UNIVERSITY OF CALIFORNIA
SANTA BARBARA

Event-Based Optimal Control of Neurons

by

Ali Nabi Bidhendi

A Dissertation submitted in partial satisfaction of the
requirements for the degree of
Doctor of Philosophy

in

Mechanical Engineering

Committee in charge:
Professor Jeffrey Moehlis, Chair
Professor Francesco Bullo
Professor Igor Mezic
Professor Andrew Teel

December 2012

The dissertation of Ali Nabi Bidhendi is approved.

Professor Francesco Bullo

Professor Igor Mezic

Professor Andrew Teel

Professor Jeffrey Moehlis, Chair

May 2012

Event-Based Optimal Control of Neurons

Copyright © 2012

by

Ali Nabi Bidhendi

Dedicated to my family and friends

Acknowledgments

First and foremost, I must express my most sincere gratitude to my advisor Professor Jeff Moehlis. It is difficult to imagine a more inspiring, encouraging, understanding, and supportive mentor. I would also like to thank other members of my committee, Professor Francesco Bullo, Professor Igor Mezic, and Professor Andrew Teel, who have generously offered their constructive feedback on my work at various times.

I also thank the exceptional individuals that I had the pleasure to collaborate with throughout my PhD. Specifically, Professor Frederic Gibou, Mohammad Mirzadeh, and Dr. Per Danzl from the UCSB campus, and Dr. Theoden Netoff from the University of Minnesota (UMN) have had a notable impact on my academic development during the past few years. I am also grateful for many insightful discussions with Professor João Hespanha, Farshad Pour Safaei, Meysam Barmi, Hosein Mahjoubi, and Anahita Mirtababaei at UCSB, and Tyler Stigen and Oscar Miranda-Dominguez from UMN.

I recognize that this research would not have been possible without the financial assistance from Institute for Collaborative Biotechnologies (ICB) and the National Science Foundation (NSF).

I also acknowledge present and past members of the Moehlis research group: Lina Kim, Margot Kimura, Allison Kolpas, Per Danzl, Gabor Orosz, Michael Busch, and Louis Van Blarigan.

A very special thanks goes my family and my friends. It would not have been possible to get through the PhD marathon without their help and continuous support. Specifically, I would like to express my gratitude to my parents and my sister for their unconditional love and support, Ralph and Barbara Green for their patience, love, and encouragement, and each and every one of my friends at both UCSB and UMN campuses for creating an environment of support and comfort for me during this time.

Vita of Ali Nabi Bidhendi

May 2012

Education

Ph.D., Mechanical Engineering, 2007-2012

University of California at Santa Barbara, Santa Barbara, California USA
Advisor: Jeff Moehlis
Thesis title: Event-Based Optimal Control of Neurons
GPA: 3.97/4

M.Sc., Mechanical Engineering, 2005-2007

Sharif University of Technology, Tehran, IRAN
Advisor: Mohammad Durali
Thesis title: System Design and Performance Evaluation of Buried Oil and Gas Pipes 3D Map Measurement Systems
GPA: 17.29/20

B.Sc., Mechanical Engineering, 2001-2005

Sharif University of Technology, Tehran, IRAN
Advisor: Saeed Sohrabpour
Thesis title: Design of Hybrid Electric Vehicles
GPA: 17.06/20

Abstract

Event-Based Optimal Control of Neurons

by

Ali Nabi Bidhendi

Motivated by issues related to treating certain neurological diseases such as Parkinson's disease by a method called electrical deep brain stimulation, we consider applying different control methods to both mathematical models of neurons and *in vitro* neurons. Patients suffering from Parkinson's disease experience involuntary tremors that typically affect the distal portion of their upper limbs. It has been hypothesized that these tremors are associated with simultaneous spiking of a cluster of neurons in the thalamus and basal ganglia regions of the brain. In a healthy situation, the periodic firing of neurons is not synchronized, but they can engage in a pathological synchrony and all fire at the same time which results in release of strong action potentials that trigger the downstream muscles with periodic shocks, manifested as tremors.

This dissertation investigates the control of different neuronal systems using methods of optimal control. The neuronal systems considered range from simple one-dimensional phase models to multi-dimensional conductance-based models, both on a single neuron level and on a population level. The optimal control methods considered produce event-based, continuous-time, typically bounded input

stimuli that can optimally achieve the desired control objective. The optimality criteria considered are minimum energy and minimum time. The control objectives of interest are the interspike interval for single neurons and desynchrony for populations of neurons.

There are three parts to this dissertation. In the first part, (Chapters 2 and 3), event-based time optimal and energy optimal control is presented to achieve desired interspike intervals for phase models of single neurons. In the second part, (Chapters 4 and 5), the problem of desynchronizing a network of pathologically synchronized coupled neurons is considered. In the third part, (Chapter 6), the theoretical method of Chapter 3 is adopted and the applicability of this method is shown in practice by testing the controller on *in vitro* pyramidal neurons in the CA1 region of rat hippocampus.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This dissertation investigates optimal control of periodically spiking neurons. Specifically, it considers several model-dependent and model-independent demand-control minimum energy and minimum time stimulation protocols for controlling the spike timing for single neurons and synchrony for populations of neurons. Much of the motivation for controlling neurons comes from the desire to treat certain neurological diseases such as epilepsy, essential tremor, and Parkinson's disease (PD). In PD, which we focus on in this dissertation, the patient experiences involuntary tremors that have been associated with the synchronization of a cluster of neurons in the thalamus and basal ganglia [4].

Patients with advanced PD have been offered partial relief through electrical Deep Brain Stimulation (DBS) [5]. In this FDA-approved surgical procedure, an electrode is implanted into the patient's brain which, in its standard form, delivers a high frequency pulsatile current stimulus that, hypothetically, breaks the

pathological synchrony among the periodically spiking neurons [4, 6, 7]. The input stimulus is generated by a neurostimulator that is implanted in the patient's chest and is powered by batteries. While the standard DBS has shown success in treating PD patients, there are a few issues that are, clinically, of concern. The high frequency nature of the stimulus is thought to have potential side-effects. These pulses are administered at around $130 - 180$ Hz and could potentially result in permanent damage to the tissue (e.g., lesioning) [8]. It is also seen that the brain adapts to the high frequency pulsatile input over time and thus the efficiency of the input drops [9]. Moreover, the accumulated charge that is inputed to the patient's brain is clinically undesirable [10] which has encouraged most of today's neurostimulators to administer a low-amplitude long-duration recharge current after each high-amplitude short-duration stimulation pulse [11]. In addition, if the amount of energy that is used in the standard DBS is reduced, it can lessen the need for battery replacement surgeries. Furthermore, the complexity and the amount of uncertainties that are present in the behavior of neurons reduce even the most comprehensive models to only approximations that, at best, capture the qualitative behavior of these systems. Therefore, while the models create platforms that enable testing of novel ideas, the results and findings that emerge from them may not meet expectations when directly applied to the biological tissue. So, another challenge is to design input stimuli that are not overly dependent on the accuracy of the model used.

To address different aspects of these issues and to find better inputs, different

control approaches have been considered in the literature both on a population level and on a single neuron level. These approaches range from applying different optimization techniques [11–23] in designing stimuli for neuron systems to methods rooted in hybrid control ideas [24–26], chaos theory [27], and delay differential equations [28–30].

The models used in these studies range from simple one-dimensional *phase models* to more complicated multi-dimensional conductance-based compartmental models. While multi-dimensional compartment models capture the spatiotemporal properties of the neurons better, the phase models have proven to be sufficiently accurate with considerably lower computational cost and the advantage of being applicable in practice.

Phase models are one-dimensional reductions of multi-dimensional (conductance-based) models that emerge from performing the so-called *phase reduction* technique on these models (see Appendix A). In a phase model, the oscillatory behavior of the neuron is represented by the evolution of this single phase variable [31–33]. The phase model for a neuron is characterized by its Phase Response Curve (PRC) which is a measure of how sensitive the neuron is to impulsive input stimulus. It characterizes the phase change of the neuron as a function of the phase at which the stimulus is applied. The PRC, and hence the phase model, can be computed for models of neurons by solving the appropriate adjoint equation [32, 34] or obtained for biological neurons by implementing the so-called direct method [33] (see Appendix B). In the direct method, a short-duration pulse (that approximates an

impulse) is injected into the periodically spiking neuron and the resulting advance in spike time is measured and recorded as a function of the phase at which the stimulus has been applied. The advantage of using phase models for controlling neurons extends beyond their computational benefits. By designing control stimuli using a neuron's PRC, one bypasses all the biological complexity of the neuron's dynamics and obtains a control law that is not so much dependent on the biological parameters of the model and can work for any neuron given its PRC. Since PRCs can be experimentally measured for real neurons, this increases the likelihood of the designed control stimuli to be also effective in practice. In this dissertation, we consider designing control stimuli for both phase models and more detailed models of neurons. We also demonstrate the applicability of some of the designed control strategies in practice.

The outline of this dissertation is as follows. In Chapter 2, we investigate the time optimal control problem for phase models of periodically firing neurons, and find analytical expressions for the minimum and maximum values of interspike intervals achievable with small bounded control stimuli. We consider two cases: with a charge-balance constraint on the input, and without it. The clinically desired charge-balance constraint ensures a zero net integral of the input stimulus. The analytical calculations are supported with numerical results for examples of qualitatively different neuron models.

In Chapter 3, we consider the minimum energy control problem for periodically firing phase neurons. The objective in this chapter is to control the interspike

interval of the neuron to a prespecified value with minimum energy, while imposing a charge-balance constraint on the input.

In Chapter 4, we consider the problem of desynchronizing a network of synchronized, globally (all-to-all) coupled neurons using an input to a single neuron. This is done by applying the discrete time dynamic programming method to reduced phase models for neural populations. This technique numerically minimizes a certain cost function over the whole state space. We evaluate the effectiveness of control inputs obtained by averaging over results obtained for different coupling strengths.

In Chapter 5, we employ optimal control theory to design an event-based, minimum energy, desynchronizing control stimulus for a network of pathologically synchronized, heterogeneously coupled neurons. This works by optimally driving the neurons to their so-called *phaseless sets*, switching the control off, and letting the phases of the neurons randomize under intrinsic background noise. For the example considered, it is shown that the proposed control causes a considerable amount of randomization in the timing of each neuron's next spike, leading to desynchronization for the network.

In Chapter 6, we demonstrate the applicability of optimal control theory for designing efficient input stimulus waveforms for single periodically-firing *in vitro* neurons from brain slices of long-evans rats. The method of control presented here is similar to that presented in Chapter 3, which uses the phase model of a neuron and does not require prior knowledge of the neuron's biological details. The PRC

for each neuron is experimentally obtained using the direct method. Based on the measured PRC, continuous-time, charge-balanced, minimum energy control waveforms are designed that apply at the onset of an action potential and can optimally change the next spike time for the neuron. The significance of this work is that it combines electrophysiology experiments with optimal control theory to achieve control waveforms for *in vitro* neurons with levels of energy that are in some cases multiple orders of magnitude smaller than those of similar past studies.

Publications

This dissertation is supported by- and covers most of- the material originally documented in the following peer reviewed publications:

- A. Nabi, T. Stigen, J. Moehlis, and T. Netoff, Minimum Energy Control for *in vitro* Neurons, In preparation.
- A. Nabi, M. Mirzadeh, F. Gibou, and J. Moehlis, Minimum Energy Desynchronizing Control for Coupled Neurons, Journal of Computational Neuroscience, doi: 10.1007/s10827-012-0419-3.
- A. Nabi and J. Moehlis, Time Optimal Control of Spiking Neurons, Journal of Mathematical Biology, Vol. 64, 981-1004, doi:10.1007/s00285-011-0441-5, 2011.
- A. Nabi and J. Moehlis, Single Input Optimal Control for Globally Coupled Neuron Networks, Journal of Neural Engineering, Vol. 8, 065008, doi:10.1088/1741-2560/8/6/065008, 2011.
- P. Danzl, A. Nabi, and J. Moehlis, Charge-Balanced Spike Timing Control for Phase Models of Spiking Neurons, Discrete and Continuous Dynamical Systems, Vol. 28, No. 4, 1413-1435, doi:10.3934/dcds.2010.28.1413, 2010.
- A. Nabi, M. Mirzadeh, F. Gibou, and J. Moehlis, Minimum Energy Spike Randomization Control for Neurons, Proceedings of the 2011 American Control Conference, June 2011, Montreal, Canada, pp. 4751-4756.

- A. Nabi and J. Moehlis, Nonlinear Hybrid Control of Phase Models for Coupled Oscillators, Proceedings of 2010 American Control Conference, July 2010, Baltimore, MD, USA, pp. 922-923.
- A. Nabi and J. Moehlis, Charge-Balanced Optimal Inputs for Phase Models of Spiking Neurons, Proceedings of 2009 ASME Dynamic Systems and Control Conference, October 2009, Hollywood, CA, USA, DSCC2009-2541.

Chapter 2

Time Optimal Control of Spiking Neurons

In this chapter, we investigate the time optimal control for a single neuron described by a phase model, and in particular find the extreme values of the next spiking time when the input is constrained between prespecified upper and lower bounds. In such time optimal control problems, which give what is known as bang-bang control, the objective is to find an input that would take the system to the target point in minimum or maximum time, without any constraint on the amount of energy used. From this analysis, one can gain insight about the maximum capability of single cell treatment procedures when input stimuli are bounded. The main focus here will be on charge-balanced input stimuli.

The organization of this chapter is as follows. In Section 2.1, we present a brief introduction to the problem. In Section 2.2, we first introduce the dynamic phase

equation and then develop and provide the solution to the time optimal control problem formulation in its general format. In Section 2.3, we present four different neural models and solve the control problem for each of these models. We give detailed analytical results for three of these models, and numerical results for all four. We summarize and discuss the results in Section 2.4.

The main results from this chapter were published in [20].

2.1 Introduction

Phase models of neurons have been used to investigate the patterns of synchrony that result from the type and architecture of coupling [35–41], and the response of large groups of oscillators to external stimuli [32, 42–44]. Phase models of neurons have also been employed in the context of controlling neurons to give a prespecified behavior [16, 19–21, 26, 43, 45].

An area of recent research interest has been to find control stimuli, in the form of electrical currents, that can effectively break this pathological synchrony; see, for example, [18, 22, 27, 43]. On a single neuron level, this objective reduces to controlling the spiking time of a neuron. Here, upon detection of a voltage spike (or a firing event), a precomputed input stimulus is injected into the neural system that would shift the next spike time to a certain prespecified value. This input stimulus may be computed under different criteria and/or constraints. By considering a time optimal control problem, one can gain insight about the maximum capability of bounded control in changing a neuron's interspike interval.

One relevant constraint that one can impose on the control stimulus is the charge-balance constraint. The charge-balance constraint ensures that the total electrical charge that is transferred to the neural tissue is zero over the course of one cycle of control input. This is important to prevent neural tissue damage. In practice, applying charge-imbalanced inputs, especially uni-sign inputs, to the neuron causes irreversible Faradaic chemical reactions to take place in the environment immediate to the electrode, which could result in permanent damage to the tissue. With charge-balanced inputs, one prevents this damage, although this could potentially cause corrosive damage to the electrode, something that can be mitigated by modifying the material and design of the electrode [10].

2.2 Model Equations

A periodically firing or spiking neuron can be considered to be a periodic oscillator with the general dynamical equation [16, 32, 46]

$$\frac{d\theta}{dt} = f(\theta) + Z(\theta)u(t). \tag{2.1}$$

This equation is referred to as the phase model for the neuron. Here, $f(\theta)$ represents the neuron's baseline dynamics, $Z(\theta)$ is the Phase Response Curve (PRC) of the neuron, and $u(t)$ is the control input which is an electrical current divided by the capacitance of the neural membrane [16, 32]. $\theta(t) \in \mathbb{R}^{\geq 0}$, is the neuron's phase, where by convention $\{\theta | \theta \ mod \ 2\pi = 0\}$ corresponds to the spiking of the neuron. We note that the typical definition of phase for neurons is such that for realistic neuron

models it yields $f(\theta) = \omega =$ constant. However, there are models in the literature (like the theta neuron model that we consider later) that have non-constant $f(\theta)$ functions; in the regime where the neuron fires periodically, we have $f(\theta) > 0$. The PRC for a neuron characterizes a measure of how sensitive the phase of the neuron is to external stimuli. The timing of the external input plays an important role in the amount of phase shift in the neuron. It is very unlikely to have a case where the neuron would be insensitive to the time of the input stimulus. This means that the PRC does not usually have a constant flat part.

For intrinsically oscillatory neurons with $u(t) = 0$, the neuron would fire (or spike) at its natural period T, determined by $f(\theta)$. Without loss of generality, we assume that the initial time of firing is $t = 0$. By inputting a control stimulus $u(t)$, the next firing time of the neuron, or the Inter-Spike Interval (ISI), can be adjusted to a desired target time $t_1 \neq T$. However, we note that $|u(t)|$ must be sufficiently small for the phase model to remain valid. Strictly speaking, the phase reduction assumes that $|u(t)|$ is infinitesimal, but in practice (2.1) is a good model for small inputs. We will only consider inputs which are small enough that $\dot{\theta} > 0$ for all times; we view this as a necessary condition for the validity of the model. As is typical for realistic neuron models, we also assume that $Z(\theta)$ has isolated roots and furthermore, we assume that there does not exist $\tau_1, \tau_2 : 0 \leq \tau_1 < \tau_2 \leq t_1$ such that $Z(\theta(t))f'(\theta(t)) = Z'(\theta(t))f(\theta(t)),\ \forall t \in [\tau_1, \tau_2]$. In the common case where $f(\theta) = \omega > 0$, this means that the PRC should not be constant on an interval which is true for almost all realistic neuron models. We use this assumption later

in Lemma 2.2.1 to prove that a particular function of $Z(\theta)$ that is of interest also has isolated roots. We note that this assumption is valid for most realistic neuron models in the literature.

The objective here is to find the control input $u(t)$ that, when bounded to be less than a certain value $\bar{u}$ in magnitude, i.e. $|u(t)| \leq \bar{u}$, would result in the minimum/maximum value of t_1. This is an optimization problem in which the next spike time t_1 needs to be extremized. This yields $C(t_1) = \int_0^{t_1} 1\, dt$ as the cost function for this system. We will solve this problem for two different cases: with and without a charge-balance constraint imposed on the control input. The charge-balance constraint can be mathematically expressed as $\int_0^{t_1} u(t)dt = 0$. In order to simplify the upcoming calculations, we restate this constraint as follows.

Let $\dot{q} = u(t)$. Integrating both sides of this equation from 0 to t_1, we obtain

$$q(t_1) - q(0) = \int_0^{t_1} u(\tau)d\tau.$$

For the charge-balance constraint to hold we need the righthand side of this equation to be zero. This means $q(t_1) = q(0)$, and assuming that the input is being applied from time $t = 0$, which implies $q(0) = 0$, we have $q(t_1) = q(0) = 0$.

Summarizing, we seek a control input $u(t)$, which extremizes

$$C(t_1) = \int_0^{t_1} 1\ dt,$$

with the following constraints:

$$\begin{aligned} \dot{\theta} &= f(\theta) + Z(\theta)u(t), \qquad \theta(0) = 0, \quad \theta(t_1) = 2\pi, \\ \dot{q} &= u(t), \quad |u(t)| \leq \bar{u}, \qquad q(0) = 0, \quad q(t_1) = 0. \end{aligned} \tag{2.2}$$

The Hamiltonian associated with this system is

$$H(\theta, q, \lambda_1, \lambda_2, u) = 1 + \lambda_1(f(\theta) + Z(\theta)u(t)) + \lambda_2 u(t), \tag{2.3}$$

where λ_1 and λ_2 are the Lagrange multipliers or the co-states for this system. To obtain the necessary conditions for optimality one can use the Hamiltonian in (2.3) and write

$$\dot{\theta} = \frac{\partial H}{\partial \lambda_1} \quad \Rightarrow \quad \dot{\theta} = f(\theta) + Z(\theta)u(t), \tag{2.4}$$

$$\dot{\lambda}_1 = -\frac{\partial H}{\partial \theta} \quad \Rightarrow \quad \dot{\lambda}_1 = -\lambda_1(f'(\theta) + Z'(\theta)u(t)), \tag{2.5}$$

$$\dot{q} = \frac{\partial H}{\partial \lambda_2} \quad \Rightarrow \quad \dot{q} = u(t), \tag{2.6}$$

$$\dot{\lambda}_2 = -\frac{\partial H}{\partial q} \quad \Rightarrow \quad \dot{\lambda}_2 = 0, \tag{2.7}$$

where prime represents differentiation with respect to θ [47, 48].

The optimal control for this problem is obtained from Pontryagin's minimum principle [47, 48] as

$$u^*(t) = \arg \mathcal{M}_{|u(t)| \leq \bar{u}} \left(1 + \lambda_1^* \left(f(\theta^*) + Z(\theta^*)u(t)\right) + \lambda_2^* u(t)\right),$$

where $\mathcal{M} \in \{\min, \max\}$. This yields the following equations for the optimal control input, $u^*(t)$, for the cases of minimizing the ISI (or t_1) of the neuron and maximizing it:

$$u^*(t) = -\text{sign}[\lambda_1^* Z(\theta^*) + \lambda_2^*]\bar{u} \quad \text{for the min. problem,} \tag{2.8}$$

$$u^*(t) = +\text{sign}[\lambda_1^* Z(\theta^*) + \lambda_2^*]\bar{u} \quad \text{for the max. problem.} \tag{2.9}$$

The star superscript indicates the optimal trajectories or functions. Equations (2.8) and (2.9) indicate that the magnitude of the optimal control is always equal to its

bound and that only its sign changes with respect to time. This solution, known as bang-bang control, is expected since the objective here is to achieve extreme final time, and thus one expects maximum effort from the control stimulus. These equations hold because $(\lambda_1 Z(\theta) + \lambda_2)$ only has isolated roots, as follows:

Lemma 2.2.1 *Suppose $Z(\theta)$ has isolated roots and that there does not exist $\tau_1, \tau_2 : 0 \leq \tau_1 < \tau_2 \leq t_1$ such that $Z(\theta(t))f'(\theta(t)) = Z'(\theta(t))f(\theta(t))$, $\forall t \in [\tau_1, \tau_2]$. Then, the roots of $\lambda_1 Z(\theta) + \lambda_2$ are isolated.*

Proof By contradiction, assume that $\exists\, \tau_1, \tau_2 : 0 \leq \tau_1 < \tau_2 \leq t_1$ such that $\lambda_1 Z(\theta) + \lambda_2 \equiv 0$ $\forall t \in [\tau_1, \tau_2]$. Taking the derivative of this with respect to time, one gets $\dot{\lambda}_1 Z + \lambda_1 Z' \dot{\theta} \equiv 0$. Substituting for $\dot{\lambda}_1$ from (2.5) and $\dot{\theta}$ from (2.4), dividing both sides by λ_1, and simplifying the results, we get $Zf' = Z'f$ which contradicts the prior assumptions.

Minimizing the ISI corresponds to speeding up the neuron dynamics, whereas maximizing it corresponds to slowing it down. Equations (2.4)-(2.7) are ordinary differential equations (ODEs) that need to be solved in order to evaluate the optimal control equations (2.8) and (2.9). When solving the system, we have these four ODEs along with one of the last two algebraic equations for $u^*(t)$, depending on which optimization problem is being considered. This makes five equations with six unknowns: $\theta^*(t)$, $\lambda_1^*(t)$, $q^*(t)$, $\lambda_2^*(t)$, $u^*(t)$, and the next spike time t_1. So we need one more equation to be able to solve this problem. This sixth equation is obtained from the fact that in the absence of any end point cost on the states of the system,

the final value of the Hamiltonian evaluated along the optimal trajectories needs to be zero, regardless of whether the minimization problem or the maximization problem is considered [47,48]. This can be stated mathematically as

$$H(\theta^*(t_1), \lambda_1^*(t_1), q^*(t_1), \lambda_2^*(t_1), u^*(t_1)) = 0. \tag{2.10}$$

Equations (2.4)-(2.8)/(2.9) together with (2.10) comprise a two point boundary value problem (TPBVP) where the boundary values for $\theta(t)$ and $q(t)$ are given in (2.2). We note that since (2.4)-(2.7) are a Hamiltonian system, (2.10) holds for all $t \in [0\,,\, t_1]$. The total input energy associated with the optimal control can be obtained by

$$E(u^*, t_1) = \int_0^{t_1} [u^*(t)]^2 dt = t_1 \bar{u}^2.$$

In order to solve this problem we substitute (2.8) and (2.9) into (2.4) and (2.5) to get

$$\dot{\theta} = f(\theta) \mp Z(\theta)\text{sign}[\lambda_1 Z(\theta) + \lambda_2]\bar{u}, \tag{2.11}$$

$$\dot{\lambda}_1 = -\lambda_1 \left(f'(\theta) \mp Z'(\theta)\text{sign}[\lambda_1 Z(\theta) + \lambda_2]\bar{u}\right), \tag{2.12}$$

where, here and elsewhere, the top signs are for the minimization problem and the bottom signs are for the maximization problem. In the minimization (resp., maximization) problem, the average angular velocity of the system with external stimulus has to be larger (resp., smaller) than that of the system running without any external stimuli. In other words, when the system is stimulated, $\theta(t)$ goes from zero to 2π in time $t_1 < T$ (resp., $t_1 > T$), where T is the natural period of the system without any stimuli. So if we integrate both the stimulated and the unstimulated

systems from $t = 0$ to $t = t_1$, while the stimulated system reaches $\theta(t_1) = 2\pi$, the unstimulated system reaches $\theta(t_1) < 2\pi$ in the minimization problem and $\theta(t_1) > 2\pi$ in the maximization problem. Therefore, from (2.11) we can write

$$\frac{1}{t_1}\int_0^{t_1} \{f(\theta) - Z(\theta)\,\mathrm{sign}[\lambda_1 Z(\theta) + \lambda_2]\,\bar{u}\}\,dt > \frac{1}{t_1}\int_0^{t_1} f(\theta)\,dt \geq 0$$

for the minimization problem, and

$$0 \leq \frac{1}{t_1}\int_0^{t_1} \{f(\theta) + Z(\theta)\mathrm{sign}[\lambda_1 Z(\theta) + \lambda_2]\,\bar{u}\}\,dt < \frac{1}{t_1}\int_0^{t_1} f(\theta)\,dt$$

for the maximization problem. In the limit as $\bar{u} \to 0$, i.e., the external stimulus vanishes, $t_1 \to T$ and the strict inequalities become equalities. For $\bar{u} \neq 0$, these equations yield

$$\int_0^{t_1} Z(\theta)\mathrm{sign}[\lambda_1 Z(\theta) + \lambda_2]dt \leq 0. \tag{2.13}$$

When the charge-balance constraint is imposed, we require the following relationship to hold as well:

$$\begin{aligned} &\textstyle\int_0^{t_1} u(t)\,dt = \mp\bar{u}\int_0^{t_1} \mathrm{sign}[\lambda_1 Z(\theta) + \lambda_2]dt = 0 \\ &\Rightarrow \quad \textstyle\int_0^{t_1} \mathrm{sign}[\lambda_1 Z(\theta) + \lambda_2]dt = 0. \end{aligned} \tag{2.14}$$

Also, equations (2.3) and (2.10) in this case yield

$$\lambda_1(t_1) = -\frac{1 + \lambda_2 u(t_1)}{f(2\pi) + Z(2\pi)u(t_1)}. \tag{2.15}$$

When the charge-balance constraint is not imposed, λ_2 and q are eliminated from the system equations. Then (2.13) becomes

$$\int_0^{t_1} Z(\theta)\text{sign}[\lambda_1 Z(\theta)]dt \leq 0. \tag{2.16}$$

We note that, from (2.5), $\lambda_1 = 0$ defines an invariant surface, so if $\lambda_1(0) < 0$ then λ_1 will remain negative, or if $\lambda_1(0) > 0$ then λ_1 will remain positive, i.e., λ_1 is a uni-sign function. Therefore, in (2.16), $\text{sign}[\lambda_1 Z(\theta)]$ is either $+\text{sign}[Z(\theta)]$ or $-\text{sign}[Z(\theta)]$, where the plus and the minus signs represent the sign of $\lambda_1(t)$. In order to have (2.16) satisfied, necessarily, we must have $\lambda_1(t) < 0$ for all t. This simplifies (2.8) and (2.9) to

$$u^*(t) = \pm\text{sign}\left[Z[\theta^*(t)]\right]\bar{u}, \tag{2.17}$$

for the case where charge-balance is not considered. This is in accordance with the argument made in [16] which shows that the time optimal control (2.17) extremizes the righthand side of (2.1) and so is a sufficient condition for achieving the extremum of ISI.

2.3 Examples

We now solve this time optimal control problem for examples of Type I and Type II neurons [49]. Type I neurons are those that have a non-negative PRC for all phases, i.e., $Z(\theta) \geq 0$ for all θ. For these neurons, any positive (resp., negative) impulsive input stimulus will advance (resp., retard) the phase regardless of the time at which the stimulus is applied. For Type II neurons, on the other hand,

the PRC takes both positive and negative values. Therefore, an impulsive input stimulus could advance or retard the phase depending on the time at which it is applied. The PRC for a neuron can be obtained experimentally, numerically, or in some cases, analytically.

Specifically, we consider four different neuron models: the SNIPER model, theta neuron model, sinusoidal model, and phase-reduced Hodgkin-Huxley model. The SNIPER model arises when the periodic orbit corresponding to periodic firing of the neuron comes from a Saddle-Node bifurcation of two fixed points on an Infinite PERiod orbit (SNIPER bifurcation). Close to the bifurcation point one can analytically approximate the associated PRC; since this turns out to be non-negative, it is a Type I PRC [32,34]. The theta neuron model generalizes the SNIPER model to include the non-oscillatory regime [34]. On the other hand, if the bifurcation which gives the periodic orbit corresponding to periodic firing of the neuron is a saddle-node bifurcation of periodic orbits, then one can approximate the PRC as sinusoidal, giving the sinusoidal model [32]; this corresponds to a Type II PRC. Finally, the phase-reduced Hodgkin-Huxley model uses the PRC for the Hodgkin-Huxley equations calculated numerically using XPPAUT (available as open source software [1]), which solves the appropriate adjoint equations [2,32].

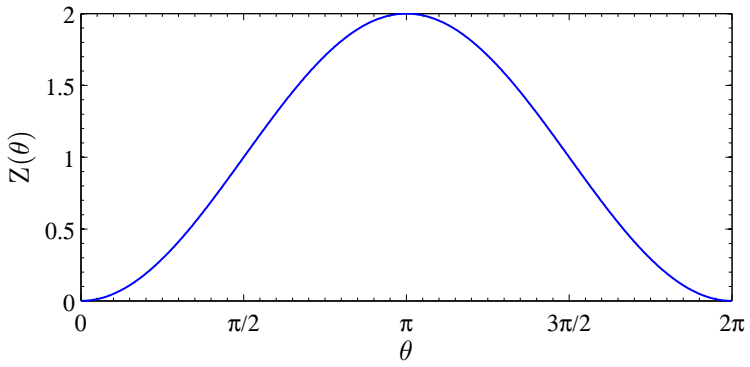


Figure 2.1: SNIPER PRC with $Z_d = 1$

2.3.1 SNIPER Model

In this model, $Z(\theta) = Z_d(1 - \cos(\theta))$ (see Figure 2.1) and $f(\theta) = \omega = \text{constant}$, where $Z_d > 0$ is a constant [32, 34]. Equations (2.11) and (2.12) become

$$\dot{\theta} = \omega \mp Z_d(1 - \cos(\theta))\text{sign}[\lambda_1 Z_d(1 - \cos(\theta)) + \lambda_2]\bar{u}, \tag{2.18}$$

$$\dot{\lambda}_1 = \pm\lambda_1 Z_d \sin(\theta)\text{sign}[\lambda_1 Z_d(1 - \cos(\theta)) + \lambda_2]\bar{u}, \tag{2.19}$$

where, as before, the signs on top are for the minimization problem, and those on the bottom are for the maximization problem. We consider two cases, namely, with the charge-balance constraint and without it.

With Charge-Balance Constraint: When the charge-balance constraint (2.14) is imposed, equation (2.13) becomes

$$\int_0^{t_1} \cos(\theta)\text{sign}[\lambda_1 Z_d(1 - \cos(\theta)) + \lambda_2]\, dt > 0. \tag{2.20}$$

The solution that satisfies (2.20) and maximizes (resp., minimizes) the average

angular velocity in the righthand side of (2.18) subject to (2.14), is one that the function sign[·] varies according to the following:

$$\begin{aligned} \text{sign}[\lambda_1(1-\cos(\theta))+\lambda_2] &= -1, \qquad \tfrac{t_1}{4} \le t \le \tfrac{3t_1}{4}, \\ \text{sign}[\lambda_1(1-\cos(\theta))+\lambda_2] &= \ \ 1, \qquad \text{otherwise.} \end{aligned} \tag{2.21}$$

To see this, we point out that maximizing (resp., minimizing) the average angular velocity in (2.18) amounts to minimizing the function $\int_0^{t_1}(1-\cos(\theta))\text{sign}[\cdot]dt$. Considering the graph of $(1-\cos(\theta))$ in Figure 2.1, it is readily inferred that the value of this integral is most effectively reduced if the sign[·] function is -1 when $(1-\cos(\theta))$ has largest area under curve, that is for intermediate values of θ. Since we also require (2.14), the sign[·] function has to change as indicated in (2.21) to achieve most effective minimization. Note that in the presence of external stimuli, even though $\theta(0)=0$ and $\theta(t_1)=2\pi$, but $\theta(\frac{t_1}{4}) \neq \frac{\pi}{2}$ and $\theta(\frac{3t_1}{4}) \neq \frac{3\pi}{2}$ as shown in Figure 2.2. Equations (2.21) yield

$$\begin{aligned} u^*(t) &= \pm\bar{u}, \qquad \tfrac{t_1}{4} \le t \le \tfrac{3t_1}{4}, \\ u^*(t) &= \mp\bar{u}, \qquad \text{otherwise.} \end{aligned} \tag{2.22}$$

From (2.21), (2.18) and (2.19) become

$$\begin{aligned} \dot{\theta} &= \omega \pm Z_d(1-\cos(\theta))\bar{u}, \qquad \tfrac{t_1}{4} \le t \le \tfrac{3t_1}{4}, \\ \dot{\theta} &= \omega \mp Z_d(1-\cos(\theta))\bar{u}, \qquad \text{otherwise,} \end{aligned} \tag{2.23}$$

$$\begin{aligned} \dot{\lambda}_1 &= \mp\lambda_1 Z_d \sin(\theta)\bar{u} \qquad \tfrac{t_1}{4} \le t \le \tfrac{3t_1}{4}, \\ \dot{\lambda}_1 &= \pm\lambda_1 Z_d \sin(\theta)\bar{u} \qquad \text{otherwise.} \end{aligned} \tag{2.24}$$

These equations are symmetric about $t = \frac{t_1}{2}$, meaning that the (θ, λ_1) trajectories take on identical values for $t = \frac{t_1}{2} - \delta$ and $t = \frac{t_1}{2} + \delta$ for all $\delta \in [0\,,\, \frac{t_1}{2}]$, which implies $\theta(\frac{t_1}{2}) = \pi$ and $\lambda_1(0) = \lambda_1(t_1)$. Since $\lambda_1 = 0$ is an invariant set, λ_1 is either always positive or always negative. In order to determine the sign of λ_1 we note that we want to satisfy (2.21). Considering the shape of the PRC function in Figure 2.1, one can verify that the only way to arrive at (2.21) is to have a negative $\lambda_1(t)$ to flip and scale $(1 - \cos(\theta))$ function and a positive scalar value λ_2 to shift the product $\lambda_1(1 - \cos(\theta))$ up along the vertical axis just enough for (2.21) to be satisfied. This amounts to having $\lambda_1(0) < 0$ with $\lambda_2 = -\lambda_1(\frac{t_1}{4})Z_d(1 - \cos(\theta_a)) > 0$, where $\theta_a = \theta(\frac{t_1}{4})$, for both the minimization and maximization problems.

From (2.15), we get

$$\lambda_1(0) = -\frac{1}{\omega}(1 \mp \lambda_2 \bar{u}), \tag{2.25}$$

where we have used the facts that $Z(2\pi) = 0$ and $\lambda_1(0) = \lambda_1(t_1)$. In order to have $\lambda_1(0) < 0$, from (2.25), we conclude that we must have $\lambda_2 < \frac{1}{\bar{u}}$, which gives an upper bound for λ_2 for the minimization problem.

By symmetry, $\theta(\frac{t_1}{2}) = \pi$. Thus, from (2.23), provided $\dot{\theta} > 0$ for all times,

$$\int_0^{t_1/2} dt = \int_0^{\theta_a} \frac{d\theta}{\omega \mp Z_d(1 - \cos(\theta))\bar{u}} + \int_{\theta_a}^{\pi} \frac{d\theta}{\omega \pm Z_d(1 - \cos(\theta))\bar{u}}, \tag{2.26}$$

where $\theta_a = \theta(\frac{t_1}{4})$. In order to solve equation (2.26) in terms of t_1, first θ_a needs to be determined. We realize that by construction, the first integral in (2.26) is valid for $t \in [0\,,\, \frac{t_1}{4})$ and the second integral is valid for $t \in [\frac{t_1}{4}\,,\, \frac{t_1}{2}]$. So solving the first

integral in (2.26), setting the result equal to $\frac{t_1}{4}$ in the limit, and solving for θ_a, one gets

$$\begin{aligned}
\theta_a &= 2\arctan\left[\sqrt{\tfrac{\omega}{\omega-2Z_d\bar{u}}}\,\tan\left(\tfrac{t_1}{8}\sqrt{\omega(\omega-2Z_d\bar{u})}\right)\right], && 0 \le \bar{u} < \tfrac{\omega}{2Z_d} \\
\theta_a &= 2\arctan\left[\tfrac{t_1}{8}\omega\right], && \bar{u} = \tfrac{\omega}{2Z_d} \\
\theta_a &= 2\arctan\left[\sqrt{\tfrac{\omega}{2Z_d\bar{u}-\omega}}\,\tanh\left(\tfrac{t_1}{8}\sqrt{\omega(2Z_d\bar{u}-\omega)}\right)\right], && \bar{u} > \tfrac{\omega}{2Z_d}
\end{aligned}$$

for the minimization problem and

$$\theta_a = 2\arctan\left[\sqrt{\tfrac{\omega}{2Z_d\bar{u}+\omega}}\,\tan\left(\tfrac{t_1}{8}\sqrt{\omega(2Z_d\bar{u}+\omega)}\right)\right], \qquad 0 \le \bar{u} < \tfrac{\omega}{2Z_d}$$

for the maximization problem. In both of these cases, when $\bar{u} \equiv 0$, $t_1 = T = \frac{2\pi}{\omega}$ and $\theta_a = \frac{\pi}{2}$. However, when $\bar{u} \neq 0$, $0 < \theta_a < \frac{\pi}{2}$ for the minimization problem and $\frac{\pi}{2} < \theta_a < \pi$ for the maximization problem. Figure 2.2 shows the evolution of θ in time for when $\bar{u} \equiv 0$ (intrinsic firing) and for when $\bar{u} = 0.2$ for both the minimization and the maximization problems. Note that the time axis is scaled so that 0, $\frac{\pi}{2}$, π, $\frac{3\pi}{2}$, and 2π points on the horizontal axis correspond to $t = 0$, $\frac{t_1}{4}$, $\frac{t_1}{2}$, $\frac{3t_1}{4}$, and t_1, respectively.

