UNIVERSITY OF CALIFORNIA
Santa Barbara

Tracking Tethered Kite via Kalman Filter and Sensor Fusion

A Thesis submitted in partial satisfaction of the
requirements for the degree Master of Science
in Electrical and Computer Engineering

by
Khanh N Huynh

Committee in charge:
Professor Bassam Bamieh, co-Chair
Professor Andrew Teel, co-Chair
Professor João Hespanha

June 2013

The thesis of Khanh N Huynh is approved.

Professor Bassam Bamieh, co-Chair

Professor Andrew Teel, co-Chair

Professor João Hespanha

May 2013

Acknowledgements

This master thesis was part of a post-doc project by Dr. Lorenzo Fagiano and under the supervision of Professor Bassam Bamieh of the Department of Mechanical Engineering at UCSB. Dr. Lorenzo Fagiano is with the Automatic Control Laboratory, ETH Zurich, Switzerland,
fagiano@control.ee.ethz.ch. The main research project of Dr. Fagiano received funding from the California Energy Commission under the EISG grant n. 56983A/10-15 under the name of "Autonomous flexible wings for high-altitude wind energy generation, and also was funded by the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement n. PIOF-GA-2009-252284 - Marie Curie project Innovative Control, Identification and Estimation Methodologies for Sustainable Energy Technologies. I was part of this project from January 2012 to November 2012 and the material in this thesis summarizes everything that I was involved in.

Abstract

Tracking Tethered Kite via Kalman Filter and Sensor Fusion

by

Khanh N Huynh

One of the keys to reducing the world dependence on fossil fuels and the emissions of greenhouse gases is the use of a suitable combination of alternative and green energy sources. In the area of wind energy, recent studies showed that there is enough potential in the total world wind power to sustain the global needs. A new class of generators denoted as HAWP (High Altitude Wind Power) has emerged recently, which exploits the aerodynamic forces generated by the flight of tethered kite to produce electric energy. The key to capture wind energy through flying tethered kite is the ability to track and control the kite to fly in a "figure-eight" pattern. Thus, an accurate reconstruction of the kite position, velocity and heading is of fundamental importance for the automatic control of the kite in the "figure-eight" pattern.

This thesis presents an approach to track a tethered kite through the Kalman

filter and fusing multiple available sensors. Two different sensor setups and the related sensor fusion algorithms are considered and implemented in real-time. The first sensor setup imply the use of a commercial Inertial Measurement Unit (IMU), installed on the kite and equipped with accelerometers, gyroscopes, magnetometers, GPS, and a barometer. The first approach uses a standard linear observer, while the second one introduces a nonlinear correction to account for the specific kinematic constraints of the considered application. In the second sensor setup, we consider the use of a line angle measurement system in addition to the accelerometers, gyroscopes and magnetometers. Lastly, the results of experimental tests carried out with a small-scale prototype capable of flying a kite at a low height is presented and analyzed.

Contents

List of Figures

List of Tables

1. Introduction

In recent year, the development of an innovative technology of high-altitude wind energy generation and the investigation of the related advanced automatic control techniques has received increasing attention in the last decade, from both industry and academia. This is due to the ongoing global energy crisis. The world energy consumption is continuously growing mainly due to the development countries and there is an expected increase of about 50-55% in energy consumption by the year 2035, see [(EIA), 2011].

1.1. Global Energy Situation

As Figure 1.1 (taken from [(EIA), 2011] indicates, most of the world energy consumption comes from non-OECD contries, i.e. from Figure 1.2 some of the OECE (Organization for Economic Cooperation and Development) countries are Canada, United States, France, Spain, Turkey, etc. According to [(IEA), 2008], most of the global energy needs are actually covered by fossil sources (i.e. oil, coal and natural gas), accounting for about 81% of the global primary energy demand in 2006.

Fossil sources are supplied by few producer countries, which own limited

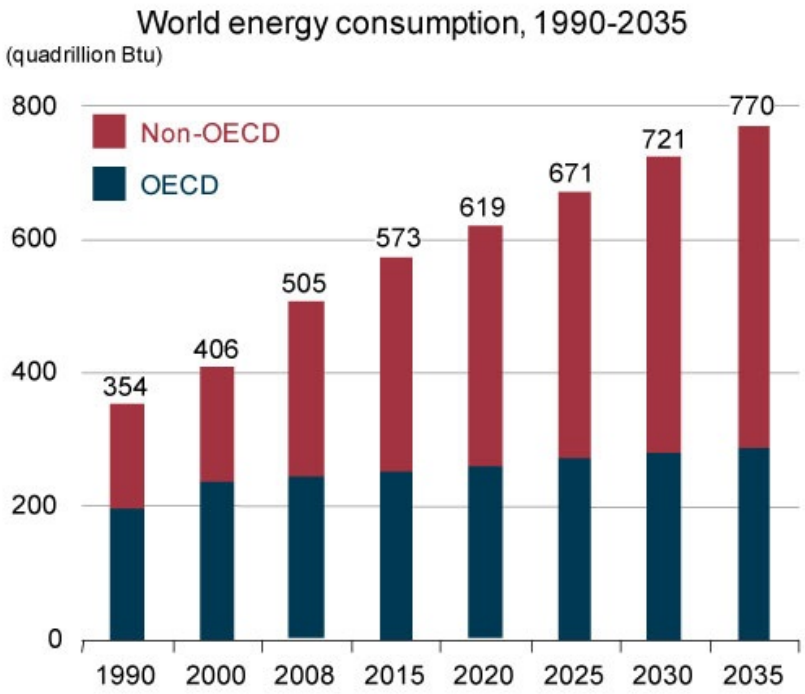


Figure 1.1.: Forecast World Energy Consumption

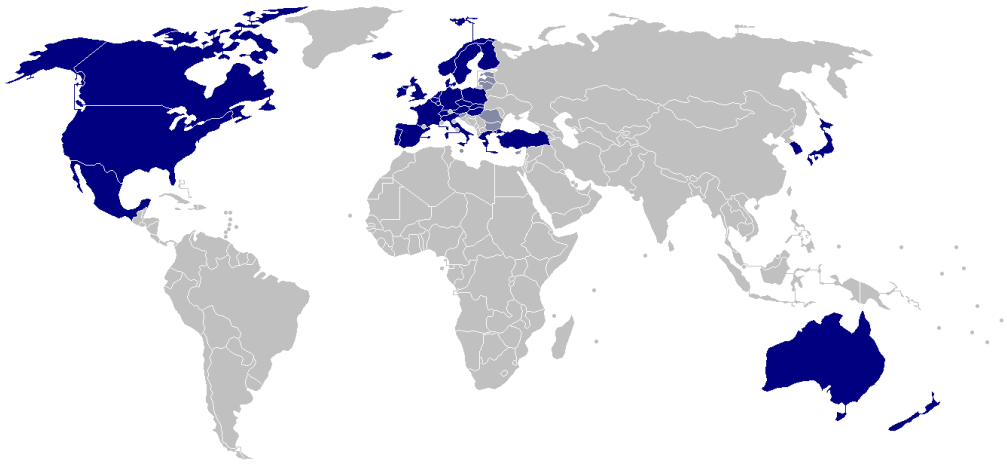


Figure 1.2.: OECD Countries

reservoirs, and the average cost of energy obtained from such sources is continuously increasing due to the increasing demand and decreasing supply, related to the rapid economy growth of the highly populated non-OECD countries. Moreover, the negative effects of energy generation from fossil

sources on global warming and climate change, due to excessive carbon dioxide emissions, and the negative impact of fossil energy on the environment are recognized worldwide and lead to additional indirect costs, see [Gurria, 2008].

