Z}, i = 0, 1, \ldots, N-1; x(N) \in \mathbb{X}_f\}$$

and $x(i) := \phi(i; (x, 0), \mathbf{u})$. Thus $\mathcal{U}(x, 0)$ is the set of admissible control sequences[1] if the initial state is x at time 0. It follows from the continuity of $f(\cdot)$ that for all $i \in \{0, 1, \ldots, N-1\}$ and all $x \in \mathbb{R}^n$, $\mathbf{u} \mapsto \phi(i; (x, 0), \mathbf{u})$ is continuous, $\mathbf{u} \mapsto V(x, 0, \mathbf{u})$ is continuous and $\mathcal{U}(x, 0)$ is compact. Hence the minimum in (C.4) exists at all $x \in \{x \in \mathbb{R}^n \mid \mathcal{U}(x, 0) \neq \emptyset\}$.

DP embeds problem $\mathbb{P}(x, 0)$ for a given state x in a whole family of problems $P(x, i)$ where, for each (x, i), problem $\mathbb{P}(x, i)$ is defined by

$$V^0(x, i) = \min_{\mathbf{u}^i} \{V(x, i, \mathbf{u}^i) \mid \mathbf{u}^i \in \mathcal{U}(x, i)\}$$

where

$$\mathbf{u}^i := (u(i), u(i+1), \ldots, u(N-1))$$

[1] An admissible control sequence satisfies all constraints.

$$V(x,i,\mathbf{u}^i) := V_f(x(N)) + \sum_{j=i}^{N-1} \ell(x(j),u(j)) \tag{C.5}$$

and

$$\mathcal{U}(x,i) := \{\mathbf{u}^i \in \mathbb{R}^{(N-i)m} \mid (x(j),u(j)) \in \mathbb{Z}, j = i,i+1,\ldots,N-1 \\ x(N) \in \mathbb{X}_f\} \tag{C.6}$$

In (C.5) and (C.6), $x(j) = \phi(j;(x,i),\mathbf{u}^i)$, the solution at time j of (C.1) if the initial state is x at time i and the control sequence is $\mathbf{u}^i$. For each i, $X(i)$ denotes the domain of $V^0(\cdot,i)$ and $\mathcal{U}(\cdot,i)$ so that

$$X(i) = \{x \in \mathbb{R}^n \mid \mathcal{U}(x,i) \neq \emptyset\}. \tag{C.7}$$

C.1.2 Dynamic Programming

One way to approach DP for discrete time control problems is the simple observation that for all (x,i)

$$\begin{aligned} V^0(x,i) &= \min_{\mathbf{u}^i}\{V(x,i,\mathbf{u}^i) \mid \mathbf{u}^i \in \mathcal{U}(x,i)\} \\ &= \min_{u}\{\ell(x,u) + \min_{\mathbf{u}^{i+1}} V(f(x,u),i+1,\mathbf{u}^{i+1}) \mid \\ &\qquad \{u,\mathbf{u}^{i+1}\} \in \mathcal{U}(x,i)\} \end{aligned} \tag{C.8}$$

where $\mathbf{u}^i = (u,u(i+1),\ldots,u(N-1)) = \left(u,\mathbf{u}^{i+1}\right)$. We now make use of the fact that $\{u,\mathbf{u}^{i+1}\} \in \mathcal{U}(x,i)$ if and only if $(x,u) \in \mathbb{Z}$, $f(x,u) \in X(i+1)$, and $\mathbf{u}^{i+1} \in \mathcal{U}(f(x,u),i+1)$ since $f(x,u) = x(i+1)$. Hence we may rewrite (C.8) as

$$\begin{aligned} V^0(x,i) = \min_{u}\{&\ell(x,u) + V^0(f(x,u),i+1) \mid \\ &(x,u) \in \mathbb{Z}, f(x,u) \in X(i+1)\} \end{aligned} \tag{C.9}$$

for all $x \in X(i)$ where

$$X(i) = \{x \in \mathbb{R}^n \mid \exists u \text{ such that } (x,u) \in \mathbb{Z} \text{ and } f(x,u) \in X(i+1)\} \tag{C.10}$$

Equations (C.9) and (C.10), together with the boundary condition

$$V^0(x,N) = V_f(x)\ \forall x \in X(N), \quad X(N) = \mathbb{X}_f$$

constitute the DP recursion for constrained discrete time optimal control problems. If there are no state constraints, i.e., if $\mathbb{Z} = \mathbb{R}^n \times \mathbb{U}$ where

$\mathbb{U} \subset \mathbb{R}^m$ is compact, then $X(i) = \mathbb{R}^n$ for all $i \in \{0, 1, \ldots, N\}$ and the DP equations revert to the familiar DP recursion:

$$V^0(x, i) = \min_u \{\ell(x, u) + V^0(f(x, u), i + 1)\} \ \forall x \in \mathbb{R}^n$$

with boundary condition

$$V^0(x, N) = V_f \ \forall x \in \mathbb{R}^n$$

We now prove some basic facts; the first is the well known *principle of optimality*.

Lemma C.1 (Principle of optimality)**.** *Let $x \in X_N$ be arbitrary, let $\mathbf{u} := (u(0), u(1), \ldots, u(N-1)) \in \mathcal{U}(x, 0)$ denote the solution of $\mathbb{P}(x, 0)$ and let $(x, x(1), x(2), \ldots, x(N))$ denote the corresponding optimal state trajectory so that for each i, $x(i) = \phi(i; (x, 0), \mathbf{u})$. Then, for any $i \in \{0, 1, \ldots, N-1\}$, the control sequence $\mathbf{u}^i := (u(i), u(i+1), \ldots, u(N-1))$ is optimal for $\mathbb{P}(x(i), i)$ (any portion of an optimal trajectory is optimal).*

Proof. Since $\mathbf{u} \in \mathcal{U}(x, 0)$, the control sequence $\mathbf{u}^i \in \mathcal{U}(x(i), i)$. If $\mathbf{u}^i = (u(i), u(i+1), \ldots, u(N-1))$ is not optimal for $\mathbb{P}(x(i), i)$, there exists a control sequence $\mathbf{u}' = (u'(i), u'(i+1), \ldots, u(N-1)') \in \mathcal{U}(x(i), i)$ such that $V(x(i), i, \mathbf{u}') < V(x(i), \mathbf{u})$. Consider now the control sequence $\tilde{\mathbf{u}} := (u(0), u(1), \ldots, u(i-1), u'(i), u'(i+1), \ldots, u(N-1)')$. It follows that $\tilde{\mathbf{u}} \in \mathcal{U}(x, 0)$ and $V(x, 0, \tilde{\mathbf{u}}) < V(x, 0, \mathbf{u}) = V^0(x, 0)$, a contradiction. Hence $\mathbf{u}(x(i), i)$ is optimal for $\mathbb{P}(x(i), i)$. ∎

The most important feature of DP is the fact that the DP recursion yields the optimal value $V^0(x, i)$ and the optimal control $\kappa(x, i) = \arg\min_u \{\ell(x, u) + V^0(f(x, u), i + 1) \mid (x, u) \in \mathbb{Z}, f(x, u) \in X(i + 1)\}$ for each $(x, i) \in X(i) \times \{0, 1, \ldots, N-1\}$.

Theorem C.2 (Optimal value function and control law from DP)**.** *Suppose that the function $\Psi : \mathbb{R}^n \times \{0, 1, \ldots, N\} \to \mathbb{R}$, satisfies, for all $i \in \{1, 2, \ldots, N-1\}$, all $x \in X(i)$, the DP recursion*

$$\Psi(x, i) = \min\{\ell(x, u) + \Psi(f(x, u), i + 1) \mid (x, u) \in \mathbb{Z}, f(x, u) \in X(i + 1)\}$$
$$X(i) = \{x \in \mathbb{R}^n \mid \exists u \in \mathbb{R}^m \text{ such that } (x, u) \in \mathbb{Z}, \ f(x, u) \in X(i + 1)\}$$

with boundary conditions

$$\Psi(x, N) = V_f(x) \ \forall x \in \mathbb{X}_f, \quad X(N) = \mathbb{X}_f$$

Then $\Psi(x, i) = V^0(x, i)$ for all $(x, i) \in X(i) \times \{0, 1, 2, \ldots, N\}$; the DP recursion yields the optimal value function and the optimal control law.

Proof. Let $(x,i) \in X(i) \times \{0,1,\ldots,N\}$ be arbitrary. Let $\mathbf{u} = (u(i), u(i+1), \ldots, u(N-1))$ be an arbitrary control sequence in $\mathcal{U}(x, i)$ and let $\mathbf{x} = (x, x(i+1), \ldots, x(N))$ denote the corresponding trajectory starting at (x,i) so that for each $j \in \{i, i+1, \ldots, N\}$, $x(j) = \phi(j; x, i, \mathbf{u})$. For each $j \in \{i, i+1, \ldots, N-1\}$, let $\mathbf{u}^j := (u(j), u(j+1), \ldots, u(N-1))$; clearly $\mathbf{u}^j \in \mathcal{U}(x(j), j)$. The cost due to initial event $(x(j), j)$ and control sequence $\mathbf{u}^j$ is $\Phi(x(j), j)$ defined by

$$\Phi(x(j), j) := V(x(j), j, \mathbf{u}^j)$$

Showing that $\Psi(x,i) \le \Phi(x,i)$ proves that $\Psi(x,i) = V^0(x,i)$ since $\mathbf{u}$ is an arbitrary sequence in $\mathcal{U}(x,i)$; because $(x,i) \in X(i) \times \{0,1,\ldots,N\}$ is arbitrary, that fact that $\Psi(x,i) = V^0(x,i)$ proves that DP yields the optimal value function.

To prove that $\Psi(x,i) \le \Phi(x,i)$, we compare $\Psi(x(j), j)$ and $\Phi(x(j), j)$ for each $j \in \{i, i+1, \ldots, N\}$, i.e., we compare the costs yielded by the DP recursion and by the arbitrary control $\mathbf{u}$ along the corresponding trajectory $\mathbf{x}$. By definition, $\Psi(x(j), j)$ satisfies for each j

$$\Psi(x(j), j) = \min_u \{\ell(x(j), u) + \Psi(f(x(j), u), j+1) \mid (x(j), u) \in \mathbb{Z}, f(x(j), u) \in X(j+1)\} \quad \text{(C.11)}$$

To obtain $\Phi(x(j), j)$ for each j we solve the following recursive equation

$$\Phi(x(j), j) = \ell(x(j), u(j)) + \Phi(f(x(j), u(j)), j+1) \quad \text{(C.12)}$$

The boundary conditions are

$$\Psi(x(N), N) = \Phi(x(N), N) = V_f(x(N)) \quad \text{(C.13)}$$

Since $u(j)$ satisfies $(x(j), u(j)) \in \mathbb{Z}$ and $f(x(j), u(j)) \in X(j+1)$ but is not necessarily a minimizer in (C.11), we deduce that

$$\Psi(x(j), j) \le \ell(x(j), u(j)) + \Psi(f(x(j), u(j)), j+1) \quad \text{(C.14)}$$

For each j, let $E(j)$ be defined by

$$E(j) := \Psi(x(j), j) - \Phi(x(j), j)$$

Subtracting (C.12) from (C.14) and replacing $f(x(j), u(j))$ by $x(j+1)$ yields

$$E(j) \le E(j+1) \ \forall j \in \{i, i+1, \ldots N\}$$

Since $E(N) = 0$ by virtue of (C.13), we deduce that $E(j) \leq 0$ for all $j \in \{i, i+1, \ldots, N\}$; in particular, $E(i) \leq 0$ so that

$$\Psi(x, i) \leq \Phi(x, i) = V(x, i, \mathbf{u})$$

for all $\mathbf{u} \in \mathcal{U}(x, i)$. Hence $\Psi(x, i) = V^0(x, i)$ for all $(x, i) \in X(i) \times \{0, 1, \ldots, N\}$. ■

Example C.3: DP applied to linear quadratic regulator

A much used example is the familiar linear quadratic regulator problem. The system is defined by

$$x^+ = Ax + Bu$$

There are no constraints. The cost function is defined by (C.2) where

$$\ell(x, u) := (1/2)x'Qx + (1/2)u'Ru$$

and $V_f(x) = 0$ for all x; the horizon length is N. We assume that Q is symmetric and positive semidefinite and that R is symmetric and positive definite. The DP recursion is

$$V^0(x, i) = \min_u \{\ell(x, u) + V^0(Ax + Bu, i + 1)\} \ \forall x \in \mathbb{R}^n$$

with terminal condition

$$V^0(x, N) = 0 \ \forall x \in \mathbb{R}^n$$

Assume that $V^0(\cdot, i + 1)$ is quadratic and positive semidefinite and, therefore, has the form

$$V^0(x, i + 1) = (1/2)x'P(i + 1)x$$

where $P(i + 1)$ is symmetric and positive semidefinite. Then

$$V^0(x, i) = (1/2) \min_u \{x'Qx + u'Ru + (Ax + Bu)'P(i + 1)(Ax + Bu)\}$$

The right-hand side of the last equation is a positive definite function of u for all x, so that it has a unique minimizer given by

$$\kappa(x, i) = K(i)x \qquad K(i) := -(B'P(i + 1)B + R)^{-1}B'P(i + 1)$$

Substituting $u = K(i)x$ in the expression for $V^0(x, i)$ yields

$$V^0(x, i) = (1/2)x'P(i)x$$

where $P(i)$ is given by:

$$P(i) = Q + K(i)'RK(i) - A'P(i+1)B(B'P(i+1)B + R)^{-1}B'P(i+1)A$$

Hence $V^0(\cdot, i)$ is quadratic and positive semidefinite if $V^0(\cdot, i+1)$ is. But $V^0(\cdot, N)$, defined by

$$V^0(x, N) := (1/2)x'P(N)x = 0 \qquad P(N) := 0$$

is symmetric and positive semidefinite. By induction $V^0(\cdot, i)$ is quadratic and positive semidefinite (and $P(i)$ is symmetric and positive semidefinite) for all $i \in \{0, 1, \ldots, N\}$. Substituting $K(i) = -(B'P(i+1)B + R)^{-1}B'P(i+1)A$ in the expression for $P(i)$ yields the more familiar matrix Riccati equation

$$P(i) = Q + A'P(i+1)A - A'P(i+1)B(B'P(i+1)B + R)^{-1}BP(i+1)A$$

□

C.2 Optimality Conditions

In this section we obtain optimality conditions for problems of the form

$$f^0 = \inf_u \{f(u) \mid u \in U\}$$

In these problems, $u \in \mathbb{R}^m$ is the *decision* variable, $f(u)$ the cost to be minimized by appropriate choice of u and $U \subset \mathbb{R}^m$ the constraint set. The value of the problem is f^0. Some readers may wish to read only Section C.2.2, which deals with convex optimization problems and Section C.2.3 which deals with convex optimization problems in which the constraint set U is polyhedral. These sections require some knowledge of tangent and normal cones discussed in Section C.2.1; Proposition C.7 in particular derives the normal cone for the case when U is convex.