Now solving the second integral in (2.26) with these θ_a values results in the following implicit expressions for t_1:

$$\begin{aligned}
\tfrac{t_1}{8}\sqrt{\omega(2Z_d\bar{u}+\omega)} &= \tfrac{\pi}{2} - \arctan\left[\sqrt{\tfrac{2Z_d\bar{u}+\omega}{\omega-2Z_d\bar{u}}}\tan\left(\tfrac{t_1}{8}\sqrt{\omega(\omega-2Z_d\bar{u})}\right)\right], && 0 \le \bar{u} < \tfrac{\omega}{2Z_d} \\
\tfrac{\sqrt{2}}{8}t_1\omega &= \tfrac{\pi}{2} - \arctan\left[\tfrac{\sqrt{2}}{8}t_1\omega\right], && \bar{u} = \tfrac{\omega}{2Z_d} \\
\tfrac{t_1}{8}\sqrt{\omega(2Z_d\bar{u}+\omega)} &= \tfrac{\pi}{2} - \arctan\left[\sqrt{\tfrac{2Z_d\bar{u}+\omega}{2Z_d\bar{u}-\omega}}\tanh\left(\tfrac{t_1}{8}\sqrt{\omega(2Z_d\bar{u}-\omega)}\right)\right], && \bar{u} > \tfrac{\omega}{2Z_d}
\end{aligned} \tag{2.27}$$

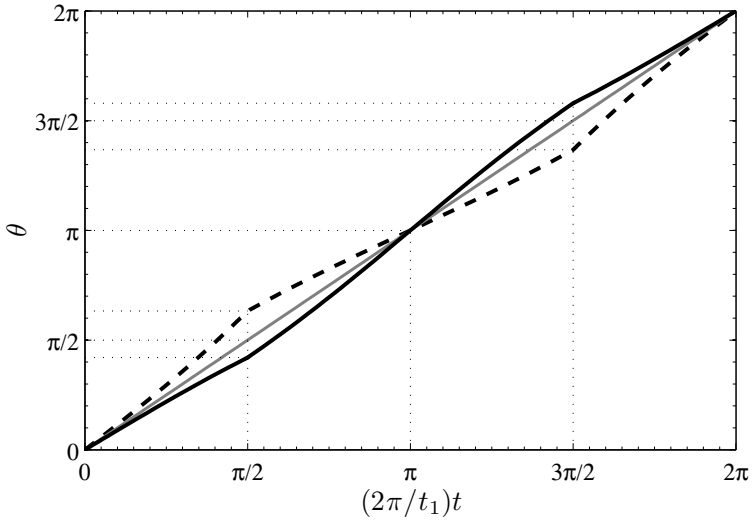


Figure 2.2: Evolution of θ in time for $\bar{u} \equiv 0$ (solid gray line) corresponding to the neuron's intrinsic firing and $\bar{u} = 0.2$ for both the minimization (solid black line) and the maximization (dashed line) problems. We see that at $t = \frac{t_1}{4}$, marked as $\pi/2$ on the scaled horizontal axis, when $\bar{u} \equiv 0$, $\theta(\frac{t_1}{4}) = \frac{\pi}{2}$, when $\bar{u} \neq 0$, for the minimization problem $\theta(\frac{t_1}{4}) < \frac{\pi}{2}$ and for the maximization problem $\theta(\frac{t_1}{4}) > \frac{\pi}{2}$.

for the minimization problem and

$$\frac{t_1}{8}\sqrt{\omega(\omega - 2Z_d\bar{u})} = \frac{\pi}{2} - \arctan\left[\sqrt{\frac{\omega-2Z_d\bar{u}}{\omega+2Z_d\bar{u}}}\tan\left(\frac{t_1}{8}\sqrt{\omega(\omega+2Z_d\bar{u})}\right)\right], \quad 0 \leq \bar{u} < \frac{\omega}{2Z_d} \tag{2.28}$$

for the maximization problem.

In order to find an explicit formula for t_1, we Taylor expand (2.27) and (2.28) for small $\bar{u}$ and solve for t_1 to obtain

$$t_1 = \frac{2\pi}{\omega} \mp \frac{4Z_d}{\omega^2}\bar{u} + \frac{\pi Z_d^2}{\omega^3}\bar{u}^2 + \mathcal{O}(\bar{u}^3), \tag{2.29}$$

where the top sign is for the minimization problem and the bottom sign for the maximization problem.

It is worth pointing out that in writing (2.26), we have assumed that $\dot{\theta} > 0$ for all times. Considering (2.23), we see that for the minimization problem, $\dot{\theta} > 0$ implies that when $t \notin [\frac{t_1}{4}, \frac{3t_1}{4}]$, we must have $\bar{u} < \frac{\omega}{Z_d(1-\cos(\theta_a))}$ for all $\bar{u}$. Lemmas 2.3.1 and 2.3.2 prove that regardless of $\bar{u}$ this statement, which is equivalent to saying $\dot{\theta} > 0$, is always true, although we recall that the phase model is only valid for sufficiently small $\bar{u}$. For the maximization problem, when $t \notin [\frac{t_1}{4}, \frac{3t_1}{4}]$, $\dot{\theta}$ is positive, but when $t \in [\frac{t_1}{4}, \frac{3t_1}{4}]$, $\dot{\theta}$ would be positive provided $\bar{u} < \frac{\omega}{Z_d(1-\cos(\pi))} = \frac{\omega}{2Z_d}$.

Lemma 2.3.1 *For the SNIPER model, for* $\omega > 0$, $\bar{u} > \frac{\omega}{Z_d}$, *and* $0 < \theta < \frac{\pi}{2}$,

$$\omega - Z_d(1-\cos(\theta))\bar{u} > 0 \quad \Leftrightarrow \quad 0 < \tan\left(\frac{\theta}{2}\right) < \sqrt{\frac{\omega}{2Z_d\bar{u} - \omega}}. \tag{2.30}$$

Proof Given the assumptions of the Lemma, one can write

$$\omega - Z_d(1-\cos(\theta))\bar{u} > 0 \quad \Leftrightarrow \quad \cos(\theta) > 1 - \frac{\omega}{Z_d\bar{u}} \quad \Leftrightarrow \quad \theta < \arccos\left(1 - \frac{\omega}{Z_d\bar{u}}\right).$$

Let $\alpha = \arccos(1 - \frac{\omega}{Z_d\bar{u}}) < \frac{\pi}{2}$, then

$$\cos^2\left(\frac{\alpha}{2}\right) = \frac{1}{2}\left(1 + \cos(\alpha)\right) = \frac{1}{2}\left(1 + 1 - \frac{\omega}{Z_d\bar{u}}\right) = 1 - \frac{\omega}{2Z_d\bar{u}},$$
$$\sin^2\left(\frac{\alpha}{2}\right) = 1 - \cos^2\left(\frac{\alpha}{2}\right) = \frac{\omega}{2Z_d\bar{u}},$$
$$\tan\left(\frac{\alpha}{2}\right) = \sqrt{\frac{\omega}{2Z_d\bar{u} - \omega}}.$$

Now one can write

$$\begin{aligned} & 0 < \theta < \alpha < \frac{\pi}{2}, \\ \Leftrightarrow \quad & 0 < \frac{\theta}{2} < \frac{\alpha}{2} < \frac{\pi}{4}, \\ \Leftrightarrow \quad & 0 \leq \tan\left(\frac{\theta}{2}\right) < \tan\left(\frac{\alpha}{2}\right) = \sqrt{\frac{\omega}{2Z_d\bar{u} - \omega}} < 1, \end{aligned}$$

so, (2.30) holds.

Lemma 2.3.2 *Consider (2.23). For the minimization problem, we have*

$$\dot{\theta} > 0 \quad \forall \bar{u} \quad \text{for} \quad 0 \leq t \leq \frac{t_1}{4} \quad \text{and} \quad \frac{3t_1}{4} \leq t \leq t_1. \tag{2.31}$$

Proof First consider $0 \leq t \leq \frac{t_1}{4}$. We consider three cases: $\bar{u} < \frac{\omega}{Z_d}$, $\bar{u} = \frac{\omega}{Z_d}$, and $\bar{u} > \frac{\omega}{Z_d}$.

If $\bar{u} < \frac{\omega}{Z_d}$, then $\min \dot{\theta} = \dot{\theta}(\frac{t_1}{4}) = \omega - Z_d(1 - \cos(\theta_a))\bar{u} > \omega - Z_d\bar{u} > 0$ since $0 < \theta_a < \pi/2$. The last inequality follows from the fact that the input over this time interval slows down the phase evolution of the neuron, so it does not reach a quarter of its total desired phase change of 2π in a quarter of the total time t_1.

If $\bar{u} = \frac{\omega}{Z_d}$, then $\min \dot{\theta} = \dot{\theta}(\frac{t_1}{4}) = \omega \cos(\theta_a) > 0$ as $\theta_a < \frac{\pi}{2}$ for the minimization problem.

If $\bar{u} > \frac{\omega}{Z_d}$, then let $\bar{\theta} = \min(\arccos(1 - \frac{\omega}{Z_d\bar{u}}), \theta_a)$, where $\arccos(1 - \frac{\omega}{Z_d\bar{u}})$ is where $\dot{\theta}$ becomes zero first and θ_a is where the control switches (before $\dot{\theta}$ becomes negative). This way, we can say that for $0 < \theta \leq \bar{\theta}$, we have $\dot{\theta} \geq 0$ and $\min \dot{\theta} = \omega - Z_d(1 -$

$\cos(\bar{\theta}))\bar{u}$. We can write:

$$\int_0^{\bar{t}} dt = \int_0^{\bar{\theta}} \frac{d\theta}{\omega - Z_d(1-\cos(\theta))\bar{u}}$$

$$\Rightarrow \qquad \bar{t} = \frac{2}{\sqrt{\omega(2Z_d\bar{u}-\omega)}} \operatorname{arctanh}\left[\sqrt{\frac{2Z_d\bar{u}-\omega}{\omega}} \tan\left(\frac{\bar{\theta}}{2}\right)\right] \quad , \tag{2.32}$$

where $\bar{t}$ is the time at which $\theta = \bar{\theta}$. Now if $\bar{\theta} = \theta_a$, meaning $\bar{t} = \frac{t_1}{4} < +\infty$ then we must have $0 < \sqrt{\frac{2Z_d\bar{u}-\omega}{\omega}} \tan(\frac{\theta_a}{2}) < 1$ for the arctanh$(\cdot)$ function to be real, which from (2.30) implies $\dot{\theta}(\frac{t_1}{4}) = \omega - Z_d(1-\cos(\theta_a))\bar{u} > 0$.

However, if $\bar{\theta} \to \arccos(1 - \frac{\omega}{Z_d\bar{u}})$ then it is inferred that $\bar{t} < \frac{t_1}{4} < +\infty$. From (2.32) we see that as $\bar{\theta} \to \arccos(1-\frac{\omega}{Z_d\bar{u}})$, $\bar{t} \to +\infty$ which contradicts $\bar{t} < \frac{t_1}{4} < +\infty$. So t always reaches $\frac{t_1}{4}$ before $\dot{\theta}$ becomes negative. We remark that by symmetry, these results also imply that $\dot{\theta} > 0$ for $\frac{3t_1}{4} \leq t \leq t_1$.

Without Charge-Balance Constraint: When the charge-balance constraint is not imposed the optimal current is given by (2.17). For the SNIPER PRC, this optimal current further simplifies to

$$u^*(t) = \pm\bar{u}, \tag{2.33}$$

resulting in

$$\dot{\theta} = \omega \pm Z_d(1-\cos(\theta))\bar{u}. \tag{2.34}$$

Since the PRC is symmetric about $\theta = \pi$, one can integrate this equation and write

$$\int_0^{t_1/2} dt = \int_0^{\pi} \frac{d\theta}{\omega \pm Z_d(1-\cos(\theta))\bar{u}},$$

which yields

$$t_1 = \frac{2\pi}{\sqrt{\omega(\omega \pm 2Z_d\bar{u})}}. \tag{2.35}$$

It should be noted that in order to make sure $\dot{\theta}$ in (2.34) is always positive, for the maximization problem, we must have $\bar{u} < \frac{\omega}{2Z_d}$.

Figure 2.3(a) shows the extreme values for the ISI (or t_1) as a function of $\bar{u}$ for this model. As can be seen in this figure, applying the charge-balance constraint has a notable effect on the value of the extreme t_1. This is due to the fact that the optimal control inputs that we achieve with this model without imposing the charge-balance constraint are always either positive or negative, and thus very different from the charge-balanced control inputs.

2.3.2 Theta Neuron Model

In this model, $Z(\theta) = 1 - \cos(\theta)$ and $f(\theta) = 1 + \cos(\theta) + I_b(1 - \cos(\theta))$. In the absence of control input, the dynamics of this neuron model are such that for $I_b > 0$ the neuron fires periodically with a natural angular velocity of $\omega = 2\sqrt{I_b}$. However, when $I_b < 0$, the neuron is said to be excitable, i.e., upon injection of some appropriate input stimuli it would start to fire periodically; otherwise it would not fire at all [16, 34].

In order to investigate the effect of a control input on the firing time of the

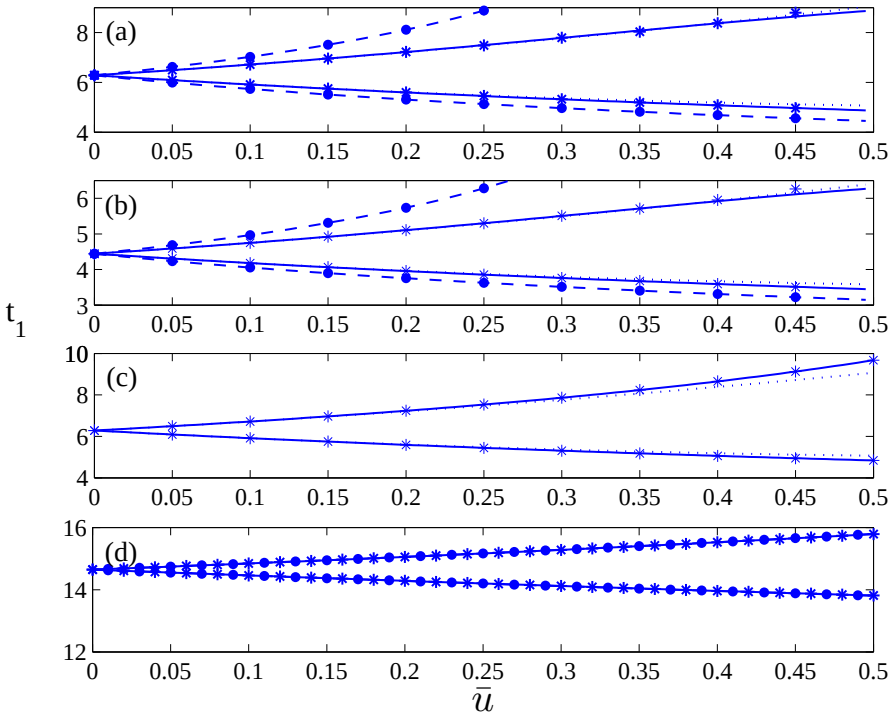


Figure 2.3: Extreme ISI (or t_1) values as functions of the control bound $\bar{u}$. (a-c) Results for the SNIPER, theta neuron, and sinusoidal models, respectively. For these results, $\omega = 1$ rad/s, $Z_d = 1$, and $I_b = 0.5$ have been chosen where applicable. The solid lines are the extreme t_1 for the charge-balanced optimal control case for the maximization (top solid line) and minimization (bottom solid line) problems obtained from the analytical formulas provided in the text. The dashed lines are the results without the charge-balance constraint. (For the sinusoidal model, since the results with or without the charge-balance constraint are identical, the dashed lines have not been shown.) The dotted lines are the second order approximations for t_1 while the star and circle markers are the results obtained from numerical simulation for the cases with and without the charge-balance constraint, respectively. (d) Numerical results for the phase-reduced Hodgkin-Huxley model with standard parameters given in Appendix C, where again the solid lines and the star markers are for the case with the charge-balance constraint and the dashed lines and the circle markers are for the case without the constraint. We see that the constraint has negligible effect on the results for this model.

neuron, we again consider two cases, namely, a control input with charge-balance constraint and one without it.

With Charge-Balance Constraint: Since this model has the same PRC as the SNIPER model (with $Z_d = 1$), similar arguments can be used to arrive at (2.21) and (2.22), giving

$$\begin{aligned} \dot{\theta} &= (1 + I_b \pm \bar{u}) + \cos(\theta)\,(1 - (I_b \pm \bar{u}))\,, && \tfrac{t_1}{4} \le t \le \tfrac{3t_1}{4}, \\ \dot{\theta} &= (1 + I_b \mp \bar{u}) + \cos(\theta)\,(1 - (I_b \mp \bar{u}))\,, && \text{otherwise,} \end{aligned} \tag{2.36}$$

$$\begin{aligned} \dot{\lambda}_1 &= -\lambda_1 \sin(\theta)(I_b \pm \bar{u} - 1), && \tfrac{t_1}{4} \le t \le \tfrac{3t_1}{4}, \\ \dot{\lambda}_1 &= -\lambda_1 \sin(\theta)(I_b \mp \bar{u} - 1), && \text{otherwise.} \end{aligned} \tag{2.37}$$

As in the case of SNIPER model, due to the symmetry that these equations have about $t = \frac{t_1}{2}$, we have $\theta(\frac{t_1}{2}) = \pi$ and $\lambda_1(0) = \lambda_1(t_1)$. Also λ_1 is either always negative or always positive. By considering the shape of the PRC, one can easily verify that we need $\lambda_1(0) < 0$ together with $\lambda_2 = -\lambda_1(\frac{t_1}{4})(1 - \cos(\theta_a)) > 0$, where $\theta_a = \theta(\frac{t_1}{4})$, in order to satisfy (2.21) for both the minimization and maximization problems.

Furthermore, we want to make sure that $\dot{\theta} > 0$ for all $0 < t < t_1$. Considering (2.36), for the minimization problem, when $t \in [\frac{t_1}{4}\,,\, \frac{3t_1}{4}]$, $\dot{\theta}$ would be positive provided $(I_b + \bar{u}) > -\frac{1+\cos(\pi)}{1-\cos(\pi)} = 0$. For $t \notin [\frac{t_1}{4}\,,\, \frac{3t_1}{4}]$, we need $(I_b - \bar{u}) > -\frac{1+\cos(\theta_a)}{1-\cos(\theta_a)}$ in order to guarantee a positive $\dot{\theta}$. Lemma 2.3.3 proves that for arbitrary $(I_b - \bar{u})$ this criterion is always satisfied. In summary, for the minimization problem, taking $I_b + \bar{u} > 0$ will guarantee that $\dot{\theta} > 0$ for all times.

Similarly, for the maximization problem, one arrives at the requirement that $I_b - \bar{u} > 0$ (which means $I_b + \bar{u} > 0$ since $\bar{u} > 0$) for a positive $\dot{\theta}$.

Lemma 2.3.3 *For the minimization problem of the theta neuron model, when* $0 \leq t \leq \frac{t_1}{4}$*, for arbitrary* $I_b - \bar{u}$ *we have*

$$\dot{\theta} = (1 + I_b - \bar{u}) + \cos(\theta)(1 - (I_b - \bar{u})) > 0 \quad \forall\, \bar{u} \geq 0. \tag{2.38}$$

Proof For $I_b - \bar{u} > 0$, this is trivial, because one can easily rearrange (2.38) and write $\dot{\theta} = 1 + \cos(\theta) + (I_b - \bar{u})(1 - \cos(\theta))$, which is exactly the equation for a non-stimulated theta neuron model with $(I_b - \bar{u})$ substituted for I_b.

For $I_b - \bar{u} \leq 0$, when $\theta = 0$ we have $\dot{\theta} = 2 > 0$. So, for $\dot{\theta}$ to become negative, it has to become zero first. Assume at $t = \bar{t} < +\infty$, $\theta(\bar{t}) = \bar{\theta} = \arccos(\frac{I_b - \bar{u} + 1}{I_b - \bar{u} - 1})$ for which $\dot{\theta}$ vanishes. Integrating the system in (2.38), we obtain

$$\bar{t} = \frac{1}{\sqrt{\bar{u} - I_b}} \operatorname{arctanh}\left[\sqrt{\bar{u} - I_b} \tan\left(\frac{\bar{\theta}}{2}\right)\right]. \tag{2.39}$$

Now if $\theta(\bar{t}) \to \arccos(\frac{I_b - \bar{u} + 1}{I_b - \bar{u} - 1})$ then, using the identity $\cos(\theta) = 2\cos^2(\frac{\theta}{2}) - 1$, one can easily verify that $\tan(\frac{\bar{\theta}}{2}) \to \sqrt{\frac{1}{\bar{u} - I_b}}$, implying $\bar{t} \to +\infty$, which in turn contradicts the assumption of $\bar{t}$ being finite. Thus, $\dot{\theta} > 0$ for $0 \leq t \leq \frac{t_1}{4}$ and $\theta_a < \bar{\theta}$ regardless of how negative $(I_b - \bar{u})$ is.

In order to obtain an analytical solution for t_1, one can perform similar calculations as for the SNIPER model case by writing

$$\int_0^{t_1/2} dt = \int_0^{\theta_a} \frac{d\theta}{(1+I_b \pm \bar{u}) + \cos(\theta)\,(1-(I_b \pm \bar{u}))} + \\ + \int_{\theta_a}^{\pi} \frac{d\theta}{(1+I_b \mp \bar{u}) + \cos(\theta)\,(1-(I_b \mp \bar{u}))},$$

where $\theta_a = \theta(\frac{t_1}{4})$, giving the following implicit formulas for t_1:

$$\begin{aligned} &\tfrac{t_1}{4}\sqrt{I_b+\bar{u}} = \tfrac{\pi}{2} - \arctan\left[\sqrt{\tfrac{I_b+\bar{u}}{I_b-\bar{u}}}\tan\left(\tfrac{t_1}{4}\sqrt{I_b-\bar{u}}\right)\right], && I_b-\bar{u}>0 \\ &\tfrac{t_1}{4}\sqrt{2I_b} = \tfrac{\pi}{2} - \arctan\left[\tfrac{t_1}{4}\sqrt{2I_b}\right], && I_b=\bar{u} \\ &\tfrac{t_1}{4}\sqrt{I_b+\bar{u}} = \tfrac{\pi}{2} - \arctan\left[\sqrt{\tfrac{\bar{u}+I_b}{\bar{u}-I_b}}\tanh\left(\tfrac{t_1}{4}\sqrt{\bar{u}-I_b}\right)\right], && I_b-\bar{u}<0 \end{aligned} \tag{2.40}$$

for the minimization problem, where $I_b + \bar{u} > 0$, and

$$\tfrac{t_1}{4}\sqrt{I_b-\bar{u}} = \tfrac{\pi}{2} - \arctan\left[\sqrt{\tfrac{I_b-\bar{u}}{I_b+\bar{u}}}\tan\left(\tfrac{t_1}{4}\sqrt{I_b+\bar{u}}\right)\right], \qquad I_b-\bar{u}>0 \tag{2.41}$$

for the maximization problem.

In order to find an explicit formula for t_1, we Taylor expand the first equation in (2.40) and (2.41) for small $\bar{u}$ and solve for t_1 to obtain

$$t_1 = \frac{\pi}{\sqrt{I_b}} \mp I_b^{-\frac{3}{2}}\bar{u} + \frac{\pi}{8} I_b^{-\frac{5}{2}}\bar{u}^2 + \mathcal{O}(\bar{u}^3), \tag{2.42}$$

where the top sign is for the minimization problem and the bottom sign for the maximization problem. We note that, for the minimization problem, for $I_b - \bar{u} \leq 0$, we do not have an approximation for t_1. We cannot use the second and third equations in (2.40) for small $\bar{u}$ as they lead to $I_b + \bar{u} \leq 0$ in the limit $\bar{u} \to 0$, violating the validity domain for (2.40).

Without Charge-Balance Constraint: Since the theta neuron model has a PRC with the same shape as for the SNIPER model, when the charge-balance constraint is not imposed the optimal control input would be the same as (2.33), resulting in:

$$\frac{t_1}{2} = \int_0^{\pi} \frac{d\theta}{1 + \cos(\theta) + (1 - \cos(\theta))(I_b \pm \bar{u})}.$$

This yields

$$t_1 = \frac{\pi}{\sqrt{I_b \pm \bar{u}}}, \qquad I_b \pm \bar{u} > 0. \tag{2.43}$$

We point out that when the charge-balance constraint is not imposed, we need to have $I_b + \bar{u} > 0$ for the minimization problem, and $I_b - \bar{u} > 0$ for the maximization problem in order to have oscillatory motion; this is analogous to the requirement for I_b in the absence of a control input.

Figure 2.3(b) shows the extreme values for ISI (or t_1) as a function of $\bar{u}$ for this model. We see that applying the charge-balance constraint has a notable effect on the value of the extreme t_1. This is due to the fact that the optimal control inputs that we achieve with this model when the charge-balance constraint is not imposed are, similar to the SNIPER model case, always either positive or negative and thus very different from the charge-balanced control inputs.

2.3.3 Sinusoidal Model

In this model, $Z(\theta) = Z_d \sin(\theta)$ and $f(\theta) = \omega = \text{constant}$, where $Z_d > 0$ is a constant. The PRC for this model is perfectly symmetric with respect to the point $\theta = \pi$. If one attempts to solve the time optimization problem without considering the charge-balance constraint, one would use (2.17) for the optimal control input. This yields

$$\begin{aligned} \dot{\theta} &= \omega \pm Z_d \sin(\theta)\bar{u}, \qquad 0 \leq \theta < \pi, \\ \dot{\theta} &= \omega \mp Z_d \sin(\theta)\bar{u}, \qquad \pi \leq \theta < 2\pi. \end{aligned} \tag{2.44}$$

In addition, considering the symmetry of the PRC about $\theta = \pi$, it follows that if the system is to evolve from $\theta(0) = 0$ to $\theta(t_1) = 2\pi$, then it would satisfy $\theta(\frac{t_1}{2}) = \pi$. This means that the optimal current in (2.17) changes sign at $t = \frac{t_1}{2}$, which implies that it is actually charge-balanced. If one solves the Euler-Lagrange equations in (2.4)-(2.8)/(2.9), one would get $\lambda_2 \equiv 0$ for this model, which implies that the optimal control inputs are always charge-balanced regardless of imposing the charge-balance constraint or not.

In this model, when $0 \leq \theta \leq \pi$, $\dot{\theta}$ takes on identical values with respect to $\theta = \frac{\pi}{2}$, i.e., $\dot{\theta}(\frac{\pi}{2} - \delta) = \dot{\theta}(\frac{\pi}{2} + \delta)$ for $\delta \in [0\,,\, \frac{\pi}{2}]$. This means that θ evolves from 0 to $\frac{\pi}{2}$ in exactly the same time as it evolves from $\frac{\pi}{2}$ to π, which implies that $\theta(t = \frac{t_1}{4}) = \frac{\pi}{2}$. Also, we want to make sure that $\dot{\theta} > 0$ for all times. This is always the case for the minimization problem, but for the maximization problem it will only hold if $\bar{u} < \frac{\omega}{Z_d}$.

One can now calculate the optimal spiking time by integrating (2.44):

$$\int_0^{t_1/4} dt = \int_0^{\pi/2} \frac{d\theta}{\omega \pm Z_d \sin(\theta)\bar{u}}.$$

This yields

$$\begin{aligned} t_1 &= \tfrac{8}{\sqrt{\omega^2 - Z_d^2\bar{u}^2}} \arctan\left[\sqrt{\tfrac{\omega - Z_d\bar{u}}{\omega + Z_d\bar{u}}}\right], && 0 \le \bar{u} < \tfrac{\omega}{Z_d} \\ t_1 &= \tfrac{4}{\omega}, && \bar{u} = \tfrac{\omega}{Z_d} \\ t_1 &= \tfrac{8}{\sqrt{Z_d^2\bar{u}^2 - \omega^2}} \operatorname{arctanh}\left[\sqrt{\tfrac{Z_d\bar{u} - \omega}{Z_d\bar{u} + \omega}}\right], && \bar{u} > \tfrac{\omega}{Z_d} \end{aligned} \tag{2.45}$$

for the minimization problem, and

$$t_1 = \tfrac{8}{\sqrt{\omega^2 - Z_d^2\bar{u}^2}} \arctan\left[\sqrt{\tfrac{\omega + Z_d\bar{u}}{\omega - Z_d\bar{u}}}\right], \qquad 0 \le \bar{u} < \tfrac{\omega}{Z_d} \tag{2.46}$$

for the maximization problem.

In order to find an approximation to t_1, we Taylor expand (2.45) and (2.46) for small $\bar{u}$ and solve for t_1 to obtain

$$t_1 = \frac{2\pi}{\omega} \mp \frac{4Z_d}{\omega^2}\bar{u} + \frac{\pi Z_d^2}{\omega^3}\bar{u}^2 + \mathcal{O}(\bar{u}^3). \tag{2.47}$$

Figure 2.3(c) shows the extreme values for the ISI (or t_1) as a function of $\bar{u}$ for this model. As mentioned before, applying the charge-balance constraint here has no effect on the value of the extreme t_1. This is due to the fact that the optimal control inputs that we achieve with the sinusoidal model without imposing the charge-balance constraint are exactly charge-balanced themselves.

2.3.4 Phase-Reduced Hodgkin-Huxley Model

The Hodgkin-Huxley equations are a conductance-based model for neurons proposed in 1952 as a result of a series of experiments on the giant axon of a squid [50]. This model has become the prototypical model for neuronal membrane dynamics. Although not representing human brain neurons, the Hodgkin-Huxley model exhibits oscillatory behavior, as do human motor control neurons in the thalamus and basal ganglia regions of the brain. The equations and parameters of this model are presented in Appendix C.

The PRC for this model has been calculated numerically using XPPAUT [2] with time steps of 0.005 ms, see Figure 2.4. The specifics of this PRC are:

$$
\begin{aligned}
&Z(0) = Z(2\pi) = 7.7 \times 10^{-5}, \\
&Z(0.354) = Z(4.120) = 0, \\
&Z(\theta) < 0 \qquad 0.354 < \theta < 4.120, \\
&Z(\theta) > 0 \qquad \text{otherwise.}
\end{aligned}
$$

Due to the complex shape of this PRC, very little can be said analytically. Thus, we present numerical results for the time optimal problem for this neuron model. In this model, for the parameters that we use, $f(\theta) = \omega = 0.429$ rad/ms $=$ constant which results in $T = 14.63$ ms.

With Charge-Balance Constraint: To solve the TPBVP in this case, given a $\bar{u}$, we initially guess arbitrary values for $\lambda_1(0)$ and t_1. We calculate λ_2 from (2.15) with $u^*(t_1) = \bar{u}$, and we solve the system of ODE's (2.4)-(2.7) for $0 \leq t \leq t_1$

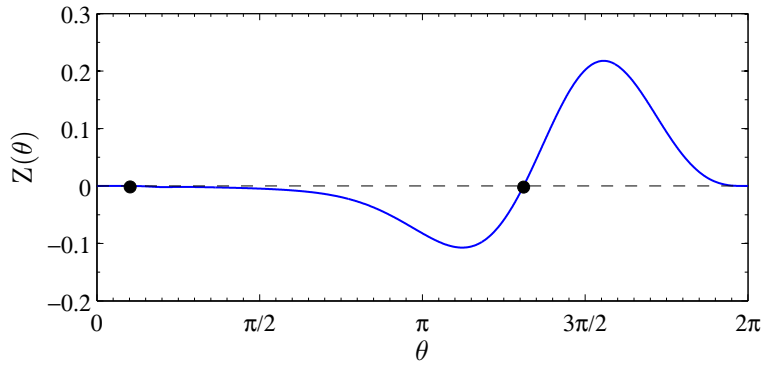


Figure 2.4: Hodgkin-Huxley PRC, obtained numerically using XPPAUT with $I_b = 10$

using a shooting method. In this shooting method, we solve the system and if $\theta(t_1) < 2\pi$, we increase t_1 by a certain dt and if $\theta(t_1) > 2\pi$ we decrease t_1 by dt until $|\theta(t_1) - 2\pi| < \epsilon_1$ where ϵ_1 is a predefined tolerance. In other words, we keep *shooting* the system with different t_1 values until we reach the t_1 that would satisfy this inequality. We note here that in this process, when an upper bound and a lower bound for t_1 is found, the actual value of t_1 is then found by employing the bisection method. Now if $q(t_1) \neq 0$, we conclude that the original guess for $\lambda_1(0)$ had been incorrect. So we perform another shooting process for $\lambda_1(0)$ exactly like the one for t_1, until we find the upper and lower bounds of $\lambda_1(0)$. We then employ the bisection method to converge to the correct $\lambda_1(0)$ for which $|q(t_1)| < \epsilon_2$, where ϵ_2 is a predefined tolerance. Figure 2.5 shows the results for this model for $\bar{u} = 0.2$. We used fourth order Runge-Kutta for numerical integration.

Without Charge-Balance Constraint: In this case, $\lambda_2 \equiv 0$ and the optimal control is simply given by (2.17).

Figure 2.3(d) shows the extreme values for the ISI (or t_1) as a function of $\bar{u}$ for this model. We see that applying the charge-balance constraint has little effect on the value of the extreme t_1. This is due to the fact that the optimal control inputs that we achieve with the phase-reduced Hodgkin-Huxley model without imposing the charge-balance constraint are *almost* charge-balanced themselves.

In order to see the performance of the phase reduction technique and the phase-reduced Hodgkin-Huxley model, we have solved the phase-reduced model for a number of different $\bar{u}$ values and have obtained the optimal input for each of them. We have then applied these inputs to the full Hodgkin-Huxley equations, given in Appendix C, and have computed the resulting ISI values $t_{1,full}$, to compare with those for the phase-reduced model $t_{1,prm}$. Figure 2.6 shows the results of this investigation. Recall that the natural period of oscillation for this model is $T = 14.63$ ms. The phase-reduced model yields accuracy for δt to within one percent for $\bar{u} \leq 0.3$ for the maximization problem and for much larger $\bar{u}$ for the minimization problem.

2.4 Discussion

We investigated the time optimal control problem for phase models of spiking neurons for which the input is constrained between prespecified upper and lower bounds. The dynamical equations were derived from the Hamiltonian for the system, and the control inputs obtained from Pontryagin's minimum principle, which gives bang-bang control. The problem was considered for two cases: with a charge-

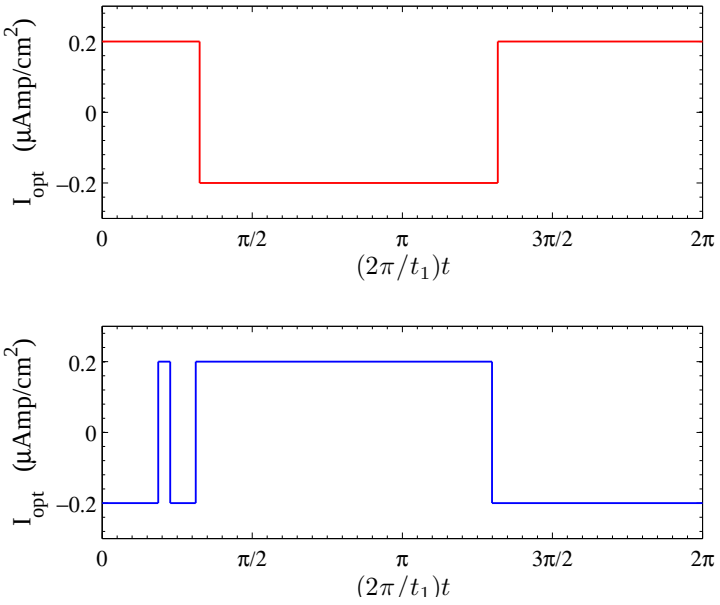


Figure 2.5: Optimal control for the Hodgkin-Huxley model with the charge-balance constraint imposed for the minimization problem (top) and the maximization problem (bottom). The control bound was set $\bar{u} = 0.2$. In the case of the maximization problem, the optimal control has a spike at $(2\pi/t_1)t \approx 0.66$ which is due to a small dent in the PRC for this model at the corresponding location.

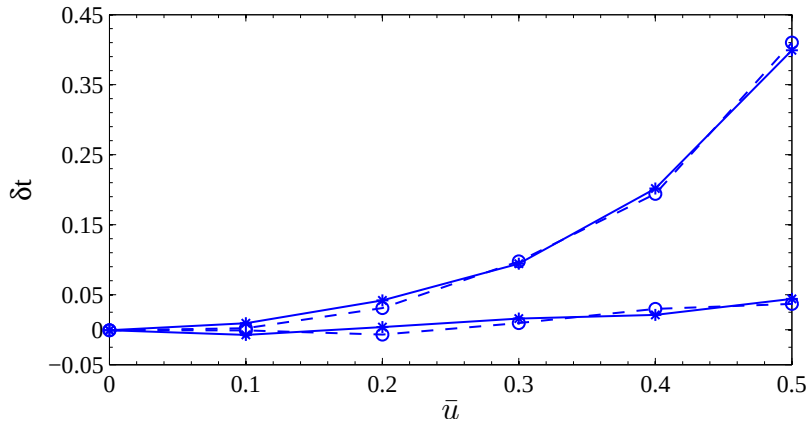


Figure 2.6: Comparison between the phase-reduced Hodgkin-Huxley model and the full Hodgkin-Huxley model. $\delta t = t_{1,full} - t_{1,prm}$ is the difference between the t_1 value calculated from the full model and the t_1 value obtained from the phase-reduced model when the $u^*(t)$ obtained from the phase-reduced model is used as the input to both models. The solid lines with asterisk markers represent the case with the charge-balance constraint and the dashed lines with circle markers represent the case without the constraint. Also, the two top lines are for the maximization problem and the two bottom lines are for the minimization problem.

balance constraint imposed on the input, and without it. Here the charge-balance constraint ensures that the total electrical charge that is transferred to the neural tissue is zero over the course of one cycle of control input; this is important to prevent neural tissue damage. Analytical expressions for the ISI were derived for these two cases for the SNIPER, theta neuron, and sinusoidal models.

We looked at the SNIPER model as a simplified version of the theta neuron model and as a platform for describing the details of our control strategy. Specifically, we found that in order to have $\dot{\theta} > 0$ at all times, we do not need to limit our control input if the minimization problem is considered. However, when considering the maximization problem, $\dot{\theta} > 0$ implies that $\bar{u} < \frac{\omega}{2Z_d}$. The ISI values are then found from (2.27)-(2.29) when the charge-balance constraint is imposed and from (2.35) when the constraint is not imposed.

In the theta neuron model, $I_b \pm \bar{u} > 0$ guarantees $\dot{\theta} > 0$ for all time. The ISI values are then found from (2.40)-(2.42) when the charge-balance constraint is imposed and from (2.43) when the constraint is not imposed.

The sinusoidal model was considered as a simple, yet very insightful, Type II neuron model. It was found that regardless of imposing the charge-balance constraint, the optimal inputs always come out charge-balanced. This is due to the symmetry of the PRC, the equations (2.44), as well as the fact that the sinusoidal PRC has equal positive and negative intervals. Due to these specifications, the control input found from the bang-bang optimal control method are always charge-balanced. We note that to ensure $\dot{\theta} > 0$ for all time, we need to have $\bar{u} < \frac{\omega}{Z_d}$ for the

maximization problem, but no such condition on $\bar{u}$ for the minimization problem. This yields (2.45)-(2.47) as the ISI values.

We also considered the phase-reduced Hodgkin-Huxley model as a more realistic model for neuronal membrane dynamics and found out that the control input in both cases of with and without the charge-balance constraint are very close to each other. The results from the sinusoidal and Hodgkin-Huxley models suggest that control inputs found for Type II neurons can be almost charge-balanced even when the constraint is not imposed.

We also presented numerical results for the phase-reduced Hodgkin-Huxley model as well as for the three other models mentioned. These numerical results agree well with the analytical computations. The performance of the phase reduction method was validated through the close agreement of the ISI values obtained by using the same computed control input for both the phase-reduced and the full Hodgkin-Huxley models.

The results for the ISI presented here are the minimum and maximum values that one can achieve with bounded input stimuli for the aforementioned neuron models on a single neuron level. However, one may consider the problem of finding the $\bar{u}$ that would achieve a prespecified ISI value t_1. We note that depending on the t_1, a $\bar{u}$ may or may not exist. If, for example, we consider a t_1 significantly larger than the neuron's natural period T, then there would not be a $\bar{u}$ that would result in t_1 while satisfying the condition $\dot{\theta} > 0$. As another example, if $t_1 \ll T$, the $\bar{u}$ needed for this would be large and hence outside of the range of validity of

phase models as these models are only valid for small inputs. If we choose $t_1 \approx T$, there will be a $\bar{u}$ for the problem, but determining the specific value is not trivial; in particular cases, one can use equations (2.29), (2.35), (2.42), (2.43), or (2.47) to find an estimate for or the exact value of the $\bar{u}$.