Such a situation gives rise to serious geopolitical and economical problems, affecting almost all of the worlds countries. One of the key points to solve these issues is the use of a suitable combination of alternative and renewable energy sources. However, the actual renewable technologies (hydro-power, solar, wind, biomass, geothermal) seem to have limited potential to replace fossil sources in the near future. Indeed, according to the projection given by [(IEA), 2008], if no political and economical measures will be adopted, the renewable share of total energy use would only increases from 10 percent in 2008 to 15 percent in 2035. Fossil sources would continue to supply much of the energy used worldwide and would still account for 78% of world energy use in 2035.

Fortunately, for the foreseeable future, renewable energy sources remain as the fastest growing source of new electricity generation worldwide. As electricity is the world's faster-growing form of end-use energy consumption, renewable energy would certainly grow to be a more important factor in worldwide energy consumption. Moreover, it is expected that about 7.5% of the whole energy consumption by non-OECD countries will be supplied by renewable energies for the next 20 years.

1.2. Wind Turbine Limitations

Excluding hydro-power (which is not likely to increase substantially in the future, because most major sites are already being exploited or are unavailable for technological and/or environmental reasons), the main issues that hamper the growth of renewable energies are the high investment costs of the related technologies, their non-uniform availability and the low generated power density per unit area. Development of modern wind technology started in the late 1970s and dramatic improvements have been obtained since that time. The present commercial wind turbines have three-bladed rotors with diameters up to 90-100 m, installed atop towers with 60-100 m of height, see Figure 1.3.

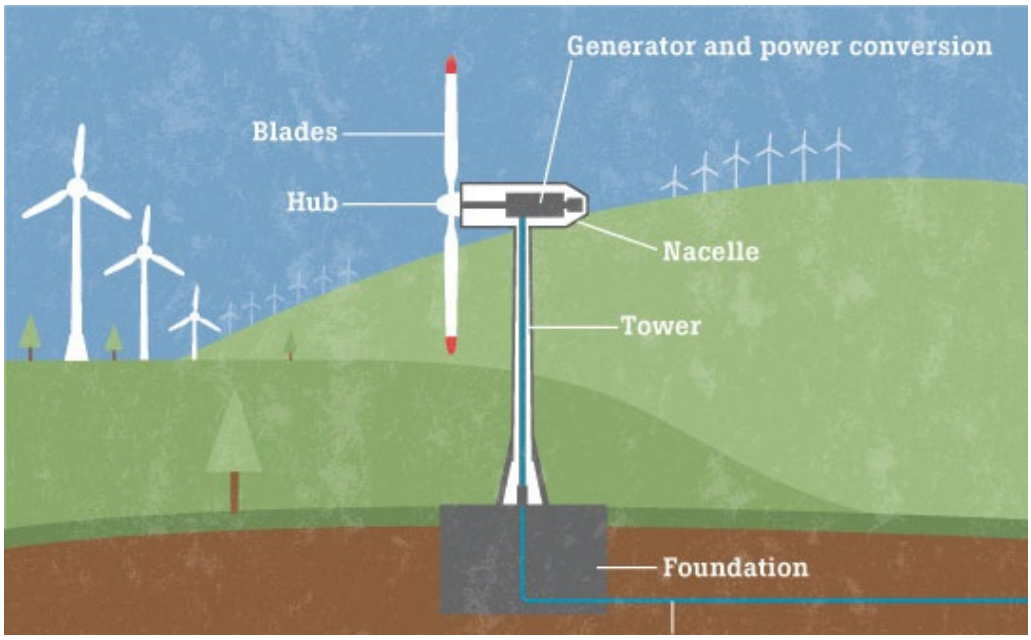


Figure 1.3.: Wind Turbine

The amount of energy in the wind available for extraction by the turbine increases with the cube of wind speed, however such increase is exploited

only to some extent, since the operation of a turbine is suitably controlled in order to not exceed the power level for which the electrical system has been designed (referred to as the "rated power"). The turbine power output is controlled by rotating the blades about their long axis to change the angle of attack with respect to the relative wind as the blades spin about the rotor hub.

Moreover, the turbine is pointed into the wind by a control system that rotates the nacelle (see Figure 1.3) about the tower, on the basis of measurements of the wind speed and direction. Almost all modern turbines operate with the rotor positioned on the windward side of the tower. Typically, a turbine starts producing power with about 3.5 m/s wind speed and reaches the rated power output at about 15 m/s. If the wind speed exceeds the cutout value (i.e. about 25 m/s), the blades are pitched to stop power production and rotation, in order to avoid possible breaking due to the excessive forces.

It is important to note that the wind energy potential is a function of the height above the ground due to the presence of the so-called "wind shear", i.e. the growth of wind speed with elevation, see [Archer and Jacobson, 2005]. The height and the size of wind turbines have increased in the past years to capture the more energetic winds at higher elevations. However, the actual limits of such a dimension growth have been almost reached, from both economical and technological points of view. In fact, in general the costs of larger turbines grow linearly with the volume of the employed material (i.e. with the cube of the diameter), while the related increase of energy output

is proportional to the rotor-swept area (the diameter squared). Therefore, at some size the cost for a larger turbine will grow faster than the resulting energy output revenue, making scaling not economically profitable. In practice, studies have shown that in recent years blade mass has been scaling at roughly an exponent of 2.3 versus the expected 3, thus delaying the achievement of dimension limit from the economical point of view.

However, it has to be also considered that much higher operation, mobilization, and demobilization costs incur to build bigger turbines. Moreover, serious constraints to size growth have been reached, related to land transportation and turbine construction. Transportation of bigger turbine parts is not cost effective and crane requirements are quite stringent because of the large nacelle mass in combination with the height of the lift and the required boom extension. For all these reasons, it is not expected that land-based turbines will become much larger than about 100 m in diameter, with corresponding power outputs of about 35 MW, see [R. Thresher and Veers, 2007].

Other important aspects of wind energy technology are the generation efficiency and the average yearly generated power. As regards efficiency, according to Betz limit, see [T. Burton and Bossanyi, 2001], a device can extract a theoretically maximum 59% of the energy in a stream with the same area as the working area of the device. The aerodynamic performance of a modern wind turbine has improved dramatically over the past 20 years and the rotor system can be expected to capture about 80% of such a theoretical upper bound. However, actually the turbine overall efficiency is such that about 40-50% of Betz limit is achieved. Furthermore, due to the

wind inconsistent flow near ground level, any wind generator cannot produce continuously its rated power. Thus the average power generated over the year is only a fraction, indicated as "Capacity Factor" (CF), of the rated one. For a given wind generator on a specific site, the CF can be evaluated knowing the generator power curve and the probability density distribution function of wind speed that flows in through the area spanned by the blades.

The issue of wind energy density per unit area of occupied land is also of paramount importance. In order to generate a noticeable amount of energy, wind turbines can be arranged in the so-called "wind farms", i.e. tens or hundreds of turbines built in the same location. The usual rule to deploy wind turbines of a given Diameter D in a wind farm is to keep a distance of $7D$ in the wind prevalent direction and $4D$ in the transverse direction. This way, considering 90m diameter, 2 MW rated power turbines, a density of about 4.4 turbines per km^2 is obtained, with a corresponding rated power density of 8.8 MW/km^2. Considering a good windy site (i.e. CF=0.4), a consequent average power density of 3.52 MW/km^2 is achieved. Thus, in order to generate an average power of 1000 MW (i.e. the power supplied by medium-to-large thermal plants), a land occupation of about 280 km^2 would be required, where more than 1200 turbines should be deployed. Such power density values are much lower than those given by thermal plants.