C.2.1 Tangent and Normal Cones

In determining conditions of optimality, it is often convenient to employ approximations to the cost function $f(\cdot)$ and the constraint set U. Thus the cost function $f(\cdot)$ may be approximated, in the neighborhood of a point $\bar{u}$, by the first order expansion $f(\bar{u}) + \langle \nabla f(\bar{u}), (u - \bar{u}) \rangle$ or by the second order expansion $f(\bar{u}) + \langle \nabla f(\bar{u}), (u - \bar{u}) \rangle + (1/2)((u - \bar{u})'\nabla^2 f(\bar{x})(u - \bar{u}))$ if the necessary derivatives exist. Thus we see that

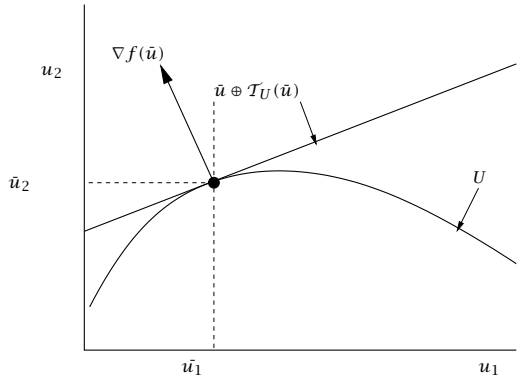


Figure C.2: Approximation of the set U.

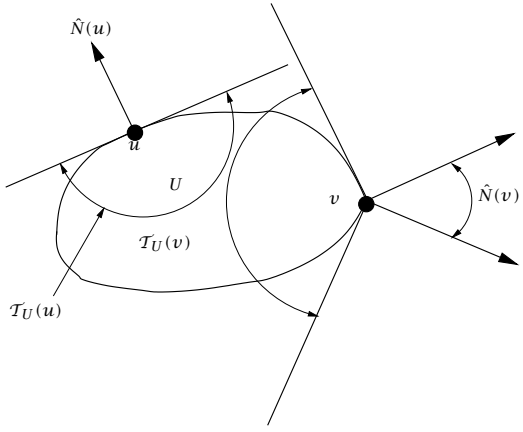


Figure C.3: Tangent cones.

in the unconstrained case, a necessary condition for the optimality of $\bar{u}$ is $\nabla f(\bar{u}) = 0$. To obtain necessary conditions of optimality for constrained optimization problems, we need to approximate the constraint set as well; this is more difficult. An example of U and its approximation is shown in Figure C.2; here the set $U = \{u \in \mathbb{R}^2 \mid g(u) = 0\}$ where $g : \mathbb{R} \to \mathbb{R}$ is approximated in the neighborhood of a point $\bar{u}$ satisfying $g(\bar{u}) = 0$ by the set $\bar{u} \oplus \mathcal{T}_U(\bar{u})$ where[2] the tangent cone $\mathcal{T}_U(\bar{u}) := \{h \in \mathbb{R}^2 \mid \nabla g(\bar{u}), u - \bar{u}\rangle = 0\}$. In general, a set U is approx-

[2]If A and B are two subsets of $\mathbb{R}^n$, say, then $A \oplus B := \{a + b \mid a \in A, b \in B\}$ and $a \oplus B := \{a + b \mid b \in B\}$.

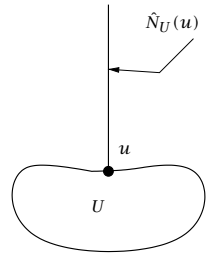


Figure C.4: Normal at u.

imated, near a point $\bar{u}$, by $\bar{u} \oplus \mathcal{T}_U(\bar{u})$ where its *tangent cone* $\mathcal{T}_U(\bar{u})$ is defined below. Following Rockafellar and Wets (1998), we use $u \xrightarrow[U]{\nu} v$ to denote that the sequence $\{u^\nu \mid \nu \in \mathbb{I}_{\geq 0}\}$ converges to v as $\nu \to \infty$ while satisfying $u^\nu \in U$ for all $\nu \in \mathbb{I}_{\geq 0}$.

Definition C.4 (Tangent vector)**.** A vector $h \in \mathbb{R}^m$ is tangent to the set U at $\bar{u}$ if there exist sequences $u \xrightarrow[U]{\nu} \bar{u}$ and $\lambda^\nu \searrow 0$ such that

$$[u^\nu - \bar{u}]/\lambda^\nu \to h$$

$\mathcal{T}_U(u)$ is the set of all tangent vectors.

Equivalently, a vector $h \in \mathbb{R}^m$ is tangent to the set U at $\bar{u}$ if there exist sequences $h^\nu \to h$ and $\lambda^\nu \searrow 0$ such that $\bar{u} + \lambda^\nu h^\nu \in U$ for all $\nu \in \mathbb{I}_{\geq 0}$. This equivalence can be seen by identifying u^ν with $\bar{u} + \lambda^\nu h^\nu$.

Proposition C.5 (Tangent vectors are closed cone)**.** *The set $\mathcal{T}_U(u)$ of all tangent vectors to U at any point $u \in U$ is a closed cone.*

See Rockafellar and Wets (1998), Proposition 6.2. That $\mathcal{T}_U(\bar{u})$ is a cone may be seen from its definition; if h is a tangent, so is αh for any $\alpha \geq 0$. Two examples of a tangent cone are illustrated in Figure C.3.

Associated with each tangent cone $\mathcal{T}_U(u)$ is a normal cone $\hat{N}(u)$ defined as follows Rockafellar and Wets (1998):

Definition C.6 (Regular normal)**.** A vector $g \in \mathbb{R}^m$ is a regular normal to a set $U \subset \mathbb{R}^m$ at $\bar{u} \in U$ if

$$\langle g, u - \bar{u} \rangle \leq o(|u - \bar{u}|) \ \forall u \in U \tag{C.15}$$

where $o(\cdot)$ has the property that $o(|u - \bar{u}|)/|u - \bar{u}| \to 0$ as $u \xrightarrow[U]{} \bar{u}$ with $u \neq \bar{u}$; $\hat{N}_U(u)$ is the set of all regular normal vectors.

Some examples of normal cones are illustrated in Figure C.3; here the set $\hat{N}_U(u) = \{\lambda g \mid \lambda \geq 0\}$ is a cone generated by a single vector g, say, while $\hat{N}_U(v) = \{\lambda_1 g_1 + \lambda_2 g_2 \mid \lambda_1 \geq 0, \lambda_2 \geq 0\}$ is a cone generated by two vectors g_1 and g_2, say. The term $o(|u - \bar{u}|)$ may be replaced by 0 if U is convex as shown in Proposition C.7(b) below but is needed in general since U may not be locally convex at $\bar{u}$ as illustrated in Figure C.4.

The tangent cone $\mathcal{T}_U(\bar{u})$ and the normal cone $\hat{N}_U(\bar{u})$ at a point $\bar{u} \in U$ are related as follows.

Proposition C.7 (Relation of normal and tangent cones)**.**

(a) At any point $\bar{u} \in U \subset \mathbb{R}^m$,

$$\hat{N}_U(\bar{u}) = \mathcal{T}_U(\bar{u})^* := \{g \mid \langle g, h\rangle \leq 0 \ \forall h \in \mathcal{T}_U(\bar{u})\}$$

where, for any cone V, $V^* := \{g \mid \langle g, h\rangle \leq 0 \ \forall h \in V\}$ denotes the polar cone of V.

(b) If U is convex, then, at any point $\bar{u} \in U$

$$\hat{N}_U(\bar{u}) = \{g \mid \langle g, u - \bar{u}\rangle \leq 0 \ \forall u \in U\} \tag{C.16}$$

Proof.

(a) To prove $\hat{N}_U(\bar{u}) \subset \mathcal{T}_U(\bar{u})^*$, we take an arbitrary point g in $\hat{N}_U(\bar{u})$ and show that $\langle g, h\rangle \leq 0$ for all $h \in \mathcal{T}(\bar{u})$ implying that $g \in \mathcal{T}_U^*(\bar{u})$. For, if h is tangent to U at $\bar{u}$, there exist, by definition, sequences $u^\nu \underset{U}{\longrightarrow} \bar{u}$ and $\lambda^\nu \searrow 0$ such that

$$h^\nu := (u^\nu - \bar{u})/\lambda^\nu \to h$$

Since $g \in \hat{N}_U(\bar{u})$, it follows from (C.15) that $\langle g, h^\nu\rangle \leq o(|(u^\nu - \bar{u})|) = o(\lambda^\nu |h^\nu|)$; the limit as $\nu \to \infty$ yields $\langle g, h\rangle \leq 0$, so that $g \in \mathcal{T}_U^*(\bar{u})$. Hence $\hat{N}_U(\bar{u}) \subset \mathcal{T}_U(\bar{u})^*$. The proof of this result, and the more subtle proof of the converse, that $\mathcal{T}_U(\bar{u})^* \subset \hat{N}_U(\bar{u})$, are given in Rockafellar and Wets (1998), Proposition 6.5.

(b) This part of the proposition is proved in (Rockafellar and Wets, 1998, Theorem 6.9). ∎

Remark. A consequence of (C.16) is that for each $g \in \hat{N}_U(\bar{u})$, the half-space $H_g := \{u \mid \langle g, u - \bar{u}\rangle \leq 0\}$ supports the convex set U at $\bar{u}$, i.e., $U \subset H_g$ and $\bar{u}$ lies on the boundary of the half-space H_g.

We wish to derive optimality conditions for problems of the form $\mathbb{P} : \inf_u \{f(u) \mid u \in U\}$. The *value* of the problem is defined to be

$$f^0 := \inf_u \{f(u) \mid u \in U\}$$

There may not exist a $u \in U$ such that $f(u) = f^0$. If, however, $f(\cdot)$ is continuous and U is compact, there exists a minimizing u in U, i.e.,

$$f^0 = \inf_u \{f(u) \mid u \in U\} = \min_u \{f(u) \mid u \in U\}$$

The minimizing u, if it exists, may not be unique so

$$u^0 := \arg\min_u \{f(u) \mid u \in U\}$$

may be a set. We say u is feasible if $u \in U$. A point u is *globally optimal* for problem $\mathbb{P}$ if u is feasible and $f(v) \geq f(u)$ for all $v \in U$. A point u is *locally optimal* for problem $\mathbb{P}$ if u is feasible and there exists a $\varepsilon > 0$ such that $f(v) \geq f(u)$ for all v in $(u \oplus \varepsilon\mathcal{B}) \cap U$ where $\mathcal{B}$ is the closed unit ball $\{u \mid \min |u| \leq 1\}$.

C.2.2 Convex Optimization Problems

The optimization problem $\mathbb{P}$ is convex if the function $f : \mathbb{R}^m \to \mathbb{R}$ and the set $U \subset \mathbb{R}^m$ are convex. In convex optimization problems, U often takes the form $\{u \mid g_j(u) \leq 0, j \in \mathcal{J}\}$ where $\mathcal{J} := \{1, 2, \ldots, J\}$ and each function $g_j(\cdot)$ is convex. A useful feature of convex optimization problems is the following result:

Proposition C.8 (Global optimality for convex problems)**.** *Suppose the function $f(\cdot)$ is convex and differentiable and the set U is convex. Any locally optimal point of the convex optimization problem* $\inf_u \{f(u) \mid u \in U\}$ *is globally optimal.*

Proof. Suppose u is locally optimal so that there exists an $\varepsilon > 0$ such that $f(v) \geq f(u)$ for all $v \in (u \oplus \varepsilon\mathcal{B}) \cap U$. If, contrary to what we wish to prove, u is *not* globally optimal, there exists a $w \in U$ such that $f(w) < f(u)$. For any $\lambda \in [0, 1]$, the point $w_\lambda := \lambda w + (1 - \lambda)u$ lies in $[u, w]$ (the line joining u and w). Then $w_\lambda \in U$ (because U is convex) and $f(w_\lambda) \leq \lambda f(w) + (1 - \lambda) f(u) < f(u)$ for all $\lambda \in (0, 1]$ (because $f(\cdot)$ is convex and $f(w) < f(u)$). We can choose $\lambda > 0$ so that $w_\lambda \in (u \oplus \varepsilon\mathcal{B}) \cap U$ and $f(w_\lambda) < f(u)$. This contradicts the local optimality of u. Hence u is globally optimal. ∎

On the assumption that $f(\cdot)$ is differentiable, we can obtain a simple necessary and sufficient condition for the (global) optimality of a point u.