We note that one can use these results as a foundation for considering the control of a population of neurons. In particular, one can gain insight about the maximum capability of treatment procedures like deep brain stimulation when input stimuli are bounded so as to account for potential practical limitations for the hardware in delivering the input stimulus, as well as the endurance of the biological tissue close to the injection apparatus.

Chapter 3

Energy Optimal Control of Spiking Neurons

In this chapter, we consider regulating the interspike interval for phase models of neurons by designing event-based minimum energy control stimuli that would be applied at the onset of an action potential and optimally change the next spiking time for the neuron. We show simulation results for implementing the designed control on the phase models considered in Chapter 2. The control is constrained to be charge-balanced, i.e., to have a zero net integral over the period of application.

The organization of this chapter is as follows. After a brief introduction in Section 3.1, the model equations are presented and the minimum energy control formulation is derived using a calculus of variations method in Section 3.2. In Section 3.3, several examples are considered for which the results of implementing the derived optimal control are shown. The results are discussed further in Section 3.4.

The main results from this chapter were partially published in [17, 19].

3.1 Introduction

The motivation for designing event-based minimum energy input stimuli comes from the desire to increase battery life for implanted neurostimulators for deep brain stimulation applications. In addition to the fact that the input stimulus is one that uses minimum energy, its event-based nature results in its application only when an action potential is detected, hence a fewer number of applications. The importance of this is more evident when a population of neurons is considered. In a population setting, the event-based feature of the control could result in considerably fewer number of applications, as the triggering event would be the crossing of some threshold by the mean field voltage.

From a clinical standpoint, charge-balanced inputs are important because they preserve the internal electrical balance in the neuron and reduce the amount of irreversible Faradaic reduction-oxidation reactions at the electrode-brain interface, thus considerably limiting the formation of toxic products over time.

3.2 Model equations

Consider the phase model presented in (2.1):

$$\frac{d\theta}{dt} = f(\theta) + Z(\theta)u(t), \tag{3.1}$$

where, $f(\theta)$ represents the neuron's baseline dynamics, $Z(\theta)$ is the Phase Response Curve (PRC) of the neuron, and $u(t) = I(t)/c$ is the input stimulus with c representing the membrane capacitance. Also, $\theta(t) \in [0, 2\pi)$ is the neuron's phase, and by convention $\theta = 0$ corresponds to the spiking of the neuron.

Without loss of generality, we assume that the neuron fires at $t = 0$. Our objective is to find the optimal input stimulus to the neuron so that it fires at the prespecified desired time t_1. This means that we require,

$$\theta(0) = 0, \qquad \theta(t_1) = 2\pi. \tag{3.2}$$

Here we present an event-based control scheme that, after detecting a spike at time t^*, stimulates the neuron with a pre-computed charge-balanced energy optimal waveform in order to drive the neuron to spike next at $t_1 > t^*$. Without loss of generality, we can take $t^* = 0$.

Consider the phase model (3.1) for a spiking neuron augmented by an additional dynamic state q as,

$$\begin{aligned} \dot{\theta} &= f(\theta) + Z(\theta)u(t), \\ \dot{q} &= u(t), \\ \theta(0) &= 0, \\ q(0) &= 0, \end{aligned} \tag{3.3}$$

where q is simply the integral of the total stimulus delivered to the neuron [17]. In order to achieve charge-balance, q must equal zero after the control waveform is applied.

The total input energy to the system is the integral of the square of the input

stimulus over the time horizon of $[0, t_1]$, scaled by the equivalent circuit impedance. The optimality criterion is to minimize this total input energy. So, for a specified spike time t_1, from the set of all stimuli $u(t)$ which evolve $\theta(t)$ via (3.1) from $\theta(0) = 0$ to $\theta(t_1) = 2\pi$, we want to find the stimulus which minimizes the following cost function:

$$\mathcal{G}[u(t)] = \int_0^{t_1} [u(t)]^2 \mathrm{d}t, \tag{3.4}$$

and yields $q(t_1) = 0$. Other optimality criteria lead to other cost functions, but can be handled similarly (cf. [51]).

We apply calculus of variations to minimize [44]

$$\mathcal{C}[\mathbf{\Phi}(t), \dot{\mathbf{\Phi}}(t), u(t)] = \int_0^{t_1} \underbrace{\left\{ [u(t)]^2 + \begin{bmatrix} \lambda_1(t) & \lambda_2(t) \end{bmatrix} \cdot \begin{bmatrix} f(\theta) + Z(\theta)u(t) - \dot{\theta} \\ u(t) - \dot{q} \end{bmatrix} \right\}}_{\mathcal{L}[\mathbf{\Phi}, \dot{\mathbf{\Phi}}, u(t)]} \mathrm{d}t, \tag{3.5}$$

where $\mathbf{\Phi}(t) = [\theta(t),\ q(t),\ \lambda_1(t),\ \lambda_2(t)]^T$. The Lagrange multipliers $\lambda_1(t)$ and $\lambda_2(t)$ force the dynamics to satisfy (3.3).

Using vector notation, the associated Euler-Lagrange equations are:

$$\frac{\partial \mathcal{L}}{\partial u} = \frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{u}}\right), \qquad \frac{\partial \mathcal{L}}{\partial \mathbf{\Phi}} = \frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\mathbf{\Phi}}}\right),$$

so that

$$
\begin{aligned}
u(t) &= -\frac{\lambda_1(t)Z(\theta)+\lambda_2(t)}{2}, && (3.6)\\
\dot{\theta} &= f(\theta)-\frac{\lambda_1(t)[Z(\theta)]^2+\lambda_2(t)Z(\theta)}{2}, && (3.7)\\
\dot{q} &= u(t)=-\frac{\lambda_1(t)Z(\theta)+\lambda_2(t)}{2}, && (3.8)\\
\dot{\lambda}_1 &= -\lambda_1(t)f'(\theta)+\frac{[\lambda_1(t)]^2Z(\theta)Z'(\theta)+\lambda_1(t)\lambda_2(t)Z'(\theta)}{2}, && (3.9)\\
\dot{\lambda}_2 &= 0, && (3.10)
\end{aligned}
$$

where $' = d/d\theta$. To find the optimal $u(t)$, (3.7)-(3.10) need to be solved subject to the conditions

$$\theta(0)=0, \quad \theta(t_1)=2\pi, \qquad q(0)=0, \quad q(t_1)=0. \tag{3.11}$$

This is a two point boundary value problem (TPBVP) where the boundary values for $\theta(t)$ and $q(t)$ are given in (3.11). In most realistic models of neurons, the baseline dynamics of the neuron is represented by a constant positive scalar ω, indicating periodic spiking at a fixed rate. For the following analytical derivations, we assume that $f(\theta)=\omega>0$. However, we present numerical results for the theta neuron model which is an example with nonconstant $f(\theta)$.

Theorem 3.2.1 *Suppose $Z(0)=0$ and $f(\theta)=\omega>0$. Then, for given values t_1 and λ_2, there is a unique trajectory solving the Euler-Lagrange equations (3.7)-(3.10) with boundary conditions (3.11).*

Proof From (3.10), λ_2 is a constant. Therefore, for the 2-dimensional system

(3.7,3.9) the Hamiltonian

$$h(\theta, \lambda_1) = \lambda_1(t)\omega - \frac{[\lambda_1(t)]^2[Z(\theta)]^2}{4} - \frac{\lambda_1(t)\lambda_2 Z(\theta)}{2} \tag{3.12}$$

is conserved along the solutions $(\theta(t), \lambda_1(t))$. Letting

$$h_0 = h(\theta(0), \lambda_1(0)) = h(0, \lambda_1(0)),$$

we have

$$\lambda_1(t)\omega - \frac{[\lambda_1(t)]^2[Z(\theta)]^2}{4} - \frac{\lambda_1(t)\lambda_2 Z(\theta)}{2} - h_0 = 0. \tag{3.13}$$

We first demonstrate that

$$\frac{d\theta}{dt} > 0, \tag{3.14}$$

as follows. Consider a trajectory $\{(\theta(t), \lambda_1(t))\}$, $0 \leq t \leq \tau$ with $\theta(\tau) = 2\pi$ and which solves (3.7),(3.9). From (3.7), we have $\frac{d\theta}{dt}|_{t=0} > 0$. Now assume in point of contradiction that there exists a time $0 < \hat{t} < \tau$ such that $\frac{d\theta}{dt}|_{t=\hat{t}} < 0$. Since $\theta(\tau) = 2\pi$, in this case there also exists a phase $\bar{\theta} < 2\pi$ such that $\theta(t) = \bar{\theta}$ for three distinct times between 0 and τ. A quick sketch in the (θ, λ_1) plane shows that, since any trajectory $\{(\theta(t), \lambda_1(t))\}$ is not self-intersecting, the trajectory under our assumption contains three distinct points $(\bar{\theta}, \lambda_1^{(j)})$, $j = 1, 2, 3$. However, the trajectory must also be a level set of the Hamiltonian; from (3.13), which is quadratic in λ_1, such a level set contains at most two points (θ, λ_1) for any value of θ. Therefore, a contradiction has been reached, and (3.14) follows.

Now, multiplying (3.13) by $[Z(\theta)]^2$ and rearranging we get

$$\left([\lambda_1(t)][Z(\theta)]^2\right)^2 + (2\lambda_2 Z(\theta) - 4\omega)\left(\lambda_1(t)[Z(\theta)]^2\right) + 4h_0[Z(\theta)]^2 = 0.$$

Solving for $\lambda_1(t)[Z(\theta)]^2$ yields

$$\lambda_1(t)[Z(\theta)]^2 = -(\lambda_2 Z(\theta) - 2\omega) \pm \sqrt{(\lambda_2 Z(\theta) - 2\omega)^2 - 4h_0[Z(\theta)]^2}. \tag{3.15}$$

From (3.7) and (3.14), $\lambda_1(t)[Z(\theta)]^2 < -(\lambda_2 Z(\theta) - 2\omega)$. Thus, in (3.15) the valid solution is the *minus* branch, i.e.,

$$\lambda_1(t)[Z(\theta)]^2 = -(\lambda_2 Z(\theta) - 2\omega) - \sqrt{(\lambda_2 Z(\theta) - 2\omega)^2 - 4h_0[Z(\theta)]^2}. \tag{3.16}$$

Now, from (3.7) we can write

$$\begin{aligned} t_1 = \int_0^{t_1} dt &= \int_0^{2\pi} \frac{d\theta}{\omega - \frac{\lambda_1(t)[Z(\theta)]^2 + \lambda_2 Z(\theta)}{2}} \\ &= \int_0^{2\pi} \frac{d\theta}{\sqrt{\left(\frac{\lambda_2 Z(\theta)}{2} - \omega\right)^2 - h_0[Z(\theta)]^2}}, \end{aligned} \tag{3.17}$$

where the last equality uses (3.16). Differentiating with respect to h_0, gives

$$\frac{dt_1}{dh_0} = \frac{1}{2}\int_0^{2\pi} \frac{[Z(\theta)]^2 d\theta}{\left[\left(\frac{\lambda_2 Z(\theta)}{2} - \omega\right)^2 - h_0[Z(\theta)]^2\right]^{3/2}} > 0, \tag{3.18}$$

Therefore, t_1 increases monotonically with h_0. Also, from (3.12), $h_0 = h(\theta(0), \lambda_1(0))$ $= h(0, \lambda_1(0)) = \omega\lambda_1(0)$. So t_1 increases monotonically with $\lambda_1(0)$. This means that, for a given t_1 and λ_2, there is a unique value of $\lambda_1(0)$, which gives a unique trajectory.

A shooting method is used to solve this boundary value problem numerically. We choose an arbitrary nonzero value of λ_2, take $\theta(0) = q(0) = 0$, and solve the system (3.7)-(3.10) iteratively for different nonzero guesses of $\lambda_1(0)$ until $\theta(t_1) = 2\pi$ with a predefined tolerance. Once an upper and a lower bound for $\lambda_1(0)$ is found, employing the bisection method guarantees an answer. From Theorem 3.2.1, there

is a unique $\lambda_1(0)$ that satisfies this. After finding this $\lambda_1(0)$ and its associated trajectory numerically, we check if the resulting $q(t_1)$ is within a small tolerance of 0. If so, the problem is considered solved. If not, we conclude that our original choice for λ_2 had been wrong and so a new value is chosen for λ_2 using the bisection method, and the process is repeated. The next choice of λ_2 is made by examining the $q(t_1)$ error gradient from the previous two simulations. The procedure continues until a pair $(\lambda_1(0), \lambda_2)$ is found for which the boundary conditions in (3.11) are achieved. A consequence of Theorem 3.2.1 is that we can effectively search for a solution to the Euler-Lagrange equations along a one-dimensional curve in $(\lambda_1(0), \lambda_2)$ space. Once the optimal trajectories for $\theta(t)$ and $\lambda_1(t)$ are found, we can find the optimal control input by evaluating (3.6).

3.3 Examples

3.3.1 SNIPER Neuron Model

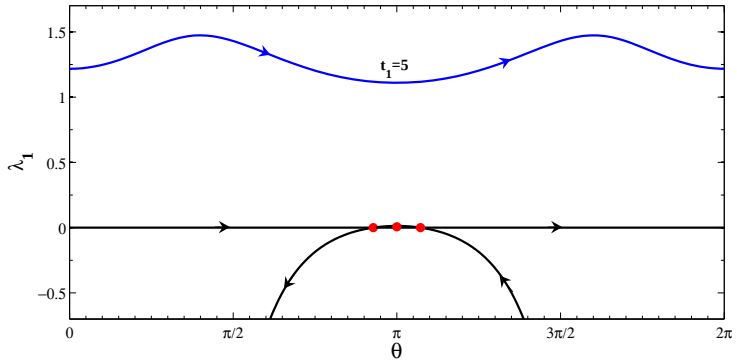
For a periodically firing neuron near a SNIPER bifurcation, $Z(\theta) = Z_d(1 - \cos(\theta))$, where Z_d is a model dependent constant, which we take to be $Z_d = 1$ for simplicity, and $f(\theta) = \omega = 1$, which gives the natural period of spiking as $T = 2\pi$ [32, 34].

The phase portraits for (3.7,3.9) with $t_1 = 5$ and $t_1 = 9$ are shown in Figures 3.1a and 3.1b, respectively. These figures also show the fixed points which exist for the (3.7,3.9) subsystem, along with their associated stable and unstable manifolds.

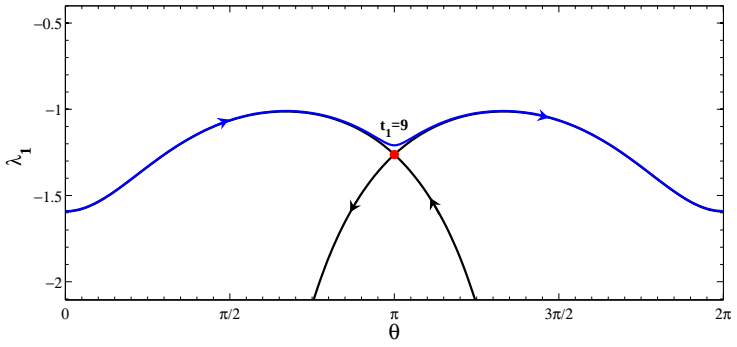
These manifolds can sometimes be used to interpret the trajectories associated with the optimal stimulus: in particular, the $t_1 = 9$ trajectory of the Euler-Lagrange equations is close to the stable and unstable manifolds of the fixed point. This forces the trajectory to spend a long time near the fixed point, delaying its arrival to $\theta = 2\pi$.

We note that Figures 3.1a and 3.1b are for different values of λ_2, the Lagrange multiplier associated with the charge-balance constraint. Each of the $t_1 = 5$ and $t_1 = 9$ trajectories shown in these Figures are the only trajectories in their respective phase planes that can take the (3.6)-(3.10) system from $\theta(0) = 0$ to $\theta(t_1) = 2\pi$ with zero charge transfer and the least amount of input energy. Without the charge-balance constraint, the phase portrait of the system would not change for different t_1 values and different trajectories corresponding to different t_1 values can be drawn in the same phase plot with different $\lambda_1(0)$ values.

A comparison of the optimal stimulus with and without the charge-balance constraint is shown in Figure 3.2. Clearly for this model, the constraint has a large effect on the form of the optimal stimulus. Note that the horizontal axis in this figure is scaled for ease of comparison. Indeed, when there is no such constraint, the optimal stimulus is always positive (resp., negative) when we want the neuron to fire earlier (resp., later) than it would in the absence of external input.



(a)



(b)

Figure 3.1: Phase portraits for the SNIPER neuron model with the charge-balance constraint for (a) $t_1 = 5$ and (b) $t_1 = 9$. When the charge-balance constraint is imposed, phase portrait of the system changes as the value of t_1 changes. Also, it is seen that the trajectory of the system for $t_1 = 9$ is closer to the fixed point of the system than that of for $t_1 = 5$. This translates into its spending more time around the fixed point, hence delaying its total time from $t_1 = 5$ to $t_1 = 9$.

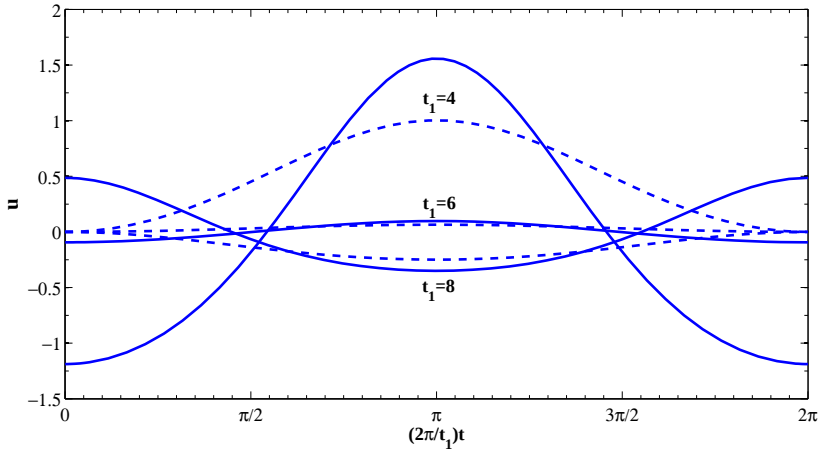


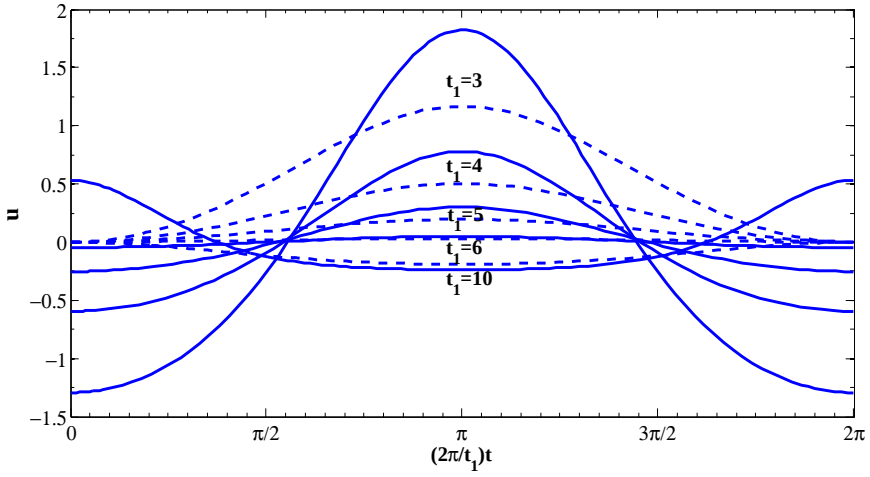
Figure 3.2: SNIPER neuron model optimal stimulus for three different values of t_1 with (solid lines) and without (dashed lines) the charge-balance constraint.

3.3.2 Theta Neuron

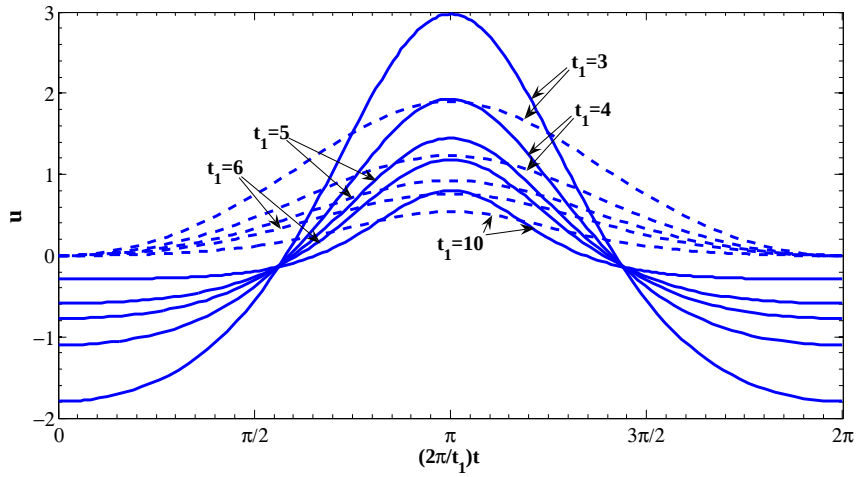
The PRC for this model is the same as the SNIPER PRC, but the baseline dynamics is quite different: $f(\theta) = 1 + \cos(\theta) + I_b(1 - \cos(\theta))$ where I_b is the baseline current. In order to see the effect of charge-balance constraint on the optimal stimuli, Figures 3.3(a) and 3.3(b) show the optimal stimuli for the two cases of with and without the constraint, for $I_b = 0.25$ and $I_b = -0.25$, respectively. It is seen that the charge-balance constraint has a notable effect for this model as well. It is worth mentioning that the horizontal axis in this figure is scaled for ease of comparison.

3.3.3 Sinusoidal PRC

In this model, we take $f(\theta) = \omega = 1$ and $Z(\theta) = \sin(\theta)$. The natural period of oscillations would then be $T = 2\pi$. The PRC for this model is a Type II PRC where it has both positive and negative values. For this particular neuronal



(a)



(b)

Figure 3.3: Theta neuron model optimal stimulus comparison for different values of t_1 with (solid lines) and without (dashed lines) the charge-balance constraint, for (a) $I_b = 0.25$ and (b) $I_b = -0.25$.

model, the results obtained with and without the charge-balance constraint are the same. This is due to the fact that the optimal stimuli obtained without the charge-balance constraint are in fact sinusoidal and thus enclose a zero-net area, i.e., the total charge transferred is zero.

Figure 3.4 shows the optimal stimulus, with and without the charge-balance constraint for three different values of t_1. Again, the horizontal axis in this figure is scaled for ease of comparison. It can be seen that for values of t_1 greater than the natural period ($T = 2\pi$), the optimal stimulus starts off negative first and then builds up to positive values, whereas for values of t_1 less than $T = 2\pi$, the result is reversed. Considering equation (3.1), we expect this to happen, because this stimulus is being multiplied by the PRC and added to the baseline dynamics of the neuron to give the phase dynamics of the neuron. Since the Sine PRC is a sinusoidal curve in θ, when multiplied by a quantity of opposite (resp., same) sign, it produces a negative (resp., positive) value which results in slowing down (resp., speeding up) the phase dynamics and thus, causing the neuron to fire later (resp., earlier) than it would have naturally.

3.3.4 Hodgkin-Huxley Neuron Model

The PRC for the Hodgkin-Huxley neuron model is numerically obtained using the XPPAUT software [1] and is shown in Figure 3.5. This PRC is another example of a Type II PRC. The solid lines in Figure 3.6 show the charge-balanced optimal inputs to the system for different t_1's. Recall that we have assumed the neuron's

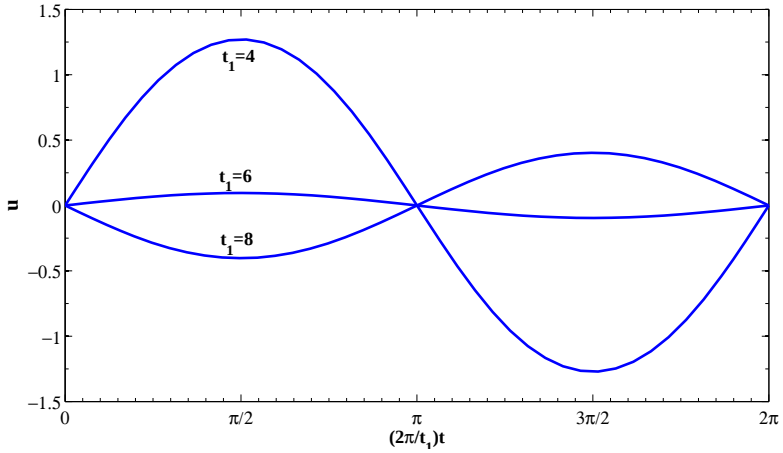


Figure 3.4: Sine neuron model optimal stimulus for three different values of t_1. The same results are obtained with or without the charge-balance constraint.

capacitance to be $c = 1$ for these simulations, and thus the optimal input is actually an electrical current stimulus. Note that the horizontal axis in this figure is scaled for ease of comparison.

By removing the charge-balance constraint we reproduce the results of [16]. In this case $\lambda_2 \equiv 0$, hence eliminating the charge-balance constraint. Again the shooting method is used to find the $\lambda_1(0)$ that would result in $\theta(t_1) = 2\pi$. The results for the optimal stimuli for this case are shown by the dashed lines in Figure 3.6. In this case, the optimal stimuli all start and end at zero, whereas in the case with the charge-balance constraint, they do not. It is worth pointing out that considering the phase-reduced model (3.1) and the shape of the PRC in Figure 3.5, one can easily verify the shape of achieved stimuli. The natural period of oscillations for this model is $T = 14.63$ ms. Therefore, for spike times $t_1 < T$ one needs to increase $\dot{\theta}$ in (3.1). Since the objective is to minimize the input energy, intuitively, one would expect the sign of the stimuli to approximately follow the PRC's sign to

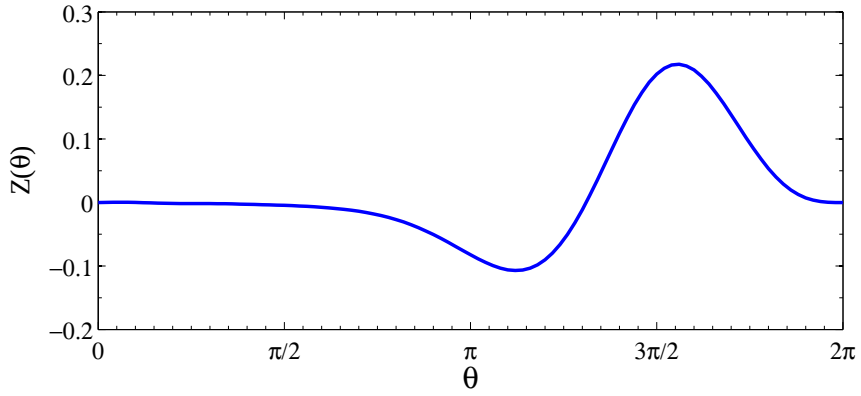


Figure 3.5: Hodgkin-Huxley phase response curve computed numerically using XPPAUT [1].

make maximum use of the injected stimulus. Conversely, for $t_1 > T$, one expects to achieve stimuli with signs opposite to that of the PRC's in most of the time. The stimuli in Figure 3.6 justify this argument. In fact, when the charge-balance constraint is not imposed, the optimal stimuli take the exact same (resp., opposite) sign of the PRC for $t_1 < T$ (resp., $t_1 > T$).

We note that it has been proven in [16] that in the case of no charge-balance constraint, the optimal solutions always exist and are unique. Furthermore, it was shown analytically that for small $|t_1 - T|$, the optimal stimulus $u(t)$ approximately takes the shape of the PRC.

Theoretically, given (3.1), one can achieve an optimal control input for any desired t_1. However, a constraining factor is the range of validity of the phase-reduced model for the large stimulus waveforms necessary to obtain extreme values of t_1. As mentioned before, the phase-reduced model is only valid when the stimuli are small, meaning the system is in a close neighborhood of the periodic orbit.

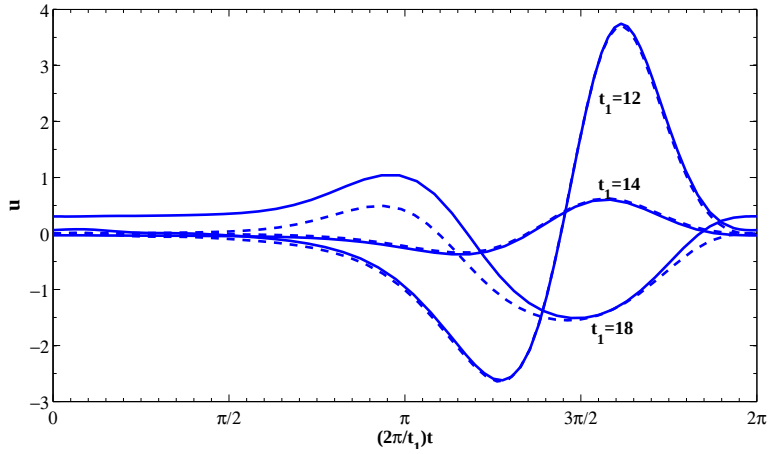


Figure 3.6: Hodgkin-Huxley phase neuron model energy optimal stimulus for three different values of t_1 with (solid lines) and without (dashed lines) the charge-balance constraint.

Therefore, for large values of $|t_1 - T|$ that require large stimuli, the phase-reduced model may not yield accurate results. In addition, there are practical limitations for the level of the current stimulus based on the capability of the hardware in delivering the current and the endurance of the biological tissue immediate to the injection probe.

3.4 Discussion

The results presented in this chapter suggest that for Type II neurons (those with both positive and negative values of PRC, like the sinusoidal and the Hodgkin-Huxley models), the optimal stimuli obtained with the charge-balance constraint are quite similar to those obtained without the constraint. This is due to the fact that the optimal stimuli approximately take the shape of the PRC, and since the PRC in a Type II neuron has both positive and negative values, the optimal stimuli

are almost charge-balanced even without the constraint having been imposed. For the sinusoidal model, because of the symmetry in the PRC, the optimal stimuli are also always charge-balanced, regardless of imposing the constraint or not. For the Hodgkin-Huxley model, the PRC is slightly asymmetric which yields optimal stimuli that are slightly charge-imbalanced when the constraint is not imposed. On the other hand, for Type I neurons (those with positive values of PRC, like the SNIPER and theta neuron models), imposing the charge-balance constraint has a notable effect on the resulting optimal stimuli. For these neurons, when the constraint is not imposed the optimal control stimuli are either non-negative or non-positive, depending on t_1. However, when the constraint is imposed, they necessarily have both negative and positive values. As a result, given a t_1, we see that the values of the stimuli with the constraint are larger than those without it.

One way to quantify the effect of the difference between the optimal stimuli obtained in the two cases of with and without the charge-balance constraint is to look at their associated energy. This energy is the cost of the optimal control problem that is minimized as the objective (see (3.4)). By plotting this input energy versus values of t_1, one can get an intuition for how the target time t_1 affects the amount of input energy required. Figure 3.7 shows this quantization for the four models considered. It is interesting to see that it takes a significantly larger amount of energy to make the neuron fire sooner than it takes for having it fire later than its natural period. This is due to the fact that, when slowing down the neuron, the input stimulus is applied over a larger period of time than when speeding up the

neuron. Hence, the magnitude of the stimuli and therefore their associated energy are smaller for t_1 values greater than the natural period, relative to those less than the natural period. Figure 3.7 also verifies that when t_1 is taken to be the natural period T, the energy is zero.

The significant difference in the optimal stimulus energy for the two cases of with and without the charge-balance constraint is clearly seen when one compares Figure 3.7 (a) and (b) with Figure 3.7 (c) and (d). Specifically, for the SNIPER model, without the charge-balance constraint we need less than half of the energy than that with the constraint, but for the sinusoidal model, the two coincide.

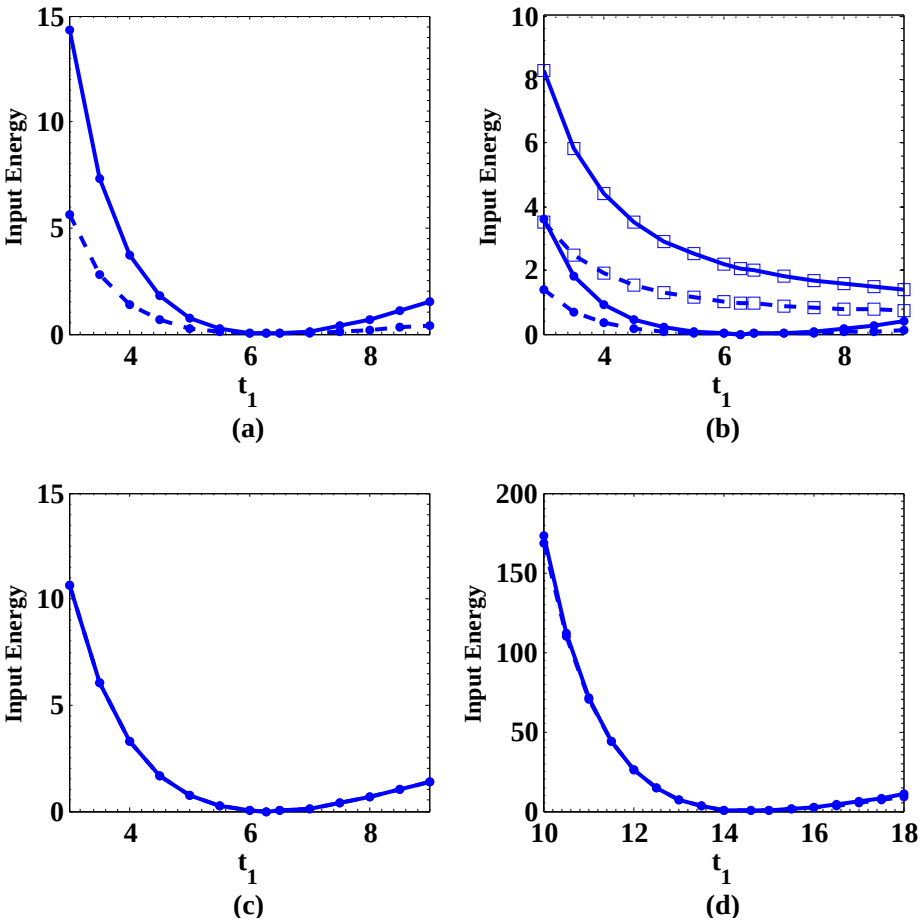


Figure 3.7: Input energy as a function of the target spike time t_1 with (solid lines) and without (dashed lines) the charge-balance constraint. (a) SNIPER model, (b) theta neuron model with $I_b = +0.25$ shown with dot markers and $I_b = -0.25$ shown with square markers, (c) sinusoidal model, and (d) Hodgkin-Huxley model. For the theta neuron with $I_b = -0.25$, shown in (b) with square markers, there is no point with zero energy. This is due to the fact that the theta neuron with negative baseline current will not fire periodically in the absence of input. So in order to make it fire periodically one has to input a stimulus, meaning that one has to spend energy. The zero energy point for this plot would be when $t_1 \to \infty$ which means the neuron not firing at all.

Chapter 4

Single Input Optimal Control for Globally Coupled Neuron Networks

In this chapter we employ discrete dynamic programming as an efficient mathematical optimization method for numerically solving the problem of desynchronizing a network of pathologically synchronized, globally (all-to-all) coupled phase neurons.

The organization of this chapter is as follows. In section 4.1, we present a brief introduction to the problem. In section 4.2, after giving the general form for phase models of networks of coupled neurons, we briefly introduce the Kuramoto model and then elaborate on deriving the Hodgkin-Huxley phase model for networks of coupled neurons. In Section 4.3, we present the setup of the discretized model and

explain, in detail, our control strategy in the dynamic programming framework. We present the results in Section 4.4 along with some discussion. In Section 4.5, we consider applying dynamic programming to a Hodgkin-Huxley phase model driven by a multiplicative control. Finally, in Section 4.6 we draw conclusions and discuss some future directions.

The main results from this chapter were published in [45].

4.1 Introduction

When it comes to controlling phase models of neurons, most of the work in the literature has been either on a single neuron level [16, 17, 29, 43, 52–54] or, if on the population level, multiple inputs have been allowed to the system [19, 55]. However, since there is typically only one electrode implanted into the brain, electrical DBS in its current state is limited by the number of input stimuli that it can deliver. In this chapter, we look at synchronized networks of coupled neurons that are subject to only one single input and investigate the ability of the input in desynchronizing different network examples using a dynamic programming approach.

In dynamic programming, a cost function is defined that is to be minimized over the entire time horizon. From this cost function, one finds the value functions $V_1(x), V_2(x), \cdots, V_K(x)$ for all states $x \in \mathcal{X}^d$, where the indices $1, 2, \cdots, K$ represent time and $\mathcal{X}^d$ is the state space. These value functions indicate the *cost-to-go* from time k at state x to the end time. Therefore, by computing the value functions, one has knowledge of the cost incurred for accomplishing a certain desired

task starting from any point in the time domain and any point in the state space. The value functions are recursively computed by defining the value of the cost-to-go at time step $K+1$, $V_{K+1}(x)$ [47, 56].

4.2 The mathematical model

For a general network of N weakly coupled phase neurons (or, more generally, oscillators) we have [57]:

$$\dot{\theta}_i = \omega_i + \sum_{j=1}^{N} \mathcal{F}_{ij}(\theta_j - \theta_i), \quad i = 1, 2, \cdots, N,$$

where $\theta_i \in (0, 2\pi]$ is the phase of neuron i, ω_i is its natural frequency of spiking, and $\mathcal{F}_{ij}(\cdot)$ is the 2π-periodic coupling function acting on neuron i from neuron j.

As mentioned earlier, we restrict the problem by only allowing a single control input, with the assumption that this control input is an additive control input that, without loss of generality, is applied to the N^{th} neuron in the network. Later, in section 4.5 we briefly explore the case of a multiplicative control as well. We also assume in this study that all neurons are identical, hence they all have identical natural frequencies ω, and that the functional form of the coupling between any pair of neurons is identical, but the strength of this coupling may be different. This yields the controlled form of the coupled phase neuron system as:

$$\dot{\theta}_i = \omega + \sum_{j=1}^{N} \alpha_{ij} f(\theta_j - \theta_i) + \delta_{iN} u(t), \tag{4.1}$$

for $i = 1, 2, \cdots, N$. Here we have assumed that $\mathcal{F}_{ij}(\cdot) = \alpha_{ij} f(\cdot)$ where α_{ij} is the coupling strength from neuron j to neuron i, $f(\cdot)$ is the 2π-periodic coupling

function acting between every pair of neurons, δ is the Kronecker delta function, and $u(t)$ is the single control input.

The coupling function $f(\cdot)$ distinguishes between different models. For the Kuramoto model, $f(\cdot) = \sin(\cdot)$ which yields

$$\dot{\theta}_i = \omega + \sum_{j=1}^{N} \alpha_{ij} \sin(\theta_j - \theta_i) + \delta_{iN} u(t). \tag{4.2}$$

Equation (4.2) characterizes a system of globally heterogeneously coupled Kuramoto phase neurons driven by a single control input. We should mention that Kuramoto's phase model can be applied to many other oscillator systems and is not specific to neurons. Applications range from biology [58–61], to physics and engineering [62–66]. A good review on the Kuramoto model is given in [57].

For the rest of this section, we focus on deriving an example coupling function for Hodgkin-Huxley's model for neurons [50]. This model, presented in 1952, was derived to model Loligo squid's giant axon. Since it is the most widely used model in the literature for modeling the dynamics of neurons, we chose to consider it in the present study. The specifics of this model are given in Appendix C. For $I_b = 10\ \mu\text{A/cm}^2$, which we will use in the following, the period of oscillations is $T_s = 14.63$ ms. In the oscillatory mode, the neuron periodically gives action potentials in the form of voltage spikes.