As regards future improvements of the present wind technology, studies on advanced rotors and drive trains and innovative towers are undergoing to try to push forward the actual technical limitations. Moreover, offshore wind turbines are being deployed at water depths of up to 30 m, see Figure

1.4, and research activities are undergoing to develop deep-water technologies (i.e. wind turbines placed in the sea with 60-90 m of depth). However, it is clear that no single component improvement in cost or efficiency can achieve significant cost reductions or dramatically improved performance in the present wind technology and it is estimated that all the projected advancements can cumulatively bring no more than 30-40% improvement in the cost effectiveness of wind energy over the next decade.

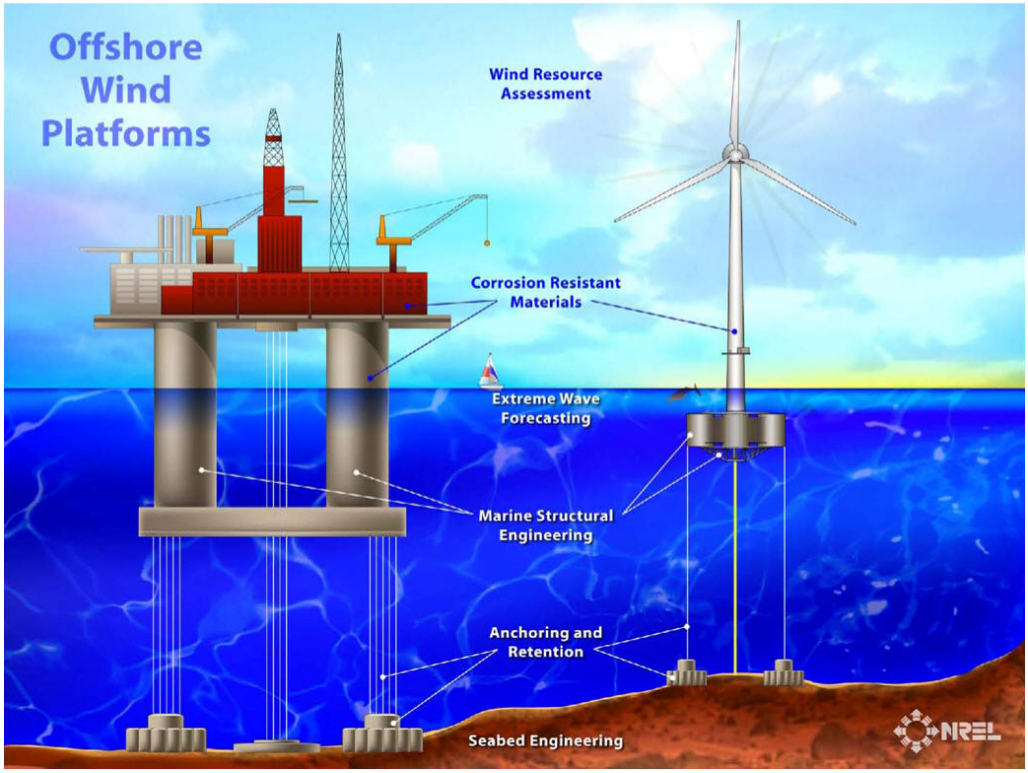


Figure 1.4.: Offshore Wind Turbine

Overall, the typical wind towers require heavy foundations and huge blades, with massive investments, and have a limited operating height of about 150 meters from the ground, where wind flows are weaker and inconsistent. [R. Thresher and Veers, 2007] gives comprehensive overview of the present

wind technology, where it is also pointed out that no dramatic improvement is expected in this field. All the issues mention by [R. Thresher and Veers, 2007] lead to wind energy production costs that are higher than those of fossil sources. Therefore, a quantum leap would be needed in wind technology to overcome the present limits and boost its application in providing green energy with competitive costs.

2. High Altitude Wind Power (HAWP)

2.1. Concept of HAWP

High-altitude wind is a vast, still untapped renewable source of energy that has received an increasing attention in the last decade, from both the arena of industry and academia. Many concepts of airborne wind energy generators were already present in patents and publications as early as the late 1970s, see [Manalis, 1976], [Fletcher and Roberts, 1979], and [Loyd, 1980]. In the field of wind energy, some recent studies such as
[Archer and Jacobson, 2005] showed that by exploiting 20% of the global land sites of "class 3" or more (i.e. with average wind speed greater than 6.9 m/s at 80 m above the ground), the entire worlds energy demand could be supplied.

However, such potential cannot be harvested with competitive costs by the current wind technology based on wind towers as we presented in Section 1.2. Such a breakthrough in wind energy generation can be realized by capturing high-altitude wind power. A possible viable approach is to use airfoils

(like power kite used for surfing or sailing), linked to the ground with one or more cables, see Figure 2.1. This concept is employed to control the kite flight and to convert the aerodynamic forces into mechanical and electrical power, using suitable rotating mechanisms and electric generators kept at the ground level. Such kites are able to exploit wind flows at higher altitudes (up to 1000 m) than those of wind turbines. At such elevations, stronger and more constant wind can be found basically everywhere in the world. Thus, this technology can be used in a much larger number of locations as oppose to the wind turbine. The potential of this concept has been theoretically investigated almost 30 years ago by [Loyd, 1980], where the study shows that if the kite is driven to fly in "crosswind" conditions, the resulting aerodynamic forces can generate surprisingly high power values. However, only in recent years more intensive studies have been carried out by quite few research groups in the world, to deeply investigate this idea from the theoretical, technological and experimental point of views.

The concept of high-altitude power generation are being studied and developed with similar ideas of exploiting high-altitude wind flows. Some research groups such as the Katholieke Universiteit of Leuven (Belgium) [A. Ilzhfer and Diehl, 2007], and the Technical University of Delft (The Netherlands) [P. Williams and Ockels, 2008], are studying and developing a concept of controlled tethered kite to extract energy from high-altitude wind flows. It is only in recent years that an increasing number of research groups and companies started to develop operating prototypes to convert the energy of high-altitude wind, blowing up to 1000 m from the ground, into electricity, see e.g. [Makani, 2013], [SkySails, 2013], [Ampyx, 2013],

[Windlift, 2010] and [Kitenergy, 2009]. The activities carried out worldwide in the last few years allowed researchers to define and assess, through theoretical, numerical and experimental research, some common grounds of airborne wind energy, see [Fagiano and Milanese, 2012] for an overview, as well as its potentials to provide cheap electricity in large quantities, available practically everywhere in the world.