Proposition C.9 (Optimality conditions—normal cone)**.** *Suppose the function $f(\cdot)$ is convex and differentiable and the set U is convex. The point u is optimal for problem $\mathbb{P}$ if and only if $u \in U$ and*

$$df(u; v-u) = \langle \nabla f(u), v-u \rangle \geq 0 \ \forall v \in U \tag{C.17}$$

or, equivalently

$$-\nabla f(u) \in \hat{N}_U(u) \tag{C.18}$$

Proof. Because $f(\cdot)$ is convex, it follows from Theorem 7 in Appendix A1 that

$$f(v) \geq f(u) + \langle \nabla f(u), v-u \rangle \tag{C.19}$$

for all u, v in U. To prove sufficiency, suppose $u \in U$ and that the condition in (C.17) is satisfied. It then follows from (C.19) that $f(v) \geq f(u)$ for all $v \in U$ so that u is globally optimal. To prove necessity, suppose that u is globally optimal but that, contrary to what we wish to prove, the condition on the right-hand side of (C.17) is not satisfied so that there exists a $v \in U$ such that

$$df(u; h) = \langle \nabla f(u), v-u \rangle = -\delta < 0$$

where $h := v-u$. For all $\lambda \in [0,1]$, let $v_\lambda := \lambda v + (1-\lambda)u = u + \lambda h$; because U is convex, each v_λ lies in U. Since

$$df(u; h) = \lim_{\lambda \searrow 0} \frac{f(u+\lambda h) - f(u)}{\lambda} = \lim_{\lambda \searrow 0} \frac{f(v_\lambda) - f(u)}{\lambda} = -\delta$$

there exists a $\lambda \in (0,1]$ such that $f(v_\lambda) - f(u) \leq -\lambda\delta/2 < 0$ which contradicts the optimality of u. Hence the condition in (C.17) must be satisfied. That (C.17) is equivalent to (C.18) follows from Proposition C.7(b). ∎

Remark. The condition (C.17) implies that the linear approximation $\hat{f}(v) := f(u) + \langle \nabla f(u), v-u \rangle$ to $f(v)$ achieves its minimum over U at u.

It is an interesting fact that U in Proposition C.9 may be replaced by its approximation $u \oplus \mathcal{T}_U(u)$ at u yielding

Proposition C.10 (Optimality conditions—tangent cone)**.** *Suppose the function $f(\cdot)$ is convex and differentiable and the set U is convex. The point u is optimal for problem $\mathbb{P}$ if and only if $u \in U$ and*

$$df(u; v-u) = \langle \nabla f(u), h \rangle \geq 0 \ \forall h \in \mathcal{T}_U(u)$$

or, equivalently

$$-\nabla f(u) \in \hat{N}_U(u) = \mathcal{T}_U^*(u).$$

Proof. It follows from Proposition C.9 that u is optimal for problem $\mathbb{P}$ if and only if $u \in U$ and $-\nabla f(u) \in \hat{N}_U(u)$. But, by Proposition C.7, $\hat{N}_U(u) = \{g \mid \langle g, h \rangle \leq 0 \ \forall h \in \mathcal{T}_U(u)\}$ so that $-\nabla f(u) \in \hat{N}_U(u)$ is equivalent to $\langle \nabla f(u), h \rangle \geq 0$ for all $h \in \mathcal{T}_U(u)$. ∎

C.2.3 Convex Problems: Polyhedral Constraint Set

The definitions of tangent and normal cones given above may appear complex but this complexity is necessary for proper treatment of the general case when U is not necessarily convex. When U is polyhedral, i.e., when U is defined by a set of linear inequalities

$$U := \{u \in \mathbb{R}^m \mid Au \leq b\}$$

where $A \in \mathbb{R}^{p \times m}$ and $b \in \mathbb{R}^p$, $\mathcal{I} := \{1, 2, \ldots, p\}$, then the normal and tangent cones are relatively simple. We first note that U is equivalently defined by

$$U := \{u \in \mathbb{R}^m \mid \langle a_i, u \rangle \leq b_i, \ i \in \mathcal{I}\}$$

where a_i is the ith row of A and b_i is the ith element of b. For each $u \in U$, let

$$\mathcal{I}^0(u) := \{i \in \mathcal{I} \mid \langle a_i, u \rangle = b_i\}$$

denote the index set of constraints *active* at u. Clearly $\mathcal{I}^0(u) = \emptyset$ if u lies in the interior of U. An example of a polyhedral constraint set is shown in Figure C.5. The next result shows that in this case, the tangent cone is the set of h in $\mathbb{R}^m$ that satisfy $\langle a_i, h \rangle \leq 0$ for all i in $\mathcal{I}^0(u)$ and the normal cone is the cone generated by the vectors a_i, $i \in \mathcal{I}^0(u)$; each normal h in the normal cone may be expressed as $\sum_{i \in \mathcal{I}^0(u)} \mu_i a_i$ where each $\mu_i \geq 0$.

Proposition C.11 (Representation of tangent and normal cones)**.** *Let $U := \{u \in \mathbb{R}^m \mid \langle a_i, u \rangle \leq b_i, \ i \in \mathcal{I}\}$. Then, for any $u \in U$:*

$$\mathcal{T}_U(u) = \{h \mid \langle a_i, h \rangle \leq 0, \ i \in \mathcal{I}^0(u)\}$$
$$\hat{N}_U(u) = \mathcal{T}_U^*(u) = \text{cone}\{a_i \mid i \in \mathcal{I}^0(u)\}$$

Proof. (i) Suppose h is any vector in $\{h \mid \langle a_i, h\rangle \le 0,\ i \in \mathcal{I}^0(u)\}$. Let the sequences u^ν and λ^ν satisfy $u^\nu = u + \lambda^\nu h$ and $\lambda^\nu \searrow 0$ with λ^0, the first element in the sequence λ^ν, satisfying $u + \lambda^0 h \in U$. It follows that $[u^\nu - u]/\lambda^\nu \equiv h$ so that from Definition C.4, h is tangent to U at u. Hence $\{h \mid \langle a_i, h\rangle \le 0,\ i \in \mathcal{I}^0(u)\} \subset \mathcal{T}_U(u)$. (ii) Conversely, if $h \in \mathcal{T}_U(u)$, then there exist sequences $\lambda^\nu \searrow 0$ and $h^\nu \to h$ such that $\langle a_i, u + \lambda^\nu h^\nu\rangle \le b_i$ for all $i \in \mathcal{I}$, all $\nu \in \mathbb{I}_{\ge 0}$. Since $\langle a_i, u\rangle = b_i$ for all $i \in \mathcal{I}^0(u)$, it follows that $\langle a_i, h^\nu\rangle \le 0$ for all $i \in \mathcal{I}^0(u)$, all $\nu \in \mathbb{I}_{\ge 0}$; taking the limit yields $\langle a_i, h\rangle \le 0$ for all $i \in \mathcal{I}^0(u)$ so that $h \in \{h \mid \langle a_i, h\rangle \le 0,\ i \in \mathcal{I}^0(u)\}$ which proves $\mathcal{T}_U(u) \subset \{h \mid \langle a_i, h\rangle \le 0,\ i \in \mathcal{I}^0(u)\}$. We conclude from (i) and (ii) that $\mathcal{T}_U(u) = \{h \mid \langle a_i, h\rangle \le 0,\ i \in \mathcal{I}^0(u)\}$. That $\hat{N}_U(u) = \mathcal{T}_U^*(u) = \text{cone}\{a_i \mid i \in \mathcal{I}^0(u)\}$ then follows from Proposition C.7 above and Proposition 9 in Appendix A1. ∎

The next result follows from Proposition C.5 and Proposition C.7.

Proposition C.12 (Optimality conditions—linear inequalities)**.** *Suppose the function $f(\cdot)$ is convex and differentiable and U is the convex set $\{u \mid Au \le b\}$. Then u is optimal for $\mathbb{P} : \min_u\{f(u) \mid u \in U\}$ if and only if $u \in U$ and*

$$-\nabla f(u) \in \hat{N}_U(u) = \text{cone}\{a_i \mid i \in \mathcal{I}^0(u)\}$$

Corollary C.13 (Optimality conditions—linear inequalities)**.** *Suppose the function $f(\cdot)$ is convex and differentiable and $U = \{u \mid Au \le b\}$. Then u is optimal for $\mathbb{P} : \min_u\{f(u) \mid u \in U\}$ if and only if $Au \le b$ and there exist multipliers $\mu_i \ge 0, i \in \mathcal{I}^0(u)$ satisfying*

$$\nabla f(u) + \sum_{i \in \mathcal{I}^0(u)} \mu_i \nabla g_i(u) = 0 \tag{C.20}$$

where, for each i, $g_i(u) := \langle a_i, u\rangle - b_i$ so that $g_i(u) \le 0$ is the constraint $\langle a_i, u\rangle \le b_i$ and $\nabla g_i(u) = a_i$.

Proof. Since any point $g \in \text{cone}\{a_i \mid i \in \mathcal{I}^0(u)\}$ may be expressed as $g = \sum_{i \in \mathcal{I}^0(u)} \mu_i a_i$ where, for each i, $\mu_i \ge 0$, the condition $-\nabla f(u) \in \text{cone}\{a_i \mid i \in \mathcal{I}^0(u)\}$ is equivalent to the existence of multipliers $\mu_i \ge 0$, $i \in \mathcal{I}^0(u)$ satisfying (C.20). ∎

The above results are easily extended if U is defined by linear equality and inequality constraints, i.e., if

$$U := \{\langle a_i, u\rangle \le b_i,\ i \in \mathcal{I},\ \langle c_i, u\rangle = d_i,\ i \in \mathcal{E}\}$$

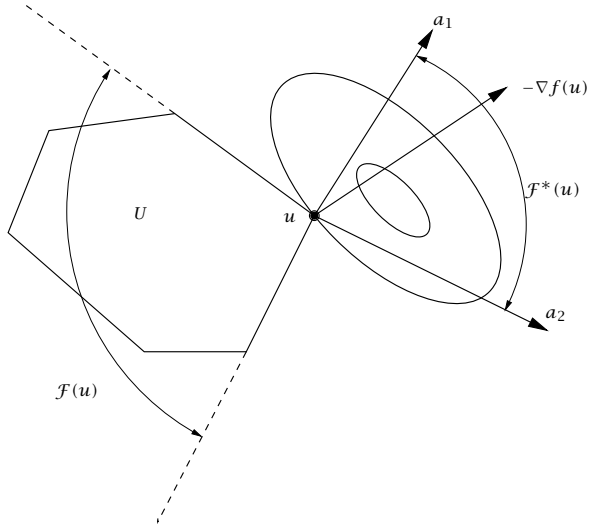


Figure C.5: Condition of optimality.

In this case, at any point $u \in U$, the tangent cone is

$$\mathcal{T}_U(u) = \{h \mid \langle a_i, h\rangle \leq 0,\ i \in \mathcal{I}^0(u),\ \langle c_i, h\rangle = 0,\ i \in \mathcal{E}\}$$

and the normal cone is

$$\hat{N}_U(u) = \{\sum_{i\in\mathcal{I}^0(u)} \lambda_i a_i + \sum_{i\in\mathcal{E}} \mu_i c_i \mid \lambda_i \geq 0\ \forall i \in \mathcal{I}^0(u), \mu_i \in \mathbb{R}\ \forall i \in \mathcal{E}\}$$

With U defined this way, u is optimal for $\min_u\{f(u) \mid u \in U\}$ where $f(\cdot)$ is convex and differentiable if and only if

$$-\nabla f(u) \in \hat{N}_U(u)$$

For each $i \in \mathcal{I}$ let $g_i(u) := \langle a_i, u\rangle - b_i$ and for each $i \in \mathcal{E}$, let $h_i(u) := \langle c_i, u\rangle - d_i$ so that $\nabla g(u_i) = a_i$ and $\nabla h_i = c_i$. It follows from the characterization of $\hat{N}_U(u)$ that u is optimal for $\min_u\{f(u) \mid u \in U\}$ if and only if there exist multipliers $\lambda_i \geq 0,\ i \in \mathcal{I}^0(u)$ and $\mu_i \in \mathbb{R},\ i \in \mathcal{E}$ such that

$$\nabla f(u) + \sum_{i\in\mathcal{I}^0(u)} \mu_i \nabla g_i(u) + \sum_{i\in\mathcal{E}} h_i(u) = 0 \tag{C.21}$$

C.2.4 Nonconvex Problems

We first obtain a necessary condition of optimality for the problem $\min\{f(u) \mid u \in U\}$ where $f(\cdot)$ is differentiable but not necessarily

convex and $U \subset \mathbb{R}^m$ is not necessarily convex; this result generalizes the necessary condition of optimality in Proposition C.9.

Proposition C.14 (Necessary condition for nonconvex problem)**.** *A necessary condition for u to be locally optimal for the problem of minimizing a differentiable function $f(\cdot)$ over the set U is*

$$df(u;h) = \langle \nabla f(u), h \rangle \geq 0, \ \forall h \in \mathcal{T}_U(u)$$

which is equivalent to the condition

$$-\nabla f(u) \in \hat{N}_U(u)$$

Proof. Suppose, contrary to what we wish to prove, that there exists a $h \in \mathcal{T}_U(u)$ and a $\delta > 0$ such that $\langle \nabla f(u), h \rangle = -\delta < 0$. Because $h \in \mathcal{T}_U(u)$, there exist sequences $h^\nu \underset{U}{\longrightarrow} h$ and $\lambda^\nu \searrow 0$ such that $u^\nu := u + \lambda^\nu h^\nu$ converges to u and satisfies $u^\nu \in U$ for all $\nu \in \mathbb{I}_{\geq 0}$. Then

$$f(u^\nu) - f(u) = \langle \nabla f(u), \lambda^\nu h^\nu \rangle + o(\lambda^\nu |h^\nu|)$$

Hence

$$[f(u^\nu) - f(u)]/\lambda^\nu = \langle \nabla f(u), h^\nu \rangle + o(\lambda^\nu)/\lambda^\nu$$

where we make use of the fact that $|h^\nu|$ is bounded for ν sufficiently large. It follows that

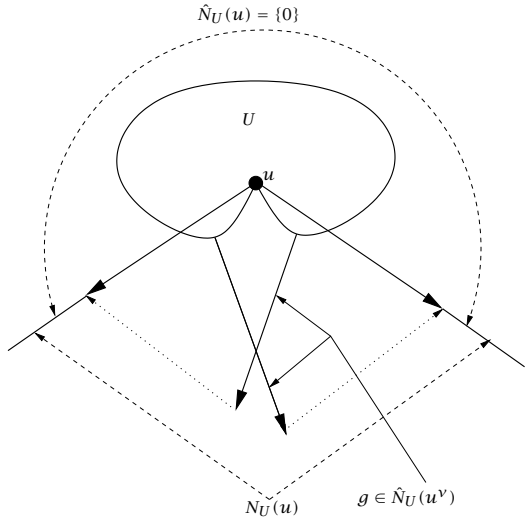
$$[f(u^\nu) - f(u)]/\lambda^\nu \to \langle \nabla f(u), h \rangle = -\delta$$

so that there exists a finite integer j such that $f(u^j) - f(u) \leq -\lambda^j \delta/2 < 0$ which contradicts the local optimality of u. Hence $\langle \nabla f(u), h \rangle \geq 0$ for all $h \in \mathcal{T}_U(u)$. That $-\nabla f(u) \in \hat{N}_U(u)$ follows from Proposition C.7. ∎

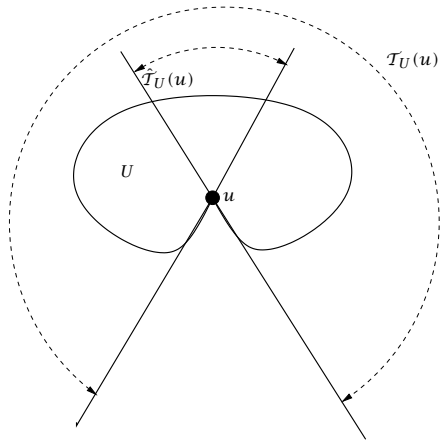
A more concise proof proceeds as follows Rockafellar and Wets (1998). Since $f(v) - f(u) = \langle \nabla f(u), v - u \rangle + o(|v - u|)$ it follows that $\langle -\nabla f(u), v - u \rangle = o(|v - u|) - (f(v) - f(u))$. Because u is locally optimal, $f(v) - f(u) \geq 0$ for all v in the neighborhood of u so that $\langle -\nabla f(u), v - u \rangle \leq o(|v - u|)$ which, by (C.15), is the definition of a normal vector. Hence $-\nabla f(u) \in \hat{N}_U(u)$.