When grouped together, the spiking of each neuron affects the voltage dynamics of the neighboring neurons as they sense the spike as an input. This interaction is referred to as *electrotonic* coupling. It can be mathematically modeled by modifying the voltage equation in (C.1):

$$\dot{V}_i = \left(I_b + I_g + I(t)\right)/c \; + \; \alpha_e \sum_{j=1}^{N} (V_j - V_i), \tag{4.3}$$

where α_e is the electrotonic coupling strength between the neurons. We assume that the network is weakly coupled and hence $\alpha_e = \mathcal{O}(\epsilon)$, where $\epsilon > 0$ is a small number. It should be pointed out that the effect of electrotonic coupling manifests itself only in the voltage dynamics and not in the dynamics of the gating variables [67]. We also note that the techniques that we consider in this chapter could also be used for synaptic coupling, and in fact any type of coupling for which a phase reduction can be performed. We chose to consider electrotonic coupling for simplicity of presentation.

In order to find the reduced phase model [31, 40, 46, 68] for the Hodgkin-Huxley coupled neuron system, we first consider (C.1) in the absence of any external input stimulus $I(t)$. This system oscillates with period T. To characterize this oscillation, following [32, 69], a phase variable $\theta \in (0, 2\pi]$ is defined such that

$$\frac{d\theta}{dt} = \omega = \frac{2\pi}{T}.$$

Now if we define $\mathbf{X} = [V, m, h, n]^{\mathrm{T}}$ as the state vector for the system, we can combine (C.1) and (4.3) and write the coupled system's equations as

$$\frac{d\mathbf{X}_i}{dt} = \mathbf{F}(\mathbf{X}_i) + \epsilon \sum_{j=1}^{N} \mathbf{p}(\mathbf{X}_i, \mathbf{X}_j),$$

for $i = 1, \cdots, N$. In this equation $\mathbf{F}(\mathbf{X}_i)$ represents the dynamics of neuron i in the absence of any external stimuli or coupling effects, and $\mathbf{p}(\mathbf{X}_i, \mathbf{X}_j)$ accounts for

the effect that neuron j has on neuron i due to coupling. We can write [40]

$$\begin{aligned}\frac{d\theta_i}{dt} &= \frac{\partial\theta_i}{\partial\mathbf{X}_i}\cdot\frac{d\mathbf{X}_i}{dt} \\ &= \frac{\partial\theta_i}{\partial\mathbf{X}_i}\cdot\left(\mathbf{F}(\mathbf{X}_i)+\epsilon\sum_{j=1}^{N}\mathbf{p}(\mathbf{X}_i,\mathbf{X}_j)\right) \\ &= \omega+\epsilon\frac{\partial\theta_i}{\partial\mathbf{X}_i}\cdot\sum_{j=1}^{N}\mathbf{p}(\mathbf{X}_i,\mathbf{X}_j),\end{aligned}$$

for $i = 1, \cdots, N$. We note that in the absence of coupling terms, we get $\frac{d\theta_i}{dt} = \omega$ where we have assumed that all neurons have identical natural frequencies.

Since we have assumed weak coupling, we argue that each neuron, even under the influence of coupling, remains close to its stable periodic orbit that characterizes its periodic spiking in its phase space, i.e., $T \approx T_s$. In addition, there is a mapping from the states to the phase variable on the periodic orbit and so with the assumption of weak coupling (small ϵ), we can consider the effect of coupling as a perturbation to the oscillatory neuron and write

$$\frac{d\theta_i}{dt} = \omega + \epsilon\mathbf{Z}(\theta_i)\cdot\sum_{j=1}^{N}\mathbf{p}(\theta_i,\theta_j), \tag{4.4}$$

where $\mathbf{p}(\theta_i,\theta_j) = \mathbf{p}(\mathbf{X}_{po}(\theta_i),\mathbf{X}_{po}(\theta_j))$, with $\mathbf{X}_{po}$ denoting the periodic orbit. $\mathbf{Z}(\theta_i)$ represents the gradient of the phase variable θ_i with respect to the state variables $[V, m, h, n]^{\mathrm{T}}$ on the periodic orbit and is defined as

$$\mathbf{Z}(\theta_i) = \left.\frac{\partial\theta_i}{\partial\mathbf{X}_i}\right|_{\mathbf{X}_{po}(\theta_i)}.$$

It turns out that since the coupling term $\mathbf{p}(\mathbf{X}_{po}(\theta_i), \mathbf{X}_{po}(\theta_j))$ is dependent only on the first state variable V (see (4.3)), only the first entry of $\mathbf{Z}(\theta_i)$ comes into play. This first entry, denoted by $Z_V(\theta_i)$, is called the Phase Response Curve (PRC) of the i^{th} neuron. For the Hodgkin-Huxley equations, as mentioned before, $Z_V(\theta_i)$ can be computed numerically using the software XPPAUT available as open source software on the web [1,2]. The PRC for the Hodgkin-Huxley equations is shown in Figure 3.5. For notational convenience, we continue using the vector form of $\mathbf{Z}(\theta_i)$ in the equations.

We can simplify (4.4) by first defining $\theta_i = \omega t + \phi_i$. Substituting this into (4.4) one can take out the mean field effect of ω and write

$$\frac{d\phi_i}{dt} = \epsilon \mathbf{Z}(\phi_i + \omega t) \cdot \sum_{j=1}^{N} \mathbf{p}(\phi_i + \omega t, \phi_j + \omega t). \tag{4.5}$$

Using the averaging theorem from [70] and [71], we get the approximate equation

$$\frac{d\phi_i}{dt} = \frac{\epsilon}{T} \int_0^T \left[\mathbf{Z}(\phi_i + \omega\tau) \cdot \sum_{j=1}^{N} \mathbf{p}(\phi_i + \omega\tau, \phi_j + \omega\tau) \right] d\tau.$$

Letting $s = \phi_i + \omega\tau$, we get

$$\frac{d\phi_i}{dt} = \frac{\epsilon}{2\pi} \sum_{j=1}^{N} \int_0^{2\pi} \left[\mathbf{Z}(s) \cdot \mathbf{p}(s, \phi_j - \phi_i + s) \right] ds,$$

which in terms of θ_i is

$$\frac{d\theta_i}{dt} = \omega + \frac{\epsilon}{2\pi} \sum_{j=1}^{N} \int_0^{2\pi} \left[\mathbf{Z}(s) \cdot \mathbf{p}(s, \theta_j - \theta_i + s) \right] ds. \tag{4.6}$$

It is worth pointing out that the righthand sides of these equations are functions

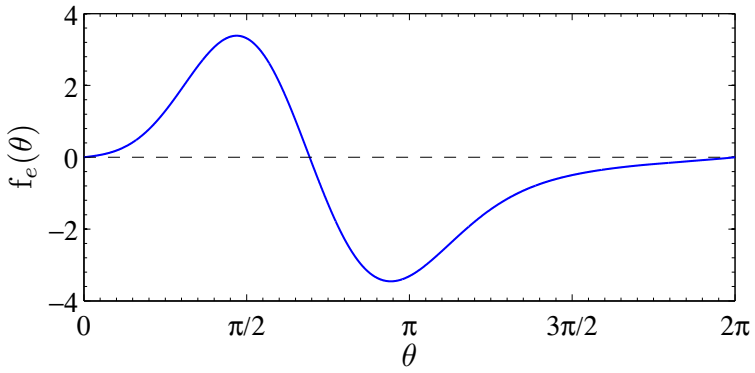


Figure 4.1: Electrotonic coupling function for the Hodgkin-Huxley equations with $I_b = 10\ \mu\text{A/cm}^2$.

of the phase differences only. If we rewrite equation (4.6) more succinctly as

$$\frac{d\theta_i}{dt} = \omega + \alpha_e \sum_{j=1}^{N} f_e(\theta_j - \theta_i), \tag{4.7}$$

with $f_e(\cdot)$ denoting the electrotonic coupling, considering (4.3) and the fact that $\alpha_e = \mathcal{O}(\epsilon)$, we find $f_e(\cdot)$ to be

$$f_e(\theta) = \frac{1}{2\pi} \int_0^{2\pi} Z_V(s)(V_j(\theta + s) - V_i(s))ds,$$

where $Z_V(\cdot)$ is the PRC for the Hodgkin-Huxley equation as shown in Figure 3.5. Figure 4.1 shows the plot of this coupling function computed numerically. $Z_V(\cdot)$ from Figure 3.5 has been used in producing this figure.

In the presence of external control, an additional term would appear in (4.7) that would be a function of the external control stimulus $I(t)$ in (C.1), as we discuss more carefully in section 4.5. For now, we simplify the problem by considering the control input to be an additive $u(t)$ to (4.7) that incorporates the appropriate functional relationship with $I(t)$. Furthermore, we restrict the problem by only allowing a single control input to the system that, without loss of generality, is

applied to the N^{th} neuron in the network. In order to incorporate heterogeneity of electrotonic coupling, instead of a common α_e, we consider different coupling strengths between different neuron pairs and rewrite (4.7) as

$$\frac{d\theta_i}{dt} = \omega + \sum_{j=1}^{N} \alpha_{ij} f_e(\theta_j - \theta_i) + \delta_{iN} u(t), \tag{4.8}$$

where δ_{iN} is the Kronecker delta function and u represents the control input. This equation is similar in form to (4.2), but with the coupling function being that shown in Figure 4.1.

Now in order to further simplify the equations, we note that the righthand side of (4.2) and (4.8), and in general (4.1), are only in terms of phase differences. This allows us to find the phase difference dynamics for these systems and hence, by defining $\psi_i = \theta_i - \theta_1$ for $i = 2, 3, \cdots, N$, reduce the system dimension by one. This yields the following general phase difference equations:

$$\begin{aligned} \dot{\psi}_i &= \alpha_{i1} f(-\psi_i) - \alpha_{11} f(0) \\ &\quad + \textstyle\sum_{j=2}^{N} \left[\alpha_{ij} f(\psi_j - \psi_i) - \alpha_{1j} f(\psi_j)\right] + \delta_{iN} u(t), \end{aligned} \tag{4.9}$$

for $i = 2, 3, \cdots, N$. In these equations, $f(\cdot)$ can be any 2π-periodic coupling function. Equations (4.9) will be the basis on which we design and present the desynchronizing control law in subsequent sections.

4.3 Discretization and Control

4.3.1 Discretization

In order to compute the desynchronizing control input for the system (4.9) numerically, we need to discretize these equations. To this end, we define $d\psi$ to be the grid size for the phase differences ψ_i, and du to be the step size for the control input u. This yields the phase differences and control spaces $\psi_i^d = \{d\psi, 2d\psi, \cdots, 2\pi\}$ and $\mathcal{U}^d = \{-u_{max}, \cdots, -du, 0, du, \cdots, u_{max}\}$, for $i = 2, 3, \cdots, N$. We define the discrete state space $\mathcal{X}^d$ such that it has a state variable for every possible vector $(\psi_2, \psi_3, \cdots, \psi_N)$. Enumerating the states in the discrete state space yields $\mathcal{X}^d = \{1, 2, \cdots, n_{\mathcal{X}^d}\}$, where $n_{\mathcal{X}^d} = (\frac{2\pi}{d\psi})^{N-1}$ is the total number of states. We assign the first state in $\mathcal{X}^d$ to the state $(\psi_2, \psi_3, \cdots, \psi_N) = (d\psi, d\psi, \cdots, d\psi)$. Subsequent states in $\mathcal{X}^d$ are assigned to vectors of $(\psi_2, \psi_3, \cdots, \psi_N)$ in which, for $j = 2, \cdots, N-1$, each ψ_j increments by $d\psi$ when ψ_{j+1} has finished marching through its minimum to its maximum with $d\psi$ increments. This way, the $n_{\mathcal{X}^d}$ state of $\mathcal{X}^d$ corresponds to $(\psi_2, \psi_3, \cdots, \psi_N) = (2\pi, 2\pi, \cdots, 2\pi)$. At each instant of time, the state of the system is one of the states in the discrete state space $\mathcal{X}^d$. We choose $d\psi$ to be a divisor of 2π to have an integer $n_{\mathcal{X}^d}$.

4.3.2 Discrete Time Dynamic Programming

Considering (4.9), one can write the following general difference equation:

$$x_{k+1} = F_k(x_k, u_k), \qquad \forall k \in \{1, 2, \cdots, K\}, \tag{4.10}$$

where $x_k \in \mathcal{X}^d$ denotes the state of the system, corresponding to a case of $(\psi_2, \psi_3, \cdots, \psi_N)$, at time k, $u_k \in \mathcal{U}^d$ is the control input at time k, and $F_k(\,\cdot\,,\,\cdot\,)$ gives the dynamics of the system at time k computed by integrating the righthand side of (4.9) numerically. K is the end time.

The objective is to find a sequence u_k for all $k \in \{1, 2, \cdots, K\}$, such that the state in (4.10) approaches a value for which the phase difference between any two neurons is at least as big as a predetermined amount $\Delta_{min} \in [0, \frac{2\pi}{N}]$. The ideal case for desynchronizing the firing times for the population is when the state x in (4.10) approaches $x_{splay} = \{x |\, \psi_i \in \{\frac{2\pi}{N}, \frac{4\pi}{N}, \cdots, 2\pi\}\ \&\ \psi_i \neq \psi_j \quad \forall i, j = 2, 3, \cdots, N\,,\, i \neq j\}$. For a system of N neurons, there are $(N-1)!$ splay states.

More concretely, we define the target set

$$\mathcal{X}_{targ} = \{x |\, (\psi_i\ \&\ (2\pi - \psi_i)\ \&\ |\psi_i - \psi_j|) > \Delta_{min}\}\,, \tag{4.11}$$

for all $i, j = 2, 3, \cdots, N$, $i \neq j$. If $x_K \in \mathcal{X}_{targ}$, then the system is considered to be desynchronized. In order to formulate the problem, we define the following time additive cost function:

$$J = \sum_{k=1}^{K} \gamma^2 u_k^2 \ + \ R(x_{K+1}), \tag{4.12}$$

where $\gamma > 0$ is a scalar penalizing factor and

$$R(x_{K+1}) = \left| 1 + \sum_{i=2}^{i=N} e^{j\psi_{i,K+1}} \right|, \tag{4.13}$$

where here $j = \sqrt{-1}$ and the $\psi_{i,K+1}$'s are the phase differences at time $K+1$ associated with state x_{K+1} as obtained from (4.10). $R(x_{K+1}) \in \mathbb{R}_{[0,N]}$ is known as the *end*

point cost. The objective is to find a sequence $u_k \in \mathcal{U}^d$ for $k = \{1, 2, \cdots, K\}$ such that (4.12) is minimized subject to (4.10). It is worth mentioning that $R(x_{K+1})$ is a variation of the order parameter for systems of coupled oscillators [57] rearranged to fit the phase difference system here, and is minimized if the neurons desynchronize fully by assuming one of the splay states, and is maximized if the neurons synchronize. Therefore, by minimizing (4.12) along the solutions of (4.10) the state x is driven towards $x_{splay} \subset \mathcal{X}_{targ}$. We note that although reaching the splay state would be ideal in maximizing the phase differences, any state within the spectrum of in-phase and splay states may be achieved and as long as the phase differences of the neurons are greater than Δ_{min}, the system is considered desynchronized. Considering the bounded control, the values of the coupling strengths, the discretization error due to meshing the phase space, and the limited time of control application, it is likely that the optimal controller results in a final state that is not one of the splay states. Another point to make here is that by defining the cost function as in (4.12), we are emphasizing desynchronization at the last time step, K. By choosing K to correspond to the spiking instant of the population, one can hope for achieving desynchronization of spikes.

To cast the problem in the dynamic programming format, we argue that the cost presented in (4.12) is composed of the cost incurred from the current time step to the next, plus the *cost-to-go* from the next time step to the final time. The cost-to-go (also known as the *value function*) from a state x at time l, denoted as $V_l(x)$, can be written as follows [47]:

$$
\begin{aligned}
V_l(x) &= \inf_{u_k \in \mathcal{U}^d, \forall k \geq l} \left(\sum_{k=l}^{K} \gamma^2 u_k^2 + R(x_{K+1}) \right) \\
&= \inf_{u_k \in \mathcal{U}^d, \forall k \geq l} \left(\gamma^2 u_l^2 + \sum_{k=l+1}^{K} \gamma^2 u_k^2 + R(x_{K+1}) \right) \\
&= \inf_{u_l \in \mathcal{U}^d} \left(\gamma^2 u_l^2 + \inf_{u_k \in \mathcal{U}^d, \forall k \geq l+1} \left(\sum_{k=l+1}^{K} \gamma^2 u_k^2 + R(x_{K+1}) \right) \right).
\end{aligned}
$$

The inner infimum on the righthand side of the above equation is exactly the cost-to-go starting at time $k = l + 1$ from the state $x_{l+1} = F_l(x, u_l)$. So we can write

$$
V_l(x) = \inf_{u_l \in \mathcal{U}^d} \left(\gamma^2 u_l^2 + V_{l+1}(F_l(x, u_l)) \right), \tag{4.14}
$$

for all $x \in \mathcal{X}^d$. This equation is valid for all $l \in \{1, 2, \cdots, K\}$ when we define

$$
V_{K+1}(x) = \begin{cases} R(x_{K+1}) & \text{if} \quad x_{K+1} \in \mathcal{X}_{targ}, \\ +\infty & \text{if} \quad x_{K+1} \notin \mathcal{X}_{targ}, \end{cases} \tag{4.15}
$$

for all $x \in \mathcal{X}^d$. With this, the optimal control and trajectory will be

$$
u_k^* = \arg \min_{u_k \in \mathcal{U}^d} \left(\gamma^2 u_k^2 + V_{k+1}(F_k(x_k^*, u_k)) \right), \tag{4.16}
$$

$$
x_{k+1}^* = F_k(x_k^*, u_k^*), \quad x_1^* = x_1, \tag{4.17}
$$

for all $k \in \{1, 2, \cdots, K\}$.

The final time K in the above formulation is chosen to be the time beyond which the control law would not be applied. We note that $x_{K+1} = F_k(x_K, u_K)$ and so if the state of the system is not going to fall within the set $\mathcal{X}_{targ}$ even by applying the optimum control at time K, u_K^*, then it is considered to be not desynchronizable and the cost-to-go that is associated to it for time step $K+1$ is infinity (see (4.15)).

However, if with the control sequence $u_1^*, u_2^*, \cdots, u_K^*$ the system falls within the set $\mathcal{X}_{targ}$ at time $K+1$, then it would be considered desynchronized and a cost-to-go of $R(x_{K+1})$ is associated to it for time step $K+1$. This formulation is known as *fixed termination time* dynamic programming.

4.3.3 Implementation in Matlab

Dynamic programming characterized by equations (4.14)-(4.17) forms a computationally efficient way to compute the cost-to-go for a system throughout the time and state domains recursively. After initializing $V_{K+1}(x)$, one can first perform a backward iteration to compute $V_1(x)$ for all $x \in \mathcal{X}^d$ [56]:

```
for k=K:-1:1
 V{k} = min(G{k}+V{k+1}(F{k}),[],2);
end
```

Then, given an initial condition $x(1)$, a forward iteration loop will yield the optimal control and state trajectories:

```
for k=1:K
 [dummy, u] = min(G{k}(x(k),:)+
                  V{k+1}(F{k}(x(k),:))',[],2);
 x(k+1) = F{k}(x(k),u);
end
```

A more detailed Matlab code is given in Appendix D.

4.4 Example

As an example, we solve this problem for a network of three neurons. In accordance with (4.9), we present the phase difference equations for a network of three neurons as

$$\begin{aligned} \dot{\psi}_2 &= -2\alpha_{12}\sin(\psi_2) \\ &+ \alpha_{23}\sin(\psi_3-\psi_2)-\alpha_{13}\sin(\psi_3), \\ \dot{\psi}_3 &= -2\alpha_{13}\sin(\psi_3) \\ &+ \alpha_{23}\sin(\psi_2-\psi_3)-\alpha_{12}\sin(\psi_2)+u, \end{aligned} \tag{4.18}$$

for the Kuramoto system and

$$\begin{aligned} \dot{\psi}_2 &= \alpha_{12}\left(f_e(-\psi_2)-f_e(\psi_2)\right) \\ &+ \alpha_{23}f_e(\psi_3-\psi_2)-\alpha_{13}f_e(\psi_3), \\ \dot{\psi}_3 &= \alpha_{13}\left(f_e(-\psi_3)-f_e(\psi_3)\right) \\ &+ \alpha_{23}f_e(\psi_2-\psi_3)-\alpha_{12}f_e(\psi_2)+u, \end{aligned} \tag{4.19}$$

for the Hodgkin-Huxley system. We note that we have assumed symmetry for the coupling strengths, i.e., $\alpha_{ij} = \alpha_{ji}$. The two splay states for this system are $(\psi_2, \psi_3) = (\frac{2\pi}{3}, \frac{4\pi}{3})$ and $(\psi_2, \psi_3) = (\frac{4\pi}{3}, \frac{2\pi}{3})$.

With positive values for α_{ij}, and in the absence of control, these systems synchronize resulting in $\psi_2 = \psi_3$ at all times. In order to find a desynchronizing control for these systems, one needs knowledge of α_{ij} values. Several experimental and numerical studies have been carried out to find these coupling strengths [72–77]. The values that are suggested in these studies generally fall within the range of $[0\,,\,1]$. Since the true values of the coupling strengths are unknown, we

allow each α to change within this interval. For the purpose of simulations we consider $0.1 \leq \alpha_{ij} \leq 1$ with 0.1 steps. This gives 10 possibilities for each α resulting in 1000 different combinations of $(\alpha_{12}, \alpha_{13}, \alpha_{23})$. We note that since we are considering all-to-all coupling, we have omitted zero values for the α's. In addition, allowing different combinations for $(\alpha_{12}, \alpha_{13}, \alpha_{23})$ results in situations where the α values are very close to (or far from) each other, which resembles the different coupling strengths among an actual neural population.

To find the desynchronizing control law for (4.18) and (4.19), we apply the fixed termination time dynamic programming formulation and find the optimal control law for every case of $(\alpha_{12}, \alpha_{13}, \alpha_{23})$. Depending on the maximum allowable value for the control input u_{max}, and the minimum acceptable phase difference Δ_{min}, a control sequence $u_1^*, u_2^*, \cdots, u_K^*$ can be found for some of the cases. This means that for those cases of $(\alpha_{12}, \alpha_{13}, \alpha_{23})$, a control input can be found that can achieve, for the system, a phase desynchronization of at least Δ_{min}. The simulations were carried out for $\Delta_{min} = 10^\circ$ and using three different values for u_{max}, namely $u_{max} \in \{1, 2, 3\}$. The penalizing constant in (4.12) is taken to be $\gamma = 10^{-4}$. The statistics for the ratio of desynchronizable cases to the total number of cases (which is 1000) for both the Kuramoto system (4.18) and the Hodgkin-Huxley system (4.19) is shown in Table 4.1. The period of control application was taken to be approximately equal to that of the uncontrolled neurons. The initial condition for all simulation results shown here is the synchronized state where $\psi_2 = \psi_3 = 2\pi$. As expected, Table 4.1 shows an increase in the percentage of desynchronizable cases

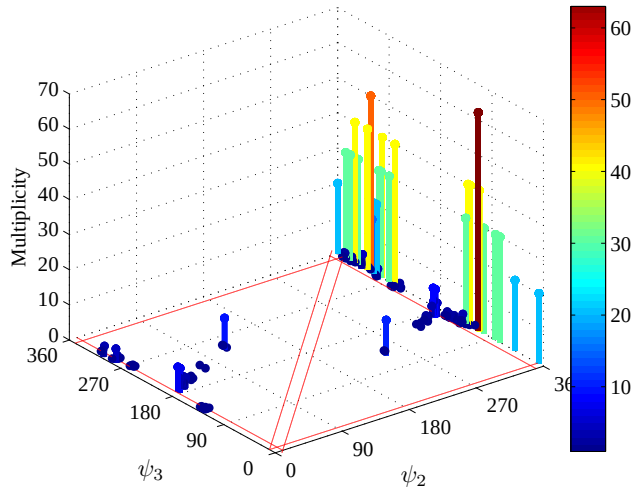
Table 4.1: The percentage of different $(\alpha_{12}, \alpha_{13}, \alpha_{23})$ cases for which a desynchronizing control law exists when employing fixed termination time dynamic programming with a synchronized initial condition. The simulations were performed for $T_f = 6.28$ time units with $dt = 0.0349$ for the Kuramoto system and $T_f = 14.6$ with $dt = 0.08$ for the Hodgkin-Huxley system. In these simulations $\Delta_{min} = 10^\circ$ and $d\psi = 2^\circ$.

u_{max}	Kuramoto	Hodgkin-Huxley
1	0%	30.7%
2	6.6%	61.1%
3	11.2%	77.7%

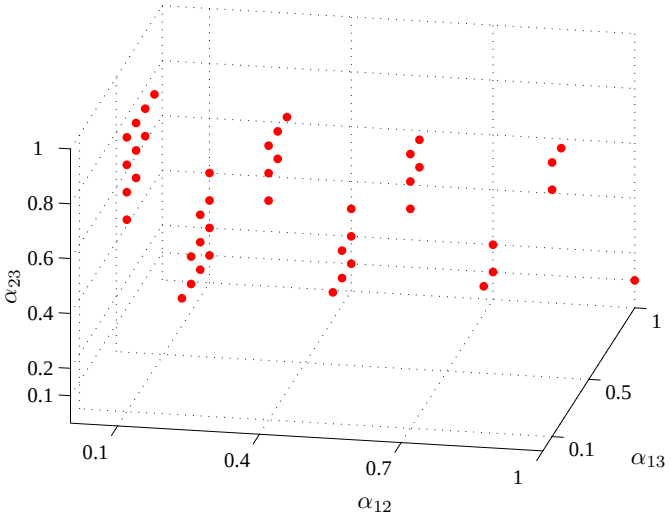
with increase in control authority u_{max}.

As an example, with $u_{max} = 3$ and $\Delta_{min} = 10^\circ$, Figure 4.2(a) shows the end state $(\psi_2(K), \psi_3(K))$ for all different cases of $(\alpha_{12}, \alpha_{13}, \alpha_{23})$ for the Kuramoto model (4.18). Each case has been subject to its own optimal control input computed through fixed termination time dynamic programming, equations (4.12), (4.14), and (4.15). We note that multiple cases can end up at a same location in the state space. Figure 4.2(b) shows four planes in the α space with points shown for those cases that were desynchronized. Figure 4.3 communicates similar information for the Hodgkin-Huxley system (4.19). It can be seen from figures 4.2(b) and 4.3(b) that as α_{12} increases, those $(\alpha_{12}, \alpha_{13}, \alpha_{23})$ combinations in which α_{13} and α_{23} values are close to each other have a lesser chance of being desynchronized. This is intuitively appealing, because the control is being applied to neuron 3 and so there needs to be a significant difference between the forces applied to neurons 1 and 2 (i.e., significant difference between α_{13} and α_{23}) for the control to even have a chance of overcoming the strong bond between neurons 1 and 2 (i.e., large α_{12}).

As an example, we have included the optimal control input and state trajectory

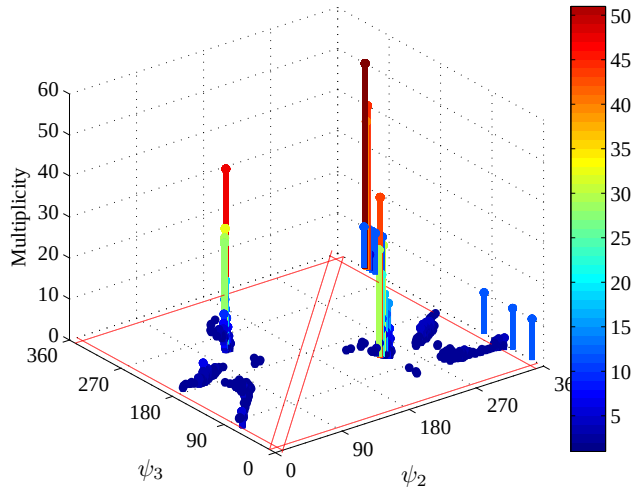


(a)

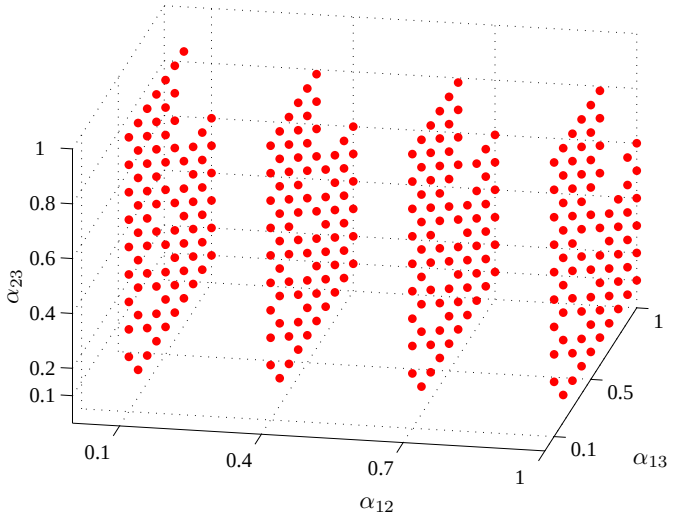


(b)

Figure 4.2: Simulation results for the Kuramoto system with $u_{max} = 3$ and $\Delta_{min} = 10°$: (a) The end state $(\psi_2(K), \psi_3(K))$ for all different cases of $(\alpha_{12}, \alpha_{13}, \alpha_{23})$. Each case has been subject to its own optimal control input computed through fixed termination time dynamic programming. (b) Four planes in the α space with points shown for those cases that were desynchronized.



(a)



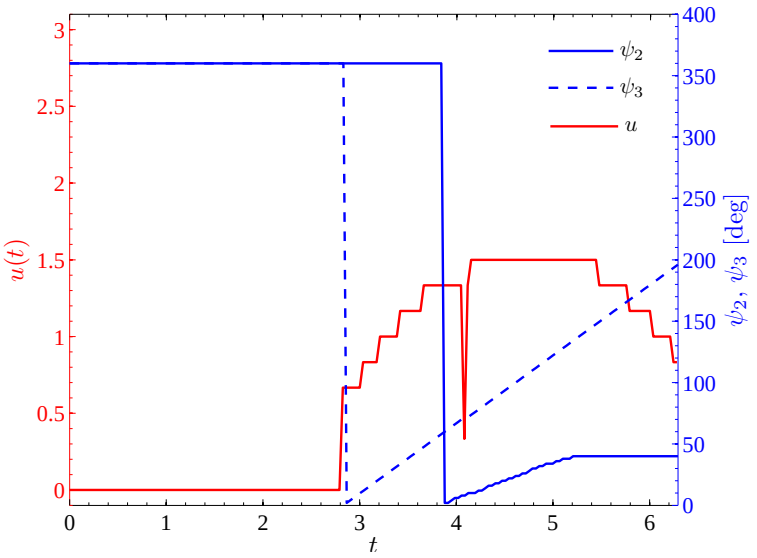
(b)

Figure 4.3: Simulation results for the Hodgkin-Huxley system with $u_{max} = 3$ and $\Delta_{min} = 10°$: (a) The end state $(\psi_2(K), \psi_3(K))$ for all different cases of $(\alpha_{12}, \alpha_{13}, \alpha_{23})$. Each case has been subject to its own optimal control input computed through fixed termination time dynamic programming. (b) Four planes in the α space with points shown for those cases that were desynchronized.

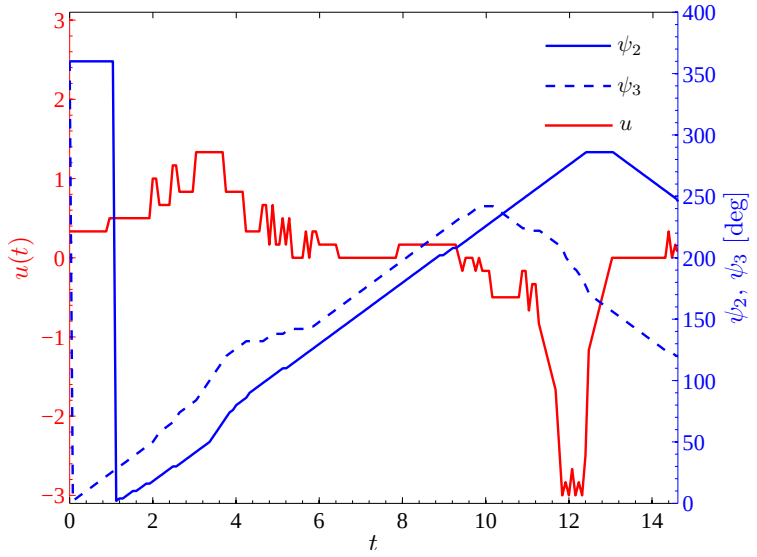
obtained for the specific case of $(\alpha_{12}, \alpha_{13}, \alpha_{23}) = (0.1, 0.1, 0.7)$ with $\Delta_{min} = 10^\circ$ and $u_{max} = 3$ for both the Kuramoto and the Hodgkin-Huxley systems in Figure 4.4(a) and (b), respectively. As it can be seen from these figures, the control has been able to take the system from the in-phase initial state to final states that are very close to the splay states.

We note that, due to the way that the cost function (4.12) is defined, if one increases the simulation time enough, the resulting control stimulus would remain zero at first before it starts to apply force at some specific point in time. This is because of the fact that the time additive cost is only on the control input and the state cost manifests itself only at the last time step. As a result, the controller does not apply any control until there is just enough time to optimally desynchronize the system with an optimal control. For the specific case shown in Figure 4.4, however, the duration of the control application has been chosen shorter and therefore, the controller has started to apply force from the beginning.

One important point in finding a control law for networks of coupled neurons is that we may not have any knowledge about the coupling strengths α_{ij}. In order to find a more general solution, given a u_{max} and Δ_{min}, we might consider all the control inputs that are able to achieve desynchronization, feed them through an averaging filter, find the average control input, and apply it to the entire 1000 different cases to investigate the probability of achieving desynchronization without having prior knowledge of α's. However, one can imagine that running a simple averaging routine on a number of different desynchronizing control sequences can



(a)



(b)

Figure 4.4: The optimal control input and state trajectory obtained for $(\alpha_{12}, \alpha_{13}, \alpha_{23}) = (0.1, 0.1, 0.7)$ with $\Delta_{min} = 10^\circ$ and $u_{max} = 3$ computed through fixed termination time dynamic programming for the (a) Kuramoto and (b) Hodgkin-Huxley systems.

result in an average sequence that is, due to potential cancellations, much smoother than each of the desynchronizing controls. This greatly reduces the chances of the averaged control in desynchronizing the network as it would, most likely, lack the important features of each control sequence.

In order to achieve a better averaged control stimulus for the general network, we first categorize the desynchronizing controls based on some *similarity index* and then find the average of each group. We then apply each of these averaged controls to the entire set of 1000 different cases and pick the one that results in the most desynchronizations as the best (final) answer. The similarity index for each desynchronizing control is a vector of length ten that is obtained as follows. We divide the time axis of the control sequence into ten equal intervals. We find the mean value of the control sequence for each of these intervals. If the mean falls in $[\frac{u_{max}}{2}, u_{max})$, we assign the number 2 to that interval. If the mean falls in $[0, \frac{u_{max}}{2})$, we assign the number 1 to that interval, if in $[-\frac{u_{max}}{2}, 0)$, we assign -1 and if in $[-u_{max}, -\frac{u_{max}}{2})$, we assign -2 to that interval. This way each desynchronizing control will have an index vector of length ten where each entry is chosen from the set $\{-2, -1, 1, 2\}$. We then group all controls that have the same index vector.

The averaged desynchronizing control inputs for different values of u_{max} are shown in Figure 4.5. The result of this investigation is summarized in Table 4.2. The initial condition for all simulation results shown here is the synchronized state where $\psi_2 = \psi_3 = 2\pi$. To show the performance of the average controls, we have included figures 4.6 and 4.7 for the Kuramoto and Hodgkin-Huxley systems, respectively.

Table 4.2: The probability of being able to desynchronize a synchronized network of three globally coupled neurons. The simulations were performed for $T_f = 6.28$ time units with $dt = 0.0349$ for the Kuramoto system and $T_f = 14.6$ with $dt = 0.08$ for the Hodgkin-Huxley system. In these simulations $\Delta_{min} = 10°$ and $d\psi = 2°$. The control input for each simulation is found by averaging the desynchronizing control inputs.

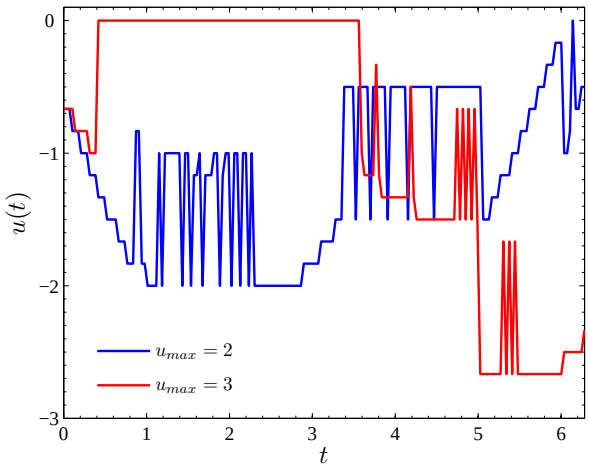
u_{max}	Kuramoto	Hodgkin-Huxley
1	0%	30.7%
2	4.9%	57.1%
3	8.4%	73.1%

These figures show simulation results for each of these systems using the $u_{max} = 3$ averaged controls shown in Figure 4.5(a) and (b).

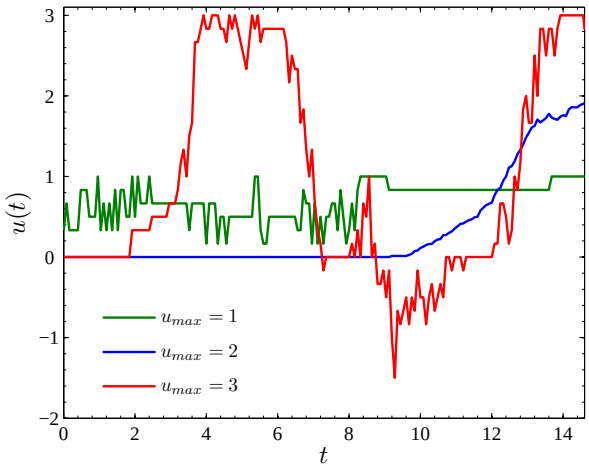
We see that for the Kuramoto and Hodgkin-Huxley systems, for $u_{max} = 3$ and $\Delta_{min} = 10°$ it is most likely to achieve desynchronization using the control input, labeled as $u_{max} = 3$, shown in figures 4.5(a) and 4.5(b), respectively.