2.2. Generate Wind Energy via Tethered Kite

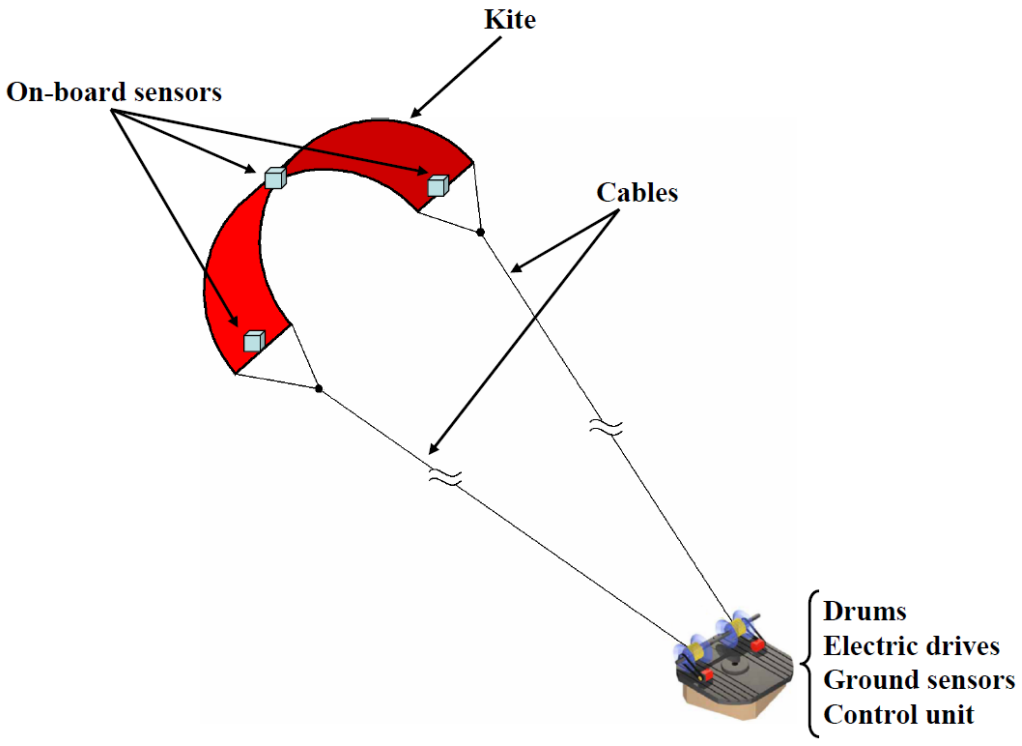


Figure 2.1.: Kite Steering Unit (KSU)

Let us imagine the removal of all the bulky structure of a wind tower and just keep the outer part of the blades, which becomes a much lighter kite flying fast in crosswind conditions while connected to the ground by only

two cables, see Figure 2.1. Thus, the rotor and the tower of the present wind technology are replaced in high-altitude wind technology by the kite and its cables, realizing a wind generator which is largely, lighter and cheaper. High efficiency, maneuverability, resistance to strain and lightness are the main characteristics that a kite should have to be employed for high-altitude wind energy production.

The kite lift force is converted into mechanical power through the traction forces acting on the lines, see Figure 2.1. Thus, the cables have to be strong enough to support high loads. At the same time, the cables have to be light and their diameter should be kept as small as possible, to limit their weight and aerodynamic drag. Lines realized in composite materials, with a traction resistance 8-10 times higher than that of steel cables of the same weight would be appropriate for this application. In order to extract energy from wind flows between 200-1000 m of elevation, about 500-1500 m long lines would be needed.

At ground level, the kite cables are rolled around two drums, linked to two electric drives which are able to act either as generators or as motors. The kite flight is tracked using onboard wireless instrumentation (e.g. GPS, magnetic and inertial sensors) as well as ground sensors, to measure the kite speed and position, the power output, the cable force and speed and the wind speed and direction. Such variables are employed for feedback by an electronic control system, able to influence the kite flight path by differentially pulling the cables, via a suitable control of the electric drives. The system composed by the electric drives, the drums, the onboard sensors and

all the hardware needed to control a single kite is denoted as Kite Steering Unit (KSU) and it is the very core of the high-altitude wind technology. The KSU can be employed in different ways to generate energy, depending on how the traction forces acting on the cables are converted into mechanical and electrical power. In particular, two different configurations have been investigated so far, namely the HE-yoyo and the HE-carousel configurations.

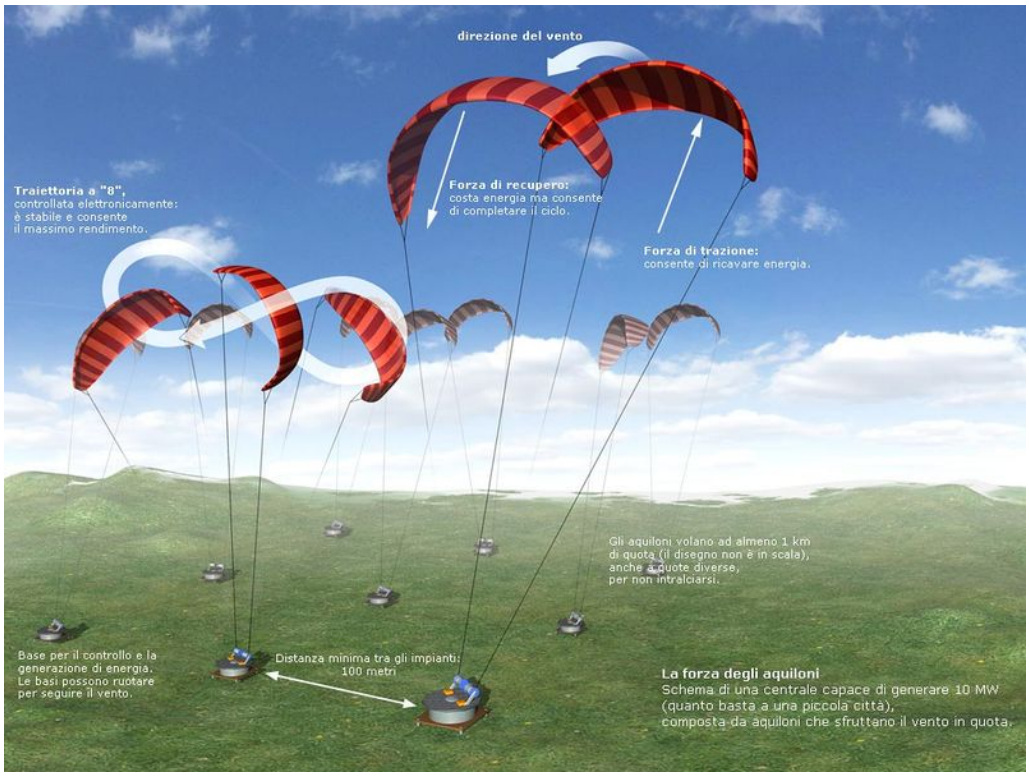


Figure 2.2.: HE-yoyo cycle

In the HE-yoyo configuration (where this project is concentrated on), the KSU is fixed on the ground and wind power is captured by unrolling the kite lines; while in the HE-carousel configuration the KSU is put on a vehicle dragged by the line forces along a circular rail path, thus generating energy by means of additional electric generators linked to the wheels. Indeed,

the described high-altitude wind energy generators are complex, open-loop unstable systems, affected by external disturbances (e.g. wind turbulence), with nonlinear dynamics and operational constraints.

The energy in the HE-yoyo configuration is obtained by continuously performing a two-phase cycle: the traction phase and the passive phase, see Figure 2.2. In the traction phase the kite exploits wind power to unroll the lines and the electric drives act as generators, driven by the rotation of the drums. In order to obtain the optimal trajectory in the air, the kite will be flying in a loop or a "figure-eight" pattern in order to generate the maximum pulling force. When the maximum line length is reached, the passive phase begins and the drives act as motors, spending a minimum amount of the previously generated energy, to recover the kite and to drive it in a position which is suitable to start another traction phase, i.e. when the kite is flying with wind advantage in a symmetric zone with respect to the nominal wind direction.