C.2.5 Tangent and Normal Cones

The material in this section is *not* required for Chapters 1-7; it is presented merely to show that alternative definitions of tangent and normal cones are useful in more complex situations than those considered



(a) Normal cones.



(b) Tangent cones.

Figure C.6: Tangent and normal cones.

above. Thus, the normal and tangent cones defined in C.2.1 have some limitations when U is not convex or, at least, not similar to the constraint set illustrated in Figure C.4. Figure C.6 illustrates the type of difficulty that may occur. Here the tangent cone $\mathcal{T}_U(u)$ is not convex, as shown in Figure C.6(b), so that the associated normal cone

$\hat{N}_U(u) = \mathcal{T}_U(u)^* = \{0\}$. Hence the necessary condition of optimality of u for the problem of minimizing a differentiable function $f(\cdot)$ over U is $\nabla f(u) = 0$; the only way a *differentiable* function $f(\cdot)$ can achieve a minimum over U at u is for the condition $\nabla f(u) = 0$ to be satisfied. Alternative definitions of normality and tangency are sometimes necessary. In Rockafellar and Wets (1998), a vector $g \in \hat{N}_U(u)$ is normal in the *regular* sense; a normal in the *general* sense is then defined by:

Definition C.15 (General normal)**.** A vector g is normal to U at u in the general sense if there exist sequences $u^\nu \xrightarrow[U]{} u$ and $g^\nu \to g$ where $g^\nu \in \hat{N}_U(u^\nu)$ for all ν; $N_U(u)$ is the set of all general normal vectors.

The cone $N_U(u)$ of general normal vectors is illustrated in Figure C.6(a); here the cone $N_U(u)$ is the union of two distinct cones each having form $\{\alpha g \mid \alpha \geq 0\}$. Also shown in Figure C.6(a) are single elements of two sequences g^ν in $\hat{N}_U(u^\nu)$ converging to $N_U(u)$. Counter intuitively, the general normal vectors in this case point into the interior of U. Associated with $N_U(u)$ is the set $\hat{\mathcal{T}}_U(u)$ of regular tangents to U at u defined, when U is locally closed,[3] in (Rockafellar and Wets, 1998, Theorem 6.26) by:

Definition C.16 (General tangent)**.** Suppose U is locally closed at u. A vector h is tangent to U at u in the regular sense if, for all sequences $u^\nu \xrightarrow[U]{} u$, there exists a sequence $h^\nu \to h$ that satisfies $h^\nu \in \mathcal{T}_u(u^\nu)$ for all ν; $\hat{\mathcal{T}}_U(u)$ is the set of all regular tangent vectors to U at u.

Alternatively, a vector h is tangent to U at u in the regular sense if, for all sequences $u^\nu \xrightarrow[U]{} u$ and $\lambda^\nu \searrow 0$, there exists a sequence $h^\nu \to h$ satisfying $u^\nu + \lambda^\nu h^\nu \in U$ for all $\nu \in \mathbb{I}_{\geq 0}$. The cone of regular tangent vectors for the example immediately above is shown in Figure C.6(b). The following result is proved in Rockafellar and Wets (1998), Theorem 6.26:

Proposition C.17 (Set of regular tangents is closed convex cone)**.** *At any $u \in U$, the set $\hat{\mathcal{T}}_U(u)$ of regular tangents to U at u is a closed convex cone with $\hat{\mathcal{T}}_U(u) \subset \mathcal{T}_U(u)$. Moreover, if U is locally closed at u, then $\hat{\mathcal{T}}_U(u) = N_U(u)^*$.*

[3] A set U is locally closed at a point u if there exists a closed neighborhood $\mathcal{N}$ of u such that $U \cap \mathcal{N}$ is closed; U is locally closed if it is locally closed at all u.

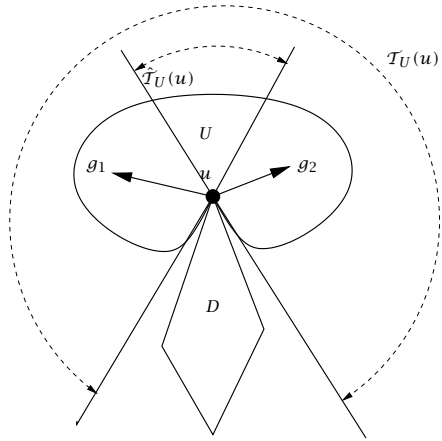


Figure C.7: Condition of optimality.

Figure C.7 illustrates some of these results. In Figure C.7, the constant cost contour $\{v \mid f(v) = f(u)\}$ of a *nondifferentiable* cost function $f(\cdot)$ is shown together with a sublevel set D passing through the point u: $f(v) \leq f(u)$ for all $v \in D$. For this example, $df(u;h) = \max\{\langle g_1, h\rangle, \langle g_2, h\rangle\}$ where g_1 and g_2 are normals to the level set of $f(\cdot)$ at u so that $df(u;h) \geq 0$ for all $h \in \hat{\mathcal{T}}_U(u)$, a necessary condition of optimality; on the other hand, there exist $h \in \mathcal{T}_U(u)$ such that $df(u;h) < 0$. The situation is simpler if the constraint set U is *regular* at u.

Definition C.18 (Regular set)**.** A set U is regular at a point $u \in U$ in the sense of Clarke if it is locally closed at u and if $N_U(u) = \hat{N}_U(u)$ (all normal vectors at u are regular).

The following consequences of Clarke regularity are established in Rockafellar and Wets (1998), Corollary 6.29:

Proposition C.19 (Conditions for regular set)**.** *Suppose U is locally closed at $u \in U$. Then U is regular at u is equivalent to each of the following.*

(a) $N_U(u) = \hat{N}_U(u)$ *(all normal vectors at u are regular).*

(b) $\mathcal{T}_U(u) = \hat{\mathcal{T}}_U(u)$ *(all tangent vectors at u are regular).*

(c) $N_U(u) = \mathcal{T}_U(u)^*$.

(d) $\mathcal{T}_U(u) = N_U(u)^*$.

(e) $\langle g, h\rangle \leq 0$ *for all* $h \in \mathcal{T}_U(u)$, *all* $g \in N_U(u)$.

It is shown in Rockafellar and Wets (1998) that if U is regular at u and a constraint qualification is satisfied, then a necessary condition of optimality, similar to (C.21), may be obtained. To obtain this result, we pursue a slightly different route in Sections C.2.6 and C.2.7.

C.2.6 Constraint Set Defined by Inequalities

We now consider the case when the set U is specified by a set of differentiable inequalities:

$$U := \{u \mid g_i(u) \leq 0 \ \forall i \in \mathcal{I}\} \tag{C.22}$$

where, for each $i \in \mathcal{I}$, the function $g_i : \mathbb{R}^m \to \mathbb{R}$ is differentiable. For each $u \in U$

$$\mathcal{I}^0(u) := \{i \in \mathcal{I} \mid g_i(u) = 0\}$$

is the index set of active constraints. For each $u \in U$, the set $\mathcal{F}_U(u)$ of feasible variations for the *linearized* set of inequalities; $\mathcal{F}_U(u)$ is defined by

$$\mathcal{F}_U(u) := \{h \mid \langle \nabla g_i(u), h\rangle \leq 0 \ \forall i \in \mathcal{I}^0(u)\} \tag{C.23}$$

The set $\mathcal{F}_U(u)$ is a closed, convex cone and is called a cone of first order feasible variations in Bertsekas (1999) because h is a descent direction for $g_i(u)$ for all $i \in \mathcal{I}^0(u)$, i.e., $g_i(u + \lambda h) \leq 0$ for all λ sufficiently small. When U is polyhedral, the case discussed in C.2.3, $g_i(u) = \langle a_i, u\rangle - b_i$ and $\nabla g_i(u) = a_i$ so that $\mathcal{F}_U(u) = \{h \mid \langle a_i, h\rangle \leq 0 \ \forall i \in \mathcal{I}^0(u)\}$ which was shown in Proposition C.11 to be the tangent cone $\mathcal{T}_U(u)$. An important question whether $\mathcal{F}_U(u)$ is the tangent cone $\mathcal{T}_U(u)$ for a wider class of problems because, if $\mathcal{F}_U(u) = \mathcal{T}_U(u)$, a condition of optimality of the form in (C.20) may be obtained. In the example in Figure C.8, $\mathcal{F}_U(u)$ is the horizontal axis $\{h \in \mathbb{R}^2 \mid h_2 = 0\}$ whereas $\mathcal{T}_U(u)$ is the half-line $\{h \in \mathbb{R}^2 \mid h_1 \geq 0,\ h_2 = 0\}$ so that in this case, $\mathcal{F}_U(u) \neq \mathcal{T}_U(u)$. While $\mathcal{F}_U(u)$ is always convex, being the intersection of a set of half-spaces, the tangent cone $\mathcal{T}_U(u)$ is not necessarily convex as Figure C.6b shows. The set U is said to be *quasiregular* at $u \in U$ if $\mathcal{F}_U(u) = \mathcal{T}_U(u)$ is which case u is said to be a quasiregular point Bertsekas (1999). The next result, due to Bertsekas (1999), shows that $\mathcal{F}_U(u) = \mathcal{T}_U(u)$, i.e., U is quasiregular at u, when a certain constraint qualification is satisfied.

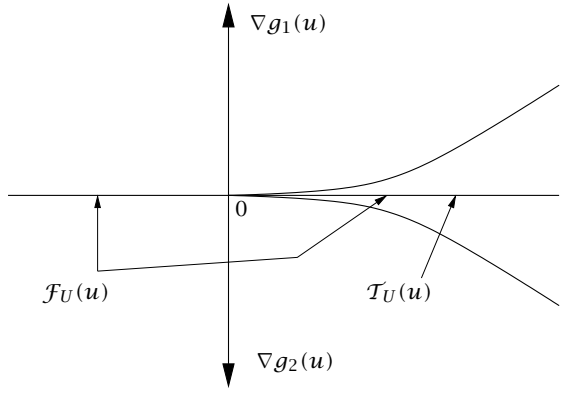


Figure C.8: $\mathcal{F}_U(u) \neq \mathcal{T}_U(u)$.

Proposition C.20 (Quasiregular set)**.** *Suppose* $U := \{u \mid g_i(u) \leq 0 \ \forall i \in \mathcal{I}\}$ *where, for each* $i \in \mathcal{I}$*, the function* $g_i : \mathbb{R}^m \to \mathbb{R}$ *is differentiable. Suppose also that* $u \in U$ *and that there exists a vector* $\bar{h} \in \mathcal{F}_U(u)$ *such that*

$$\langle \nabla g_i(u), \bar{h} \rangle < 0, \ \forall \ i \in \mathcal{I}^0(u) \tag{C.24}$$

Then

$$\mathcal{T}_U(u) = \mathcal{F}_U(u)$$

i.e., U *is quasiregular at* u*.*

Equation (C.24) is the constraint qualification; it can be seen that it precludes the situation shown in Figure C.8.

Proof. It follows from the definition (C.23) of $\mathcal{F}_U(u)$ and the constraint qualification (C.24) that:

$$\langle \nabla g_i(u), h + \alpha(\bar{h} - h) \rangle < 0, \ \forall h \in \mathcal{F}_U(u), \alpha \in (0,1], i \in \mathcal{I}^0(u)$$

Hence, for all $h \in \mathcal{F}_U(u)$, all $\alpha \in (0,1]$, there exists a vector $h_\alpha := h + \alpha(\bar{h} - h)$, in $\mathcal{F}_U(u)$ satisfying $\langle \nabla g_i(u), h_\alpha \rangle < 0$ for all $i \in \mathcal{I}^0(u)$. Assuming for the moment that $h_\alpha \in \mathcal{T}_U(u)$ for all $\alpha \in (0,1]$, it follows, since $h_\alpha \to h$ as $\alpha \to 0$ and $\mathcal{T}_U(u)$ is closed, that $h \in \mathcal{T}_U(u)$, thus proving $\mathcal{F}_U(u) \subset \mathcal{T}_U(u)$. It remains to show that h_α is tangent to U at u. Consider the sequences h^ν and $\lambda^\nu \searrow 0$ where $h^\nu := h_\alpha$ for all $\nu \in \mathbb{I}_{\geq 0}$. There exists a $\delta > 0$ such that $\langle \nabla g_i(u), h_\alpha \rangle \leq -\delta$ for all $i \in \mathcal{I}^0(u)$ and $g_i(u) \leq -\delta$ for all $i \in \mathcal{I} \setminus \mathcal{I}^0(u)$. Since

$$g_i(u + \lambda^\nu h^\nu) = g_i(u) + \lambda^\nu \langle \nabla g_i(u), h_\alpha \rangle + o(\lambda^\nu) \leq -\lambda^\nu \delta + o(\lambda^\nu)$$

for all $i \in \mathcal{I}^0(u)$, it follows that there exists a finite integer N such that $g_i(u + \lambda^\nu h^\nu) \leq 0$ for *all* $i \in \mathcal{I}$, all $\nu \geq N$. Since the sequences $\{h^\nu\}$ and $\{\lambda^\nu\}$ for all $\nu \geq N$ satisfy $h^\nu \to h_\alpha$, $\lambda^\nu \searrow 0$ and $u + \lambda^\nu h^\nu \in U$ for all $i \in \mathcal{I}$, it follows that $h_\alpha \in \mathcal{T}_U(u)$, thus completing the proof that $\mathcal{F}_U(u) \subset \mathcal{T}_U(u)$.