4.5 Multiplicative Control in Hodgkin-Huxley Phase Model

In this section, we briefly investigate the control laws that one obtains by implementing dynamic programming on a different phase model for Hodgkin-Huxley neurons which allows closer comparison to the full Hodgkin-Huxley model. In writing (4.1), we simplified the problem by assuming that the effect of the input current stimulus $I(t)$ on the phase dynamics is like an additive control $u(t)$. However, when one carefully does the phase reduction, the input current stimulus appears in the phase reduced model having been multiplied by the PRC of the neuron to which

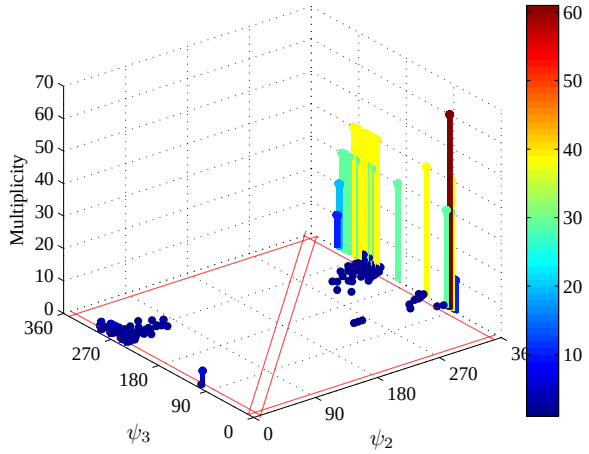


(a)

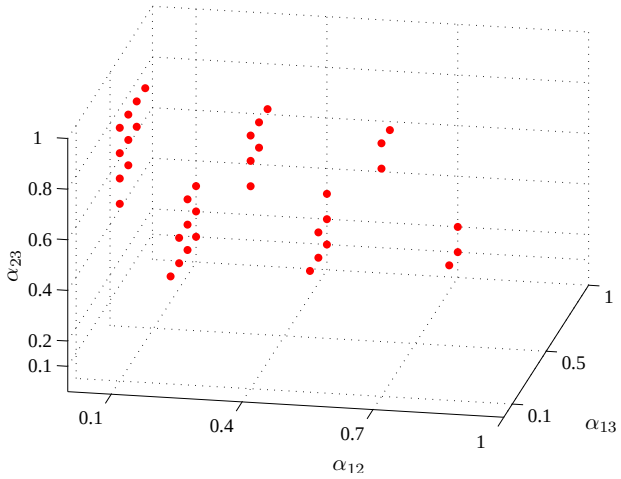


(b)

Figure 4.5: The averaged desynchronizing control inputs for $u_{max} = 3$ for (a) the Kuramoto system and (b) the Hodgkin-Huxley system.

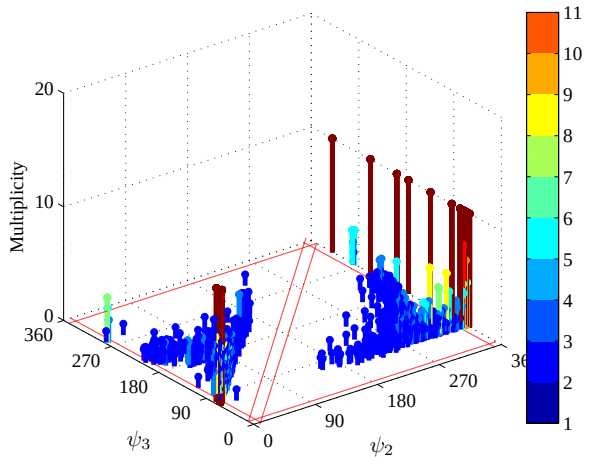


(a)

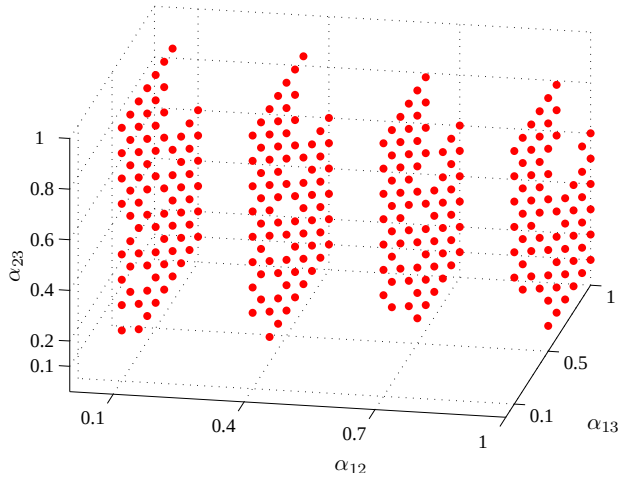


(b)

Figure 4.6: Simulation results for the Kuramoto system using the $u_{max} = 3$ averaged control shown in Figure 4.5(a) and with $\Delta_{min} = 10°$: (a) The end state $(\psi_2(K), \psi_3(K))$ for all different cases of $(\alpha_{12}, \alpha_{13}, \alpha_{23})$. (b) Four planes in the α space with points shown for those cases that were desynchronized using the common averaged control.



(a)



(b)

Figure 4.7: Simulation results for the Hodgkin-Huxley system using the $u_{max} = 3$ averaged control shown in Figure 4.5(b) and with $\Delta_{min} = 10°$: (a) The end state $(\psi_2(K), \psi_3(K))$ for all different cases of $(\alpha_{12}, \alpha_{13}, \alpha_{23})$. (b) Four planes in the α space with points shown for those cases that were desynchronized using the common averaged control.

it is being applied, that is, $u(\theta, t) = Z_V(\theta_N)I(t)$. Indeed, using a generalization of the general averaging theorem (Theorem 4.3.6 in [78]) when $I(t) = \mathcal{O}(\epsilon)$, one can show that when the coupling $\mathbf{p}$ in (4.4) and the external input $u(\theta, t)$ are Lipschitz continuous in the state variables and continuous in time, on the time scale of $\mathcal{O}(1/\epsilon)$ the solutions of (4.5) can be approximated with the solutions for the system in which only the coupling term is averaged [79]. So instead of (4.1) one gets

$$\frac{d\theta_i}{dt} = \omega + \sum_{j=1}^{N} \alpha_{ij} f_e(\theta_j - \theta_i) + \delta_{iN} Z_V(\theta_i) I(t). \tag{4.20}$$

We note that models of this form have been considered elsewhere, for example [43], but to our knowledge dynamic programming has not been applied to such a model before. A computational challenge is that (4.20) cannot be rewritten in terms of phase differences only, so it is necessary to discretize θ_1, θ_2, and θ_3 separately, which leads to a dynamic programming problem with one more dimension than the analogous problem for additive control considered above.

We have applied the fixed termination time dynamic programming formulation to find the optimal desynchronizing control law for (4.20) for the case of $(\alpha_{12}, \alpha_{13}, \alpha_{23}) = (0.2, 0.6, 0.2)$ and $I_{max} = 2\,\mu\text{A/cm}^2$, with the results shown in Figure 4.8(a). When this input is applied to the full model we see that the voltage traces of the three neurons become slightly desynchronized (Figure 4.8(b)). This result serves as a validating example for the phase reduction in the presence of external stimulus that is presented (without proof) in (4.20).

We have also found that when one increases the bound on $I(t)$ to $I_{max} = 10\,\mu\text{A/cm}^2$, the resulting control only achieves a 2-1 state, in which two neurons

have the same phase and the other has a different phase. It turns out that the algorithm determines that going to a 2-1 state that is spaced out on the phase circle results in less cost than going to any of the 1-1-1 states (in which all neurons have different phases) within the domain of capability of the control law. Interestingly, when the optimal control for this case is applied to the full Hodgkin-Huxley equations, it achieves an appreciable desynchronization (see Figure 4.9).

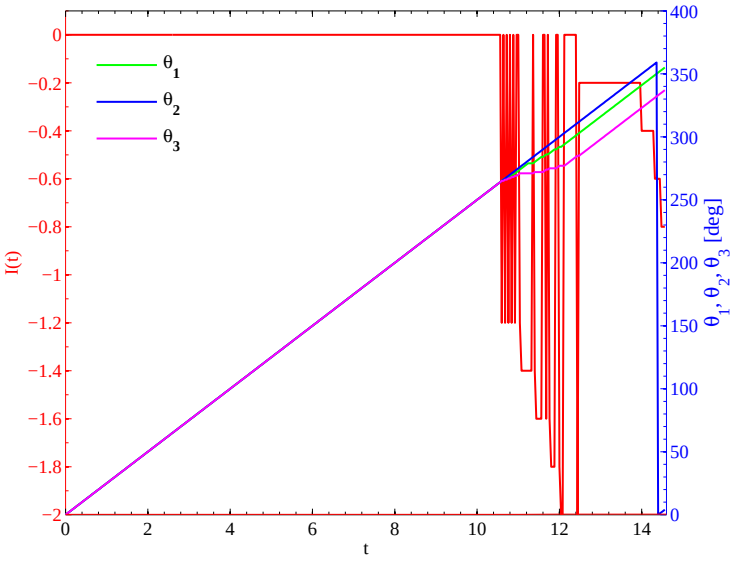
We note that the major drawback for dynamic programming is that in this method the size of the state vector grows exponentially with the number of neurons in the system. Therefore, given one's computational power and resources, there are limitations as to how small the mesh size can be. In these simulations we have set the mesh size to be reasonably small, but we can still see the effect of the mesh size on the accuracy of the output results. For example, we have observed cases where, in the absence of any input, the weak coupling that should be acting as a synchronizing force for the neurons is actually unable to fully synchronize them. This is because as the phases of the neurons get closer to each other, the dynamical contribution from the coupling becomes smaller, eventually becoming so small that it is unable to exert enough force to move the neurons over the boundaries of their bins. An obvious future direction for this method would be to find efficient ways to perform these computations with reduced mesh size to avoid such issues in these systems. One observation that might help is that the computed controls that we find tend to be zero for the first half of the period of the neurons, starting right after they have fired. Intuitively, one can let the system evolve according to its

natural dynamics until the control is needed and most effective. This suggests that we could start our initial condition at $\theta_1 = \theta_2 = \theta_3 = 180$ degrees at a time equal to half the period of the periodic orbit, assuming zero input for the evolution from $\theta_1 = \theta_2 = \theta_3 = 0$ up to this point, thereby freeing up computational resources for the consideration of smaller bins.

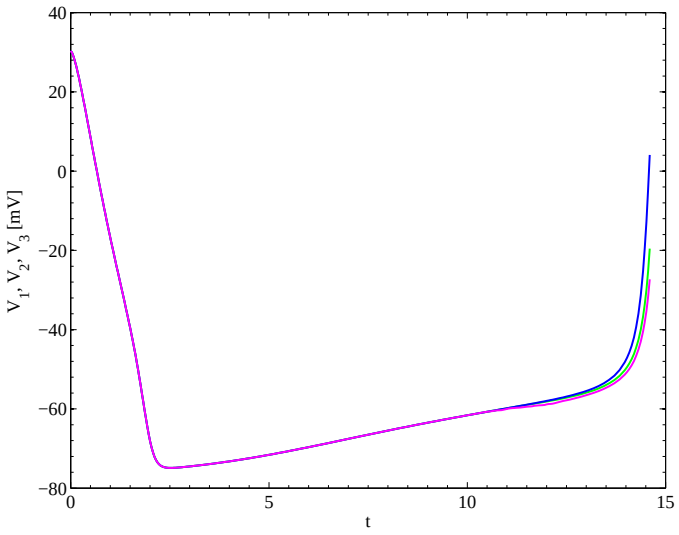
4.6 Conclusion and Future Directions

We have considered the problem of desynchronizing a network of pathologically synchronized globally coupled phase neurons using optimal control techniques. We used the Kuramoto model and a reduced phase model derived for a network of N Hodgkin-Huxley neurons under weak global electrotonic coupling as the basis for our control design. We only allowed one bounded control input to one of the neurons in the system, and we have assumed observability of all phases at all times. We introduced discrete dynamic programming as an efficient mathematical optimization method for numerically solving this problem.

For both the Kuramoto and Hodgkin-Huxley models, the desynchronization problem was solved for a network of three coupled neurons. Since the coupling strengths between the neurons are in practice unknown, a spectrum of different coupling strengths was considered. For some combinations of coupling strengths there exists a desynchronizing control, while for some there does not. The period of control application was taken to be approximately equal to that of the uncontrolled neurons. The different desynchronizing control laws were then categorized and

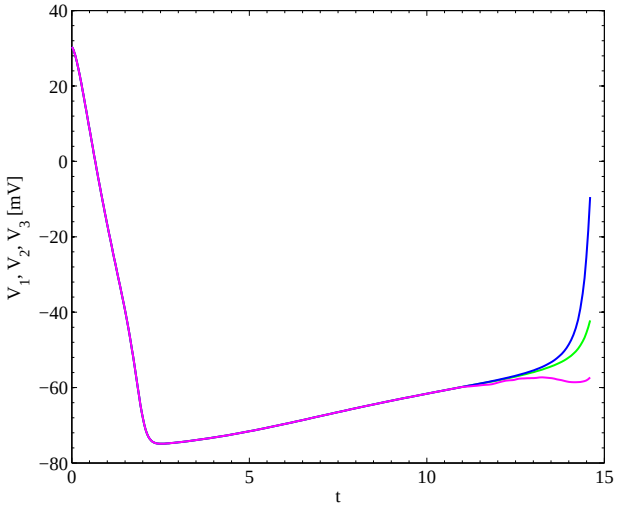


(a)

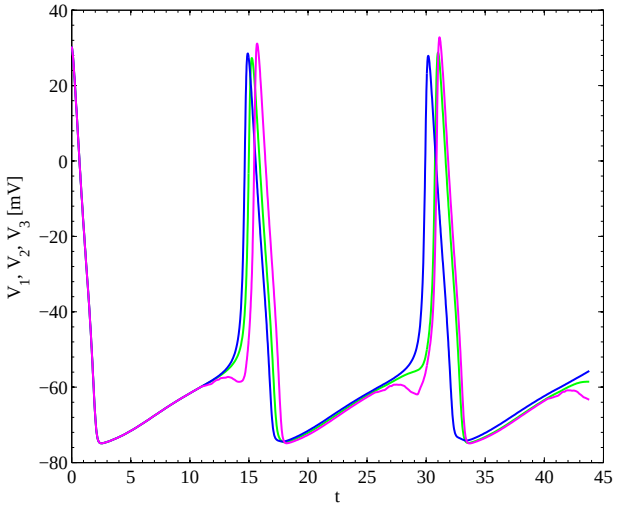


(b)

Figure 4.8: (a) The optimal control input and state trajectory obtained for $(\alpha_{12}, \alpha_{13}, \alpha_{23}) = (0.2, 0.6, 0.2)$ and $I_{max} = 2\,\mu\text{A/cm}^2$ computed through fixed termination time dynamic programming for the Hodgkin-Huxley coupled phase model, (b) Voltage variable evolution for three coupled full Hodgkin-Huxley neurons under the control shown in part (a). For this simulation, we have considered a mesh size of 1 degree.



(a)



(b)

Figure 4.9: (a) Voltage traces for the full Hodgkin-Huxley model obtained for $(\alpha_{12}, \alpha_{13}, \alpha_{23}) = (0.2, 0.6, 0.2)$ by applying the optimal control input from the phase model (not shown) with $I_{max} = 10$ computed through fixed termination time dynamic programming. (b) Voltage traces as a result of applying three copies of the optimal input. For this simulation, we have considered a mesh size of 2 degrees.

averaged, which resulted in a single control law for the entire system regardless of what the coupling strengths may be. When the bound on the controller is set to be $u_{max} = 3$, this final control law can desynchronize the system, under the mentioned assumptions, with a probability of 8.4% for the Kuramoto network and 73.1% for the Hodgkin-Huxley network. Figure 4.5 and Table 4.2 present the final control for each control bound and the probablity of it achieving desynchronization. We argue that the large difference between the percentages obtained for the Kuramoto and Hodgkin-Huxley systems are due to the notable difference in the slope of the their respective coupling functions at 0. The Kuramoto coupling function, $\sin(\theta)$ has slope 1 at $\theta = 0$, but the Hodgkin-Huxley coupling function has a slope very close to zero at $\theta = 0$ as is seen from Figure 4.1. The smaller slope in the Hodgkin-Huxley coupling functions allows for notably easier desynchronization and hence higher percentages.

The control method was also tested on a more realistic modeling of the control for the Hodgkin-Huxley model in which the stimulus is multiplied by the PRC. It was observed that there is a good agreement between the simulation results and the theory on which the model is based. However, the curse of dimensionality and the effect of discretization error due to relatively large mesh size proved to be limiting factors that need more consideration in the future.

The control approach presented in this study can be viewed as an event-based control approach where the controller starts to apply input upon occurrence of an event. The event here would be the simultaneous spiking of all neurons. When the

controller is triggered, it applies the precomputed traces of Figure 4.5 and waits until the next event triggers it.

In order to improve the accuracy of the results presented in chapter, one can reduce the mesh size when discretizing the phases. This would substantially improve the precision of the solutions. However, by reducing the mesh size, the number of states grows exponentially, which can be detrimental given one's available computational power. We believe that by decreasing the mesh size, the results from the full Hodgkin-Huxley model will match more closely to the results from the phase model presented in Section 4.5. In addition, one can optimize the averaging process that leads to the final control laws, so that more systems, as characterized by their $(\alpha_{12}, \alpha_{13}, \alpha_{23})$, can be desynchronized with an averaged final control. Also, it would be beneficial to consider more combinations of $(\alpha_{12}, \alpha_{13}, \alpha_{23})$ in order to have a better probabilistic estimate for any control input.

The main drawback in dynamic programming is that it demands exponentially higher computational power as the number of states in the system is increased. An interesting future direction for this work would be to find reasonable approximations to networks of higher dimension to overcome the curse of dimensionality without greatly sacrificing the accuracy of the results. As a suggestion, one can think of splitting larger networks in several smaller networks that are each all-to-all coupled, but only communicate with each other through a mean field effect. This way, it might be possible to take advantage of parallel programming techniques and distribute the computational burden onto several processors.

Other interesting future directions would be to add uncertainties to the models and investigate the extent of applicability of current control laws in the presence of noise, and to compare the results obtained by dynamic programming to those for other control schemes.

Chapter 5

Minimum Energy Desynchronizing Control for Coupled Neurons

In this chapter, we design an event-based, minimum energy, desynchronizing control stimulus for a network of pathologically synchronized, heterogeneously coupled neurons. As previously mentioned, applying event-based minimum energy stimuli in a deep brain stimulation setting for treatment of Parkinson's disease is clinically desirable in that it could reduce the number of stimulus applications and the amount of energy needed per stimulation. The proposed design works by optimally driving the neurons to their *phaseless sets*, switching the control off, and letting the phases of the neurons randomize under the intrinsic background noise (cf., [46]). This is a non-trivial extension of work done in [18]. In [18] a minimum

time control problem with the same objective has been considered that has resulted in a *bang-bang* control input. We find that, even with considerably stronger coupling between the neurons, the method of the present study substantially reduces the amount of energy that is needed to effectively achieve desynchronization when compared to the method of [18].

The organization of this chapter is as follows. In Section 5.1, a brief introduction is presented. In Section 5.2 we describe the model used for the neuron. Then we lay out the control method and derive the necessary equations in Section 5.3. In Section 5.4, we describe the essentials of the numerical method that was used to solve the equations. Results and discussion are presented in Section 5.5 for both a single neuron and a network of neurons. Finally, we present concluding remarks in Section 5.6.

The main results from this chapter are given in [23].

5.1 Introduction

Pathological synchronization among the spiking neurons of the basal ganglia and the thalamus regions of the brain is thought to be one cause for the involuntary tremors that patients with Parkinson's disease experience [7]. Deep Brain Stimulation (DBS), an FDA-approved surgical treatment procedure, has shown success in alleviating these tremors by administration of high frequency pulsatile stimuli through an electrode implanted deep into the patient's brain which, hypothetically, desynchronizes the neurons [4,6,27]. This has motivated researchers to adopt

control theory and investigate alternative desynchronizing stimuli with less possible side-effects such as tissue damage or adaptation, and with less energy consumption. Various control methods have been investigated and applied to different models in the past. Among these control methods, feedback control and optimal control are more prominent. These methods are attractive from a clinical perspective in that the control stimulus is designed to be applied only when needed (characterized by the feedback signal) and in an optimal way (characterized by the optimality criteria). For example, in [43] a system of noisy coupled phase neurons is studied and a demand-controlled deep-brain double-pulse stimulation has been suggested, where a double-pulse stimulus is administered when a feedback signal indicates occurrence of synchronization. In [28] and [80] nonlinear delay feedback control has been considered that can achieve desynchronization for systems of globally coupled limit-cycle oscillators. In [12] and [13] the authors used a genetic algorithm to design a model-independent optimal input specific to each patient's brain. In [14] and [15], unscented Kalman filtering has been shown to achieve success in optimally estimating the unobservable states of a neuron through the feedback information from the observable state, which may be important for designing DBS control. In [18], a Hamilton-Jacobi-Bellman approach has been taken to design a minimum time desynchronizing control law for a globally coupled network. In recent work [45], we have considered the problem of desynchronization for a deterministic system of coupled Hodgkin-Huxley phase neurons driven by a single constrained input. Dynamic programming was used to find a minimum energy desynchronizing control. In ad-

dition to those mentioned, a number of other studies have also shown potential in desynchronizing a population of pathologically synchronized neurons [26, 29]. On a single neuron level, various event-based optimal control ideas have been considered as well [3, 16, 17, 19–21].

As described in Appendix A, the phaseless set for an oscillatory system is where the isochrons of the system converge together. One can randomize the phase of a noisy oscillator by steering its state to its phaseless set, and letting it randomly fall on an isochron due to its intrinsic noise. In neuronal systems, this yields randomization of the next spike time for each neuron, and hence, desynchronization in a network structure. As a proof of concept, Figure 5.1 shows the voltage traces and spike time histogram for a network of 100 all-to-all coupled and synchronized neurons, more formally introduced in the following section, that are all initialized at their phaseless set after the control is switched off. It is seen that despite the synchronizing force of the coupling and the fact that the network is initialized coherently, the next spike time for the neurons has been effectively randomized due to an intrinsic zero-mean, variance 2, Gaussian white background noise.

5.2 Model

The model considered for the neurons in the population is as follows:

$$\begin{aligned} \dot{V}_i &= f_V(V_i, n_i) + \eta_i(t) + \tfrac{1}{N}\textstyle\sum_{j=1}^{N} \alpha_{ij}(V_j - V_i) + u, \\ \dot{n}_i &= f_n(V_i, n_i). \end{aligned} \tag{5.1}$$

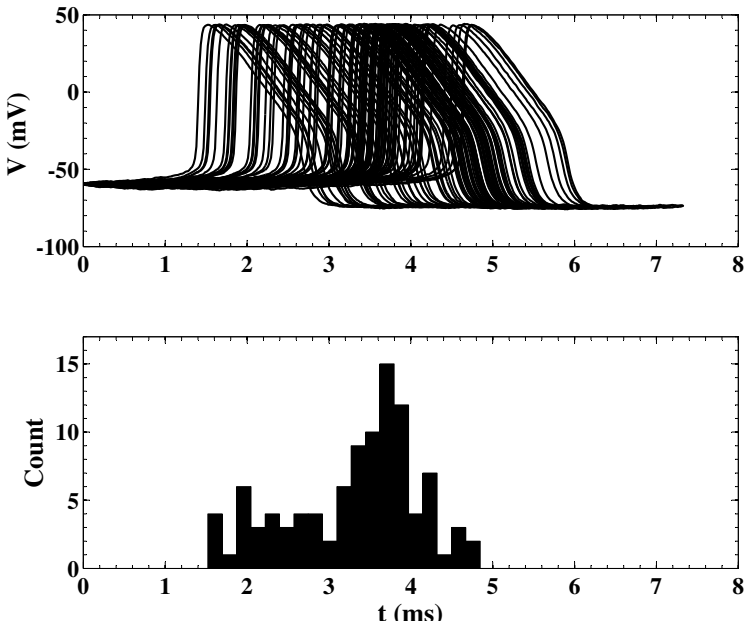


Figure 5.1: Voltage traces and spike time histogram for a network of 100 coupled and synchronized neurons (described by (1) in the absence of any control u) that are initialized at their phaseless points, subject to i.i.d. Gaussian white background noise with zero-mean and variance 2.

Here, $i, j = 1, \cdots, N$, where N is the total number of neurons in the network, V_i and n_i are the membrane voltage and the gating variable for neuron i, α_{ij} is the coupling strength between neurons i and j, which are assumed to be electrotonically coupled [67] with $\alpha_{ij} = \alpha_{ji}$ and $\alpha_{ii} = 0$ for all i, j, $\eta_i(t) = \sqrt{2D}\mathcal{N}(0,1)$ is the intrinsic noise for each neuron taken as zero-mean Gaussian white noise with variance $2D$, $u = I(t)/c$ is the common control input where $I(t)$, in μA/cm^2, gives the DBS input current, and $c = 1$ μF/cm^2 is the constant membrane capacitance. Also,

$$\begin{aligned} f_V &= (I_b - \bar{g}_{Na}[m_\infty(V)]^3(0.8-n)(V-V_{Na}) \\ &\quad -\bar{g}_K n^4(V-V_K) - \bar{g}_L(V-V_L))/c, \\ f_n &= a_n(V)(1-n) - b_n(V)n, \end{aligned}$$

are the state dynamics for each neuron in the absence of noise, coupling, and control. This is a two-dimensional reduction of the celebrated four-dimensional Hodgkin-Huxley (HH) model [50] that captures the essentials of a neuron's dynamical behavior (cf., [81, 82]). The full HH model was originally developed for the Loligo squid's giant axon through a series of experiments. The other functions and parameters in this reduced model are

$$
\begin{aligned}
m_\infty(V) &= \frac{a_m(V)}{a_m(V)+b_m(V)}\,, \\
a_m(V) &= 0.1(V+40)/(1-\exp(-(V+40)/10))\,, \\
b_m(V) &= 4\exp(-(V+65)/18)\,, \\
a_n(V) &= 0.01(V+55)/(1-\exp(-(V+55)/10))\,, \\
b_n(V) &= 0.125\exp(-(V+65)/80)\,,
\end{aligned}
$$

$$
\begin{aligned}
&V_{Na} = 50 \text{ mV},\ V_K = -77 \text{ mV},\ V_L = -54.4 \text{ mV}, \\
&\bar{g}_{Na} = 120 \text{ mS/cm}^2,\ \bar{g}_K = 36 \text{ mS/cm}^2, \\
&\bar{g}_L = 0.3 \text{ mS/cm}^2,\ c = 1\ \mu\text{F/cm}^2.
\end{aligned}
$$

Also, I_b, in μA/cm^2, is the neuron's baseline current which represents the effect of other parts of the brain on the neuron under consideration and can be viewed as a bifurcation parameter in the model that controls whether the neuron is in an excitable or an oscillatory regime. We consider $I_b = 10\ \mu$A/cm^2 to ensure oscillatory (periodic spiking) behavior for the neuron. With this, the period of spiking is $T_s = 11.85$ ms. $\bar{g}_{Na}$, $\bar{g}_K$, and $\bar{g}_L$ are the conductances of the sodium,

potassium, and leakage channels, respectively. Also, V_{Na}, V_K, and V_L represent their respective reversal potentials.

In the absence of noise, coupling, and control, the oscillatory behavior of (5.1) is seen as a periodic orbit in the $V-n$ phase plane of the system, shown in Figure 5.2 as the thick solid black line. The isochrons for this system are shown as gray lines. The isochrons converge at the unstable fixed point where the $V-$ and $n-$nullclines intersect. This unstable fixed point is the phaseless set for this system [83]. In the present article, we first find the optimal control stimulus that, when applied to a single neuron, drives the system to its phaseless set. As mentioned before, the idea here is that once the state of the system is at the phaseless set, the intrinsic background noise could cause the system to fall on a random isochron, thereby randomizing the phase of the neuron and its next spiking time. We then apply this optimal control to the population of synchronized, coupled, and noisy neurons and evaluate its performance in desynchronizing the population. Although we consider this specific model in this chapter, we expect that a similar approach to that described below can be used to find optimal control inputs for other neuron models.

In order to better stabilize the numerical simulation, we scale down the V dimension in (5.1) by a factor of $K = 100$ so that the two states are of same order of magnitude. Consider the change of variables $z \equiv (x, y) = (\frac{1}{K}V, n)$. In view of (5.1), for a single deterministic neuron under control, we get

$$\dot{z} = F(z) + Bu, \tag{5.2}$$

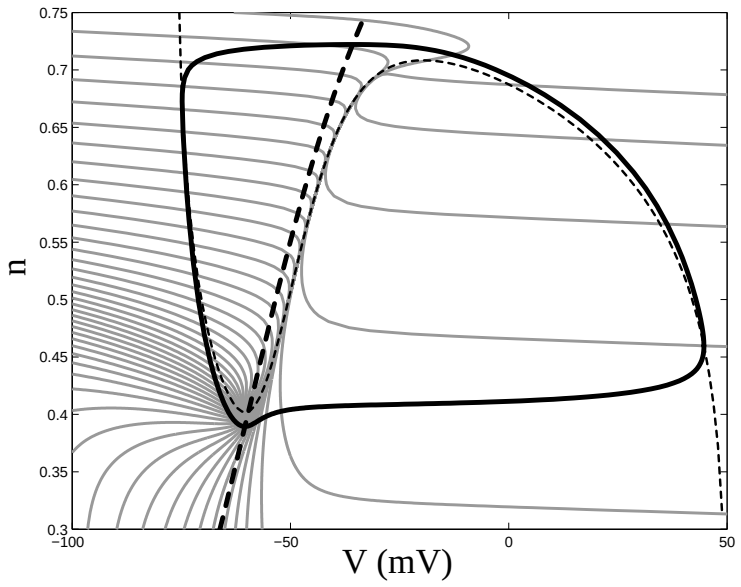


Figure 5.2: Periodic orbit (thick solid), V-nullcline (thin dashed), and n-nullcline (thick dashed), and fifty isochrons equally spaced in phase for the two-dimensional reduced Hodgkin-Huxley model in the absence of noise, coupling, and control. The location of the unstable fixed point (phaseless point) for this system is at the intersection of the nullclines.

where $B = [\frac{1}{K}, 0]^T$ and

$$F(z) = \begin{bmatrix} f_x(z) \\ \\ f_y(z) \end{bmatrix} = \begin{bmatrix} \frac{1}{K} f_V(Kx, y) \\ \\ f_n(Kx, y) \end{bmatrix}. \tag{5.3}$$

We note that this scaling is only for the sake of numerical stability and the results that we present later are all in the original $V - n$ coordinates.

5.3 Optimal Control

We consider the system (5.2). The objective is to find the optimal control law that would take the system to its phaseless set in some prespecified length of time

$[0, T_{end}]$, while minimizing the cost function

$$J(z, u(t)) = \int_0^{T_{end}} u^2 dt + \gamma q(z(T_{end})). \tag{5.4}$$

This cost function is composed of a time-additive portion, $\int_0^{T_{end}} u^2 dt$, that characterizes the total input energy being used, and an end-point cost, $q(z(T_{end}))$, that discriminates between different possible outcomes for the end states. γ is a penalizing scalar. We consider bounded inputs, i.e., $|u| \leq u_{max}$, as would be the case in practice due to hardware limitations as well as tissue sensitivity.

Employing a Hamilton-Jacobi-Bellman (HJB) approach [47, 56], we define the cost-to-go function, also known as the value function, from state z and time $\tau \in [0, T_{end}]$ to be

$$\mathcal{V}(z, \tau) = \min_{\substack{|u(t)| \leq u_{max} \\ \forall\, t \in [\tau, T_{end}]}} J = \min_{\substack{|u(t)| \leq u_{max} \\ \forall\, t \in [\tau, T_{end}]}} \left[\int_\tau^{T_{end}} u^2 dt + \gamma q(z(T_{end})) \right]. \tag{5.5}$$

With this definition, following classical optimal control theory [47, 56], we can write

$$\mathcal{V}(z, \tau) = \min_{\substack{|u(t)| \leq u_{max} \\ \forall\, t \in [\tau, T_{end}]}} \left[\int_\tau^{\tau+h} u^2 dt + \int_{\tau+h}^{T_{end}} u^2 dt + \gamma q(z(T_{end})) \right],$$

where, $h \in [0, T_{end} - \tau)$. Now we realize that the first integral in this equation is dependent only on $u(t)$ for $t \in [\tau, \tau + h]$, whereas the second integral and the value of the end point cost are dependent on $u(t)$ for $t \in [\tau, T_{end}]$. The reason for this is that the u for $t \in [\tau, \tau + h]$ determines the state of the system at $t = \tau + h$, $z(\tau + h)$, which in turn comes into play when finding the u for $t \in [\tau + h, T_{end}]$. With this, we can write

$$\mathcal{V}(z, \tau) = \min_{\substack{|u(t)| \leq u_{max} \\ \forall\, t \in [\tau, \tau+h]}} \left[\int_\tau^{\tau+h} u^2 dt + \min_{\substack{|u(t)| \leq u_{max} \\ \forall\, t \in [\tau+h, T_{end}]}} \left(\int_{\tau+h}^{T_{end}} u^2 dt + \gamma q(z(T_{end})) \right) \right],$$

where the inner minimum is exactly the cost-to-go function from state $z(\tau+h)$ and time $t = \tau + h$: $\mathcal{V}(z(\tau + h), \tau + h)$. By subtracting $\mathcal{V}(z(\tau), \tau)$ from both sides of the above equation and dividing by $h > 0$, we get

$$0 = \min_{\substack{|u(t)| \leq u_{max} \\ \forall\, t \in [\tau, \tau+h]}} \left[\frac{1}{h} \int_{\tau}^{\tau+h} u^2 dt + \frac{\mathcal{V}(z(\tau + h), \tau + h) - \mathcal{V}(z(\tau), \tau)}{h} \right], \tag{5.6}$$

for any $\tau \in [0, T_{end}]$ and $h \in [0, T_{end} - \tau)$. Taking the limit as $h \to 0$, yields the well-known HJB equation

$$0 = \frac{\partial \mathcal{V}}{\partial t}(z(\tau), \tau) + \min_{|u(\tau)| \leq u_{max}} \left[u^2 + \frac{\partial \mathcal{V}}{\partial z}(z(\tau), \tau) \Big(F(z(\tau)) + Bu(\tau) \Big) \right], \tag{5.7}$$

which is a partial differential equation (PDE) with the boundary condition

$$\mathcal{V}(z(T_{end}), T_{end}) = \gamma q(z(T_{end})). \tag{5.8}$$

It should be noted that by letting $h \to 0$ in (5.6), the minimization with respect to the control function in (5.6) reduces from choosing values of a curve, $|u(t)| \leq u_{max}, \ \forall t \in [\tau, \tau + h]$, to choosing a single value on the curve, $|u(\tau)| \leq u_{max}$ at $t = \tau$ [84]. By defining,

$$\mathcal{H}(z, \nabla \mathcal{V}, u) = u^2 + \nabla \mathcal{V}^T(z(t), t)(F(z(t)) + Bu(t)), \tag{5.9}$$

as the Hamiltonian for the system, one can rewrite (5.7) more succinctly as

$$\frac{\partial \mathcal{V}}{\partial t} + \min_{|u| \leq u_{max}} \mathcal{H}(z, \nabla \mathcal{V}, u) = 0, \tag{5.10}$$

where $\nabla \mathcal{V}$ is the gradient of the value function with respect to $z, (\frac{\partial \mathcal{V}}{\partial x}, \frac{\partial \mathcal{V}}{\partial y})^T$. The optimal control that globally minimizes $\mathcal{H}$ is obtained as

$$u^*(t) = \arg \min_{|u| \leq u_{max}} [u^2 + \nabla \mathcal{V}^T(z^*(t), t)(F(z^*(t)) + Bu(t))],$$

where $z^*(t)$ represents the optimal trajectory.

In order to find the optimal control, we can set the derivative of the Hamiltonian (5.9) with respect to u equal to zero and solve for the extremal u. This is true as long as the magnitude of the control remains smaller than the predetermined bound u_{max}. When the magnitude of the optimal control reaches the bound u_{max}, it saturates in accordance to Pontryagin's minimum principle [47, 85]. Considering (5.9), $\frac{\partial \mathcal{H}}{\partial u} = 0$ results in $u^*(t) = -\frac{1}{2}\nabla \mathcal{V}^T B$ as long as $|u^*(t)| \leq u_{max}$. Equivalently, one can write $u^*(t) = -\frac{1}{2}\nabla \mathcal{V}^T B$ for when $|\nabla \mathcal{V}^T B| \leq 2u_{max}$. When $|\nabla \mathcal{V}^T B| = 2u_{max}$, the optimal control reaches its bound and if $|\nabla \mathcal{V}^T B| > 2u_{max}$, it gets saturated, in which case, considering (5.9) with $|u^*(t)| = u_{max}$, the minimizing optimal control becomes $u^*(t) = -\text{sign}(\nabla \mathcal{V}^T B) u_{max}$. So in summary, considering the fact that $B = [\frac{1}{K}, 0]^T$, we get the optimal control as

$$\begin{aligned} u^*(t) &= -\tfrac{1}{2K}\mathcal{V}_x \qquad && |\mathcal{V}_x| \leq 2K u_{max}, \\ u^*(t) &= -\text{sign}(\mathcal{V}_x) u_{max} \qquad && |\mathcal{V}_x| > 2K u_{max}, \end{aligned} \tag{5.11}$$

where $\mathcal{V}_x = \frac{\partial \mathcal{V}}{\partial x}$. With this optimal control, the Hamiltonian can be written as

$$\begin{aligned} \mathcal{H} &= \nabla \mathcal{V}^T F(z) - \tfrac{1}{4K^2}\mathcal{V}_x^2, \qquad && |\mathcal{V}_x| \leq 2K u_{max}, \\ \mathcal{H} &= \nabla \mathcal{V}^T F(z) + u_{max}^2 - |\mathcal{V}_x|\tfrac{u_{max}}{K}, \qquad && |\mathcal{V}_x| > 2K u_{max}. \end{aligned} \tag{5.12}$$

In order to find the optimal control $u^*(t)$ in (5.11), we need to find the cost-to-go, $\mathcal{V}(z,t)$ from the HJB PDE (5.7) with boundary condition (5.8). This is done numerically as explained in the following.

5.4 Numerical Method

The HJB PDE (5.7) is a special form of a broader class of equations known as the Hamilton-Jacobi (HJ) equations which for a scalar variable $\mathcal{V}(z,t)$ are given by

$$\frac{\partial \mathcal{V}}{\partial t} + \mathcal{H}(z,t,\mathcal{V},\nabla\mathcal{V}) = 0,$$

where $\mathcal{H}$ is the Hamiltonian and $\nabla\mathcal{V}$ denotes spatial gradients. These equations frequently appear in different areas of research such as optimal control theory, image processing and computational physics [47, 86, 87] and thus have been well studied in the past.

The numerical solution of the HJ equation is deeply rooted in the methods that already had existed for the solution of nonlinear hyperbolic conservation laws (HCL) [86]. Originally, Crandall and Lions [88] proposed their first-order accurate numerical algorithm for the solution of the HJ equation and a few years later, Osher and Sethian [89] used the connection between HJ and HCL to derive higher order accurate algorithms. For a more complete list of references, one may consult standard texts such as [86] and [87].

Generally, a convergent, high-order approximation to the HJ equation consists of three steps: (i) Computing the solution gradient $\nabla\mathcal{V}$, which is typically achieved with essentially non-oscillatory (ENO) schemes [90,91]. These schemes are designed such that they do not produce oscillatory results when the solution gradients are evaluated close to the discontinuities that are inherent to nonlinear HJ equations. (ii) Evaluating the Hamiltonian function, which is straightforward only for linear

problems. For nonlinear problems it is necessary to compute the so-called *numerical* Hamiltonian (see $\hat{\mathcal{H}}$ in [92]) in such a way as to account for the nonlinear shock and rarefaction phenomena; Godunov or local Lax-Friedrichs (LLF) schemes are traditionally used. In this study, we use the LLF scheme to obtain the numerical Hamiltonian. (iii) Given an initial condition $\mathcal{V}(z,0)$, time integration is needed to obtain the solution $\mathcal{V}(z,t)$ at later times. To prevent non-physical oscillations in the solution, this is done using a total variation diminishing (TVD) method [86].

The details regarding the careful implementation of these steps are included in Appendix E. In this study we have used the Matlab toolbox, "Level Set Methods Toolbox" written by Ian Mitchell [93], which is a working example of such implementations.