Two different controllers can be designed to control the kite in the traction and passive phases. For the whole cycle to be generative, the total amount of energy produced in the traction phase has to be greater than the energy spent in the passive one. Therefore, the controller employed in the traction phase must maximize the produced energy, while in the passive phase the objective is to maneuver the kite in a suitable position and to minimize, at the same time, the spent energy. The potential of the HE-yoyo configuration has also been investigated in [Houska and Diehl, 2007] for the cases of one and two kites linked to a single cable: optimal kite periodic loops, which

maximize the generated energy, have been computed considering as inputs, the derivatives of the kite roll angle, lift coefficient and cable winding speed.

Control engineering plays a crucial role in airborne wind energy technologies, since, differently from conventional wind energy based on wind turbines, there is no passive and rigid structure that constrains the motion of the kite in the air and imposes its crosswind motion. Rather, this task has to be accomplished by an active control system that keeps track of the kite motion and issues suitable control inputs through actuators. In the last six years, the problem of control design for airborne wind energy generators has been studied by several research groups and companies, leading to a quite significant series of theoretical and numerical studies, see e.g. [A. Ilzhfer and Diehl, 2007], [M. Canale and Milanese, 2010], [P. Williams and Ockels, 2008], and [Fagiano et al., 2011].

All of the mentioned control approaches rely on the availability of a series of variables to be used for feedback, most notably the kite three-dimensional position and velocity. The problem of estimating with sufficiently good accuracy and limiting the lag of these quantities is therefore of paramount importance in the field. However in the literature there are only few works on this topic, e.g. [Erhard and Strauch, 2013] and most of the literature works highlight the specific issues that need to be addressed and provide either numerical or experimental results.

In this project, we contribute to fill the gap by providing a formulation of the estimation problem and by analyzing two different sensor setups and

the related sensor fusion algorithms. The first sensor setup imply the use of a commercial Inertial Measurement Unit (IMU), installed on the kite and equipped with accelerometers, gyroscopes, magnetometers, GPS, and a barometer. The first approach uses a standard linear observer, while the second one introduces a nonlinear correction to account for the specific kinematic constraints of the considered application. In the second sensor setup, we consider the use of a line angle measurement system in addition to the accelerometers, gyroscopes and magnetometers. This thesis is focused on the estimation of the kite position and of its "velocity angle", which is a variable well suited for automatic control strategies, when the wing is flying fast in crosswind conditions (see [L. Fagiano and Khammash, 2013] for more details on the automatic control design, which was another part of the main project by Dr. Lorenzo Fagiano).

3. Probability

3.1. Probability, Random Processes and Stochastic Systems

The theory of random processes and stochastic systems represents the evolution over time of the uncertainty of our knowledge about physical systems. This representation includes the effects of any measurements (or observations) that we make of the physical process and the effects of uncertainties about the measurement processes and dynamic processes involved. The uncertainties in the measurement and dynamic processes are modeled by random processes and stochastic systems.

Properties of uncertain dynamic systems are characterized by statistical parameters such as means, correlations, and covariances. By using these numerical parameters, one can obtain a finite representation of the problem, which is important for implementing the solution. This representation depends upon statistical properties such as orthogonality, stationarity, ergodicity, and Markovianness of the random processes involved and the probability distributions.

Most of us have some notion of what is meant by a "random" occurrence, or the probability that some event in a sample space will occur. Formally, the probability that the outcome of a discrete event (e.g., a coin flip) will favor a particular event is defined as:

$$p(A) = \frac{\text{Possible outcome favoring event A}}{\text{Total number of possible outcomes}}. \tag{3.1}$$

If two events are disjoint or mutually exclusive, the probability of an outcome favoring either event A or event B is given by:

$$p(A \cup B) = p(A) + p(B). \tag{3.2}$$

If the probability of two outcomes is independent (one does not affect the other) then the probability of both occurring is the product of their individual probabilities:

$$p(A \cap B) = p(A)p(B). \tag{3.3}$$

For example, if the probability of seeing a heads on a coin flip is $1/2$, then the probability of seeing "heads" on both of two coins flipped at the same time is $1/4$. Clearly, the outcome of one coin flip does not affect the other.

In addition, the probability of outcome A given an occurrence of outcome B is called the conditional probability of A given B:

$$p(A|B) = \frac{p(A \cap B)}{p(B)}. \tag{3.4}$$

Finally, from Bayes' rule, the relationship between the probabilities of A and B and their conditional probabilities is given as:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}. \tag{3.5}$$

3.2. Random Variables

A random variable is essentially a function that maps all points in the sample space to real numbers. A continuous random variable is a random variable where the data can take infinitely many values. In the case of continuos random variables, the probability of any single discrete event A is in fact zero. That is $p(A) = 0$. Instead we can only evaluate the probability of events within some interval. A common function representing the probability of random variables is defined as the "cumulative distribution function":

$$F_X(x) = p(-\infty, x]. \tag{3.6}$$

This function represents the cumulative probability of the continuous random variable X for all (uncountable) events up to and including x. Some important properties of the cumulative density function are:

1. $F_X(x) \to 0$ as x $\to -\infty$.
2. $F_X(x) \to 1$ as x $\to +\infty$.
3. $F_X(x)$ is a non-decreasing function of x.

The derivative of the cumulative distribution function is more popularly known as the "probability density function" :

$$f_X(x) = \frac{d}{dx} F_X(x). \tag{3.7}$$

With the above given properties of the cumulative probability function, the probability density function $f_X(x)$ also has the following properties:

1. $f_X(x)$ is a non-negative function.
2. $\int_{-\infty}^{\infty} f_X(x)dx = 1$.

Finally note that the probability over any interval $[a, b]$ is defined as:

$$p_X[a, b] = \int_a^b f_X(x)dx. \tag{3.8}$$

So rather than summing the probabilities of discrete events as in Equation (3.2), for continuous random variables one can integrate the probability density function over the interval of interest.

3.3. Mean and Variance

The average of a sequence of numbers, or the sample mean is give by:

$$\bar{X} = \frac{X_1 + X_2 + \cdots + X_N}{N}. \tag{3.9}$$

Because in tracking we are dealing with continuous signals it is useful to think in terms of the outcome we would expect to see if we sampled the random variable infinitely, each time seeing one of n possible outcomes $x_1 \ldots x_n$. In this case, the expected value of the discrete random variable could be approximated by averaging probability-weighted events:

$$\bar{X} = \frac{(p_1 N)x_1 + (p_2 N)x_2 + \cdots + (p_n N)x_N}{N}, \tag{3.10}$$

where out of N trials, we would expect to see $(p_1 N)$ occurrences of event x_1. This notion of infinite trials (samples) leads to the conventional definition of expected value for discrete random variables:

$$\text{Expected value of } X = E(X) = \sum_{i=1}^{n} p_i x_i, \tag{3.11}$$

for n possible outcomes $x_1 \ldots x_n$ and corresponding probabilities $p_1 \ldots p_n$. Similarly for the continuous random variable the expected value is defined as:

$$\text{Expected value of } X = E(X) = \int_{-\infty}^{\infty} x f_X(x) dx, \tag{3.12}$$

where $f_X(x)$ is the probability density function introduced in Section 3.2. Finally, note that Equation (3.11 - 3.12) can be applied to function of the random variable X as follow:

$$E(g(X)) = \sum_{i=1}^{n} p_i g(x_i), \tag{3.13}$$

and

$$E(g(X)) = \int_{-\infty}^{\infty} g(x) f_X(x) dx. \tag{3.14}$$

The expected value of a random variable is also known as the first statistical moment. We can apply the notion of equation (3.13) or (3.14) by letting $g(X) = X^k$, to obtain the k^{th} statistical moment. The k^{th} statistical moment of a continuous random variable X is given by:

$$E(X^k) = \int_{-\infty}^{\infty} x^k f_X(x) dx. \tag{3.15}$$

Of particular interest in general, and to us in particular, is the second moment of the random variable. The second moment is given by:

$$E(X^2) = \int_{-\infty}^{\infty} x^2 f_X(x) dx. \tag{3.16}$$

When we let $g(X) = X - E(X)$ and apply equation (3.16), we get the variance of the signal about the mean. In other words,

$$\begin{aligned} \text{Variance of } X &= E[(X - E(X))^2] \\ &= E(X^2) - E(X)^2. \end{aligned} \tag{3.17}$$