Suppose now that $h \in \mathcal{T}_U(u)$. There exist sequences $h^\nu \to h$ and $\lambda^\nu \to 0$ such that $u + \lambda^\nu h^\nu \in U$ so that $g(u + \lambda^\nu h^\nu) \leq 0$ for all $\nu \in \mathbb{I}_{\geq 0}$. Since $g(u + \lambda^\nu h^\nu) = g(u) + \langle \nabla g_j(u), \lambda^\nu h^\nu \rangle + o(\lambda^\nu |h^\nu|) \leq 0$, it follows that $\langle \nabla g_j(u), \lambda^\nu h^\nu \rangle + o(\lambda^\nu) \leq 0$ for all $j \in \mathcal{I}^0(u)$, all $\nu \in \mathbb{I}_{\geq 0}$. Hence $\langle \nabla g_j(u), h^\nu \rangle + o(\lambda^\nu)/\lambda^\nu \leq 0$ for all $j \in \mathcal{I}^0(u)$, all $\nu \in \mathbb{I}_{\geq 0}$. Taking the limit yields $\langle \nabla g_j(u), h^\nu \rangle \leq 0$ for all $j \in \mathcal{I}^0(u)$ so that $h \in \mathcal{F}_U(u)$ which proves $\mathcal{T}_U(u) \subset \mathcal{F}_U(u)$. Hence $\mathcal{T}_U(u) = \mathcal{F}_U(u)$. ∎

The existence of a $\bar{h}$ satisfying (C.24) is, as we have noted above, a constraint qualification. If u is locally optimal for the inequality constrained optimization problem of minimizing a differentiable function $f(\cdot)$ over the set U defined in (C.22) and, if (C.24) is satisfied thereby ensuring that $\mathcal{T}_U(u) = \mathcal{F}_U(u)$, then a condition of optimality of the form (C.20) may be easily obtained as shown in the next result.

Proposition C.21 (Optimality conditions nonconvex problem)**.** *Suppose u is locally optimal for the problem of minimizing a differentiable function $f(\cdot)$ over the set U defined in* (C.22) *and that $\mathcal{T}_U(u) = \mathcal{F}_U(u)$. Then*

$$-\nabla f(u) \in \text{cone}\{\nabla g_i(u) \mid i \in \mathcal{I}^0(u)\}$$

and there exist multipliers $\mu_i \geq 0$, $i \in \mathcal{I}^0(u)$ satisfying

$$\nabla f(u) + \sum_{i \in \mathcal{I}^0(u)} \mu_i \nabla g_i(u) = 0 \tag{C.25}$$

Proof. It follows from Proposition C.14 that $-\nabla f(u) \in \hat{\mathcal{N}}_U(u)$ and from Proposition C.7 that $\hat{\mathcal{N}}_U(u) = \mathcal{T}_U^*(u)$. But, by hypothesis, $\mathcal{T}_U(u) = \mathcal{F}_U(u)$ so that $\hat{\mathcal{N}}_U(u) = \mathcal{F}_U^*(u)$, the polar cone of $\mathcal{F}_U(u)$. It follows from (C.23) and the definition of a polar cone, given in Appendix A1, that

$$\mathcal{F}_U^*(u) = \text{cone}\{\nabla g_i(u) \mid i \in| I^0(u)\}$$

Hence

$$-\nabla f(u) \in \text{cone}\{\nabla g_i(u) \mid i \in \mathcal{I}^0(u)\}$$

The existence of multipliers μ_i satisfying (C.25) follows from the definition of a cone generated by $\{\nabla g_i(u) \mid i \in \mathcal{I}^0(u)\}$. ∎

C.2.7 Constraint Set Defined by Equalities and Inequalities

Finally, we consider the case when the set U is specified by a set of differentiable equalities *and* inequalities:

$$U := \{u \mid g_i(u) \leq 0 \ \forall i \in \mathcal{I},\ h_i(u) = 0 \ \forall i \in \mathcal{E}\}$$

where, for each $i \in \mathcal{I}$, the function $g_i : \mathbb{R}^m \to \mathbb{R}$ is differentiable and for each $i \in \mathcal{E}$, the function $h_i : \mathbb{R}^m \to \mathbb{R}$ is differentiable. For each $u \in U$

$$\mathcal{I}^0(u) := \{i \in \mathcal{I} \mid g_i(u) = 0\}$$

the index set of active inequality constraints is defined as before. We wish to obtain necessary conditions for the problem of minimizing a differentiable function $f(\cdot)$ over the set U. The presence of equality constraints makes this objective more difficult than for the case when U is defined merely by differentiable inequalities. The result we wish to prove is a natural extension of Proposition C.21 in which the equality constraints are included in the set of active constraints:

Proposition C.22 (Fritz-John necessary conditions)**.** *Suppose u is a local minimizer for the problem of minimizing $f(u)$ subject to the constraint $u \in U$ where U is defined in* (C.22)*. Then there exist multipliers μ_0, $\mu_i,\ i \in \mathcal{I}$ and $\lambda_i,\ i \in \mathcal{E}$, not all zero, such that*

$$\mu_0 \nabla f(u) + \sum_{i \in \mathcal{I}} \mu_i \nabla g_i(u) + \sum_{j \in \mathcal{E}} \lambda_j \nabla h_j(u) = 0 \tag{C.26}$$

and

$$\mu_i g_i(u) = 0 \ \forall i \in \mathcal{I}$$

where $\mu_0 \geq 0$ and $\mu_i \geq 0$ for all $i \in \mathcal{I}^0$.

The condition $\mu_i g_i(u) = 0$ for all $i \in \mathcal{I}$ is known as the *complementarity* conditions and implies $\mu_i = 0$ for all $i \in \mathcal{I}$ such that $g_i(u) < 0$. If $\mu_0 > 0$, then (C.26) may be normalized by dividing each term by μ_0 yielding the more familiar expression

$$\nabla f(u) + \sum_{i \in \mathcal{I}} \mu_i \nabla g_i(u) + \sum_{j \in \mathcal{E}} \nabla h_j(u) = 0$$

We return to this point later. Perhaps the simplest method for proving Proposition C.22 is the penalty approach adopted by Bertsekas (1999), Proposition 3.3.5. We merely give an outline of the proof. The constrained problem of minimizing $f(v)$ over U is approximated, for each

$k \in \mathbb{I}_{\geq 0}$ by a penalized problem defined below; as k increases the penalized problem becomes a closer approximation to the constrained problem. For each $i \in \mathcal{I}$, we define

$$g_i^+(v) := \max\{g_i(v), 0\}$$

For each k, the penalized problem $\mathbb{P}^k$ is then defined as the problem of minimizing $F^k(v)$ defined by

$$F^k(v) := f(v) + (k/2)\sum_{i\in\mathcal{I}}(g_i^+(v))^2 + (k/2)\sum_{j\in\mathcal{E}}(h_j(v))^2 + (1/2)|v-u|^2$$

subject to the constraint

$$S := \{v \mid |v-u| \leq \varepsilon\}$$

where $\epsilon > 0$ is such that $f(u) \leq f(v)$ for all v in $S \cap U$. Let v^k denote the solution of $\mathbb{P}^k$. Bertsekas shows that $v^k \to u$ as $k \to \infty$ so that for all k sufficiently large, v^k lies in the interior of S and is, therefore, the unconstrained minimizer of $F^k(v)$. Hence for each k sufficiently large, v^k satisfies $\nabla F^k(v^k) = 0$, or

$$\nabla f(v^k) + \sum_{i\in\mathcal{I}} \bar{\mu}_i^k \nabla g(v^k) + \sum_{i\in\mathcal{E}} \bar{\lambda}_i^k \nabla h(v^k) = 0 \tag{C.27}$$

where

$$\bar{\mu}_i^k := k g_i^+(v^k), \quad \bar{\lambda}_i^k := k h_i(v^k)$$

Let μ^k denote the vector with elements μ_i^k, $i \in \mathcal{I}$ and λ^k the vector with elements λ_i^k, $k \in \mathcal{E}$. Dividing (C.27) by δ^k defined by

$$\delta^k := [1 + |\mu^k|^2 + |\lambda^k|^2]^1/2$$

yields

$$\mu_0^k \nabla f(v^k) + \sum_{i\in\mathcal{I}} \mu_i^k \nabla g(v^k) + \sum_{j\in\mathcal{E}} \lambda_j^k \nabla h(v^k) = 0$$

where

$$\mu_0^k := \bar{\mu}_i^k/\delta^k, \quad \mu_i^k := \bar{\mu}_i^k/\delta^k, \quad \lambda_j^k := \bar{\lambda}_i^k/\delta^k$$

and

$$(\mu_0^k)^2 + |\mu^k|^2 + |\lambda^k|^2 = 1$$

Because of the last equation, the sequence $\left(\mu_0^k, \mu^k, \lambda^k\right)$ lies in a compact set, and therefore has a subsequence, indexed by $K \subset \mathbb{I}_{\geq 0}$, converging to some limit (μ_0, μ, λ) where μ and λ are vectors whose elements are,

respectively, $\mu_i,\ i \in \mathcal{I}$ and $\lambda_j,\ j \in \mathcal{E}$. Because $v^k \to u$ as $k \in K$ tends to infinity, it follows from (C.27) that

$$\mu_0 \nabla f(u) + \sum_{i \in \mathcal{I}} \mu_i \nabla g_i(u) + \sum_{j \in \mathcal{E}} \lambda_j \nabla h_j(u) = 0$$

To prove the complementarity condition, suppose, contrary to what we wish to prove, that there exists a $i \in \mathcal{I}$ such that $g_i(u) < 0$ but $\mu_i > 0$. Since $\mu_i^k \to \mu_i > 0$ and $g_i(v^k) \to g_i(u)$ as $k \to \infty$, $k \in K$, it follows that $\mu_i \mu_i^k > 0$ for all $k \in K$ sufficiently large. But $\mu_i^k = \bar{\mu}_i^k / \delta^k = k g_i^+(v^k)/\delta^k$ so that $\mu_i \mu_i^k > 0$ implies $\mu_i g_i^+(v^k) > 0$ which in turn implies $g_i^+(v^k) = g_i(v^k) > 0$ for all $k \in K$ sufficiently large. This contradicts the fact that $g_i(v^k) \to g_i(u) < 0$ as $k \to \infty$, $k \in K$. Hence we must have $g_i(u) = 0$ for all $i \in \mathcal{I}$ such that $\mu_i > 0$.

The Fritz-John condition in Proposition C.22 is known as the Karush-Kuhn-Tucker (KKT) condition if $\mu_0 > 0$; if this is the case, μ_0 may be normalized to $\mu_0 = 1$. A constraint qualification is required for the Karush-Kuhn-Tucker condition to be a necessary condition of optimality for the optimization problem considered in this section. A simple constraint qualification is linear independence of $\{\nabla g_i(u),\ i \in \mathcal{I}^0(u),\ \nabla h_j(u),\ j \in \mathcal{E}\}$ at a local minimizer u. For, if u is a local minimizer and $\mu_0 = 0$, then the Fritz-John condition implies that $\sum_{i \in \mathcal{I}^0(u)} \mu_i \nabla g_i(u) + \sum_{j \in \mathcal{E}} \lambda_j \nabla h_j(u) = 0$ which contradicts the linear independence of $\{\nabla g_i(u),\ i \in \mathcal{I}^0(u),\ \nabla h_j(u),\ j \in \mathcal{E}\}$ since not all the multipliers are zero. Another constraint qualification, used in Propositions C.20 and C.21 for an optimization problem in which the constraint set is $U := \{u \mid g_i(u) \le 0,\ i \in \mathcal{I}\}$, is the existence of a vector $\bar{h}(u) \in \mathcal{F}_U(u)$ such that $\langle \nabla g_i(u), \bar{h} \rangle < 0$ for all $i \in \mathcal{I}^0(u)$; this condition ensures $\mu_0 = 1$ in C.25. Many other constraint qualifications exist; see, for example, Bertsekas (1999), Chapter 3.