We set $T_{end} = 7$ ms and use a 321×321 uniform grid for the states to solve the HJB equation (5.7) for the cost-to-go function $\mathcal{V}(z,t)$. The control bound is set to be $u_{max} = 10\ \mu\text{A}/\mu\text{F}$. We also set the end point cost to be

$$\mathcal{V}(z(T_{end}), T_{end}) = \gamma \left(1 - e^{-\left(\frac{(x-x_{pl})^2}{\sigma_x^2} + \frac{(y-y_{pl})^2}{\sigma_y^2} \right)} \right),$$

where $\gamma = 1000$, $\sigma_x^2 = \sigma_y^2 = 0.001$, and $(x_{pl}, y_{pl}) = (\frac{1}{K} V_{pl}, n_{pl})$ where $K = 100$ and $(V_{pl}, n_{pl}) = (-59.6, 0.403)$ is the phaseless target point. This Gaussian end point cost function has a minimum of zero at the phaseless point that encourages the evolution of the controlled system towards this point. We note that we solve the HJB equation backward in time and treat this end point cost as the initial condition for the equations.

Once the solution $\mathcal{V}(z,t)$ is computed, the optimal control is found as a func-

tion of the state at all time steps using (5.11). Given this data in time and space, the optimal control sequence $u(t)$, and the optimal trajectories can be found by forward integrating (5.1) in the absence of noise and coupling, and for any initial condition including the spiking point $(V_0, n_0) = (44.8, 0.459) \equiv (V_s, n_s)$, as we consider here. A fourth order Runge-Kutta method is used for the integration. We note that since $\mathcal{V}(z,t)$ is available on spatial grid points, a simple bilinear interpolation scheme is used to obtain the input off grid points at each time step. This optimal control sequence is then applied first to the noisy single neuron system to evaluate its performance in randomizing the noisy neuron's next spiking time, and then, to a population of 100 initially synchronized, coupled neurons to evaluate its performance in desynchronizing the population in the presence of noise and heterogeneities in the coupling. Since introducing noise into the equations makes the problem a stochastic differential equation (SDE) problem, care must be taken in choosing a proper numerical algorithm for integration. A simple and straightforward algorithm is the Honeycutt's second order stochastic Runge-Kutta method [94] which was chosen in this study.

5.5 Results and Discussion

The top panel in Figure 5.3 shows the minimum energy control law for the deterministic single neuron. As can be seen in Figure 5.3, the control has saturated at the bound value $u_{max} = 10\ \mu\text{A}/\mu\text{F}$ (equivalent to $I_{max} = 10\ \mu\text{A}/\text{cm}^2$). Figure 5.3 also shows the evolution of the system states (V, n) in time as well as the optimal

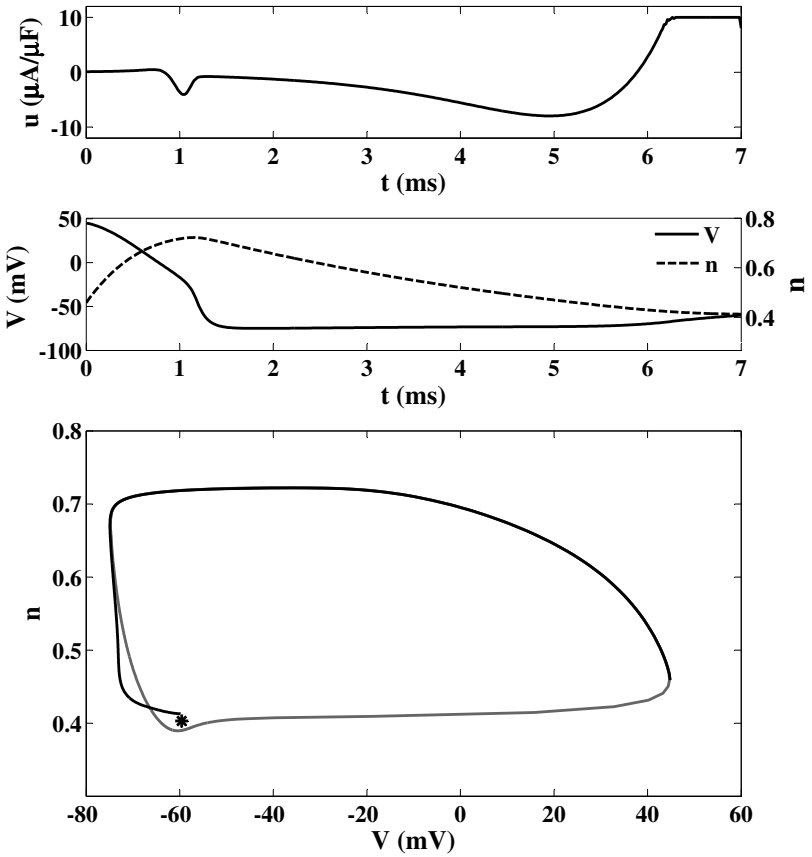


Figure 5.3: Results for the deterministic single neuron system (equation (5.1) with $\eta_i(t) \equiv 0$ and $\alpha_{ij} = 0, \forall i, j$). The system is initialized at the spiking point $(V_s, n_s) = (44.2, 0.465)$ and the target point for the control is the phaseless set for the neuron which is its unstable fixed point $(V_{pl}, n_{pl}) = (-59.6, 0.403)$. Top: minimum energy control law that is bounded to $|u| \leq 10\ \mu\text{A}/\mu\text{F}$. Middle: the time evolution of the states of the system. Bottom: the state space representation of the trajectory of the system under the control shown in top panel. We see that the control has been able to take the system close to the phaseless point shown with asterisk marker.

trajectory in the state space when driven by the control. It is worth pointing out that we set the initial condition to be the spiking state as this is a practical observable which can be used as a trigger for the control, hence producing an event-based control.

5.5.1 Single neuron level

To evaluate the performance of the control for phase randomization, we apply it to the single neuron in the presence of noise and integrate the noisy system forward in time. Figure 5.4 shows the results obtained for this case for 100 different numerical realizations. We have included three different cases in this figure for comparison. The top row in this figure shows the case of the neuron under its natural dynamics in the absence of both the noise and the control. As expected, the neuron spikes at its natural period $T_s = 11.85$ ms for all 100 different trials. The second panel shows the case where noise is active, but the control is not. As can be seen, the spiking instant of the neuron varies due to the effect of different noise realizations. In the third panel, both noise and the control are acting on the neuron. We see that applying the control causes the next spiking instant of the neuron to randomize over a considerable time interval. We note that the control has only been applied for one cycle and has been set to zero for $t > 7$ ms.

5.5.2 Population level

We now apply the minimum energy control that is found for a single neuron to the network of $N = 100$ coupled synchronized noisy neurons with common coupling strength $\alpha_{ij} = 0.1$ and i.i.d. noise with $D = 1$. We note that, in this study, we consider coupling strengths that are 10 times greater than those considered in [18].

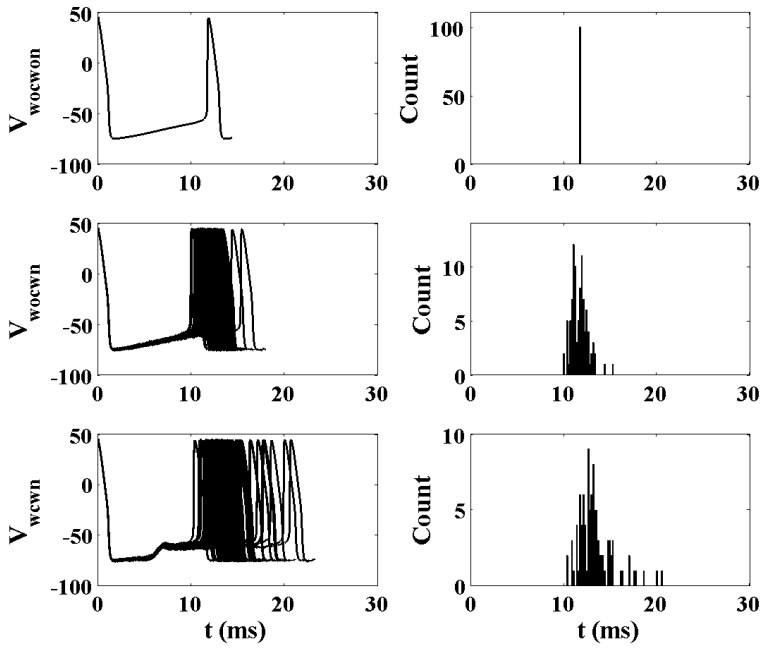


Figure 5.4: Results for 100 different simulations for the system (5.1) with initial condition $(V_0, n_0) = (44.8, 0.459)$. Top row: voltage trace (left) and histogram (right) for the case of without noise and without external control; middle row: voltage trace (left) and histogram (right) for the case of 100 different noise realizations, without control; bottom row: voltage trace (left) and histogram (right) for the case of 100 different noise realizations, with one cycle of control.

For this system, we define the mean voltage as the observable for the network

$$\bar{V}(t) = \frac{1}{N}\sum_{i=1}^{N} V_i(t), \tag{5.13}$$

and set $\bar{V} = -20$ mV as the event that triggers one cycle of control administration. After one cycle, the control turns off until the next event triggers it. Figure 5.5 shows the result for this with different noise for each of the neurons in the system. The first panel shows the individual voltages and mean voltage for the coupled system with activated noise, but without control. We see that the mean voltage spikes are always above the -20 mV threshold that is shown as dotted line. The second panel shows the individual voltages and mean voltage for the coupled system

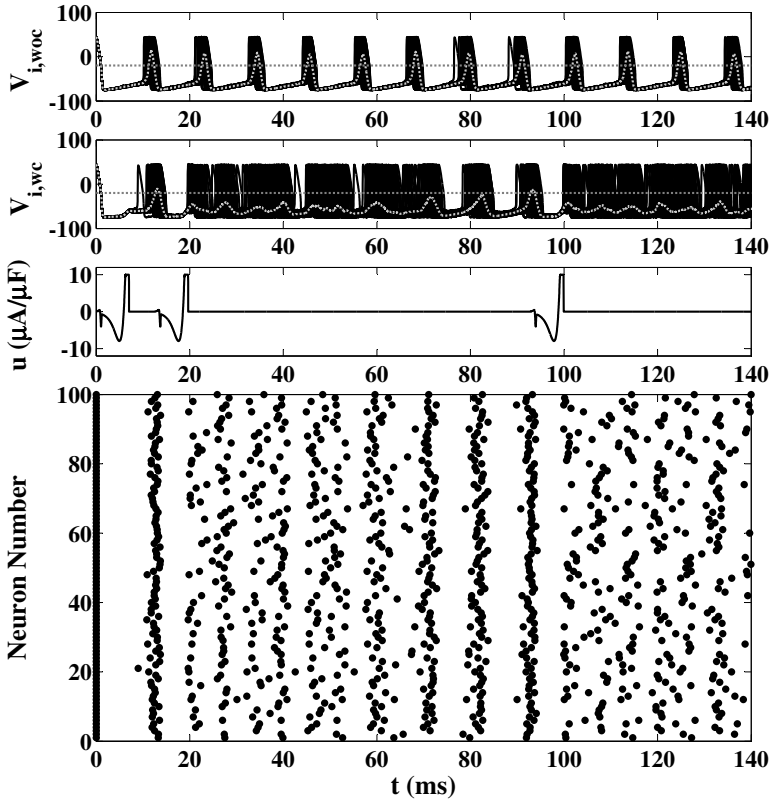


Figure 5.5: Results for a population of $N = 100$ coupled neurons with $\eta_i = \sqrt{2D}\mathcal{N}(0,1)$, $D = 1$, and coupling strength $\alpha = 0.1$. First panel shows the result for the noisy network without control. The second panel shows the results for the same network with active event-based control. The dotted gray traces show the mean voltage for each case and the horizontal dotted lines mark the control activation threshold. We see that the control (shown in third panel) has only been applied when the mean voltage has reached the $\bar{V} = -20$ mV threshold, and has been able to substantially desynchronize the network as communicated by the raster plot.

when both the noise and the control are present. The control input is shown in the third panel. The desynchronizing effect of the control is clearly seen from the raster plot. The event-based nature of the control is also apparent from the fact that the control has only been turned on when the mean voltage has crossed the threshold line.

To model more realistic networks, we now take into account the effect of network heterogeneities on the performance of the optimal controller. First, instead of

a common coupling strength, we take the coupling strengths from a normal distribution with mean $\bar{\alpha} = 0.1$ and standard deviation $\sigma_\alpha = 0.02$, i.e., $\alpha_{ij} = \mathcal{N}(\bar{\alpha}, \sigma_\alpha) = \mathcal{N}(0.1, 0.02)$. The left panel in Figure 5.6 shows this distribution. Simulating the network with this variability in the coupling strengths, we get the result shown in the second panel of Figure 5.7. The first panel in this figure is for the case of network with common coupling strength $\alpha_{ij} = 0.1$ as shown in Figure 5.5 and is reillustrated here to facilitate comparison. When the coupling strengths are different, we see very similar behavior as before except that there is a delay in resynchronization after the network is desynchronized. We note that although this difference in timing of the resynchronization is seen, it is dependent on the particular realization of the α values. Next, we used the same realization of the coupling strengths, but randomly set 20% of them equal to zero so that the network is not an all-to-all coupled network. Again only a further delay in resynchronization is observed, as shown in the third panel of Figure 5.7. This is reasonable since now some of the connections are broken and thus the overall drive for resynchronizing the network is smaller. We also considered the baseline current I_b of the neurons to be drawn from a normal distribution so that the neurons are not all identical. The distribution considered for the I_b values is shown in the right panel of Figure 5.6. With this additional variability added to those described before, we get the result shown in the fourth panel of Figure 5.7.

It should be noted that when increasing the mean coupling strength between the neurons to $\bar{\alpha} = 0.2$, we see the same qualitative results, but with more instances

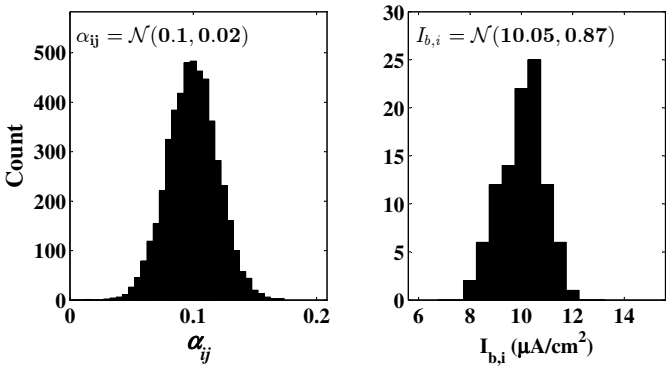


Figure 5.6: Normal distributions for the coupling strengths (left) and the baseline currents (right) for inducing heterogeneity into the network of neurons.

of control application due to the network's higher tendency for synchronization (results not shown).

5.6 Conclusion

We have considered the design of an event-based, minimum energy, desynchronizing control stimulus for a network of pathologically synchronized coupled neurons. The control drives the neurons to their phaseless sets, and lets the phases of the neurons randomize under intrinsic background noise. The minimum energy optimality criterion is desirable for practical purposes, as it may increase the battery life of implanted stimulus generators in patients with Parkinson's disease treated by DBS. By employing the minimum energy formulation, the total input energy for the control input shown in Figure 5.3 is computed to be $\int_0^{T_{end}} u^2 dt \approx 194$ which is about 70% less than the minimum time approach considered in [18].

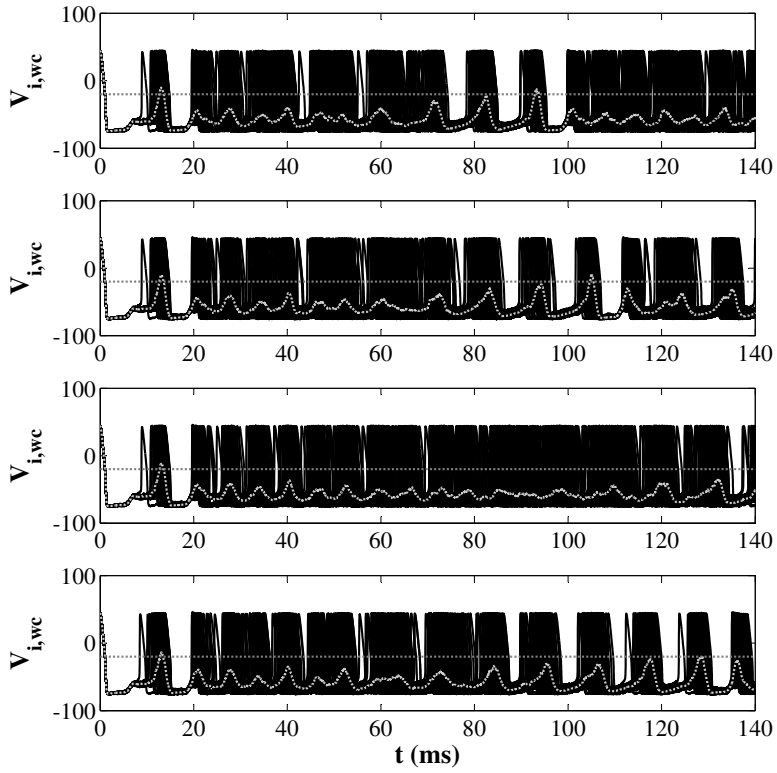


Figure 5.7: Results for a population of $N = 100$ coupled neurons with $\eta_i = \sqrt{2D}\mathcal{N}(0, 1)$, $D = 1$; first panel: the network has a common coupling strength $\alpha = 0.1$ and the neurons all have the same baseline current $I_b = 10$ μA/cm^2; second panel: the coupling strengths are drawn from the distribution shown in Figure 5.6, but the baseline current is the same for all neurons $I_b = 10$ μA/cm^2; third panel: the same coupling strengths and baseline current values are used as in the second panel, except that 20% of the coupling strengths have been randomly chosen and set equal to zero; fourth panel: the coupling strengths are exactly the same as those used to produce panel three, but the baseline current values have been drawn from the normal distribution shown in Figure 5.6. The seed for the random number generators have been set such that the same value for the random vectors were produced across all four experiments to facilitate comparison. The dotted gray traces show the mean voltage for each case and the horizontal dotted lines mark the control activation threshold. The control is only active when the mean voltage has crossed the threshold.

Chapter 6

Minimum Energy Control for *in vitro* Neurons

In this chapter, we consider optimal regulation of interspike intervals (ISI) for periodically spiking *in vitro* neurons. Revisiting the approach presented in Chapter 3, this is done by employing optimal control theory to design charge-balanced, continuous-time, minimum energy input stimuli that can change the ISI of a neuron to pre-specified values. The optimal control algorithm uses the neuron's phase response curve (PRC) which is measured experimentally using the so-called direct method. In Chapter 3, following [16], we have applied optimal control theory to phase models of neurons to derive charge-balanced minimum energy ISI regulatory input stimuli. In this chapter, we extend the theoretical results from Chapter 3 and show the applicability of this method in practice by testing the controller on *in vitro* pyramidal neurons in the CA1 region of rat hippocampus. In addition to

being minimum energy, charge-balanced, and continuous-time, the designed input stimuli are also low amplitude and of event-based nature. The event-based nature of the input means that it applies only when a specific event, e.g., an action potential, occurs. This way, the onset of an action potential could be considered as a feedback signal that triggers the input.

This algorithm is independent of the biological details in neuron models and is applicable to any neuron whose PRC can be measured. We first demonstrate the theory through computational simulations using a model of cortical pyramidal neuron [95]. We then show experimental evidence that the ISI of a periodically spiking neuron can be optimally changed with energy levels that are in some cases multiple orders of magnitude smaller than those previously considered in [3].

The main results from this chapter are given in [96].

6.1 Introduction

This work is motivated by the treatment of neurological diseases such as Parkinson's disease, epilepsy, and essential tremor. When extended to a population level, the ability to optimally control the spike timing of *in vitro* neurons could prove useful in the treatment of these diseases by deep brain stimulation (DBS) [5]. In recent years, various studies have considered different control approaches to neuron systems both on a population level and on a single neuron level [11–23, 26–29, 45]. However, very few papers have shown experimental evidence of the applicability of their method in practice: In [80], the authors present experimental results for

model-independent control of electrochemical reactions of nickel electrode arrays in sulfuric acid via a mild model-engineered feedback control method that suggests applicability to networks of coupled neurons. In [97], a model is developed to describe thalamic DBS for patients with essential tremor and the authors present experimental results that support the idea that for high frequency pulsatile stimulation of the Ventral Intermediate nucleus of the thalamus, there is an optimal voltage for maximum tremor suppression. In [98], the stabilization of the ISI of periodically firing *in vitro* neurons is tested by designing and implementing a feedback PI controller that changes the input current to the neuron based on the history of previous ISIs. In another work, [3], the authors use a model-independent control algorithm that regulates an *in vitro* neuron's ISI by inputting an appropriately sized and timed pulse. In all of these studies, phase models have been employed to bypass the need for detailed biological characteristics for modeling of the cells.

6.2 Methods

6.2.1 Optimal Control

The control method used here is similar to that presented in Chapter 3. We briefly repeat it here for convenience. The phase model for a neuron under an arbitrary, small external stimulus $u(t)$ is written as [32, 34],

$$\dot{\theta} = \omega + Z(\theta)u(t), \tag{6.1}$$

where $\theta \in [0, 2\pi)$ is the neuron's phase. By convention, $\theta = 0$ is associated with the onset of a spike for the neuron and the neuron is periodically spiking in the absence of external input with period $T_s = \frac{2\pi}{\omega}$. The natural frequency of the neuron is determined by the variable ω, and $Z(\theta)$ is its PRC, which determines how the phase is changed due to an impulsive input at phase θ.

The objective of the optimal stimulus is to make the neuron fire at a desired spike time. The algorithm finds a bounded input waveform $|u(t)| \leq u_{max}$, that starts at the onset of a spike and steers the neuron (6.1) from $\theta(0) = 0$ to $\theta(t_1) = 2\pi$, where t_1 is the desired next spiking time for the neuron. We also require the input to be charge-balanced and to use minimum energy. In order to ensure charge-balance for the input, we define the variable $q(t) : \mathbb{R} \to \mathbb{R}$ as the total accumulated charge in the neuron at time t due to the external input $u(t)$. Then one can write

$$\dot{q} = u(t), \tag{6.2}$$

with the boundary conditions $q(0) = q(t_1) = 0$. Minimizing the total input energy is equivalent to minimization of the cost function

$$C = \int_0^{t_1} u^2(t)dt. \tag{6.3}$$

Following standard optimal control theory [47], one can write the Hamiltonian for this system as

$$\mathcal{H}(\theta, q, \lambda_1, \lambda_2, u) = u^2(t) + \lambda_1(t)(\omega + Z(\theta)u(t)) + \lambda_2(t)u(t), \tag{6.4}$$

where $\lambda_1(t)$ and $\lambda_2(t)$ are the Lagrange multipliers (or co-states) associated with the θ-dynamics (6.1) and the q-dynamics (6.2), respectively. Using this Hamiltonian,

the necessary conditions for optimality are written as

$$\dot{\theta} = \frac{\partial H}{\partial \lambda_1} \quad \Rightarrow \quad \dot{\theta} = \omega + Z(\theta)u(t), \tag{6.5}$$

$$\dot{\lambda}_1 = -\frac{\partial H}{\partial \theta} \quad \Rightarrow \quad \dot{\lambda}_1 = -\lambda_1(t)Z'(\theta)u(t), \tag{6.6}$$

$$\dot{q} = \frac{\partial H}{\partial \lambda_2} \quad \Rightarrow \quad \dot{q} = u(t), \tag{6.7}$$

$$\dot{\lambda}_2 = -\frac{\partial H}{\partial q} \quad \Rightarrow \quad \dot{\lambda}_2 = 0, \tag{6.8}$$

where prime represents differentiation with respect to θ [47, 48]. From Pontryagin's minimum principle, an optimal control stimulus is one that minimizes the Hamiltonian (6.4)

$$u^*(t) = \arg \min_{|u(t)| \leq u_{max}} \left(u(t)^2 + \lambda_1^*(t) \left(\omega + Z(\theta^*)u(t) \right) + \lambda_2^*(t)u(t) \right),$$

where the search for the control stimulus is constrained to values bounded in magnitude by u_{max} and the asterisk superscript denotes optimal values. We note that $\lambda_2(t)$ is a constant according to (6.8). This equation yields

$$\begin{aligned} u^*(t) &= -\tfrac{1}{2}(\lambda_1(t)Z(\theta) + \lambda_2), && |\lambda_1(t)Z(\theta) + \lambda_2| \leq 2u_{max}, \\ u^*(t) &= -\text{sign}(\lambda_1(t)Z(\theta) + \lambda_2)u_{max}, && |\lambda_1(t)Z(\theta) + \lambda_2| > 2u_{max}. \end{aligned} \tag{6.9}$$

By substituting (6.9) in the system of equations (6.5)-(6.8) we arrive at a two point boundary value problem (TPBVP) which we solve using the shooting method. The boundary values for this system are $\theta(0) = q(0) = q(t_1) = 0$ and $\theta(t_1) = 2\pi$. This formulation can be solved to yield the minimum energy control stimulus for any oscillatory system as long as a PRC can be obtained for it.

6.2.2 Estimation of PRC

For computational models of periodically spiking neurons, the PRC, and hence the phase model, can be obtained either by solving the appropriate adjoint equation or by implementing the so-called direct method [33]. For biological neurons however, one can only use the direct method to obtain the PRC experimentally (see Appendix B). We refer to the PRC found through solving the adjoint equation as the *adjoint PRC* and that found from the direct method, the *fit PRC*.

In the direct method, a short-duration pulse (that approximates an impulse) is injected into the periodically spiking neuron at various phase values, that results in a change in the next spike time for the neuron (see Figure B.1). In measuring the PRC with the direct method, it is necessary to choose pulse amplitudes that are high enough to produce spike perturbation values beyond those due to the neuron's intrinsic noise. However, if the pulse amplitudes are too high, they can induce instantaneous spikes which means that the neuron has been perturbed too far off of its periodic orbit to the point where the linearity assumption of the PRC analysis does not hold. This is usually seen when the input pulse is applied in the last $20\% - 30\%$ of the phase cycle. At these phases, the amount of remaining phase to advance may be less than what the pulse stimulus could achieve given the sensitivity of the neuron in that phase: $(2\pi - \theta_{st}) < Z(\theta_{st})Q_p/c$, with Q_p being the pulse electrical charge, c the membrane capacitance, and θ_{st} being the phase at which the stimulus is applied. The effect is that the data points in the last $20\% - 30\%$ of the phase cycle may fall along a straight line of slope $-c/Q_p$, that

passes through the $(2\pi, 0)$ point in the PRC graph (see, for example, Figure 6.1a). This line is referred to as the causality line (cf. [99]) and indicates the upper limit for the phase advance in the PRC. Although some points will inevitably be near or on the line of causality at the end of the phase cycle, a pronounced causality line is an indicator that the stimulation has reached saturation, and thus has induced nonlinearities in the computation of the PRC. In practice, the pulse amplitudes are set by trial and error and on a case by case basis.

To measure the level of induced nonlinearity for the PRCs we define the *nonlinearity coefficient*, C_{NL}, as the percentage of the data points that fall in a band neighborhood of the causality line. The width of the band neighborhood is set to be 0.03, i.e., any data point that falls in the band $(2\pi - \theta_{st}) - Z(\theta_{st})Q_p/c \leq 0.03$ is considered to be under the effect of induced nonlinearity. This way, the more pronounced the causality line, the higher the C_{NL}.

6.2.3 GA Model

To bridge the theory described in the previous sections to practice, we use a model of a pyramidal neuron introduced by Golomb and Amitai [95], hereafter referred to as the *GA model*, and evaluate the performance of the optimal control (6.9) in simulation. The GA model is a conductance-based Hodgkin-Huxley [50] type model with five dimensions that incorporates three potassium currents (I_{Kdr}, I_{KA}, $I_{K,slow}$), two sodium currents (I_{Na}, I_{NaP}), one leak current (I_L), and an externally applied current (I_{app}). The specifics of this model can be found in [95]. We

present a slightly modified version for the voltage equation where a noise term is added and the external current stimulus in the original equations, $I_{app}(t)$, is split into two parts, I_b and $I_c(t)$, as indicated below:

$$\begin{aligned} \dot{V} &= \frac{1}{c}(-I_{Kdr}(V,n) - I_{KA}(V,b) \\ &- I_{K,slow}(V,z) - I_{Na}(V,h) - I_{NaP}(V) \\ &- I_L(V) + I_b + I_c(t)) + \eta(t), \end{aligned} \tag{6.10}$$

where V is the voltage difference across the neuron's membrane in mV, $c = 1\ \mu\text{F/cm}^2$ is the membrane capacitance, and I_b is a constant baseline current that induces stable periodic spiking of the neuron in the absence of any control current I_c and any noise $\eta(t)$. For $I_b \approx 0.93\ \mu\text{A/cm}^2$, the neuron spikes with a period of $T_s = 100$ ms which gives $\omega = \frac{2\pi}{T_s} = 0.0628$ rad/ms. The noise process $\eta(t)$ is added to generate uncertainties in the ISI seen in real neurons. We take $\eta(t)$ to be a zero-mean Gaussian white noise process with standard deviation 0.15, i.e., $\eta(t) = \mathcal{N}(0, 0.15)$, to generate approximately 10% variability in the ISI for the GA model.

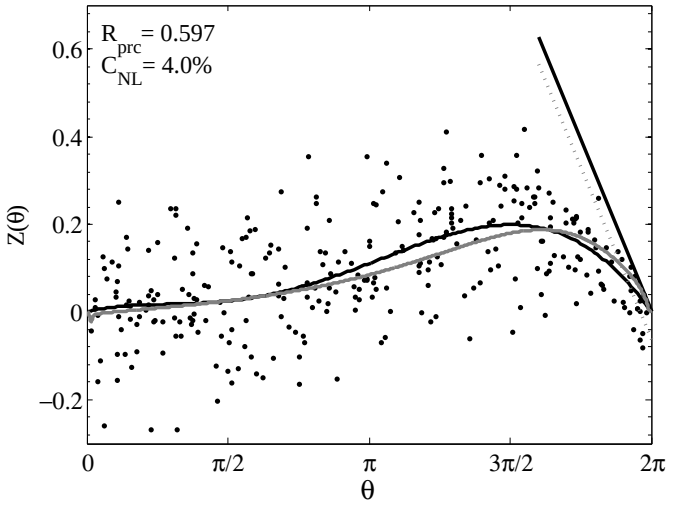
In the presence of small external perturbations $I_c = \mathcal{O}(\epsilon)$, the behavior of the model can be approximated by its phase model. Following standard phase reduction techniques (see, e.g., [32,34]), one can write the phase dynamics for this model as in (6.1) with the unperturbed angular velocity being $\omega = 0.0628$ rad/ms, the control input being $u(t) = I_c/c$ (with units $\mu\text{A}/\mu\text{F}$), and the PRC $Z(\theta)$ being as shown in Figure 6.1a. The two PRC traces shown in Figure 6.1a represent the adjoint PRC (gray) for the noiseless GA model, obtained using XPPAUT, and the fit PRC (black) for the noisy GA model, obtained from the direct method. Pulsatile inputs

of amplitude 2 pA and duration 1 ms have been used for the direct method. This figure also shows the data points, scaled by c/Q_p, with black markers. We see that the fit PRC found using the direct method approximates the adjoint PRC with reasonable accuracy.

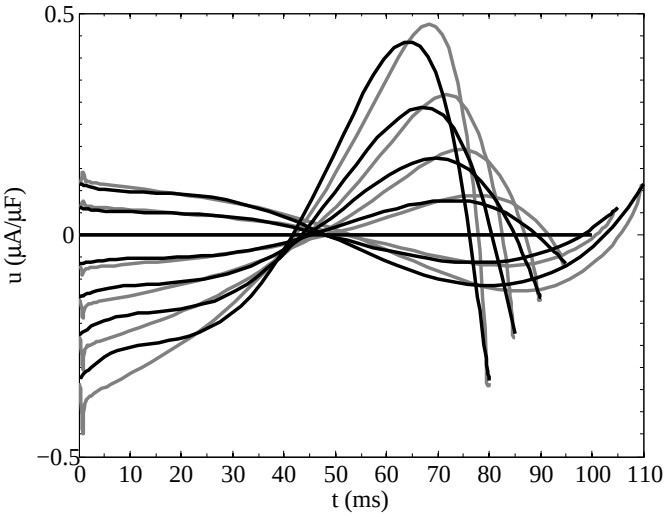
For the simulation shown in Figure 6.1a, the causality line is shown as the straight black line and the neighborhood that is considered for calculating the nonlinearity coefficient is shown with gray dotted line below it. For this simulation, $C_{NL} = 4.0\%$.

6.2.4 Experimental Preparation

The experiments were performed on brain slices of long-evans rats of 14-21 day postnatal age. The rats were deeply anesthetized using isoflurane before extraction of the brain. Once extracted, the brain was bathed in chilled artificial cerebral spinal fluid (aCSF) composed of 124 mM NaCl, 2 mM KCl, 2 mM $MgSO_4$, 1.25 mM NaH_2PO_4, 2 mM $CaCl_2$, 26 mM $NaHCO_3$, and 10 mM D-glucose at pH 7.4, 295 mosM [3]. Transverse slices of the ventral horn of the hipocampal region were sectioned 350 μm thick on a vibratome (Leica Microsystems, Bannockburn, IL). The slices were placed under the microscope with circulating aCSF and neurons were visualized using differential interference contrast optics (Olympus, Center Valley, PA). Whole cell patch-clamp recordings were performed in the CA1 region of the hipocampus using pyramidal cells. Borosilicate capillary pipettes were pulled to 8 MΩ and filled with intracellular recording fluid (ICF) composed of 120 mM K-



(a)



(b)

Figure 6.1: PRC and optimal stimuli for the GA model. (a): Shown in gray, the adjoint PRC for the noiseless GA model (6.10) computed mathematically by solving the adjoint equation using XPPAUT [2]; shown in black, the fit PRC for the noisy GA model (6.10) computed by stimulating the neuron with a pulse every 6^{th} cycle with a 2 pA, 1 ms pulse, measuring the resulting phase change, and fitting a sixth order polynomial to the data that is constrained to zero at both ends and scaled by the pulse charge. The markers show the data points for the noisy system from which the black PRC fit is obtained. An advance in the next spike time corresponds to a positive value in the PRC graph. The diagonal black line on the right represents the line of causality, above which a neuron would be firing prior to the stimulus, and on which represents the stimulus instantaneously evoking an action potential. The dashed line offset from the diagonal represents the area in which action potentials are considered to be evoked by the pulse stimulus, representing a nonlinear response. The nonlinearity coefficient, C_{NL}, is the percentage of the data points that fall in this band neighborhood of the causality line. (b) Charge-balanced minimum energy input stimuli for the GA model found from the adjoint PRC (gray lines) and the fit PRC (black lines) for different values of t_1.

glucose, 10 mM HEPES, 1 mM EGTA, 20 mM KCl, 2 mM $MgCl_2$, 2 mM Na_2ATP, and 0.25 mM Na_3GTP at pH 7.3, 290 mosM. The neuron's membrane potential was amplified and low-pass filtered at 2.4 kHz (Axon 700B; Molecular Devices, Sunnyvale, CA) and digitized on a real-time Linux computer (NiDAQ 6259; National Instruments, Austin, TX).

6.2.5 Dynamic Clamp

Dynamic clamp is a low latency closed-loop control system that connects a computer or an analog device to one or several (virtual or *in vitro*) neurons. In this setting, a patch-clamp amplifier is connected to a data acquisition card (DAQ) and in turn to a computer through a real-time interface [100]. We use the Real-Time eXperiment Interface (RTXI) software, an open source program for real time experiments (www.rtxi.org). RTXI can be used with a variety of DAQ cards through Comedi project (www.comedi.org) which runs on the Real-Time Application Interface (RTAI) real-time Linux nanokernel (www.rtai.org). RTXI is a modular software with a freely available software repository. We carried out experiments at a rate of 5 kHz, which corresponds to a time step of 0.2 ms. The dynamic clamp is used for measuring PRCs in both the virtual neuron and the *in vitro* neurons. It is also used when the Matlab-calculated optimal control waveforms are applied to the neurons.

6.3 Results and Discussion

6.3.1 Simulations

Once the PRC for the GA model is available (see Figure 6.1a), equations (6.5)-(6.9) can be solved to give the optimal input stimulus $u(t)$ for a given target ISI. We consider a set of seven different target ISI values around the neuron's natural ISI, $T_s = 100$ ms, $\text{ISI}_{\mathcal{T}} = [80,\ 85,\ 90,\ 95,\ 100,\ 105,\ 110]$ ms, and find the charge-balanced minimum energy stimuli corresponding to each of the target ISI values using both the adjoint and the fit PRCs of Figure 6.1a. The resulting optimal input stimuli are shown in Figure 6.1b. These optimal controls were applied to the full GA model (6.10) in the following manner. At the onset of an action potential, a target ISI, $\text{ISI}_{\text{targ}} \in \text{ISI}_{\mathcal{T}}$ is randomly selected and its corresponding optimal stimulus waveform is applied to (6.10) as $I_c(t) = c\,u(t)$ for $t \in [0, T_{end}]$, where T_{end} is the minimum of ISI_{targ} and the time of next neuron spiking. In other words, the control input $I_c(t)$ is reset to zero at the onset of the next action potential or when it has been applied fully for one cycle. Once $I_c(t)$ is reset to zero, it will remain zero for three cycles of the neuron firing before it comes back on at the onset of the fourth action potential with another randomly selected ISI_{targ}. The reason for holding the input zero for three cycles is to allow the neuron to return to its original periodic orbit which the PRC was originally computed for. Figure 6.2a shows an example of how the stimulus waveform is applied to the neuron. The optimal stimulus waveform (black) is determined from the fit PRC measured using the direct

method. As shown in the figure, first a target ISI (of, in this case, $\mathrm{ISI}_{\mathrm{targ}} = 90$ ms) is set and its corresponding optimal input is applied. The measured ISI (for this case, $\mathrm{ISI}_{\mathrm{act}} = 87.8$ ms) is labeled below, indicating that the neuron spiked early. Then, the input is set to zero for three cycles before turning back on for another randomly selected target ISI of, in this case, 105 ms and the actual $\mathrm{ISI}_{\mathrm{act}} = 105.4$ ms. Note that for the first application of the input shown, the input is reset to zero at the onset of the next spike, whereas for the second application, the input has been reset to zero when the full cycle of the input is applied. Also, it is seen that due to the presence of noise, during the middle resetting period, the actual ISIs fluctuate around the nominal 100 ms spiking period.

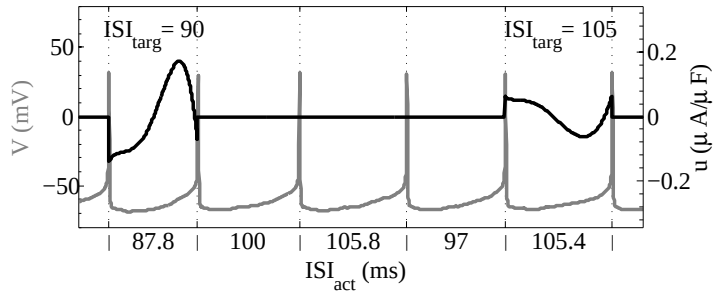
Performance of the controller is measured by correlating the target ISI against the measured actual ISI. The black and gray optimal inputs of Figure 6.1b were applied to the noisy and noiseless full GA model to produce the results shown in figures 6.2b and 6.2c, respectively. From Figure 6.2c, one can see that the controller has been able to achieve the target ISI with a Pearson R_{cont} value of 0.998 for perturbations over a range of 30 ms, which confirms the suitability, and justifies the use, of phase models for controlling neurons. Figure 6.2b shows the same results but for the noisy GA model with the fit PRC. It is seen that even though there is noise in the system and the optimal stimuli are only accurate up to the accuracy of the PRC, the controller has been able to achieve a Pearson R_{cont} value of 0.951 in controlling the ISI of the neuron to the targeted values. We note that the accuracy of the controller is also dependent on the target ISI: the further it is away from the

natural period at which the PRC is measured, the less accurate it becomes.