Variance is a very useful statistical property for random signals, because if we knew the variance of a signal that was otherwise supposed to be "constant" around some value (i.e. the mean), then the magnitude of the variance would give us a sense how much jitter or "noise" is in the signal.

The square root of the variance, known as the standard deviation, is also a useful statistical unit of measure because while being always positive, it has (as opposed to the variance) the same units as the original signal. The standard deviation is given by:

$$\text{Standard deviation of } X = \sigma_X = \sqrt{\text{Variance of } X}. \tag{3.18}$$

3.4. Gaussian (Normal) Distribution and Conditional Probability

A special probability distribution known as the Normal or Gaussian distribution has historically been popular in modeling random systems for a variety of reasons. As it turns out, many random processes occurring in nature actually appear to be normally distributed, or very close. In fact, under some moderate conditions, it can be proved that a sum of random variables with any distribution tends toward a normal distribution. The theorem that formally states this property is called the central limit theorem, see [Maybeck, 1974]. Finally, the normal distribution has some nice properties that make it mathematically tractable and even attractive.

Given a random process $X \sim N(\mu, \sigma^2)$, i.e. a continuous random process X that is normally distributed with mean μ and varince σ^2 (standard deviation σ), the probability density function for X is given by:

$$f_X(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2}\right), \tag{3.19}$$

for $-\infty < x < \infty$. Any linear function of a normally distributed random process (variable) is also a normally distributed random process. In particular if $X \sim N(\mu, \sigma^2)$ and $Y = aX + b$, then

$$Y \sim N(a\mu + b, a^2\sigma^2). \tag{3.20}$$

The probability density function for Y is then given by:

$$f_Y(y) = \frac{1}{\sqrt{2\pi a^2\sigma^2}} \exp\left(-\frac{1}{2}\frac{(y-(a\mu+b))^2}{a^2\sigma^2}\right). \tag{3.21}$$

Finally, if X_1 and X_2 are independent, where $X_1 \sim N(\mu_1, \sigma_1^2)$ and $X_2 \sim N(\mu_2, \sigma_2^2)$, then:

$$X_1 + X_2 \sim N(\mu_1 + \mu_2, \sigma_1^2 + \sigma_2^2), \tag{3.22}$$

and the density function becomes:

$$f_X(x_1 + x_2) = \frac{1}{\sqrt{2\pi(\sigma_1^2 + \sigma_2^2)}} \exp\left(-\frac{1}{2}\frac{(x-(\mu_1+\mu_2))^2}{\sigma_1^2+\sigma_2^2}\right), \tag{3.23}$$

see [Kelly, 1994] for further explanation and proofs of the result density function. Graphically, the normal distribution is what is likely to be familiar as the "bell-shaped" curve.

Two continuous random variables X and Y are said to be statistically independent if their joint probability $f_{XY}(x, y)$ is equal to the product of their individual probabilities. In other words, they are considered independent if

$$f_{XY}(x, y) = f_X(x) f_Y(y). \tag{3.24}$$

In addition, Bayes' rule offers a way to specify the probability density of the random variable X given (in the presence of) random variable Y. Bayes' rule is given as:

$$f_{X|Y}(x) = \frac{f_{Y|X}(y) f_X(x)}{f_Y(y)}. \tag{3.25}$$

4. Quaternion

4.1. Quaternion

Euler's theorem states that two arbitrarily oriented bases with common origin can be made to coincide with one another by rotating one of them through a certain angle about an axis that is passing through the origin, see [Wittenburg, 1977]. The manner in which complex numbers were used to represent 2D rotations can also be adapted to using quaternions to represent 3D rotations, see [LaValle, 2006].

William Rowan Hamilton invented the quaternions in 1843, in his effort to construct hypercomplex numbers or higher dimensional generalizations of the complex numbers. Failing to construct a generalization in three dimensions in such a way that division would be possible, he considered a systems with four complex units and arrived at the quaternion. He realized that, just as multiplication by i is a rotation by 90° in the complex plane, each one of his complex units could also be associated with a rotation in space. Vectors were introduced by Hamilton for the first time as "pure quaternion" and vector calculus was first developed as part of this theory.

Let the quaternion be represented as:

$$q = q_1 + q_2 i + q_3 j + q_4 k, \tag{4.1}$$

where $(q_1, q_2, q_3, q_4) \in \mathbb{R}$. A quaternion can be considered as a four-dimensional vector. The symbols i, j and k are used to denote three "imaginary" components of the quaternion.

The magnitude of the quaternion can be defined as:

$$\|q\| = \sqrt{q_1^2 + q_2^2 + q_3^2 + q_4^2}. \tag{4.2}$$

The quaternion can be viewed as a scalar coupled to a 3D vector:

$$q = q_1 + \mathbf{v}, \tag{4.3}$$

where $\mathbf{v}$ is a 3D vector that represents the final three quaternion components, $\mathbf{v} = [q_2\, q_3\, q_4]$.

Let consider two quaternions:

$$q_a = q_{a1} + q_{a2} i + q_{a3} j + q_{a4} k = q_{a1} + \mathbf{v}_a, \tag{4.4}$$

$$q_b = q_{b1} + q_{b2} i + q_{b3} j + q_{b4} k = q_{b1} + \mathbf{v}_b. \tag{4.5}$$

The dot product between two quaternions is:

$$q_a \cdot q_b = q_{a1} q_{b1} + q_{a2} q_{b2} + q_{a3} q_{b3} + q_{a4} q_{b4} = q_{a1} q_{b1} + \mathbf{v}_a \cdot \mathbf{v}_b. \tag{4.6}$$

In addition, using the definition of the quaternion dot product and quaternion magnitude, we can see that:

$$\|q\|^2 = q \cdot q. \tag{4.7}$$

A quaternion is defined as a normalized quaternion if its magnitude is equal to 1. Therefore, the normalization of a quaternion is conducted by dividing each component with its magnitude. The normalized quaternion, q_N, is the unit quaternion of q given by the expression:

$$q_N = \frac{q}{\|q\|} = \frac{|q_1, q_2, q_3, q_4|}{\sqrt{q_1^2 + q_2^2 + q_3^2 + q_4^2}}. \tag{4.8}$$