C.3 Set-Valued Functions and Continuity of Value Function

A set-valued function $U(\cdot)$ is one for which, for each value of x, $U(x)$ is a set; these functions are encountered in parametric programming. For example, in the problem $\mathbb{P}(x) : \inf_u \{f(x,u) \mid u \in U(x)\}$ (which has the same form as an optimal control problem in which x is the state and u is a control sequence), the constraint set U is a set-valued function of the state. The solution to the problem $\mathbb{P}(x)$ (the value of u that achieves the minimum) can also be set-valued. It is important to

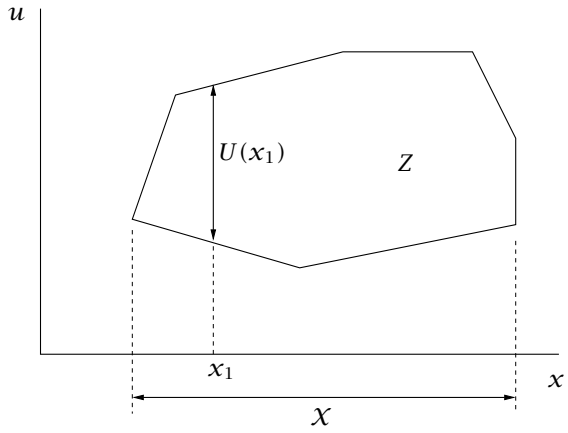
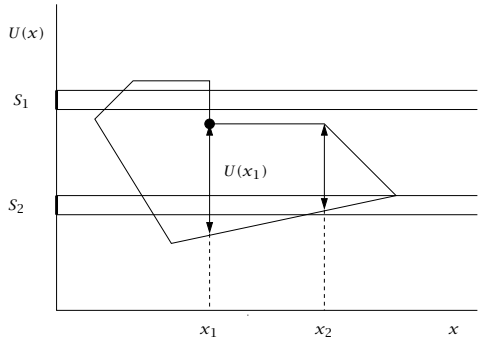


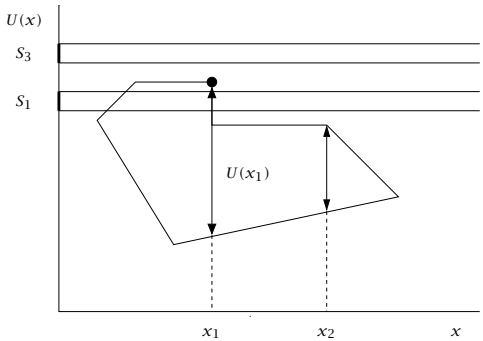
Figure C.9: Graph of set-valued function $U(\cdot)$.

know how smoothly these set-valued functions vary with the parameter x. In particular, we are interested in the continuity properties of the value function $x \mapsto f^0(x) = \inf_u\{f(x,u) \mid u \in U(x)\}$ since, in optimal control problems we employ the value function as a Lyapunov function and robustness depends, as we have discussed earlier, on the continuity of the Lyapunov function. Continuity of the value function depends, in turn, on continuity of the set-valued constraint set $U(\cdot)$. We use the notation $U : \mathbb{R}^n \rightsquigarrow \mathbb{R}^m$ to denote the fact that $U(\cdot)$ maps points in $\mathbb{R}^n$ into subsets of $\mathbb{R}^m$.

The *graph* of a set-valued functions is often a useful tool. The graph of $U : \mathbb{R}^n \rightsquigarrow \mathbb{R}^m$ is defined to be the set $Z := \{(x,u) \in \mathbb{R}^n \times \mathbb{R}^m \mid u \in U(x)\}$; the *domain* of the set-valued function U is the set $\mathcal{X} := \{x \in \mathbb{R}^n \mid U(x) \neq \emptyset\} = \{x \in \mathbb{R}^n \mid \exists u \in \mathbb{R}^m \text{ such that } (x,u) \in Z\}$; clearly $\mathcal{X} \subset \mathbb{R}^n$. Also $\mathcal{X}$ is the *projection* of the set $Z \subset \mathbb{R}^n \times \mathbb{R}^m$ onto $\mathbb{R}^n$, i.e., $(x,u) \in Z$ implies $x \in \mathcal{X}$. An example is shown in Figure C.9. In this example, $U(x)$ varies continuously with x. Examples in which $U(\cdot)$ is discontinuous are shown in Figure C.10. In Figure C.10(a), the set $U(x)$ varies continuously if x increases from its initial value of x_1, but jumps to a much larger set if x decreases an infinitesimal amount (from its initial value of x_1); this is an example of a set-valued function that is inner semicontinuous at x_1. In Figure C.10(b), the set $U(x)$ varies continuously if x decreases from its initial value of x_1, but jumps to a much smaller set if x increases an infinitesimal amount (from its initial value of x_1); this is an example of a set-valued function that is



(a) Inner semicontinuous set-valued function.



(b) Outer semicontinuous set-valued function.

Figure C.10: Graphs of discontinuous set-valued functions.

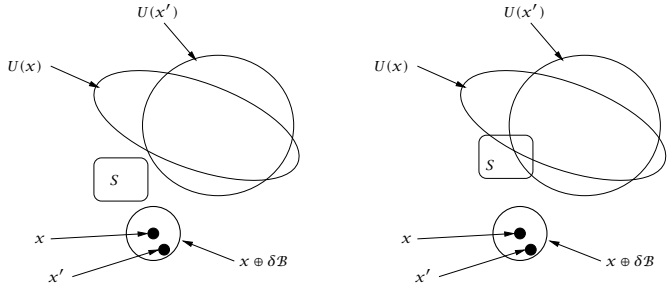
outer semicontinuous at x_1. The set-valued function is continuous at x_2 where it is both outer and inner semicontinuous.

We can now give precise definitions of inner and outer semicontinuity.

C.3.1 Outer and Inner Semicontinuity

The concepts of inner and outer semicontinuity were introduced by Rockafellar and Wets (1998, p. 144) to replace earlier definitions of lower and upper semicontinuity of set-valued functions. This section is based on the useful summary provided by Polak (1997, pp. 676-682).

Definition C.23 (Outer semicontinuous function)**.** A set-valued function $U : \mathbb{R}^n \rightsquigarrow \mathbb{R}^m$ is said to be outer semicontinuous (osc) at x if $U(x)$



(a) Outer semicontinuity. (b) Inner semicontinuity.

Figure C.11: Outer and inner semicontinuity of $U(\cdot)$.

is closed and if, for every compact set S such that $U(x) \cap S = \emptyset$, there exists a $\delta > 0$ such that $U(x') \cap S = \emptyset$ for all $x' \in x \oplus \delta\mathcal{B}$.[4] The set-valued function $U : \mathbb{R}^n \rightsquigarrow \mathbb{R}^m$ is outer semicontinuous if it is outer semicontinuous at each $x \in \mathbb{R}^n$.

Definition C.24 (Inner semicontinuous function)**.** A set-valued function $U : \mathbb{R}^n \rightsquigarrow \mathbb{R}^m$ is said to be inner semicontinuous (isc) at x if, for every open set S such that $U(x) \cap S \neq \emptyset$, there exists a $\delta > 0$ such that $U(x') \cap S \neq \emptyset$ for all $x' \in x \oplus \delta\mathcal{B}$. The set-valued function $U : \mathbb{R}^n \rightsquigarrow \mathbb{R}^m$ is inner semicontinuous if it is inner semicontinuous at each $x \in \mathbb{R}^n$.

These definitions are illustrated in Figure C.11. Roughly speaking, a set-valued function that is outer semicontinuous at x cannot explode as x changes to x' arbitrarily close to x; similarly, a set-valued function that is inner semicontinuous at x cannot collapse as x changes to x' arbitrarily close to x.

Definition C.25 (Continuous function)**.** A set-valued function is continuous (at x) if it is both outer and inner continuous (at x).

If we return to Figure C.10(a) we see that $S_1 \cap U(x_1) = \emptyset$ but $S_1 \cap U(x) \neq \emptyset$ for x infinitesimally less than x_1 so that $U(\cdot)$ is not outer semicontinuous at x_1. For all S_2 such that $S_2 \cap U(x_1) \neq \emptyset$, however, $S_2 \cap U(x) \neq \emptyset$ for all x in a sufficiently small neighborhood of x_1 so that $U(\cdot)$ is inner semicontinuous at x_1. If we turn to Figure C.10(b) we see that $S_1 \cap U(x_1) \neq \emptyset$ but $S_1 \cap U(x) = \emptyset$ for x infinitesimally greater than x_1 so that in this case $U(\cdot)$ is not inner semicontinuous at x_1. For all S_3 such that $S_3 \cap U(x_1) = \emptyset$, however, $S_3 \cap U(x) = \emptyset$ for

[4] Recall that $\mathcal{B} := \{x \mid |x| \leq 1\}$ is the closed unit ball in $\mathbb{R}^n$.

all x in a sufficiently small neighborhood of x_1 so that $U(\cdot)$ is outer semicontinuous at x_1.

The definitions of outer and inner semicontinuity may be interpreted in terms of infinite sequences (Rockafellar and Wets, 1998, p. 152), (Polak, 1997, pp. 677-678).

Theorem C.26 (Equivalent conditions for outer and inner semicontinuity)**.**

(a) A set-valued function $U : \mathbb{R}^n \rightsquigarrow \mathbb{R}^m$ is outer semicontinuous at x if and only if for every infinite sequence (x_i) converging to x, any accumulation point[5] u of any sequence (u_i), satisfying $u_i \in U(x_i)$ for all i, lies in $U(x)$ ($u \in U(x)$).

(b) A set-valued function $U : \mathbb{R}^n \rightsquigarrow \mathbb{R}^m$ is inner semicontinuous at x if and only if for every $u \in U(x)$ and for every infinite sequence (x_i) converging to x, there exists an infinite sequence (u_i), satisfying $u_i \in U(x_i)$ for all i, that converges to u.

Proofs of these results may be found in Rockafellar and Wets (1998); Polak (1997). Another result that we employ is:

Proposition C.27 (Outer semicontinuity and closed graph)**.** *A set-valued function $U : \mathbb{R}^n \rightsquigarrow \mathbb{R}^m$ is outer semicontinuous in its domain if and only if its graph Z is closed in $\mathbb{R}^n \times \mathbb{R}^m$.*

Proof. Since $(x, u) \in Z$ is equivalent to $u \in U(x)$, this result is a direct consequence of the Theorem C.26. ∎

In the above discussion we have assumed, as in Polak (1997), that $U(x)$ is defined everywhere in $\mathbb{R}^n$; in constrained parametric optimization problems, however, $U(x)$ is defined on $\mathcal{X}$, a closed subset of $\mathbb{R}^n$; see Figure C.9. Only minor modifications of the above definitions are then required. In definitions C.23 and C.24 we replace the closed set $\delta\mathcal{B}$ by $\delta\mathcal{B} \cap \mathcal{X}$ and in Theorem C.26 we replace "every infinite sequence (in $\mathbb{R}^n$)" by "every infinite sequence in $\mathcal{X}$." In effect, we are replacing the topology of $\mathbb{R}^n$ by its topology relative to $\mathcal{X}$.

C.3.2 Continuity of the Value Function

Our main reason for introducing set-valued functions is to provide us with tools for analyzing the continuity properties of the value function and optimal control law in constrained optimal control problems.

[5] Recall, u is the limit of (u_i) if $u_i \to u$ as $i \to \infty$; u is an accumulation point of (u_i) if it is the limit of a subsequence of (u_i).

These problems have the form

$$V^0(x) = \min\{V(x,u) \mid u \in U(x)\} \tag{C.28}$$
$$u^0(x) = \arg\min\{V(x,u) \mid u \in U(x)\} \tag{C.29}$$

where $U : \mathbb{R}^n \rightsquigarrow \mathbb{R}^m$ is a set-valued function and $V : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ is continuous; in optimal control problems arising from MPC, u should be replaced by $\mathbf{u} = (u(0), u(1), \ldots, u(N-1))$ and m by Nm. We are interested in the continuity properties of the value function $V^0 : \mathbb{R}^n \to \mathbb{R}$ and the control law $u^0 : \mathbb{R}^n \to \mathbb{R}^m$; the latter may be set-valued (if the minimizer in (C.28) is not unique).

The following max problem has been extensively studied in the literature

$$\phi^0(x) = \max\{\phi(x,u) \mid u \in U(x)\}$$
$$\mu^0(x) = \arg\max\{\phi(x,u) \mid u \in U(x)\}$$

If we define $\phi(\cdot)$ by $\phi(x,u) := -V(x,u)$, we see that $\phi^0(x) = -V^0(x)$ and $\mu^0(x) = u^0(x)$ so that we can obtain the continuity properties of $V^0(\cdot)$ and $u^0(\cdot)$ from those of $\phi^0(\cdot)$ and $\mu^0(\cdot)$ respectively. Using this transcription and Corollary 5.4.2 and Theorem 5.4.3 in Polak (1997) we obtain the following result:

Theorem C.28 (Minimum theorem)**.** *Suppose that $V : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ is continuous, that $U : \mathbb{R}^n \rightsquigarrow \mathbb{R}^m$ is continuous, compact-valued and satisfies $U(x) \subset \mathbb{U}$ for all $x \in \mathcal{X}$ where $\mathbb{U}$ is compact. Then $V^0(\cdot)$ is continuous and $u^0(\cdot)$ is outer semicontinuous. If, in addition, $u^0(x) = \{\mu^0(x)\}$ (there is a unique minimizer $\mu^0(x)$), then $\mu^0(\cdot)$ is continuous.*

It is unfortunately the case, however, that due to state constraints, $U(\cdot)$ is often not continuous in constrained optimal control problems. If $U(\cdot)$ is constant, which is the case in optimal control problem if state or mixed control-state constraints are absent, then, from the above results, the value function $V^0(\cdot)$ is continuous. Indeed, under slightly stronger assumptions, the value function is Lipschitz continuous.

Lipschitz continuity of the value function. If we assume that $V(\cdot)$ is Lipschitz continuous and that $U(x) \equiv U$, we can establish Lipschitz continuity of $V^0(\cdot)$. Interestingly the result does not require, nor does it imply, Lipschitz continuity of the minimizer $u^0(\cdot)$.

Theorem C.29 (Lipschitz continuity of the value function, constant U)**.** *Suppose that $V : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ is Lipschitz continuous on bounded sets*[6] *and that $U(x) \equiv U$ where U is a compact subset of $\mathbb{R}^m$. Then $V^0(\cdot)$ is Lipschitz continuous on bounded sets.*

Proof. Let S be an arbitrary bounded set in $\mathcal{X}$, the domain of the value function $V^0(\cdot)$, and let $R := S \times \mathbb{U}$; R is a bounded subset of $\mathcal{Z}$. Since R is bounded, there exists a Lipschitz constant L_S such that

$$\left| V(x', u) - V(x'', u) \right| \leq L_S |x' - x''|$$

for all $x', x'' \in S$, all $u \in U$. Hence,

$$V^0(x') - V^0(x'') \leq V(x', u'') - V(x'', u'') \leq L_S |x' - x''|$$

for all $x', x'' \in S$, any $u'' \in u^0(x'')$. Interchanging x' and x'' in the above derivation yields

$$V^0(x'') - V^0(x') \leq V(x'', u') - V(x', u') \leq L_S |x'' - x'|$$

for all $x', x'' \in S$, any $u' \in u^0(x')$. Hence $V^0(\cdot)$ is Lipschitz continuous on bounded sets. ■

We now specialize to the case where $U(x) = \{u \in \mathbb{R}^m \mid (x, u) \in \mathcal{Z}\}$ where $\mathcal{Z}$ is a polyhedron in $\mathbb{R}^n \times \mathbb{R}^m$; for each x, $U(x)$ is a polytope. This type of constraint arises in constrained optimal control problems when the system is linear and the state and control constraints are polyhedral. What we show in the sequel is that, in this special case, $U(\cdot)$ is continuous and so, therefore, is $V^0(\cdot)$. An alternative proof, which many readers may prefer, is given in Chapter 7 where we exploit the fact that if $V(\cdot)$ is strictly convex and quadratic and $\mathcal{Z}$ polyhedral, then $V^0(\cdot)$ is piecewise quadratic and continuous. Our first concern is to obtain a bound on $d(u, U(x'))$, the distance of any $u \in U(x)$ from the constraint set $U(x')$.