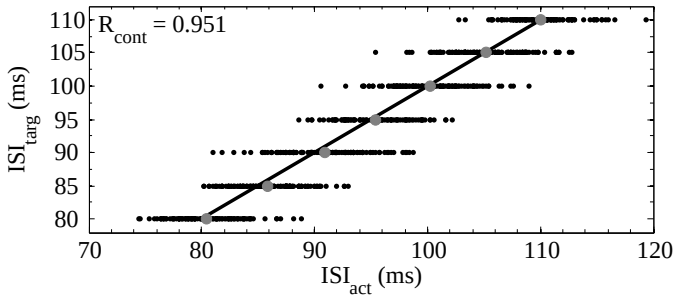
6.3.2 Electrophysiology Experiments

After testing the control algorithm for a virtual neuron, we then tested it for real neurons. This was done by patch-clamping neurons in the CA1 region of hippocampus in brain slices from rats. Neurons were stimulated and recorded using whole cell patch-clamp recording techniques. The applied current was determined using a dynamic clamp and implemented using the RTXI platform. The implementation consists of three parts. In the first part, we control a neuron to spike periodically at an average rate of 100 ms. This is done by implementing an auxiliary closed-loop controller that regulates the baseline current I_b for the neuron. It is important to have stable periodic spiking behavior in the neuron, as the underlying assumption in the formulation developed previously is that the system is on a stable limit cycle. We note that the necessity of having a controller to maintain the periodicity of the neuron comes from the empirical fact that the dynamics of *in vitro* neurons change slowly in time when under laboratory conditions [98]. To compensate for this slow change, a PI controller, previously designed and tested in [98], that makes small perturbations to the DC I_b value at each spike time is used. The DC I_b value is typically around 100 pA, and the amount of perturbation from the PI controller is less than 1 pA per action potential.

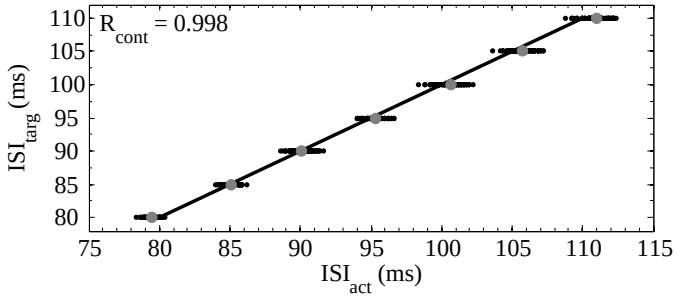
Once the neuron has stabilized around a baseline firing rate, the PRC of the neuron is obtained using the direct method. The neuron is stimulated with a pulse



(a)



(b)



(c)

Figure 6.2: Results for the GA model. (a): An example illustrating the application of the optimal stimulus to the full (noisy) system (6.10). The inputs are from those shown in Figure 6.1b in black. (b): Performance of the phase model and the control characterized by the difference between the actual ISIs and the target ISIs for the noisy full system. (c) Similar results as in (b), but for the noiseless system (6.10) and under charge-balanced minimum energy control stimuli shown as gray lines in Figure 6.1b. We see that the Pearson correlation factor, R_{cont}, between the target ISI values and the actual ISI values increases due to higher accuracy in computing the PRC and the lack of noise in the system.

stimulus every 6^{th} cycle, and the advance in spike time relative to the nominal 100 ms is measured and recorded to obtain the PRC. The phase at which the pulse is injected is randomly chosen every time. As explained before, the reason for applying the pulse stimulus every 6^{th} cycle is to let the neuron return back to its natural periodic orbit before applying the next stimulus; this is to minimize interactions between stimuli which could affect the estimate of the PRC. The input pulse that is used has 1 ms duration and 150 pA amplitude. Although we used pulses with amplitude 150 pA for most of the experiments, we investigated the effect of other pulse amplitudes as reported later.

The phase advance data points are then transferred into Matlab and are fit with a sixth order polynomial constrained to zero at both ends and scaled by c/Q_p to estimate the fit PRC for the neuron under study. In Matlab, we use the fit PRC to solve equations (6.5)-(6.9) for the minimum energy control stimuli for each of the desired target ISI values in the $\text{ISI}_{\mathcal{T}}$ set. The optimal input waveforms are then fed back into RTXI where, at the onset of a spike, a target ISI value from $\text{ISI}_{\mathcal{T}}$ is randomly chosen and its corresponding optimal waveform is applied to the neuron for one cycle. The actual ISI of the neuron for that cycle is measured to evaluate the performance of the controller. We note that since a unit membrane capacitance, c, is assumed, the resulting optimal control inputs $u^*(t)$ can be thought as electrical current stimuli, since $u = I_c/c$.

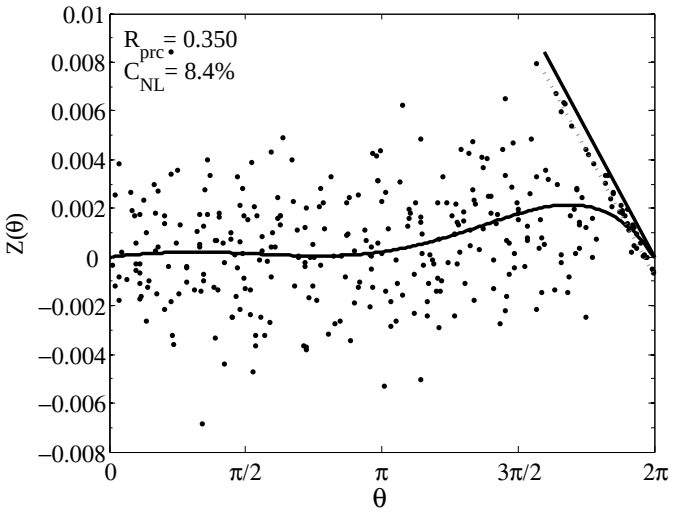
In Figure 6.3 the measured PRC and the calculated optimal inputs for an *in vitro* neuron are shown. The correlation factor for this fit is obtained to be $R_{prc} = 0.350$.

One can see the formation of the causality line in Figure 6.3a. For the example shown in this figure, $C_{NL} = 8.4\%$, which is indicative of reasonable data set for obtaining the PRC. As mentioned before, to ensure realistic PRCs, one must choose pulse amplitudes that are small enough to avoid large nonlinear responses, while at the same time are large enough that can produce a notable variability in the spike time of the neuron.

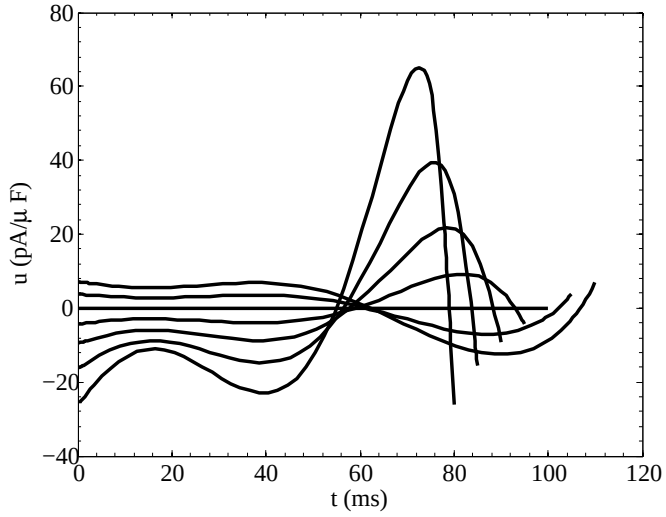
When the optimal control inputs found using the PRC (shown in Figure 6.3b) are applied to the *in vitro* neuron, we get the results shown in Figure 6.4. We note that in the results presented in this figure, we have eliminated any ISI_{act} value less than 30 ms and greater than 200 ms, which amount to less than 1% of the data, as outliers. The first subplot in this figure is a summary result figure, while the others are the histograms for each of the target ISI values of interest. The statistics of the actual ISIs achieved by the neuron are reported in each case in the form of a mean and standard deviation: $\text{ISI}_{\text{act}} = \text{mean} \pm \text{std}$. It is seen that for the case with $\text{ISI}_{\text{targ}} = 100$ ms, the mean is 100.7 ms and the standard deviation is 8.639 ms. Considering the optimal control for this case, which is practically zero according to Figure 6.3b, we can conclude that the standard deviation is a result of the intrinsic noise in the neuron. We see that the standard deviation in all of the other cases is of the same order as in this case, which is indicative of consistency across the different cases. By looking at the mean values, however, one observes that in general, as the target ISI is moved away from the nominal (unstimulated) ISI of $T_s = 100$ ms, the error in the mean values tend to increase. We argue that

the main reason for this is that by targeting ISIs that are further away from T_s, the magnitude of the optimal control inputs increases, which in turn pushes the phase model of the neuron toward the edge of the parameter range for which it is valid.

The method was applied to a total of 9 different neurons from four different rats. A total of 15 recordings were made, as summarized in Table 6.1. It is seen that in finding the PRC, there is a relationship between the pulse amplitude, the PRC correlation coefficient R_{prc}, and the associated PRC nonlinearity coefficient C_{NL}, in that when the pulse amplitude is high (e.g., 200 pA), the R_{prc} and C_{NL} values are also higher. The reason for this is that with higher amplitude pulses, the neuron is being overstimulated, which results in saturation and increased importance of nonlinearities for those pulses applied in the last 20%-30% of the phase. This gives a higher C_{NL}. Moreover, since the PRC fit function is a least squares fit to the data and, by design, it approaches zero at 2π, it produces a higher R_{prc} value as it easily hugs the dense causality line due to overstimulation. A PRC that is overridden by nonlinearities is not reliable and oftentimes results in inefficient control characterized by low R_{cont} values. This is better seen in Figure 6.5a. From the left panel, one sees that higher R_{cont} values are obtained for C_{NL} values less than 20%. These values of C_{NL} correspond to R_{prc} values that are mostly less than 0.5, as shown in the right panel. The data point with $C_{NL} = 37.8\%$ in this figure (shown with solid marker) indicates overstimulation in measuring the PRC, shown in Figure 6.5b. Although this recording gives a very high value of $R_{prc} = 0.645$, the resulting $R_{cont} = 0.423$ value is not very high, proving the unreliability of the PRC.



(a)



(b)

Figure 6.3: PRC and optimal control inputs for an *in vitro* neuron. (a): PRC data points and fit PRC computed from the direct method by stimulating the neuron with a 150 pA, 1 ms pulse every 6^{th} cycle. The fitted curve is a sixth order polynomial that is constrained to be zero at both ends. The data shown here are scaled by the pulse area. The dotted box shows the neighborhood around the causality line (shown as straight dashed line) that is used to compute the nonlinearity coefficient. (b) Charge-balanced minimum energy input stimuli for the neuron found using the fit PRC for different values of t_1.

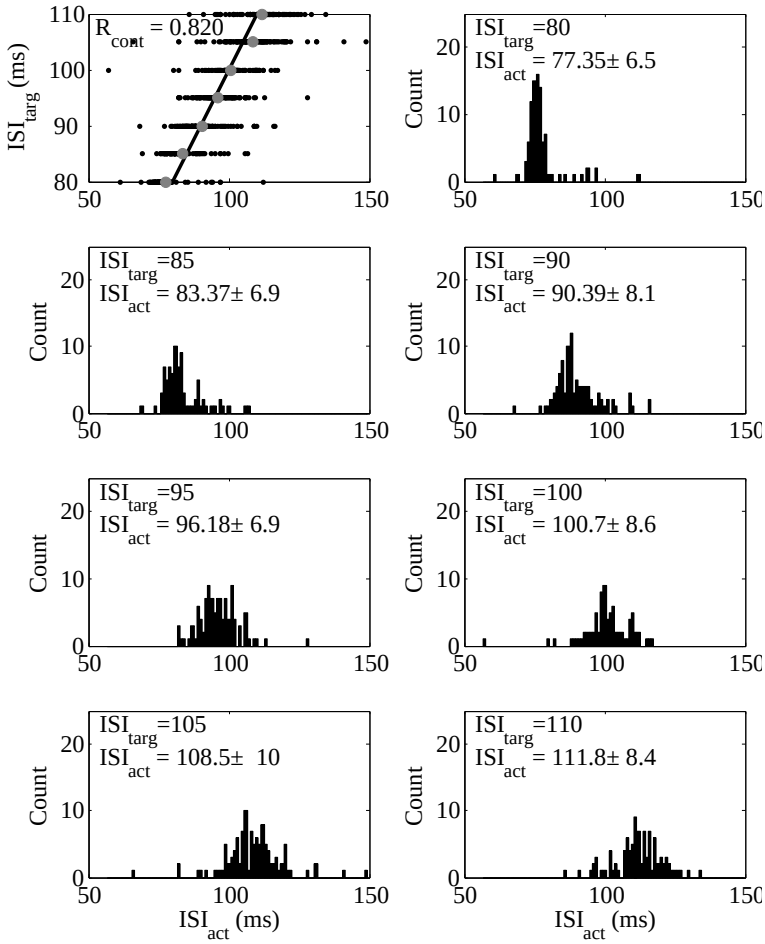


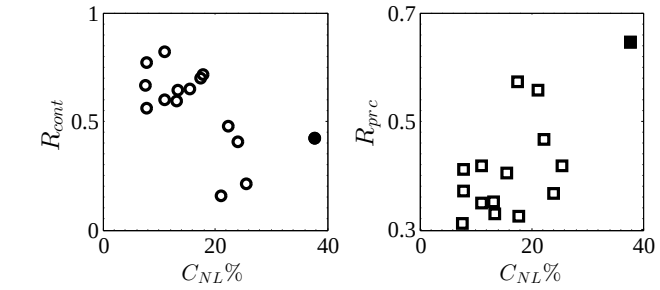
Figure 6.4: Results for an *in vitro* neuron. The upper right subplot shows a summary plot for the control where the data points are the actual ISIs achieved for each of the target ISIs. The mean values of the actual ISIs are shown with gray markers around the straight unit slope black line. The correlation coefficient for the overall controller output across all target ISIs is also reported. Other subplots show histograms and statistical results for control to each of the target ISIs. The actual ISIs achieved for each case is reported in the form: $\mathrm{ISI_{act}} = \mathrm{mean} \pm \mathrm{std}$.

Table 6.1: Results for 9 different *in vitro* neurons and a total of 15 different recordings. Pls Amp gives the pulse amplitude in pA that was used to obtain the PRC through the direct method. R_{prc} gives the correlation coefficient for the PRC. $\%C_{NL}$ give the value of the nonlinearity coefficient for each of the PRC recordings. R_{cont} gives the correlation coefficient for administration of the optimal controls across all seven different target ISIs. Cont. Hold indicates the number of reset cycles that was allowed between every two consecutive control applications.

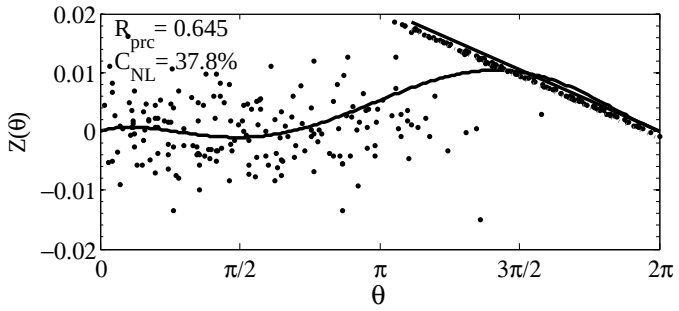
Cell No./Recording	Pls Amp	R_{prc}	$\%C_{NL}$	R_{cont}	Cont. Hold
1/1	200	0.559	21.1	0.160	0
1/2	100	0.352	13.2	0.594	0
1/3	150	0.325	17.8	0.717	0
2/1	200	0.573	17.4	0.697	0
2/2	100	0.312	7.6	0.666	0
2/3	150	0.404	15.5	0.651	0
3/1	200	0.467	22.3	0.478	0
3/2	100	0.330	13.4	0.644	0
4/1	150	0.412	7.8	0.559	6
5/1	150	0.645	37.8	0.423	6
5/2	100	0.419	25.6	0.214	3
6/1	150	0.350	11.0	0.820	3
7/1	150	0.368	24.1	0.404	0
8/1	150	0.373	7.8	0.773	3
9/1	150	0.418	11.1	0.599	3

We did not observe a notable effect from changing the number of holding cycles in the last column of the table. The results shown in figures 6.3 and 6.4 are for the 6/1 row in the table.

To evaluate the performance of the minimum energy control method presented in this chapter, we compare a measure of the input energy obtained for this method with the same measure calculated for a phase change achieved by applying a single pulse input that is optimally timed to occur when the PRC of the neuron is most effective at $\theta \approx 0.7 \times 2\pi$, as used previously for control in [3]. The duration of the input pulse in [3] is fixed at 0.2 ms, but the amplitude is determined as a function of the target ISI. Even though the timing of the pulse control method is optimized



(a)



(b)

Figure 6.5: The nonlinearity coefficient versus the correlation coefficients for the control (top left) and the measured (fit) PRC (top right). It is seen that better control, i.e., a higher R_{cont} value, is achieved when C_{NL} values are less than 20% which have mostly been obtained in cases where the R_{prc} is less than 0.5. If the neuron is overstimulated when measuring the PRC, a high value of C_{NL} is obtained which makes the PRC unreliable despite a high R_{prc} as shown by the solid markers in (a) corresponding to the PRC shown in (b), where the neuron has been overstimulated with pulse amplitudes of 200 pA.

to yield minimum pulse amplitudes, we find that the (continuous-time) minimum energy control method presented here achieves the same task with levels of energy that are, in most cases, two orders of magnitude smaller, which is a significant achievement. The measure of the input energy that is used to do the comparison is of the form of the cost function written in (6.3). To find an average input energy for each of the seven target ISI values, we calculate the quantity $\frac{1}{N_k}\sum_{i=1}^{N_k}(\int_0^{T_{end,k}} u^2 dt)$, where N_k is the number of times that the neuron is controlled to spike at a specific

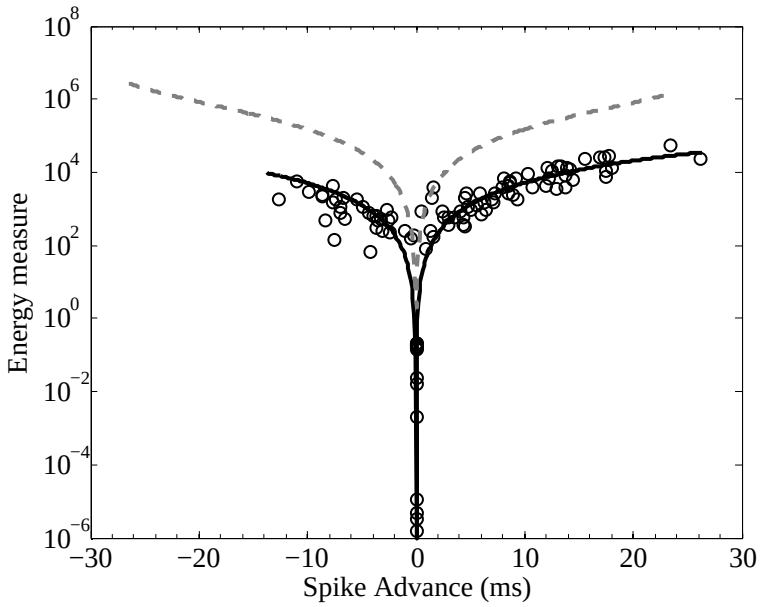


Figure 6.6: Input energy comparison for the method presented in this chapter (markers with solid black fit) with that of [3] (gray). The x-axis presents the spike time advance ($T_s - T_{st}$ in Figure B.1) for all 15 recordings of Table 6.1. The data points for each recording, their least squares quadratic fit (solid black line), and the dashed gray line have been shifted horizontally to yield a minimum at zero for ease of comparison. We see that the method presented in this chapter can reduce the level of the input energy by two orders of magnitude as the ISI of the neuron is controlled to values further away from its natural period.

target ISI, and $T_{end,k}$ is the total time that the corresponding optimal input is applied in the k^{th} round of control application. To evaluate the performance of the presented minimum energy control method, Figure 6.6 shows the results of this comparison between the experimental results reported here and those reported in [3].

6.4 Conclusion

The optimal control method presented here uses a neuron's phase model and PRC to compute the minimum energy waveform for a specific target ISI that is

charge-balanced, continuous, low frequency and low amplitude. We see that the control inputs found from this method reduce the amount of energy required to achieve a target ISI by, in most cases, two orders of magnitude compared to the optimally timed pulsatile input considered in [3]. The results show that the accuracy of the controller declines as the target ISI is set further away from the natural ISI of the uncontrolled neuron. This is due to the fact that the phase model, on which the control algorithm is based, loses validity as the system receives larger control inputs and is pushed further away from its natural periodic orbit. The data also suggest a correlation between the R_{prc}, C_{NL}, and R_{cont} values in that higher R_{cont} values are achieved for lower R_{prc} and C_{NL} values (see top row in Figure 6.5).

We argue that by overstimulating the neuron when measuring the PRC, more nonlinearities are induced which result in unreliable PRCs that can prove inefficient in control. Nonlinearities seem to also result in spurious higher R_{prc} values due to the fit PRC's adherence to the unreliable data points along the causality line. This highlights the importance of not overstimulating the neuron when measuring the PRC. Since much of the accuracy for this method of control results from measuring a good PRC for the neuron, designing more efficient protocols for measuring PRCs would be of interest. It is worth mentioning that by employing the direct method for computing the PRC for an *in vitro* neuron, the control method becomes model-independent and can easily be applied to any neuron. Furthermore, since the resulting control inputs are charge-balanced, there would be no concern for an accumulated charge if the control is applied for long periods of time. Also, when

extended to a population level, the low amplitude and frequency features of this type of control may prove desirable as they can reduce concerns about lesioning due to high amplitudes or other possible side effects due to the high frequency nature of traditional DBS inputs. In addition, since the control method is event-based, the control input is only applied when an event (such as a spike) is elicited, which reduces the number of applications.

Chapter 7

Conclusion and Future Directions

In this dissertation, we have studied different event-based optimal control methods for controlling single and populations of periodically spiking neurons. There were three main parts to this disseration. First, we considered controlling the interspike interval for single neurons represented by phase models. Specifically, in Chapter 2, we considered the time optimal control problem and found analytical expressions for the minimum and maximum values of interspike intervals achievable with small bounded control stimuli. In Chapter 3, we studied the minimum energy control for regulating the interspike interval of a neuron. In both of these chapters, we studied the effect of imposing a charge-balance constraint on the control.

In the second part, we focused on the problem of desynchronization with minimum input energy for populations of pathologically synchronized neurons using a Hamilton-Jacobi-Bellman formulation and dynamic programming techniques. Specifically, in Chapter 4, we achieved desynchronization by applying discrete time

dynamic programming to a network of globally (all-to-all) coupled, synchronized, phase neurons. The formulation was presented using a single input to the network. In Chapter 5, we employed an HJB approach to optimally drive the state of a pathologically synchronized network of neurons to its phaseless set, switch the control off, and let the phases of the neurons randomize under intrinsic background noise. For the example considered, it is shown that the proposed control causes a considerable amount of randomization in the timing of each neuron's next spike, leading to desynchronization for the network.

In the third part, we investigated the suitability of optimal control in practical applications. Specifically, in Chapter 6, we demonstrated the applicability of minimum energy control for designing efficient input stimulus waveforms for single periodically-firing *in vitro* neurons from brain slices of long-evans rats. The PRC for each neuron was experimentally obtained using the direct method. Based on the measured PRC, continuous-time, charge-balanced, minimum energy control waveforms were designed and shown to be effective.

One significant achievement of the work presented in this dissertaion is that it is shown that for a single neuron under intracellular stimulation, one can combine electrophisiology experiments with optimal control theory to achieve control waveforms for *in vitro* neurons with levels of energy that are in some cases multiple orders of magnitude smaller than those of similar past studies. The minimum energy design of the controller may result in longer battery life for neurostimulators.

In the future, it would be interesting to extend the minimum energy control al-

gorithm to populations of real neurons. However, for population level applications, the stimulation would be extracellular and in this setting, one needs to account for the irreversible faradaic reactions that happen at the electrode interface that are not desirable and should be prevented. To prevent irreversible faradaic reactions, the input stimulus should be designed such that it does not remain positive or negative for long periods of time. If the input remains in the same polarity for too long, this would allow the products of the chemical reactions to move away from the stimulating electrode, thus not being available for reverse reactions upon the change in the stimulus polarity [10]. One mathematical challenge is to formulate the optimal control scheme that incorporates this restriction. Also, obtaining a population phase response curve can prove challenging both in practice and in theory. One also needs to account for the location of stimulation and potential correlations between the frequency and the amplitude of the input waveform to ensure an effective input (cf. [97]). Apart from those mentioned here, there are other interesting future directions as well, some of which have been mentioned at the end of each chapter.

Finally, we note that there may be other applications of deep brain stimulation for which the results presented in this dissertation are relevant, in particular treatments of disorders for which the increase in the neurons' firing rate is desirable; this might also be useful for enhancing a person's performance, for example by heightening their attention through targeted stimulation. The results might also be relevant for other stimulated oscillators for which a phase reduction can be per-

formed. In the biological context, this could include the heart as stimulated by an artificial pacemaker, or an organisim's circadian rhythm as stimulated by light or chemicals [44, 101, 102].

Bibliography

[1] G. B. Ermentrout. XPPAUT. Available at `http://www.math.pitt.edu/~bard/xpp/xpp.html`.

[2] G. B. Ermentrout. *Simulating, Analyzing, and Animating Dynamical Systems: A Guide to XPPAUT for Researchers and Students*. SIAM, Philadelphia, 2002.

[3] T. Stigen, P. Danzl, J. Moehlis, and T. Netoff. Controlling spike timing and synchrony in oscillatory neurons. *J. Neurophysiol.*, 105:2074–2082, 2011.

[4] D. Pare, R. Curro'Dossi, and M. Steriade. Neuronal basis of the Parkinsonian resting tremor: a hypothesis and its implications for treatment. *Neuroscience*, 35:217–226, 1990.

[5] A. L. Benabid, P. Pollak, C. Gervason, D. Hoffmann, D. M. Gao, M. Hommel, J. E. Perret, and J. De Rougemont. Long-term suppression of tremor by chronic stimulation of the ventral intermediate thalamic nucleus. *The Lancet*, 337:403–406, 1991.

[6] A. Nini, A. Feingold, H. Slovin, and H. Bergman. Neurons in the globus

pallidus do not show correlated activity in the normal monkey, but phase-locked oscillations appear in the MPTP model of Parkinsonism. *Journal of Neurophysiology*, 74(4):1800–1805, 1995.

[7] J. Volkmann, M. Joliot, A. Mogilner, A. Ioannides, F. Lado, E. Fazzini, U. Ribary, and R. Llinàs. Central motor loop oscillations in Parkinsonian resting tremor revealed magnetoencephalography. *Neurology*, 46(5):1359, 1996.

[8] A. L. Benabid, A. Benazzous, and P. Pollak. Mechanisms of deep brain stimulation. *Movement Disorders*, 17:S73S74, 2002.

[9] R. Kumar, A. Lozano, E. Sime, and A. Lang. Long-term followup of thalamic deep brain stimulation for essential and Parkinsonian tremor. *Neurology*, 61:16011604, 2003.

[10] D. Merrill, M. Bikson, and J. Jefferys. Electrical stimulation of excitable tissue: design of efficacious and safe protocols. *Journal of Neuroscience Methods*, 141(2):171–198, 2005.

[11] T. Foutz and C. McIntyre. Evaluation of novel stimulus waveforms for deep brain stimulation. *J. Neural Eng.*, 7:066008, 2010.

[12] X. Feng, E. Shea-Brown, B. Greenwald, R. Kosut, and H. Rabitz. Optimal deep brain stimulation of the subthalamic nucleus - a computational study. *Journal of Computational Neuroscience*, 23:265–282, 2007.

[13] X. Feng, B. Greenwald, H. Rabitz, E. Shea-Brown, and R. Kosut. Toward

closed-loop optimization of deep brain stimulation for Parkinson's disease: concepts and lessons from a computational model. *J. Neural Eng.*, 4:L14–21, 2007.

[14] S. J. Schiff and T. Sauer. Kalman filter control of a model of spatiotemporal cortical dynamics. *J. Neural Eng.*, 5:1–8, 2008.

[15] S. J. Schiff. Towards model-based control of Parkinson's disease. *Phil. Trans. R. Soc. A*, 368:2269–2308, 2010.

[16] J. Moehlis, E. Shea-Brown, and H. Rabitz. Optimal inputs for phase models of spiking neurons. *ASME J. Comp. Nonlin. Dyn.*, 1:358–367, 2006.

[17] A. Nabi and J. Moehlis. Charge-balanced optimal inputs for phase models of spiking neurons. In *Proceedings of the 2009 ASME Dynamic Systems and Control Conference*, Hollywood, CA, 2009. Article No. DSCC2009-2541.

[18] P. Danzl, J. Hespanha, and J. Moehlis. Event-based minimum-time control of oscillatory neuron models. *Biol. Cybern.*, 101:387399, 2009.

[19] P. Danzl, A. Nabi, and J. Moehlis. Charge-balanced spike timing control for phase models of spiking neurons. *Discrete and Continuous Dynamical Systems Series A*, 28:1413–1435, 2010.

[20] A. Nabi and J. Moehlis. Time optimal control of spiking neurons. *J. Math. Biol.*, 64:981–1004, 2011. DOI: 10.1007/s00285-011-0441-5.

[21] I. Dasanayake and J.-S. Li. Optimal design of minimum-power stimuli for phase models of neuron oscillators. *Physical Review E*, 83:061916, 2011.

[22] A. Nabi, M. Mirzadeh, F. Gibou, and J. Moehlis. Minimum energy spike randomization for neurons. In *Proceedings of the 2012 American Control Conference*, pages 4751–4756, Montreal, Canada, 2012.

[23] A. Nabi, M. Mirzadeh, F. Gibou, and J. Moehlis. Minimum energy desynchronizing control for coupled neurons. *J. Comp. Neuroscience*, 2012. DOI: 10.1007/s10827-012-0419-3.

[24] E. M. Izhikevich. Hybrid spiking models. *Phil. Trans. R. Soc. A*, 368:5061–5070, 2010.

[25] K. Aihara and H. Suzuki. Theory of hybrid dynamical systems and its applications to biological and medical systems. *Phil. Trans. R. Soc. A*, 368:4893–4914, 2010.

[26] A. Nabi and J. Moehlis. Nonlinear hybrid control of phase models for coupled oscillators. In *Proceedings of the 2010 American Control Conference*, pages 922–923, Baltimore, MD, 2010.

[27] C. Wilson, B. Beverlin II, and T. Netoff. Chaotic desynchronization as the therapeutic mechanism of deep brain stimulation. *Front. Syst. Neurosci.*, 5, 2011. Art. No. 50.

[28] O. V. Popovych, C. Hauptmann, and P. A. Tass. Control of neuronal synchrony by nonlinear delayed feedback. *Biological Cybernetics*, 95(1), 2006.

[29] E. Schöll, G. Hiller, P. Hövel, and M. A. Dahlem. Time-delayed feedback in neurosystems. *Phil. Trans. R. Soc. A*, 367:1079–1096, 2009.

[30] G. Orosz, J. Moehlis, and R. Murray. Controlling biological networks by time-delayed signals. *Phil. Trans. R. Soc. A*, 368:439454, 2010.

[31] Y. Kuramoto. *Chemical Oscillations, Waves, and Turbulence*. Springer, New York, 1984.

[32] E. Brown, J. Moehlis, and P. Holmes. On the phase reduction and response dynamics of neural oscillator populations. *Neural Comp.*, 16:673–715, 2004.

[33] E. M. Izhikevich. *Dynamical Systems in Neuroscience: The Geometry of Excitability and Bursting*. The MIT Press, London, 2007.

[34] G. Ermentrout. Type I membranes, phase resetting curves, and synchrony. *Neural Comp.*, 8:979–1001, 1996.

[35] E. Brown, P. Holmes, and J. Moehlis. Globally coupled oscillator networks. In *Perspectives and Problems in Nonlinear Science: A Celebratory Volume in Honor of Larry Sirovich*, pages 183–215. Springer-Verlag, New York, 2003.

[36] R. M. Ghigliazza and P. Holmes. A minimal model of a central pattern generator and motoneurons for insect locomotion. *SIAM J. on Applied Dynamical Systems*, 3(4):671–700, 2004.

[37] A. Cohen, P. Holmes, and R. Rand. The nature of coupling between segmental oscillators of the lamprey spinal generator for locomotion: a model. *J. Math Biol.*, 13:345–369, 1982.

[38] N. Kopell and G. Ermentrout. Phase transitions and other phenomena in chains of coupled oscillators. *SIAM J. Appl. Math.*, 50:1014–1052, 1990.

[39] P. Ashwin and J. Swift. The dynamics of N weakly coupled identical oscillators. *J. Nonlin. Sci.*, 2:69–108, 1992.

[40] D. Hansel, G. Mato, and C. Meunier. Phase dynamics for weakly coupled Hodgkin-Huxley neurons. *Europhys. Lett.*, 25(5):367–372, 1993.

[41] W. Gerstner, L. Van Hemmen, and J. Cowan. What matters in neuronal locking? *Neural Comp.*, 8:1653–1676, 1996.

[42] E. Brown, J. Moehlis, P. Holmes, E. Clayton, J. Rajkowski, and G. Aston-Jones. The influence of spike rate and stimulus duration on noradrenergic neurons. *J. Comp. Neuroscience*, 17:13–29, 2004.

[43] P. A. Tass. *Phase Resetting in Medicine and Biology.* Springer, New York, 1999.

[44] D. B. Forger and D. Paydarfar. Starting, stopping, and resetting biological oscillators: in search of optimal perturbations. *J. theor. Biol.*, 230:521–532, 2004.

[45] A. Nabi and J. Moehlis. Single input optimal control for globally coupled neuron networks. *J. Neural Eng.*, 8, 2011. DOI: 10.1088/1741-2560/8/6/065008.

[46] A. Winfree. *The Geometry of Biological Time, Second Edition.* Springer, New York, 2001.

[47] D. E. Kirk. *Optimal Control Theory: An Introduction.* Dover Publications Inc., New York, 1970.

[48] S. Lenhart and J. T. Workman. *Optimal Control Applied to Biological Models.* Chapman and Hall/CRC, London, 2007.

[49] D. Hansel, G. Mato, and C. Meunier. Synchrony in excitatory neural networks. *Neural Comp.*, 7:307–337, 1995.

[50] A. L. Hodgkin and A. F. Huxley. A quantitative description of membrane current and its application to conduction and excitation in nerve. *J. Physiol.*, 117:500–544, 1952.

[51] A. Bryson and Y. Ho. *Applied Optimal Control.* Halsted Press, Washington, D.C., 1975.

[52] P. Danzl and J. Moehlis. Event-based feedback control of nonlinear oscillators using phase response curves. In *Proceedings of the 46th IEEE Conference on Decision and Control*, pages 5806–5811, New Orleans, LA, 2007.

[53] P. Danzl and J. Moehlis. Spike timing control of oscillatory neuron models

using impulsive and quasi-impulsive charge-balanced inputs. In *Proceedings of the 2008 American Control Conference*, pages 171–176, Seattle, WA, 2008.

[54] G. Ullah and S. J. Schiff. Tracking and control of neuronal Hodgkin-Huxley dynamics. *Phys. Rev. E*, 79:040901(R), 2009.

[55] P. Tass. A model of desynchronizing deep brain stimulation with a demand-controlled coordinated reset of neural subpopulations. *Biol. Cybern.*, 89:81–88, 2003.

[56] J. Hespanha. *An Introductory Course in Noncooperative Game Theory*. 2007. Available at `http://www.ece.ucsb.edu/~hespanha/published`.

[57] S. H. Strogatz. From Kuramoto to Crawford: exploring the onset of synchronization in populations of coupled oscillators. *Physica D*, 143:1–20, 2000.

[58] D. C. Michaels, E. P. Matyas, and J. Jalife. Mechanisms of sinoatrial pacemaker synchronization: a new hypothesis. *Circ. Res.*, 61:704–714, 1987.

[59] J. Buck. Synchronous rhythmic flashing of fireflies II. *The Quaterly Review of Biology*, 63:265–289, 1988.

[60] J. Buck and E. Buck. Synchronous fireflies. *Scientific Am.*, 234:74–85, 1976.

[61] T. Walker. Acoustic synchrony: Two mechanisms in snowy tree cricket. *Science*, 166:891, 1969.

[62] J. Buck and E. Buck. Numerical simulation of a large number of coupled lasers. *J. Opt. Soc. Am.*, 10:155–163, 1993.

[63] S. Yu. Kourtchatov, V. V. Likhanskii, A. P. Napartovich, F. T. Arecchi, and A. Lapucci. Theory of phase locking of globally coupled laser arrays. *Phys. Rev. A*, 52:4089–4094, 1995.

[64] R. A. York and R. C. Compton. Quasi-optical power combining using mutually synchronized oscillator arrays. *IEEE Trans. Microwave Theory Tech.*, 39:1000–1009, 1991.

[65] K. Wiesenfeld, P. Colet, and S.H. Strogatz. Synchronization transitions in a disordered Josephson series array. *Phys. Rev. Lett.*, 76:404–407, 1996.

[66] K. Wiesenfeld, P. Colet, and S.H. Strogatz. Frequency locking in Josephson arrays: Connection with the Kuramoto model. *Phys. Rev. E.*, 57:1563–1569, 1998.

[67] D. Johnston and S. M.-S. Wu. *Foundations of Cellular Neurophysiology*. MIT Press, Cambridge, MA, 1995.

[68] A. Winfree. Patterns of phase compromise in biological cycles. *J. Math. Biol.*, 1:73–95, 1974.

[69] J. Guckenheimer. Isochrons and phaseless sets. *J. Math. Biol.*, 1:259–273, 1975.

[70] J. Guckenheimer and P. J. Holmes. *Nonlinear Oscillations, Dynamical Systems and Bifurcations of Vector Fields*. Springer-Verlag, New York, 1983.

[71] H. K. Khalil. *Nonlinear Systems.* Prentice Hall, Upper Saddle River, NJ, 2002.

[72] A. Sherman and J. Rinzel. Rhythmogenic effects of weak electrotonic coupling in neuronal models. *Proc. Natl. Acad. Sci. USA*, 89:2471–2474, 1992.

[73] J. E. Blankenship and J. T. Haskins. Electrotonic coupling among neuroendocrine cells in aplysia. *J Neurophysiol*, 42:347–355, 1979.

[74] G. B. Ermentrout, J. W. Wang, J. Flores, and A. Gelperin. Model for olfactory discrimination and learning in limax procerebrum incorporating oscillatory dynamics and wave propagation. *J Neurophysiol*, 85:1444–1452, 2001.

[75] X. L. Zhang, L. Zhang, and P. L. Carlen. Electrotonic coupling between stratum oriens interneurones in the intact in vitro mouse juvenile hippocampus. *J Physiol*, 558:825–839, 2004.

[76] P. Mann-Metzer and Y. Yarom. Electrotonic coupling interacts with intrinsic properties to generate synchronized activity in cerebellar networks of inhibitory interneurons. *Journal of Neuroscience*, 19:3298–3306, 1999.

[77] B. Levavi-Sivan, C. L. Bloch, M. J. Gutnick, and I. A. Fleidervish. Electrotonic coupling in the anterior pituitary of a teleost fish. *Endocrinology*, 146:1048–1052, 2005.

[78] J. A. Sanders and F. Verhulst. *Averaging Methods in Nonlinear Dynamical Systems, 2nd Edition.* Springer, New York, 2007.

[79] H. Kori, Y. Kawamura, H. Nakao, K. Arai, and Y. Kuramoto. Collective-phase description of coupled oscillators with general network structure. *Phys. Rev. E*, 80:036207, 2009.

[80] I. Kiss, C. G. Rusin, H. Kori, and J. Hudson. Engineering complex dynamical structures: Sequential patterns and desynchronization. *Science*, 316:1886–1889, 2007.

[81] J. Keener and J. Sneyd. *Mathematical Physiology*. Springer, New York, 1998.

[82] J. Moehlis. Canards for a reduction of the Hodgkin-Huxley equations. *J. Math. Biol.*, 52:141–153, 2006.