Furthermore, the conjugate of a quaternion is defined as:

$$\bar{q} = q_1 - q_2 i - q_3 j - q_4 k. \tag{4.9}$$

The multiplication of two quaternions under the above rules leads to the definition:

$$q_a \circ q_b = q_c = q_{c1} + q_{c2} i + q_{c3} j + q_{c4} k, \tag{4.10}$$

where

$$\begin{aligned} q_{c1} &= q_{a1}q_{b1} - q_{a2}q_{b2} - q_{a3}q_{b3} - q_{a4}q_{b4}, \\ q_{c2} &= q_{a1}q_{b2} + q_{b1}q_{a2} + q_{a3}q_{b4} - q_{b3}q_{a4}, \\ q_{c3} &= q_{a1}q_{b3} + q_{b1}q_{a3} + q_{b2}q_{a4} - q_{a2}q_{b4}, \\ q_{c4} &= q_{a1}q_{b4} + q_{b1}q_{a4} + q_{a2}q_{b3} - q_{b2}q_{a3}. \end{aligned} \tag{4.11}$$

Using the above definitions, the quaternion multiplication can be express in

a more simpler form:

$$q_a \circ q_b = q_{a1}q_{b1} - \mathbf{v}_a \cdot \mathbf{v}_b + q_{a1}\mathbf{v}_b + q_{b1}\mathbf{v}_a + \mathbf{v}_a \times \mathbf{v}_b. \tag{4.12}$$

where $\times$ is the cross product two vectors. The next step is to describe a mapping from unit quaternion to a rotational matrix. If the quaternion q is normalized, it can be converted to a 3×3 rotational matrix, often called Directional Cosine Matrix (see [Klumpp, 1976]):

$$R(q) = \begin{pmatrix} 2(q_1^2 + q_2^2) - 1 & 2(q_2q_3 - q_1q_4) & 2(q_2q_4 + q_1q_3) \\ 2(q_2q_3 + q_1q_4) & 2(q_1^2 + q_3^2) - 1 & 2(q_3q_4 - q_1q_2) \\ 2(q_2q_4 - q_1q_3) & 2(q_3q_4 + q_1q_2) & 2(q_1^2 + q_4^2) - 1 \end{pmatrix}, \tag{4.13}$$

which can be verified as orthogonal and $\det(R(q)) = 1$. [Markley, 1994] states that a quaternion is not a unique representation of the attitude and can be shown that q and $\bar{q}$ represent the same attitude.

Note that a matrix M is said to be orthogonal if and only if:

$$MM^T = I, \tag{4.14}$$

and the inverse of a matrix in defined as:

$$MM^{-1} = I, \tag{4.15}$$

thus it follows that the transpose of an orthogonal matrix is identical to the inverse of that orthogonal matrix:

$$M^T = M^{-1}, \tag{4.16}$$

therefore, the inverse of the rotational matrix $R(h)$ is:

$$R(h)^{-1} = R(h)^T. \tag{4.17}$$

4.2. Quaternion Rotation Rates

A constant vector in the body frame, $\vec{x}_B$, is described in the inertial frame by:

$$\vec{x}_I = q\vec{x}_B\bar{q}, \tag{4.18}$$

where $\vec{x}_I$ is a vector in the inertial frame. By taking the time derivative of definition (4.18) introduced above:

$$\begin{aligned} \dot{x}_I &= \dot{q}x_B\bar{q} + qx_B\dot{\bar{q}} \\ &= (\dot{q}\bar{q})\vec{x}_I + \vec{x}_I(q\dot{\bar{q}}), \end{aligned} \tag{4.19}$$

and since $(\overline{\dot{q}\bar{q}}) = -(\dot{q}\bar{q})$ and $\bar{\vec{x}}_I = -\vec{x}_I$, we have:

$$\begin{aligned} \dot{x}_I &= (\dot{q}\bar{q})\vec{x}_I - \vec{x}_I(\dot{q}\bar{q}) \\ &= 2(\dot{q}\bar{q}) \times \vec{x}_I, \end{aligned} \tag{4.20}$$

which can be related to the angular velocites, ω by:

$$\omega = 2\dot{q}\bar{q}. \tag{4.21}$$

This gives the equation that relate the angular velocities ω to the quaternion:

$$
\begin{aligned}
\dot{q} &= \tfrac{1}{2}\omega q \\
&= \tfrac{1}{2}\begin{bmatrix} 0 & -\omega_1 & -\omega_2 & -\omega_3 \\ \omega_1 & 0 & -\omega_3 & \omega_2 \\ \omega_2 & \omega_3 & 0 & -\omega_1 \\ \omega_3 & -\omega_2 & \omega_1 & 0 \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_3 \end{bmatrix}.
\end{aligned} \tag{4.22}
$$

By using the quaternion to describe spatial rotation, the representation of rotation matrix become more compact and quicker to compute. In addition, unlike Euler angles they are not susceptible to "gimbal lock", where one degree of freedom is lost when two axes are driven into a parallel configuration. Furthermore, [Lefferts and Shuster, 1982] shows that the quaternion prediction equation is linear and quaternion representation is free from singularities. That is why we decided to use the quaternion over the Euler angles in this project.

5. State-Space, Kalman Filter & Luenberger Observer

The Kalman filter was introduced in 1960 [Kalman, 1960] by Rudolph E. Kalman, who published his famous paper describing a recursive solution to the discrete-data linear filtering problem. The Kalman filter is essentially a set of mathematical equations that implement a predictor-corrector type estimator that is optimal in the sense that it minimizes the estimated error covariance when some presumed conditions are met. Since the time of its introduction, the Kalman filter has been the subject of extensive research and application, see e.g. [Lefferts and Shuster, 1982], particularly in the area of autonomous or assisted navigation, which is why it will be used extensively for this project.

5.1. State-Space Models

A state-space model is essentially a notational convenience for estimation and control problems. It was developed to make tractable what would otherwise be a notationally-intractable analysis. In order that the future state of a system, e.g. position of a flying kite, may be determinable from its

current state and future inputs, the behavior of each state of the system must be a know function of the instantaneous values of the other state variables and system inputs. The state-space model for a system represents these functional dependencies in terms of differential equations (in continuous case) or difference equations (in discrete case).

Consider a process described by an n-th order difference equation (similarly a differential equation) of the form:

$$y_{k+1} = a_{0,k}y_k + \cdots + a_{n-1,i}y_{k-n+1} + \omega_k, \tag{5.1}$$

where $k \geq 0$, ω_k is a zero-mean (statistically) white random "noise" process with the autocorrelation of:

$$E(\omega_k, \omega_j) = Q_k\delta_{kj}, \tag{5.2}$$

and initial values $\{y_0, y_{-1}, \ldots, y_{-n+1}\}$ are zero-mean random variables with a know $n \times n$ covariance matrix:

$$P_0 = E(y_{-j}, y_{-i}), \tag{5.3}$$

where $j, k \in \{0, n-1\}$. Also assuming that:

$$E(\omega_k, y_k) = 0 \quad \text{for} - n + 1 \leq j \geq 0 \text{ and } k \geq 0, \tag{5.4}$$

which ensure that:

$$E(\omega_k, y_k) = 0, \tag{5.5}$$

for $k \geq j \geq 0$. In other words, the noise is statistically independent from the process to be estimated. Under some other basic conditions, see [Kailath and Hassibi, 2000], this difference equation can be re-written as:

$$\hat{x}_{k+1} = \begin{bmatrix} y_{k+1} \\ y_k \\ y_{k-1} \\ \vdots \\ y_{k-n+2} \end{bmatrix} = \underbrace{\begin{bmatrix} a_0 & a_1 & \dots & a_{n-2} & a_{n-1} \\ 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \dots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & 0 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} y_k \\ y_{k-1} \\ y_{k-2} \\ \vdots \\ y_{k-n+1} \end{bmatrix}}_{\hat{x}_k} + \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \omega_k, \quad (5.6)$$