A bound on $d(u, U(x'))$, $u \in U(x)$. The bound we require is given by a special case of a theorem due to Clarke, Ledyaev, Stern, and Wolenski (1998, Theorem 3.1, page 126). To motivate this result, consider a differentiable convex function $f : \mathbb{R} \to \mathbb{R}$ so that $f(u) \geq f(v) + \langle \nabla f(v), u - v \rangle$ for any two points u and v in $\mathbb{R}$. Suppose also that there exists a nonempty interval $U = [a, b] \subset \mathbb{R}$ such that $f(u) \leq 0$ for

[6] A function $V(\cdot)$ is Lipschitz continuous on bounded sets if, for any bounded set S, there exists a constant $L_S \in [0, \infty)$ such that $|V(z') - V(z)| \leq L_S |z - z'|$ for all $z, z' \in S$.

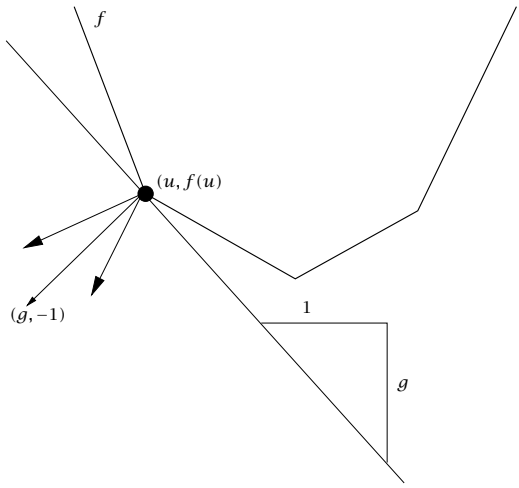


Figure C.12: Subgradient of $f(\cdot)$.

all $u \in U$ and that there exists a $\delta > 0$ such that $\Delta f(u) > \delta$ for all $u \in \mathbb{R}$. Let $u > b$ and let $v = b$ be the closest point in U to u. Then $f(u) \geq f(v) + \langle \nabla f(v), u - v \rangle \geq \delta |v - u|$ so that $d(u, U) \leq f(u)/\delta$. The theorem of Clarke et al. (1998) extends this result to the case when $f(\cdot)$ is not necessarily differentiable but requires the concept of a subgradient of a convex function

Definition C.30 (Subgradient of convex function)**.** Suppose $f : \mathbb{R}^m \to \mathbb{R}$ is convex. Then the subgradient $\delta f(u)$ of $f(\cdot)$ at u is defined by

$$\delta f(u) := \{g \mid f(v) \geq f(u) + \langle g, v - u \rangle \ \forall v \in \mathbb{R}^m\}$$

Figure C.12 illustrates a subgradient. In the figure, g is one element of the subgradient because $f(v) \geq f(u) + \langle g, v - u \rangle$ for all v; g is the slope of the line passing through the point $(u, f(u))$. To obtain a bound on $d(u, U(x))$ we require the following result which is a special case of the much more general result of the theorem of Clarke *et al.*:

Theorem C.31 (Clarke et al. (1998))**.** *Take a nonnegative valued, convex function* $\psi : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$. *Let* $U(x) := \{u \in \mathbb{R}^m \mid \psi(x, u) = 0\}$ *and* $\mathcal{X} := \{x \in \mathbb{R}^n \mid U(x) \neq \emptyset\}$. *Assume there exists a* $\delta > 0$ *such that*

$$u \in \mathbb{R}^m,\ x \in \mathcal{X}, \psi(x, u) > 0 \text{ and } g \in \partial_u \psi(x, u) \implies |g| > \delta$$

where $\partial_u \psi(x, u)$ *denotes the convex subgradient of* ψ *with respect to*

the variable u. Then, for each $x \in \mathcal{X}$, $d(u, U(x)) \leq \psi(x,u)/\delta$ for all $u \in \mathbb{R}^m$.

The proof of this result is given in the reference cited above. We next use this result to bound the distance of u from $U(x)$ where, for each x, $U(x)$ is polyhedral.

Corollary C.32 (A bound on $d(u, U(x'))$ for $u \in U(x)$)**.** [7] *Suppose $\mathcal{Z}$ is a polyhedron in $\mathbb{R}^n \times \mathbb{R}^m$ and let $\mathcal{X}$ denote its projection on $\mathbb{R}^n$ ($\mathcal{X} = \{x \mid \exists u \in \mathbb{R}^m$ such that $(x,u) \in \mathcal{Z}\}$). Let $\mathcal{U}(x) := \{u \mid (x, u) \in \mathcal{Z}\}$. Then there exists a $K > 0$ such that for all $x, x' \in \mathcal{X}$, $d(u, U(x')) \leq K|x' - x|$ for all $u \in U(x)$ (or, for all $x, x' \in \mathcal{X}$, all $u \in U(x)$, there exists a $u' \in U(x')$ such that $|u' - u| \leq K|x' - x|$).*

Proof. The polyhedron $\mathcal{Z}$ admits the representation $\mathcal{Z} = \{(x,u) \mid \langle m^j, u\rangle - \langle n^j, x\rangle - p^j \leq 0,\ j \in \mathcal{J}\}$ for some $m^j \in \mathbb{R}^m$, $n^j \in \mathbb{R}^n$ and $p^j \in \mathbb{R}$, $j \in \mathcal{J} := \{1, \ldots, J\}$. Define $\mathcal{D}$ to be the collection of all index sets $I \subseteq \mathcal{J}$ such that $\sum_{j\in I} \lambda^j m^j \neq 0,\ \forall \lambda \in \Lambda_I$ in which, for a particular index set I, Λ_I is defined to be $\Lambda_I := \{\lambda \mid \lambda^j \geq 0,\ \sum_{j\in I} \lambda^j = 1\}$. Because $\mathcal{D}$ is a finite set, there exists a $\delta > 0$ such that for all $I \in \mathcal{D}$, all $\lambda \in \Lambda_I$, $|\sum_{j\in I} \lambda^j m^j| > \delta$. Let $\psi(\cdot)$ be defined by $\psi(x,u) := \max\{\langle m^j, u\rangle - \langle n^j, x\rangle - p^j, 0 \mid j \in \mathcal{J}\}$ so that $(x,u) \in \mathcal{Z}$ (or $u \in \mathcal{U}(x)$) if and only if $\psi(x,u) = 0$. We now claim that, for every $(x,u) \in \mathcal{X} \times \mathbb{R}^m$ such that $\psi(x,u) > 0$ and every $g \in \partial_u \psi(x,u)$, the subgradient of ψ with respect to u at (x,u), we have $|g| > \delta$. Assuming for the moment that the claim is true, the proof of the Corollary may be completed with the aid of Theorem C.31. Assume, as stated in the Corollary, that $x, x' \in \mathcal{X}$ and $u \in \mathcal{U}(x)$; the theorem asserts

$$d(u, \mathcal{U}(x')) \leq (1/\delta)\psi(x', u),\ \forall x' \in \mathcal{X}$$

But $\psi(x,u) = 0$ (since $u \in \mathcal{U}(x)$) so that

$$d(u, \mathcal{U}(x')) \leq (1/\delta)[\psi(x', u) - \psi(x,u)] \leq (c/\delta)|x' - x|$$

where c is the Lipschitz constant for $x \mapsto \psi(x,u)$ ($\psi(\cdot)$ is piecewise affine and continuous). This proves the Corollary with $K = c/\delta$.

It remains to confirm the claim. Take any $(x,u) \in \mathcal{X} \times \mathbb{R}^m$ such that $\psi(x,u) > 0$. Then $\max_j\{\langle m^j, u\rangle - \langle n^j, x\rangle - p^j, 0 \mid j \in \mathcal{J}\} > 0$. Let

[7] The authors wish to thank Richard Vinter and Francis Clarke for providing this result.

$I^0(x,u)$ denote the active constraint set (the set of those constraints at which the maximum is achieved). Then

$$\langle m^j, u\rangle - \langle n^j, x\rangle - p^j > 0,\ \forall j \in I^0(x,u)$$

Since $x \in \mathcal{X}$, there exists a $\bar{u} \in \mathcal{U}(x)$ so that

$$\langle m^j, \bar{u}\rangle - \langle n^j, x\rangle - p^j \leq 0,\ \forall j \in I^0(x,u)$$

Subtracting these two inequalities yields

$$\langle m^j, u - \bar{u}\rangle > 0,\ \forall j \in I^0(x,u)$$

But then, for all $\lambda \in \Lambda_{I^0(x,u)}$, it follows that $|\sum_{j\in I^0(x,u)} \lambda^j m^j (u-\bar{u})| > 0$, so that

$$\sum_{j\in I^0(x,u)} \lambda^j m^j \neq 0$$

It follows that $I^0(x,u) \in \mathcal{D}$. Hence

$$|\sum_{j\in I^0(x,u)} \lambda^j m^j| > \delta,\ \forall \lambda \in \Lambda_{I^0(x,u)}$$

Now take any $g \in \partial_u f(x,u) = \text{co}\{m^j \mid j \in I^0(x,u)\}$ (co denotes "convex hull"). There exists a $\lambda \in \Lambda_{I^0(x,u)}$ such that $g = \sum_{j\in I^0(x,u)} \lambda^j m_j$. But then $|g| > \delta$ by the inequality above. This proves the claim and, hence, completes the proof of the Corollary. ■

Continuity of the value function when $U(x) = \{u \mid (x,u) \in \mathcal{Z}\}$. In this section we investigate continuity of the value function for the constrained linear quadratic optimal control problem $\mathbb{P}(x)$; in fact we establish continuity of the value function for the more general problem where the cost is continuous rather than quadratic. We showed in Chapter 2 that the optimal control problem of interest takes the form

$$V^0(x) = \min_u \{V(x,u) \mid (x,u) \in \mathcal{Z}\}$$

where $\mathcal{Z}$ is a polyhedron in $\mathbb{R}^n \times \mathbb{U}$ where $\mathbb{U} \subset \mathbb{R}^m$ is a polytope and, hence, is compact and convex; in MPC problems we replace the control u by the sequence of controls $\mathbf{u}$ and m by Nm. Let $u^0 : \mathbb{R}^n \rightsquigarrow \mathbb{R}^m$ be defined by

$$u^0(x) := \arg\min_u \{V(x,u) \mid (x,u) \in \mathcal{Z}\}$$

and let $\mathcal{X}$ be defined by

$$\mathcal{X} := \{x \mid \exists u \text{ such that } (x,u) \in \mathcal{Z}\}$$

so that $\mathcal{X}$ is the projection of $\mathcal{Z} \subset \mathbb{R}^n \times \mathbb{R}^m$ onto $\mathbb{R}^n$. Let the set-valued function $U : \mathbb{R}^n \rightsquigarrow \mathbb{R}^m$ be defined by

$$U(x) \coloneqq \{u \in \mathbb{R}^m \mid (x, u) \in Z\}$$

The domain of $V^0(\cdot)$ and of $U(\cdot)$ is $\mathcal{X}$. The optimization problem may be expressed as $V^0(x) = \min_u\{V(x, u) \mid u \in U(x)\}$. Our first task is establish the continuity of $U : \mathbb{R}^n \rightsquigarrow \mathbb{R}^m$.

Theorem C.33 (Continuity of $U(\cdot)$)**.** *Suppose $\mathcal{Z}$ is a polyhedron in $\mathbb{R}^n \times \mathbb{U}$ where $\mathbb{U} \subset \mathbb{R}^m$ is a polytope. Then the set-valued function $U : \mathbb{R}^n \rightsquigarrow \mathbb{R}^m$ defined above is continuous in $\mathcal{X}$.*

Proof. By Proposition C.27, the set-valued map $U(\cdot)$ is outer semicontinuous in $\mathcal{X}$ because its graph, $\mathcal{Z}$, is closed. We establish inner semicontinuity using Corollary C.32 above. Let x, x' be arbitrary points in $\mathcal{X}$ and $U(x)$ and $U(x')$ the associated control constraint sets. Let S be any open set such that $U(x) \cap S \neq \emptyset$ and let u be an arbitrary point in $U(x) \cap S$. Because S is open, there exist an $\varepsilon > 0$ such that $u \oplus \varepsilon\mathcal{B} \subset S$. Let $\varepsilon' \coloneqq \varepsilon/K$ where K is defined in Corollary 1. From Corollary C.32, there exists a $u' \in U(x')$ such that $|u' - u| \leq K|x' - x|$ which implies $|u' - u| \leq \varepsilon$ ($u' \in u \oplus \varepsilon\mathcal{B}$) for all $x' \in \mathcal{X}$ such that $|x' - x| \leq \varepsilon'$ ($x' \in (x \oplus \varepsilon'\mathcal{B}) \cap \mathcal{X}$). This implies $u \in \mathcal{U}(x') \cap S$ for all $x' \in \mathcal{X}$ such that $|x' - x| \leq \varepsilon'$ ($x' \in (x \oplus \varepsilon'\mathcal{B}) \cap \mathcal{X}$). Hence $\mathcal{U}(x') \cap S \neq \emptyset$ for all $x' \in (x \oplus \varepsilon'\mathcal{B}) \cap \mathcal{X}$, so that $\mathcal{U}(\cdot)$ is inner semicontinuous in $\mathcal{X}$. Since $U(\cdot)$ is both outer and inner semicontinuous in $\mathcal{X}$, it is continuous in $\mathcal{X}$. ∎

We can now establish continuity of the value function.