[83] H. Osinga and J. Moehlis. A continuation method for computing global isochrons. *SIAM Journal on Applied Dynamical Systems*, 9:1201–1228, 2010.

[84] M. Caputo. *Foundations of dynamic economic analysis: optimal control theory and applications*. Cambridge University Press, Cambridge, 2005.

[85] L. Pontryagin, K. Trirogoff, and L. Neustadt. *The Mathematical Theory of Optimal Processes*. Wiley New York, 1962.

[86] S. Osher and R. Fedkiw. *Level Set Methods and Dynamic Implicit Surfaces*. Springer, New York, first edition, 2003.

[87] J. Sethian. *Level Set Methods and Fast Marching Methods*. Cambridge University Press, second edition, 1999.

[88] M. Crandall and P. Lions. Two approximations of solutions of Hamilton-Jacobi equations. *Mathematics of Computation*, 43(167):1, 1984.

[89] S. Osher and J. Sethian. Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations. *Journal of Computational Physics*, 79(1):12–49, 1988.

[90] A. Harten, B. Engquist, S. Osher, and S. Chakravarthy. Uniformly high order accurate essentially non-oscillatory schemes, III. *Journal of Computational Physics*, 303:231–303, 1987.

[91] C. Shu and S. Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes, II. *Journal of Computational Physics*, 83(1):32–78, 1989.

[92] S. Osher and C. Shu. High-order essentially nonoscillatory schemes for Hamilton-Jacobi equations. *SIAM Journal on Numerical Analysis*, 28(4):907–922, 1991.

[93] I. Mitchell. A toolbox of level set methods. Technical Report UBC CS TR-2007-11, University of British Columbia, 2007.

[94] R. Honeycutt. Stochastic Runge-Kutta algorithms. I. White noise. *Physical Review A*, 45:600–603, 1992.

[95] D. Golomb and Y. Amitai. Propagating neuronal discharges in neocortical

slices: Computational and experimental study. *J. Neurophysiol.*, 78:1199–1211, 1997.

[96] A. Nabi, T. Stigen, J. Moehlis, and T. Netoff. Minimum energy control for *in vitro* neurons. 2012. In preparation.

[97] S. E. Cooper, A. M. Kuncel, B. R. Wolgamuth, A. Rezai, and W. M. Grill. A model predicting optimal parameters for deep brain stimulation in essential tremor. *J Clin Neurophysiol*, 25:265–273, 2008.

[98] O. Miranda-Dominguez, J. Gonia, and T. Netoff. Firing rate control of a neuron using a linear proportional-integral controller. *J. Neural Eng.*, 7, 2010.

[99] T. I. Netoff, M. I. Banks, A. D. Dorval, C. D. Acker, J. S. Haas, N. Kopell, and J. A. White. Synchronization in hybrid neuronal networks of the hippocampal formation. *J Neurophysiol*, 93:1197–1208, 2005.

[100] A. Dorval, D. Christini, and J. White. Real-time linux dynamic clamp: a fast and flexible way to construct virtual ion channels in living cells. *Annals of Biomedical Engineering*, 29:897–907, 2001.

[101] V. L. Revell. How to trick mother nature into letting you fly around or stay up all night. *J Biol Rhythms*, 20:353–365, 2005.

[102] O. S. Shaik, S. Sager, O. Slaby, and D. Lebiedz. Phase tracking and restora-

tion of circadian rhythms by model-based optimal control. *Systems Biology, IET*, 2:16–23, 2008.

[103] J. Rinzel and R. N. Miller. Numerical calculations of stable and unstable periodic solutions to the Hodgkin-Huxley equations. *Math. Biosci.*, 49:27–59, 1980.

[104] R. Rose and J. Hindmarsh. The assembly of ionic currents in a thalamic neuron I. The three-dimensional model. *Proc. R. Soc. Lond. B*, 237:267–288, 1989.

[105] J. Rinzel and G. B. Ermentrout. Analysis of neural excitability and oscillations. In *Methods in Neuronal Modeling*, pages 251–291. MIT Press, 1998.

[106] C. Shu and S. Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes. *Journal of Computational Physics*, 77(2):439–471, 1988.

[107] X. Liu, S. Osher, and T. Chan. Weighted essentially non-oscillatory schemes. *Journal of Computational Physics*, 115(1):200–212, 1994.

[108] S. Gottlieb, C. Shu, and E. Tadmor. Strong stability-preserving high-order time discretization methods. *SIAM Review*, 43:89, 2001.

Appendix A

Phase Models for Neurons

Phase models can be derived for any oscillatory system, such as a neuron. In this appendix, we review the reduction of neuron models to phase models.

There are different mathematical models for neurons developed in the literature. One class of these models are the conductance-based models. As examples of this group we can mention the Hodgkin-Huxley model [50, 103], Rose-Hindmarsh model [104], Fitzhugh-Nagumo model [81], and Morris-Lecar model [105]. A general conductance-based model of a single neuron is written as:

$$\begin{cases} C\dot{V} = I^g(V, \mathbf{n}) + I^b + I(V, t), \\ \dot{\mathbf{n}} = \mathbf{N}(V, \mathbf{n}), \quad (V, \mathbf{n})^T \in \mathbb{R}^n, \end{cases} \tag{A.1}$$

where V is the voltage across the membrane of the neuron, $\mathbf{n}$ is an $(n-1)$ dimensional vector referred to as the vector of the gating variables (gating variables control the ionic flow across the membrane of the neuron), $I^g(V, \mathbf{n})$ represents the

membrane currents, I^b is the baseline inward current which effectively sets the spiking frequency of the neuron, and $I(V,t)$ represents the effect of all synaptic and external currents from other areas of the brain or a stimulation electrode which stimulate the neuron [32]. If we neglect reversal potentials we can write $I(V,t)$ as $I(t)$, without any dependence on V.

The system (A.1) can more compactly be written as:

$$\dot{\mathbf{X}} = \mathbf{F}(\mathbf{X}) + \epsilon \mathbf{G}(\mathbf{X}, t), \quad \mathbf{X} = (V, \mathbf{n})^T \in \mathbb{R}^n, \tag{A.2}$$

where $\mathbf{F}(\mathbf{X})$ is the baseline vector field and $\epsilon\mathbf{G}(\mathbf{X},t)$ is the stimulus effect in the form of a disturbance. The important assumption here is that when $\mathbf{G}(\mathbf{X},t) \equiv 0$ the neuron fires (or spikes) periodically with a frequency determined by the baseline vector field $\mathbf{F}(\mathbf{X})$. In other words, the neural oscillator has a hyperbolic attracting limit cycle γ in its phase portrait. We assume that such a limit cycle always exists for each individual neuron. Note that the hyperbolic assumption guarantees structural stability to small perturbations $\mathbf{G}(\mathbf{X}$,t), and attractivity basically means local asymptotic stability of the limit cycle.

Now the objective here is to further simplify (A.2) by reducing it to a one-dimensional scalar equation. To this end, we introduce the following definition.

Definition The scalar phase variable of the neuron system (A.2) is defined as $\theta(\mathbf{X}) : \mathbb{R}^n \to \mathbb{R}$ where $\theta(\mathbf{X}) \in [0, 2\pi)$ for all $\mathbf{X}$ in some neighborhood $\mathcal{U}$ of γ, where

$\mathcal{U} \subset \mathcal{B}_\gamma$ with $\mathcal{B}_\gamma$ denoting the basin of attraction of γ, such that the phase evolution takes the simple form:

$$\frac{d\theta(\mathbf{X})}{dt} = \omega \quad when \quad \mathbf{G(X,t)} \equiv 0, \quad \forall \mathbf{X} \in \mathcal{U}. \tag{A.3}$$

In this definition the baseline frequency of the neural oscillator is $\omega = \frac{2\pi}{T}$ where T is the period of equation (A.2) with $\mathbf{G(X,t)} \equiv 0$. Note that $\theta(\mathbf{X})$ is a scalar field.

We can associate another definition with this definition for $\theta(\mathbf{X})$ and that is the definition of *isochrons*. For a system with a stable periodic orbit (or attracting limit cycle) isochrons are defined as follows.

Definition Let γ be a stable periodic orbit. Let $\mathbf{X}(0)$ be a point on γ; the isochron associated with $\mathbf{X}(0)$ is the set of all initial conditions $\mathbf{Y}(0)$ such that

$$\lim_{t\to\infty} \|\mathbf{X}(t) - \mathbf{Y}(t)\| \to 0.$$

That is, at each t, $\mathbf{Y}(t)$ has the same constant ω as $\mathbf{X}(t)$, but its radial distance to γ is reducing due to the stability of the periodic orbit. It has been proven that isochrons exist for any smooth vector field with a hyperbolic stable limit cycle [69]. Figure A.1 shows a schematic of the periodic orbit of the firing neuron in the phase

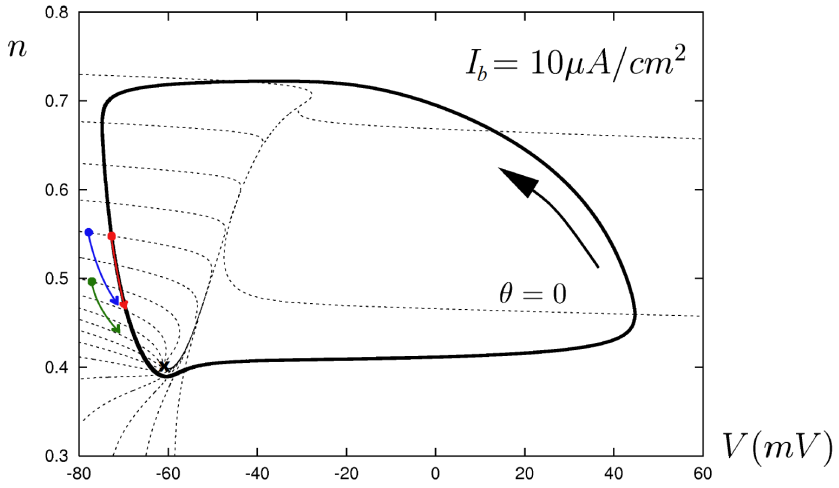


Figure A.1: Two-dimensional phase portrait of a neural oscillator in its periodic firing mode. The periodic orbit is locally asymptotically stable with the basin of attraction $\mathcal{B}_\gamma$ being the entire state space minus the unstable fixed point shown with an $\mathbf{x}$ marker. The phase variable θ along with the isochrons are shown. It is shown, schematically, that if two states intiate on the same isochron, they converge to the same point on γ, whereas if they intiate on different isochrons, the maintains their initial phase difference due to having the same ω. Although the concept of isochrons is defined for everywhere within the basin of attraction, the phase model is only valid in a neighborhood close to the periodic orbit.

space along with its basin of attraction and the isochrons. By convention, $\theta = 0$ corresponds to the maximum value of V, i.e., when the neuron fires.

We also note that, according to our definition of θ and the isochrons, we have less dense isochrons when the dynamics of the system are fast, and more dense isochrons when the dynamics are slow. So in figure A.1, the isochrons, equally spaced in phase, are far apart on the righthand side of the periodic orbit where the neuron fires and closer on the left.

Having defined $\theta(\mathbf{X})$, we can now write:

$$\frac{d\theta(\mathbf{X})}{dt} = \frac{\partial\theta(\mathbf{X})}{\partial\mathbf{X}} \cdot \frac{d\mathbf{X}}{dt} = \frac{\partial\theta}{\partial\mathbf{X}} \cdot \mathbf{F}(\mathbf{X}) + \frac{\partial\theta}{\partial\mathbf{X}} \cdot \epsilon\mathbf{G}(\mathbf{X}, t). \tag{A.4}$$

Since, by construction $\frac{d\theta}{dt} = \omega$ when $\mathbf{G}(\mathbf{X}, t) \equiv 0$, we can write

$$\frac{d\theta(\mathbf{X})}{dt} = \omega + \frac{\partial \theta}{\partial \mathbf{X}} \cdot \epsilon \mathbf{G}(\mathbf{X}, t). \tag{A.5}$$

This is a first order partial differential equation that the scalar field $\theta(\cdot)$ must satisfy. Even after $\theta(\cdot)$ has been found, this equation is not a phase-only (and hence self-contained) description of the oscillator dynamics. This is because it is dependent on $\mathbf{X}$ rather than just θ. However, evaluating the vector field at the periodic orbit $\mathbf{X}^\gamma(\theta)$, which is defined as the intersection of γ and the $\theta(\mathbf{X})$ level sets (i.e., isochrons) gives an equation which is only in terms of θ and t:

$$\frac{d\theta(\mathbf{X})}{dt} = \omega + \frac{\partial \theta}{\partial \mathbf{X}}\big(\mathbf{X}^\gamma(\theta)\big) \cdot \epsilon \mathbf{G}\big(\mathbf{X}^\gamma(\theta), t\big) + \mathcal{E},$$

where $\mathcal{E}$ represents the error terms of $\mathcal{O}(|\mathbf{G}|^2)$. We can think of this as a linearization about $\mathbf{G} = 0$ for small perturbations $\mathbf{G}(\mathbf{X}, t)$. Neglecting the error term, for small perturbations close to the periodic orbit we can write:

$$\frac{d\theta(\mathbf{X})}{dt} = \omega + \frac{\partial \theta}{\partial \mathbf{X}}(\mathbf{X}^\gamma(\theta)) \cdot \epsilon \mathbf{G}(\mathbf{X}^\gamma(\theta), t). \tag{A.6}$$

We define $\mathbf{Z}(\theta) = \frac{\partial\theta}{\partial\mathbf{X}}(\mathbf{X}^\gamma(\theta))$, and realize that for the system under consideration (i.e., system (A.1)), $\epsilon\mathbf{G}(\mathbf{X},t)$ only has one entry: $\epsilon\mathbf{G}(\mathbf{X},t) = [I(t)/C \quad 0 \quad 0 \quad \cdots \quad 0]^T$. This means that only the first entry of $\mathbf{Z}(\theta)$, $Z_V(\theta)$ comes into play. $Z_V(\theta)$, or as more commonly denoted, $Z(\theta)$ is called the Phase Response Curve (PRC) for the neuron. Also, we can replace ω by a more general function $f(\theta)$. So, in summary, the phase model is written as

$$\frac{d\theta}{dt} = f(\theta) + Z(\theta)u(t), \tag{A.7}$$

where $f(\theta)$ represents the neuron's baseline dynamics, $Z(\theta)$ is the PRC of the neuron, and $u(t) = I(t)/C$ is the input stimulus. By convention $\theta = 2k\pi$, $k = 0, 1, 2, \cdots$ corresponds to the spiking of the neuron. For $u(t) = 0$, the neuron would fire at its natural period T determined by $f(\theta)$.

The concept of phase is extended from the periodic orbit to its entire basin of attraction by defining isochrons using the idea of asymptotic phase [32, 69]. Isochrons are level sets of the phase within the basin of attraction [46]. The points at which all isochrons converge are where a phase value can not be defined and hence, the collection of these points make the phaseless set for the oscillator.

Appendix B

Adjoint and Direct methods for PRCs

For computational models of periodically spiking neurons, the PRC, and hence the phase model, can be obtained either by solving the appropriate adjoint equation or by implementing the so-called direct method [33]. For biological neurons however, one can only use the direct method to obtain the PRC experimentally.

In the adjoint method, for an n-dimensional model of an oscillator (such as a periodically spiking neuron represented by (A.2)), the PRC is obtained as the first element of the solution of the following adjoint equation which is an n-dimensional system of first order linear differential equations obtained for the system in the presence of small external impulsive inputs [32, 34]:

$$\frac{d\mathbf{Z}(t)}{dt} = \left[-\frac{\partial \mathbf{F}}{\partial \mathbf{X}}(\mathbf{X}^{\gamma}(t))\right]^{T} \mathbf{Z}(t), \tag{B.1}$$

where $\mathbf{Z}(t) = \frac{\partial \theta}{\partial \mathbf{X}}(\mathbf{X}^\gamma(t))$. The condition for this equation can be written as

$$\frac{\partial \theta}{\partial \mathbf{X}}(\mathbf{X}^\gamma(0)) \cdot \mathbf{F}(\mathbf{X}^\gamma(0)) = \omega, \tag{B.2}$$

where, ω is the constant angular velocity of oscillations in the absence of external stimuli. This equation supplies only one condition for the n-dimensional system (B.1). The rest arise from requiring periodicity for the solutions of $\mathbf{Z}(t)$ in (B.1).

In the adjoint method, it is required that the system's periodic orbit be a stable periodic orbit that would attract the state of the system when perturbed away from it as a result of applying a small external input. Also, the underlying assumption in deriving the linear adjoint equation is that the external input is small enough so that when perturbed, the system remains close to its periodic orbit. This perturbation is quantified by the oscillator's PRC: the change in the oscillator's phase as a function of the phase at which the stimulus is applied. To solve the adjoint equation for different models numerically, we use the software XPPAUT [1]. We refer to the PRC found through solving the adjoint equation as the *adjoint PRC*.

In the direct method [33], a short-duration pulse (that approximates an impulse) is injected into the periodically spiking neuron at various phase values, that results in a change in the next spike time for the neuron (see Figure B.1). This change in the next spike time is measured and recorded as a function of the phase at which the stimulus was applied to give the PRC. For the results shown in this dissertation, the pulsatile input has been applied once every 6^{th} cycle of the neuron's oscillation to allow time for the neuron to settle back on its periodic orbit before the next pulsatile input. Once the data points for the phase advances are collected, a sixth

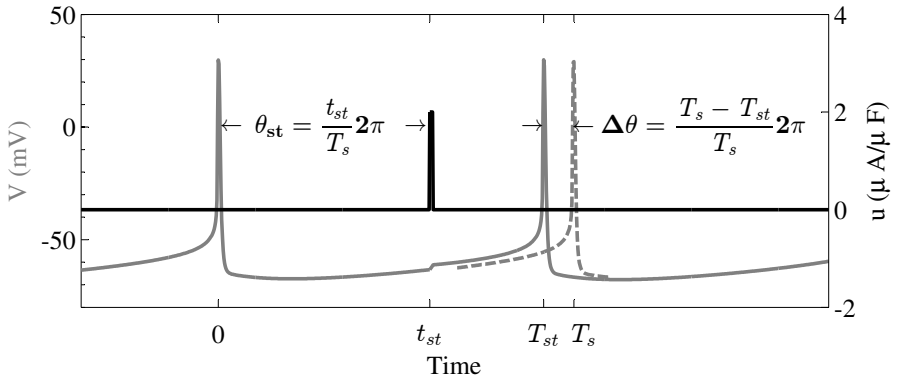


Figure B.1: Measuring PRC with the direct method. Every 6^{th} cycle, a pulse stimulus (solid black) is applied at a random time t_{st}, equivalent to a random phase θ_{st}, in the neuron's cycle which changes the next spiking time of the neuron from its natural ISI T_s (indicated with the dashed gray voltage trace) to a stimulated ISI T_{st} (as indicated with the solid gray voltage trace). The resulting advance in the sdspiking time, $(T_s - T_{st})$, is measured and converted to phase $\Delta\theta$. This phase advance in then normalized by Q_p/c, where Q_p is the total charge of the pulse stimulus and c is the membrane capacitance, to give a single PRC data point $(\theta_{st}, Z(\theta_{st}))$, where $Z(\theta_{st}) = c\Delta\theta/Q_p$. The GA model, in the absence of noise, with $T_s = 100$ ms, pulse amplitude of 2 μA/cm^2 and duration 1 ms is used for this figure.

order polynomial is fitted to the data which is constrained to be zero at both ends of the phase, i.e., at $\theta = 0$ and $\theta = 2\pi$, and is scaled by c/Q_p, where Q_p is the pulse electrical charge and c is the membrane capacitance. The resulting curve is the PRC from the direct method, which we refer to as the *fit PRC*. We note that the reason for constraining the fit PRC to be zero at both ends is to account for the fact that biological neurons generally show little or no sensitivity to inputs right at their spiking instance [99].

Appendix C

Hodgkin-Huxley's Full Model

The specifics of the Hodgkin-Huxley equations are as follows [32, 50]:

$$\begin{aligned}
\dot{V} &= \left(I_b + I(t)\right)/c + \\
&\underbrace{(-\bar{g}_{Na}h(V - V_{Na})m^3 - \bar{g}_K(V - V_K)n^4 - \bar{g}_L(V - V_L))}_{I_g(V,m,h,n)}/c, \\
\dot{m} &= a_m(V)(1 - m) - b_m(V)m, \\
\dot{h} &= a_h(V)(1 - h) - b_h(V)h, \\
\dot{n} &= a_n(V)(1 - n) - b_n(V)n,
\end{aligned} \tag{C.1}$$

where,

$$\begin{aligned}
a_m(V) &= 0.1(V + 40)/(1 - \exp(-(V + 40)/10)), \\
b_m(V) &= 4\exp(-(V + 65)/18), \\
a_h(V) &= 0.07\exp(-(V + 65)/20), \\
b_h(V) &= 1/(1 + \exp(-(V + 35)/10)),
\end{aligned}$$

$$a_n(V) = 0.01(V + 55)/(1 - \exp(-(V + 55)/10)),$$

$$b_n(V) = 0.125 \exp(-(V + 65)/80),$$

$$V_{Na} = 50 \text{ mV},\ V_K = -77 \text{ mV},\ V_L = -54.4 \text{ mV},$$

$$\bar{g}_{Na} = 120 \text{ mS/cm}^2,\ \bar{g}_K = 36 \text{ mS/cm}^2,$$

$$\bar{g}_L = 0.3 \text{ mS/cm}^2,\ c = 1\ \mu\text{F/cm}^2,$$

in which $V \in \mathbb{R}$ is the voltage across the neuron membrane, $[m, h, n]^{\mathrm{T}} \in \mathbb{R}^3_{[0,1]}$ is the vector of *gating* variables which correspond to the state of the membrane's ion channels, $c \in \mathbb{R}^+$ is the constant membrane capacitance, $I_g : \mathbb{R} \times \mathbb{R}^3 \mapsto \mathbb{R}$ is the sum of the membrane currents, and $I : \mathbb{R} \mapsto \mathbb{R}$ is the stimulus current. $I_b \in \mathbb{R}$ is the baseline current, which represents the effect of other parts of the brain on the neuron under consideration and can be viewed as a bifurcation parameter in the model that controls whether the neuron is in an excitable or an oscillatory regime. In, e.g., [82] it is discussed that in this model for $I_b < 6.26\ \mu\text{A/cm}^2$ the neuron would be in excitable mode where it does not show spontaneous periodic spiking. For $6.26 \leq I_b \leq 9.78\ \mu\text{A/cm}^2$ there is a bistable regime in which the neuron can be excitable or oscillatory, and for $I_b > 9.78\ \mu\text{A/cm}^2$ the neuron would be in oscillatory regime where it has a stable periodic orbit and oscillates with period $T_s < +\infty$.

Appendix D

Dynamic Programming in Matlab

In accordance with Section 4.3.1, let `dpsi` and `du` be the grid size for the phase differences and the control, respectively. For a system of three Kuramoto neurons, we have two phase difference angles that comprise a state space of size `nx` when discretized. The following demonstrates how one can implement dynamic programming for the Kuramoto network example presented in Section 4.4 using vectorized Matlab coding.

```
% total number of states for the discretized system
 nx = (360/dpsi)^2;
 sqnx = sqrt(nx);
% states for the system
 x = (1:nx)';
```

```
% converting state x to angle (in degrees):  Section 4.3.1
 angle(:,2)=2*((mod(x,360/dpsi)==0)*360/dpsi
        + (mod(x,360/dpsi)~=0).*mod(x,360/dpsi));
 angle(:,1)=2*(round((x-((mod(x,360/dpsi)==0)*360/dpsi
        + (mod(x,360/dpsi)~=0).*mod(x,360/dpsi)))/(360/dpsi))+1);
% meshgrid over all possible u's and x's
 [umesh,xmesh] = meshgrid(-umax:du:umax, 1:1:nx);
% create meshgrid-ed matrices Psi_3 and Psi_2 (in radians) using xmesh
 Psi_3 = 2*((mod(xmesh,360/dpsi)==0)*360/dpsi
 +(mod(xmesh,360/dpsi)~=0).*mod(xmesh,360/dpsi))*pi/180;
 Psi_2 = 2*(round((xmesh-((mod(xmesh,360/dpsi)==0)*360/dpsi
 +(mod(xmesh,360/dpsi)~=0).*mod(xmesh,360/dpsi)))/(360/dpsi))+1)*pi/180;
% solving the dynamics
 Psi_2dot=(-2*a12*sin(Psi_2)+a23*sin(Psi_3-Psi_2)-a13*sin(Psi_3));
 Psi_3dot=(-2*a13*sin(Psi_3)-a23*sin(Psi_3-Psi_2)-a12*sin(Psi_2))+umesh;
% forward integration and phase wrapping
 Psi_2plus = mod(Psi_2+Psi_2dot*delta_t,2*pi)*180/pi;
 Psi_3plus = mod(Psi_3+Psi_3dot*delta_t,2*pi)*180/pi;
```

```
% quantization
 rem1 = mod(Psi_2plus,dpsi);
 rem2 = mod(Psi_3plus,dpsi);
 Psi_2plus = mod(fix(Psi_2plus/dpsi)+(rem1>=dpsi/2),(360/dpsi));
 Psi_3plus = mod(fix(Psi_3plus/dpsi)+(rem2>=dpsi/2),(360/dpsi));
 c1 = Psi_2plus<1e-2;
 c2 = Psi_3plus<1e-2;
 Psi_2plus(c1) = sqnx;
 Psi_3plus(c2) = sqnx;
% time-independent state transition matrix:  constant for all time steps
% converting angle values to corresponding state numbers x
 F{1} = ((Psi_2plus-1)*sqnx+Psi_3plus);
% end point cost R and time additive cost G: constant for all time steps
 R{1} = sqrt((1+cos(Psi_2)+cos(Psi_3)).^2 + (sin(Psi_2)+sin(Psi_3)).^2 );
 gamma = 0.0001;
 G{1} = gamma^2*umesh.^2;
% end time step
 K = round(T/delta_t+1);
% target set
 ISImin = 10;
 X_targ = x(Dangle(:,1)>ISImin & (360-Dangle(:,1))>ISImin &
           angle(:,2)>ISImin & (360-angle(:,2))>ISImin &
           abs(angle(:,1)-angle(:,2))>ISImin);
```

```
% target set in terms of phase differences
 Psi_3targ = 2*((mod(X_targ,360/dpsi)==0)*360/dpsi
          +(mod(X_targ,360/dpsi)~=0).*mod(X_targ,360/dpsi))*pi/180;
 Psi_2targ = 2*(round( ( X_targ- ((mod(X_targ,360/dpsi)==0)*360/dpsi
          +(mod(X_targ,360/dpsi)~=0).* ...
          mod(X_targ,360/dpsi)))/(360/dpsi) )+1)*pi/180;
% dynamic programming backward loop
 V{K+1} = inf*ones(size(G{1},1),1);
 V{K+1}(X_targ) = sqrt((1+cos(Psi_2targ)+cos(Psi_3targ)).^2
          + (sin(Psi_2targ)+sin(Psi_3targ)).^2);
for k=K:-1:1
 V{k} = min(G{k}+V{k+1}(F{k}),[],2);
end


% initial condition:  in-phase condition (in degrees)
IC=[360 360];
x(1)=(IC(1)/dpsi-1)*sqnx+IC(2)/dpsi;
% control space
 uplot = -umax:du:umax;
```

```
% dynamic programming forward loop

for k=1:K

 [dummy, u_ind] = min(G{k}(x(k),:)+V{k+1}(F{k}(x(k),:))',[],2);

 x(k+1) = F{k}(x(k),u_ind);

 control(k) = uplot(u_ind);

end
```

Appendix E

Solving the Hamilton-Jacobi Equation

This appendix describes careful implementation of steps that one needs to take to obtain a convergent, high-order approximation to the HJ equation.

A. Essentially Non-Oscillatory (ENO) schemes

Finite difference approximations of derivatives of a function $\mathcal{V} : \mathbb{R}^n \mapsto \mathbb{R}$, are essentially equivalent to choosing an interpolation polynomial for the function $\mathcal{V}$, and performing exact differentiation. In traditional finite difference methods, the polynomial *stencil* is fixed, i.e., to approximate $\mathcal{V}_x \equiv \frac{\partial \mathcal{V}}{\partial x}$, at the grid point x_i, one assumes,

$$\mathcal{V}_x|_i \approx f(\mathcal{V}_{i-m}, \mathcal{V}_{i-m+1}, \ldots, \mathcal{V}_i, \ldots, \mathcal{V}_{i+n-1}, \mathcal{V}_{i+n}),$$

where constants m and n are fixed in space and are the same for all points (except maybe at the boundaries). Here, the subscript indices refer to the grid points, i.e., $\mathcal{V}_i \equiv \mathcal{V}(x_i)$. Problems arise when the function $\mathcal{V}$ is not sufficiently smooth and this interpolation, when combined with time integration, results in spurious oscillations and even divergence.

To remedy this problem, Harten et al. [90] first introduced the idea of *essentially non-oscillatory* (ENO) schemes. Unlike traditional finite difference methods, in ENO schemes the polynomial stencil is not fixed, and at each point, one chooses the smoothest possible polynomial. When combined with a TVD time integration method (see below), the scheme is guaranteed not to produce any spurious solutions.

This idea was further improved, from the implementation point of view, by Shu [91, 106] and later applied to the numerical solution of the HJ equation [92]. Here we merely consider the method in one spatial dimension. Extension of the method to higher spatial dimensions is possible through a dimension-by-dimension approach and we leave the details to the appropriate references mentioned above.

Consider the one-dimensional HJ equation written as,

$$\frac{\partial \mathcal{V}}{\partial t} + \mathcal{H}(\mathcal{V}_x) = 0,$$

where the explicit dependence on other variables have been dropped for brevity. To construct a polynomial function, $\mathcal{Q}_j^{j+\mathcal{M}}(x)$, of degree $\mathcal{M}$ that interpolates through points $x_j, x_{j+1}, \ldots, x_i, \ldots, x_{j+\mathcal{M}}$, one first needs to define the m^{th}-order Newton divided differences coefficients, f_j^{j+m}, for $m = 0, 1, \ldots, \mathcal{M}$. This is done via the

recursive formula,

$$f_j^{j+m} = \frac{f_{j+1}^{j+m} - f_j^{j+m-1}}{x_{j+m} - x_j},$$

with $f_j^j = \mathcal{V}_j = \mathcal{V}(x_j)$. Here the subscript index, j, and superscript index, $j+m$, refer to the lower and upper bounds of the interval used to compute the coefficient, i.e., to compute f_j^{j+m}, $m+1$ grid points, x_j through x_{j+m}, are required. Using the divided differences coefficients, the polynomial $\mathcal{Q}_j^{j+\mathcal{M}}(x)$ may be written as,

$$\mathcal{Q}_j^{j+\mathcal{M}}(x) = \sum_{m=0}^{\mathcal{M}} f_j^{j+m} \phi_j^m(x), \tag{E.1}$$

where, by definition, $\phi_j^0(x) = 1$ and, for $m \geq 1$,

$$\phi_j^m(x) = \prod_{n=0}^{m-1} (x - x_{j+n}).$$

Once the interpolating polynomial, $\mathcal{Q}_j^{j+\mathcal{M}}(x)$, is known, equation (E.1) may be differentiated with respect to x, to obtain the following approximation to the spatial derivative:

$$\mathcal{V}_x|_i = \left.\frac{d\mathcal{Q}_j^{j+\mathcal{M}}}{dx}\right|_{x_i} + \mathcal{O}\left(h^{\mathcal{M}}\right),$$

where $h \equiv \max_{1 \leq n \leq \mathcal{M}} |x_{j+n} - x_{j+n-1}|$, is the maximum grid spacing in the interpolation interval and determines the order of truncation error.

Constructing the polynomial $\mathcal{Q}_j^{j+\mathcal{M}}(x)$ also requires the knowledge of the interpolation interval, $\mathcal{I} \equiv [x_j, x_{j+1}, \ldots, x_i, \ldots, x_{j+\mathcal{M}}]$. To build the interval, and starting at point x_i, one has two possible options for choosing the next point, x_{i-1} or x_{i+1}. In fact, these are both valid, and as we shall see, required for the next step of the algorithm. Thus at each point, x_i, two different approximations for $\mathcal{V}_x$ exist which we denote as $\mathcal{V}_x^+$ and $\mathcal{V}_x^-$ based on whether x_{i+1} or x_{i-1} are chosen,

respectively. In fact, if one chooses $\mathcal{M} = 1$, these are simply the classical first-order upwind methods.

What comes next is the core idea of the ENO scheme. At each remaining step, toward finding the interval, one will face two options in choosing the next point. The idea is to choose the point that will result in the smoothest polynomial. In fact, it is easy to note that the divided differences coefficients are a good measure of the variations in the interpolating function. A coefficient with a large magnitude alerts the existence of a rapid change, or even a discontinuity in function $\mathcal{V}$ around point x_i, and thus should be avoided in constructing the polynomial.

For example, let's consider the second-order correction to $\mathcal{V}_x^-$. The two possible options are x_{i-2} and x_{i+1}. There are also two second-order divided differences coefficients, f_{i-2}^i and f_{i-1}^{i+1}. As a result one either chooses x_{i-2} if

$$\left|f_{i-2}^i\right| \leq \left|f_{i-1}^{i+1}\right|,$$

or x_{i+1} otherwise. This same idea is repeated for all higher-order coefficients until all remaining points in the interval are found, after which the interpolation polynomial is uniquely determined. Note that in this algorithm, one first determines the interval and then constructs the polynomial. It is possible to combine the two in one single pass as suggested in [86].

Finally let us briefly mention that, since the ENO scheme always chooses the smoothest polynomial with minimal variations, it may dampen out even slightest gradients in the solution where no shock or discontinuity exist. To remedy this problem, [107] introduced the idea of the weighted ENO (WENO) schemes in which

one obtains a higher order numerical approximation to $\mathcal{V}_x$ by appropriate weighting of all possible $\mathcal{M}^{th}$-order ENO schemes. The weighting coefficients are usually chosen so that they inversely depend on the smoothness of corresponding ENO scheme in a potential interval. This idea then results in a WENO scheme that automatically switches to an ENO scheme in parts of the domain that solution is non-smooth while obtaining higher-order approximations in smooth part of the domain. We do not present the algorithm in detail here and refer the interested reader to [86] for a discussion of the implementation and also the literature review on WENO schemes.

B. Numerical Hamiltonian

The second part in constructing a high-order algorithm for solving the HJ equation is the high-order construction of a *numerical Hamiltonian*, $\hat{\mathcal{H}}$. As noted in the previous section, at each point there are two approximations to the gradient, denoted by $\mathcal{V}_x^+$ and $\mathcal{V}_x^-$, depending on the initial stencil bias. As such, in general, the numerical Hamiltonian may generally be written as,

$$\hat{\mathcal{H}} = \hat{\mathcal{H}}\left(\mathcal{V}_x^+, \mathcal{V}_x^-\right).$$

To get a numerical Hamiltonian that correctly accounts for the nonlinear shock and rarefaction phenomena (see [92]), three criteria must be satisfied. First, the Hamiltonian needs to be Lipschitz continuous in both $\mathcal{V}_x^+$ and $\mathcal{V}_x^-$. Second, the numerical Hamiltonian needs to be a non-increasing function of $\mathcal{V}_x^+$ and a non-

decreasing function of $\mathcal{V}_x^-$. Symbolically, this is usually denoted as $\hat{\mathcal{H}}(\downarrow,\uparrow)$. Third, the numerical Hamiltonian needs to be consistent with the analytical Hamiltonian, i.e., $\hat{\mathcal{H}}(\mathcal{V}_x,\mathcal{V}_x)=\mathcal{H}(\mathcal{V}_x)$.

Usually, what makes one numerical Hamiltonian better than the other is the degree of numerical dissipation it adds to the problem. Different constructs have been proposed over the years that vary, not only in the degree of dissipation they add, but also how hard they are to implement. One of the easiest ones, though more dissipative, is the Lax-Friedrichs Hamiltonian. In one spatial dimension, this is written as,

$$\hat{\mathcal{H}}(\mathcal{V}_x^+,\mathcal{V}_x^-)=\mathcal{H}\left(\frac{\mathcal{V}_x^+ + \mathcal{V}_x^-}{2}\right)-\frac{1}{2}\alpha_x\left(\mathcal{V}_x^+ - \mathcal{V}_x^-\right), \tag{E.2}$$

where $\alpha_x = \max|\partial\mathcal{H}/\partial\mathcal{V}_x|$. If the maximum is computed globally through the whole computational domain, the method is usually termed global Lax-Friedrichs (LF). This, however, is usually unnecessarily too restrictive and the maximum, at any point, is usually computed locally for adjacent grid points and the method is termed local Lax-Friedrichs (LLF).

Although both LF and LLF Hamiltonians are very easy to implement, they usually over-dampen the solution and distort sharp gradients. Better results may be obtained via the Godunov's Hamiltonian. The Godunov's Hamiltonian, unfortunately, is usually hard to obtain for complicated Hamiltonians and may be quite computationally expensive. We do not go into the details of finding these, and other, numerical Hamiltonians and refer the interested reader to [86] and [92] for more details. Finally we note that to avoid the computational cost of evaluating

the Godunov's Hamiltonian, and to obtain solutions that are not overly damped when using the LLF Hamiltonian, one can generally use a high-order ENO scheme along with a high order TVD Runge-Kutta (see below) scheme and use a sufficiently refined grid.

C. Total Variation Diminishing (TVD) Runge-Kutta schemes

A TVD Runge-Kutta (TVD-RK) method is merely a Runge-Kutta method that is ensured to decrease the total variation (TV) in the solution as integrated in time. By definition, total variation of a differentiable function, $f(x)$, is defined as,

$$TV(f) = \int |f_x| \, dx,$$

while for a discrete function, u_j, this definition changes to

$$TV(u) = \sum_j |u_{j+1} - u_j| \, .$$

In both cases, TV of a function simply is a measure of the amount of variation in the function. To have a convergent solution, one requires a time integration method that decreases the total variation in the function since otherwise it may lead to non-physical oscillatory results. A TVD method is then any time integration scheme that satisfies the following condition:

$$TV(u^{n+1}) \leq TV(u^n).$$

Many different TVD methods exist in the literature, from both the Runge-Kutta [106] and the linear multi-step families [108]. Without going into much detail, here we present the third-order TVD-RK method and leave other TVD methods to the references mentioned above.

Just like normal Runge-Kutta methods, the third-order TVD-RK method consists of three consecutive forward Euler parts. For the semi-discrete HJ equation, i.e., already discretized in the space variable, x, using the ENO scheme,

$$\frac{\partial \mathcal{V}}{\partial t} = -\hat{\mathcal{H}}(\mathcal{V}_x^+, \mathcal{V}_x^-),$$

the third-order TVD-RK method is written as,

$$\begin{aligned} \mathcal{V}^{n+1/3} &= \mathcal{V}^n - \Delta t\, \hat{\mathcal{H}}^n, \\ \mathcal{V}^{n+2/3} &= \frac{3}{4}\mathcal{V}^n + \frac{1}{4}\mathcal{V}^{n+1/3} - \frac{1}{4}\Delta t\, \hat{\mathcal{H}}^{n+1/3}, \\ \mathcal{V}^{n+1} &= \frac{1}{3}\mathcal{V}^n + \frac{2}{3}\mathcal{V}^{n+2/3} - \frac{2}{3}\Delta t\, \hat{\mathcal{H}}^{n+2/3}. \end{aligned} \tag{E.3}$$

Note that to ensure stability, and the TVD property, it is required to impose a restriction on the time step according to,

$$\frac{\Delta t}{h_{min}}\alpha_{max} \leq c,$$

where $\alpha_{max} = \max|\partial\mathcal{H}/\partial\mathcal{V}_x|$ and $h_{min} = \min|x_{i+1} - x_i|$. These can also be local evaluations (i.e., for each grid point, x_i, within the stencil used to compute $\mathcal{V}_x^{\pm}$). Here, $0 < c \leq 1$ is called the *CFL* number [86, 92].