Theorem C.34 (Continuity of the value function)**.** *Suppose that $V : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ is continuous and that $\mathcal{Z}$ is a polyhedron in $\mathbb{R}^n \times \mathbb{U}$ where $\mathbb{U} \subset \mathbb{R}^m$ is a polytope. Then $V^0 : \mathbb{R}^n \to \mathbb{R}$ is continuous and $u^0 : \mathbb{R}^n \rightsquigarrow \mathbb{R}^m$ is outer semicontinuous in $\mathcal{X}$. Moreover, if $u^0(x)$ is unique (not set-valued) at each $x \in \mathcal{X}$, then $u^0 : \mathbb{R}^n \to \mathbb{R}^m$ is continuous in $\mathcal{X}$.*

Proof. Since the real-valued function $V(\cdot)$ is continuous (by assumption) and since the set-valued function $U(\cdot)$ is continuous in $\mathcal{X}$ (by Theorem C.33), it follows from Theorem C.28 that $V^0 : \mathbb{R}^n \to \mathbb{R}$ is continuous and $u^0 : \mathbb{R}^n \rightsquigarrow \mathbb{R}^m$ is outer semicontinuous in $\mathcal{X}$; it also follows that if $u^0(x)$ is unique (not set-valued) at each $x \in \mathcal{X}$, then $u^0 : \mathbb{R}^n \to \mathbb{R}^m$ is continuous in $\mathcal{X}$. ∎

Lipschitz continuity when $U(x) = \{u \mid (x,u) \in \mathcal{Z}\}$. Here we establish that $V^0(\cdot)$ is Lipschitz continuous if $V(\cdot)$ is Lipschitz continuous and $U(x) := \{u \in \mathbb{R}^m \mid (x,u) \in \mathcal{Z}\}$; this result is more general than Theorem C.29 where it is assumed that U is constant.

Theorem C.35 (Lipschitz continuity of the value function—$U(x)$)**.** *Suppose that $V : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ is continuous, that $\mathcal{Z}$ is a polyhedron in $\mathbb{R}^n \times \mathbb{U}$ where $\mathbb{U} \subset \mathbb{R}^m$ is a polytope. Suppose, in addition, that $V : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ is Lipschitz continuous on bounded sets.*[8] *Then $V^0(\cdot)$ is Lipschitz continuous on bounded sets.*

Proof. Let S be an arbitrary bounded set in $\mathcal{X}$, the domain of the value function $V^0(\cdot)$, and let $R := S \times \mathbb{U}$; R is a bounded subset of $\mathcal{Z}$. Let x, x' be two arbitrary points in S. Then

$$V^0(x) = V(x, \kappa(x))$$
$$V^0(x') = V(x', \kappa(x'))$$

where $V(\cdot)$ is the cost function, assumed to be Lipschitz continuous on bounded sets, and $\kappa(\cdot)$, the optimal control law, satisfies $\kappa(x) \in U(x) \subset \mathbb{U}$ and $\kappa(x') \in U(x') \subset \mathbb{U}$. It follows from Corollary C.32 that there exists a $K > 0$ such that for all $x, x' \in \mathcal{X}$, there exists a $u' \in U(x') \subset \mathbb{U}$ such that $|u' - \kappa(x)| \le K|x' - x|$. Since $\kappa(x)$ is optimal for the problem $\mathbb{P}(x)$, and since $(x, \kappa(x))$ and (x', u') both lie in $R = S \times \mathbb{U}$, there exists a constant L_R such that

$$\begin{aligned} V^0(x') - V^0(x) &\le V(x', u') - V(x, \kappa(x)) \\ &\le L_R(|(x', u') - (x, \kappa(x))|) \\ &\le L_R|x' - x| + L_R K|x' - x| \\ &\le M_S|x' - x|, \quad M_S := L_R(1 + K) \end{aligned}$$

Reversing the role of x and x' we obtain the existence of a $u \in \mathcal{U}(x)$ such that $|u - \kappa(x')| \le K|x - x'|$; it follows from the optimality of $\kappa(x')$ that

$$\begin{aligned} V^0(x) - V^0(x') &\le V(x, u) - V(x', \kappa(x')) \\ &\le M_S|x - x'| \end{aligned}$$

where, now, $u \in U(x)$ and $\kappa(x') \in U(x')$. Hence $|V^0(x') - V^0(x)| \le M_S|x - x'|$ for all x, x' in S. Since S is an arbitrary bounded set in $\mathcal{X}$, $V^0(\cdot)$ is Lipschitz continuous on bounded sets. ∎

[8] A function $V(\cdot)$ is Lipschitz continuous on bounded sets if, for any bounded set S, there exists a constant $L_S \in [0, \infty)$ such that $|V(z') - V(z)| \le L_S|z - z'|$ for all $z, z' \in S$.

C.4 Exercises

Exercise C.1: Nested optimization and switching order of optimization

Consider the optimization problem in two variables

$$\min_{(x,y)\in\mathbb{Z}} V(x,y)$$

in which $x \in \mathbb{R}^n$, $y \in \mathbb{R}^m$, and $V : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$. Assume this problem has a solution. This assumption is satisfied, for example, if V is continuous and $\mathbb{Z}$ is compact, but, in general, we do not require either of these conditions.

Define the following four sets

$$\mathbb{X}(y) = \{x \mid (x,y) \in \mathbb{Z}\} \qquad \mathbb{Y}(x) = \{y \mid (x,y) \in \mathbb{Z}\}$$
$$\mathbb{B} = \{y \mid \mathbb{X}(y) \neq \emptyset\} \qquad \mathbb{A} = \{x \mid \mathbb{Y}(x) \neq \emptyset\}$$

Note that $\mathbb{A}$ and $\mathbb{B}$ are the projections of $\mathbb{Z}$ onto $\mathbb{R}^n$ and $\mathbb{R}^m$, respectively. Projection is defined in Section C.3. Show the solutions of the following two nested optimization problems exist and are equal to the solution of the original problem

$$\min_{x\in\mathbb{A}} \left(\min_{y\in\mathbb{Y}(x)} V(x,y) \right)$$
$$\min_{y\in\mathbb{B}} \left(\min_{x\in\mathbb{X}(y)} V(x,y) \right)$$

Exercise C.2: DP nesting

Prove the assertion made in Section C.1.2 that $\mathbf{u}^i = \{u, \mathbf{u}^{i+1}\} \in \mathcal{U}(x,i)$ if and only if $(x,u) \in \mathbb{Z}$, $f(x,u) \in X(i+1)$, and $\mathbf{u}^{i+1} \in \mathcal{U}(f(x,u), i+1)$.

Exercise C.3: Recursive feasibility

Prove the assertion in the proof of Theorem C.2 that $(x(j), u(j)) \in \mathbb{Z}$ and that $f(x(j), u(j)) \in X(j+1)$.

Exercise C.4: Basic minmax result

Consider the following two minmax optimization problems in two variables

$$\inf_{x\in\mathbb{X}} \sup_{y\in\mathbb{Y}} V(x,y) \qquad \sup_{y\in\mathbb{Y}} \inf_{x\in\mathbb{X}} V(x,y)$$

in which $x \in \mathbb{X} \subseteq \mathbb{R}^n$, $y \in \mathbb{Y} \subseteq \mathbb{R}^m$, and $V : \mathbb{X} \times \mathbb{Y} \to \mathbb{R}$.

(a) Show that the values are ordered as follows

$$\inf_{x\in\mathbb{X}} \sup_{y\in\mathbb{Y}} V(x,y) \geq \sup_{y\in\mathbb{Y}} \inf_{x\in\mathbb{X}} V(x,y)$$

or, if the solutions to the problems exist,

$$\min_{x\in\mathbb{X}} \max_{y\in\mathbb{Y}} V(x,y) \geq \max_{y\in\mathbb{Y}} \min_{x\in\mathbb{X}} V(x,y)$$

A handy mnemonic for this result is that the player who optimizes first (inner problem) has the advantage.[9]

(b) Use your results to order these three problems

$$\sup_{x\in\mathbb{X}} \inf_{y\in\mathbb{Y}} \sup_{z\in\mathbb{Z}} V(x,y,z) \qquad \inf_{y\in\mathbb{Y}} \sup_{z\in\mathbb{Z}} \sup_{x\in\mathbb{X}} V(x,y,z) \qquad \sup_{z\in\mathbb{Z}} \sup_{x\in\mathbb{X}} \inf_{y\in\mathbb{Y}} V(x,y,z)$$

Exercise C.5: Lagrange multipliers and minmax

Consider the constrained optimization problem

$$\min_{x\in\mathbb{R}^n} V(x) \qquad \text{subject to } g(x) = 0 \tag{C.30}$$

in which $V : \mathbb{R}^n \to \mathbb{R}$ and $g : \mathbb{R}^n \to \mathbb{R}^m$. Introduce the Lagrange multiplier $\lambda \in \mathbb{R}^m$ and Lagrangian function $L(x,\lambda) = V(x) - \lambda' g(x)$ and consider the following minmax problem

$$\min_{x\in\mathbb{R}^n} \max_{\lambda\in\mathbb{R}^m} L(x,\lambda)$$

Show that if (x_0,λ_0) is a solution to this problem with finite $L(x_0,\lambda_0)$, then x_0 is also a solution to the original constrained optimization (C.30).

Exercise C.6: Dual problems and duality gap

Consider again the constrained optimization problem of Exercise C.5

$$\min_{x\in\mathbb{R}^n} V(x) \qquad \text{subject to } g(x) = 0$$

and its equivalent minmax formulation

$$\min_{x\in\mathbb{R}^n} \max_{\lambda\in\mathbb{R}^m} L(x,\lambda)$$

Switching the order of optimization gives the maxmin version of this problem

$$\max_{\lambda\in\mathbb{R}^m} \min_{x\in\mathbb{R}^n} L(x,\lambda)$$

Next define a new (dual) objective function $q : \mathbb{R}^m \to \mathbb{R}$ as the inner optimization

$$q(\lambda) = \min_{x\in\mathbb{R}^n} L(x,\lambda)$$

Then the maxmin problem can be stated as

$$\max_{\lambda\in\mathbb{R}^m} q(\lambda) \tag{C.31}$$

Problem (C.31) is known as the *dual* of the original problem (C.30), and the original problem (C.30) is then denoted as the *primal* problem in this context (Nocedal and Wright, 2006, p. 343–345), (Boyd and Vandenberghe, 2004, p. 223).

[9] Note that different conventions are in use. Boyd and Vandenberghe (2004, p. 240) say that the player who "goes" *second* has the advantage, meaning that the inner problem is optimized *after* the outer problem has selected a value for its variable. "Going" first or second is not a clear indicator of who has the advantage; *optimizing* first confers the advantage, and provides a natural mnemonic to recall this result.

(a) Show that the solution to the dual problem is a lower bound for the solution to the primal problem

$$\max_{\lambda \in \mathbb{R}^m} q(\lambda) \leq \min_{x \in \mathbb{R}^n} V(x) \qquad \text{subject to } g(x) = 0 \tag{C.32}$$

This property is known as *weak duality* (Nocedal and Wright, 2006, p. 345), (Boyd and Vandenberghe, 2004, p. 225).

(b) The difference between the dual and the primal solutions is known as the duality gap. *Strong duality* is defined as the property that equality is achieved in (C.32) and the duality gap is zero (Boyd and Vandenberghe, 2004, p. 225).

$$\max_{\lambda \in \mathbb{R}^m} q(\lambda) = \min_{x \in \mathbb{R}^n} V(x) \qquad \text{subject to } g(x) = 0 \tag{C.33}$$

Show that strong duality is equivalent to the existence of λ_0 such that

$$\min_{x \in \mathbb{R}^n} V(x) - \lambda_0' g(x) = \min_{x \in \mathbb{R}^n} V(x) \qquad \text{subject to } g(x) = 0 \tag{C.34}$$

Characterize the set of all λ_0 that satisfy this equation.

Exercise C.7: Example with duality gap

Consider the following function and sets (Peressini, Sullivan, and Uhl, Jr., 1988, p. 34)

$$V(x, y) = (y - x^2)(y - 2x^2) \qquad \mathbb{X} = [-1, 1] \qquad \mathbb{Y} = [-1, 1]$$

Make a contour plot of $V(\cdot)$ on $\mathbb{X} \times \mathbb{Y}$ and answer the following question. Which of the following two minmax problems has a nonzero duality gap?

$$\min_{y \in \mathbb{Y}} \max_{x \in \mathbb{X}} V(x, y)$$

$$\min_{x \in \mathbb{X}} \max_{y \in \mathbb{Y}} V(x, y)$$

Notice that the two problems are different because the first one minimizes over y and maximizes over x, and the second one does the reverse.

Exercise C.8: The Heaviside function and inner and outer semicontinuity

Consider the (set-valued) function

$$H(x) = \begin{cases} 0, & x < 0 \\ 1, & x > 0 \end{cases}$$

and you are charged with deciding how to define $H(0)$.

(a) Characterize the choices of set $H(0)$ that make H outer semicontinuous. Justify your answer.

(b) Characterize the choices of set $H(0)$ that make H inner semicontinuous. Justify your answer.

(c) Can you define $H(0)$ so that H is both outer and inner semicontinuous? Explain why or why not.

Bibliography

R. E. Bellman. *Dynamic Programming.* Princeton University Press, Princeton, New Jersey, 1957.

D. P. Bertsekas. *Nonlinear Programming.* Athena Scientific, Belmont, MA, second edition, 1999.

D. P. Bertsekas, A. Nedic, and A. E. Ozdaglar. *Dynamic Programming and Optimal Control.* Athena Scientific, Belmont, MA 02478-0003, USA, 2001.

S. P. Boyd and L. Vandenberghe. *Convex Optimization.* Cambridge University Press, 2004.

F. Clarke, Y. S. Ledyaev, R. J. Stern, and P. R. Wolenski. *Nonsmooth analysis and control theory.* Springer-Verlag, New York, 1998.

J. Nocedal and S. J. Wright. *Numerical Optimization.* Springer, New York, second edition, 2006.

A. L. Peressini, F. E. Sullivan, and J. J. Uhl, Jr. *The Mathematics of Nonlinear Programming.* Springer-Verlag, New York, 1988.

E. Polak. *Optimization: Algorithms and Consistent Approximations.* Springer Verlag, New York, 1997. ISBN 0-387-94971-2.

R. T. Rockafellar and R. J.-B. Wets. *Variational Analysis.* Springer-Verlag, 1998.