$$y_k = \underbrace{\begin{bmatrix} 1 & 0 & \dots & 0 \end{bmatrix}}_{C} \underbrace{\begin{bmatrix} y_k \\ y_{k-1} \\ y_{k-2} \\ \vdots \\ y_{k-n+1} \end{bmatrix}}_{\hat{x}_k}, \quad (5.7)$$

which leads to the state space model:

$$x_{k+1} = Ax_k + \omega_k, \quad (5.8)$$

$$y_k = Cx_k, \quad (5.9)$$

Equation (5.8) represents the way a new state at the next time step x_{k+1} is modeled as a linear combination of both the previous state x_k and some process noise ω_k. Equation (5.9) of the state space model describes the way the process measurements or observations y_k are derived from the internal

state x_k. In other word, Equation (5.9) provides the measurements from the system comprised of some linear combination of the states. These two equations are often referred to respectively as the process model and the measurement model, and they serve as the basis for virtually all linear estimation methods.

5.2. Measurement and Process Noise

Suppose that a measurement has been made at time t_k and that the information provided is to be applied in updating the estimate of the state x of a stochastic system at time t_k. The many approaches to this basic problem are typically based on the state-space model presented in the previous Section 5.1. There is typically a process model that models the transformation of the process state. This can usually be represented, similar to Equation (5.8), as a linear stochastic difference equation:

$$x_k = Ax_{k-1} + Bu_k + w_{k-1}, \tag{5.10}$$

where u_k is the system input at time k. In addition there is some form of measurement model that describes the relationship between the process state and the measurements. This can usually be represented, similar to equation (5.9) with a linear expression:

$$z_k = Cx_k + v_k. \tag{5.11}$$

The terms w_k and v_k are random variables representing the process and measurement noise respectively. Note that in Equation (5.11) we changed

the dependent variable to z_k instead of y_k as in equation (5.9). The rationale is to reinforce the notion that the measurements do not have to be of elements of the state specifically, but can be any linear combination of the state elements plus some measurement noise. We consider here the common case of noisy sensor measurements. There are many sources of noise in such measurements. For example, each type of sensor has fundamental limitations related to the associated physical medium, and when pushing the envelope of these limitations the signals are typically degraded.

In addition, some amount of random electrical noise is added to the signal via the sensor and the electrical circuits. The time varying ratio of "pure" signal to the electrical noise continuously affects the quantity and quality of the information. There is the additional problem that the actual state transform model is completely unknown. While we can make predictions over relatively short intervals using models based on recent state transforms, such predictions assume that the transforms are predictable, which is not always the case. The result is that information obtained from any one sensor must be qualified as it is interpreted as part of an overall sequence of estimates, and analytical measurement models typically incorporate some notion of random measurement noise or uncertainty.

5.3. Basic Kalman Filter

The Kalman filter can be characterized as an algorithm for computing the conditional mean and covariance of the probability distributions of the state

of a linear stochastic system with uncorrelated Gaussian process and measurement noise. The conditional mean is the unique unbiased and it is propagated in feedback form by a system of linear differential equations or by the corresponding discrete-time equations (presented in Section 5.2):

$$x_k = Ax_{k-1} + Bu_k + w_{k-1}, \tag{5.12}$$

with a measurement equation:

$$z_k = Cx_k + v_k. \tag{5.13}$$

The random variables w_k and v_k represent the process and measurement noise and they are assumed to be independent (of each other), white, and with Gaussian (normal) probability distributions:

$$p(w) \sim N(0, Q), \tag{5.14}$$

$$p(v) \sim N(0, R). \tag{5.15}$$

The relevant covariance matrices are thus:

$$\begin{aligned} E(w_k v_j^T) &= 0 \text{ for all } k, j \\ E(w_k w_j^T) &= Q\delta_{k,j} \\ E(v_k v_j^T) &= R\delta_{k,j}. \end{aligned} \tag{5.16}$$

In other word, the process and measurement noise are independent of each other. In practice, the process noise covariance Q and measurement noise covariance R matrices might change with each time step or measurement,

however here we assume they are constant.

Next, we will introduce the definition of the innovations process, see [Mendel, 1995], which is the key to deriving the Kalman filter. In general the innovations is defined as:

$$\tilde{z}_k = z_k - E(z_k \mid z^{k-1}), \tag{5.17}$$

where $z^k = \{z_k, z_{k-1}, \ldots, z_0\}$ denotes the cumulative measurement history. For the state-space model presented above, the estimate $E(z_k \mid z^{k-1})$ can be written as:

$$\begin{aligned} E(z_k \mid z^{k-1}) &= E(Cx_k + v_k \mid z^{k-1}) \\ &= CE(x_k \mid z^{k-1}) + E(v_k \mid z^{k-1}). \end{aligned} \tag{5.18}$$

By using the fact that $E(v_k) = 0$ (the measurement noise is zero-mean), and the measurement noise v_k are independent of prior measurement; $E(z_k \mid z^{k-1})$ can be further re-written as:

$$\begin{aligned} E(z_k \mid z^{k-1}) &= CE(x_k \mid z^{k-1}) + 0 \\ &= C\hat{x}_{k|k-1}. \end{aligned} \tag{5.19}$$

The quantity $E(x_k \mid z^{k-1}) = \hat{x}_{k|k-1}$ is very important in the Kalman filter derivation and can be thought of as a minimum variance estimate of x_k given prior measurements (a one step prediction). The innovations in Equation (5.17) reflects the discrepancy between the predicted measurement $\hat{x}_{k|k-1}$ and the actual measurement z_k. An innovations of zero implies that the two are in complete agreement. In addition, there are two properties of the innovations to be considered:

1. The innovations $\tilde{z}_k$ is zero mean, $E(\tilde{z}_k \mid z^{k-1}) = 0$ when conditioned on z^{k-1}. To show this we have:

$$\begin{aligned} E(\tilde{z}_k \mid z^{k-1}) &= E(z_k - E(z_k \mid z^{k-1}) \mid z^{k-1}) \\ &= E(z_k \mid z^{k-1}) - E(E(z_k \mid z^{k-1}) \mid z^{k-1}) = 0. \end{aligned} \tag{5.20}$$

2. The innovations is an uncorrelated sequence, $E(\tilde{z}_k \tilde{z}_j^T) = 0$. To show this, assume that $j < k$, and note that $z_j - E(z_j \mid z^{j-1})$ is a determistic function of z^{k-1}, then:

$$\begin{aligned} E(\tilde{z}_k \tilde{z}_j^T) &= E([z_k - E(z_k \mid z^{k-1})][z_j - E(z_j \mid z^{j-1})]^T) \\ &= E(E([z_k - E(z_k \mid z^{k-1})][z_j - E(z_j \mid z^{j-1})]^T \mid z^{k-1})) \\ &= E(E([z_k - E(z_k \mid z^{k-1})] \mid z^{k-1})[z_j - E(z_j \mid z^{j-1})]^T) \\ &= E(0[z_j - E(z_j \mid z^{j-1})]^T) = 0, \end{aligned} \tag{5.21}$$

where we have used the fact that z_j and z^{j-1} are effectively constants when conditioned on z^{k-1} for $j < k$. In general, the innovations $z_k - E(z_k \mid z^{k-1})$ is orthogonal to any linear or nonlinear vector function of z^{k-1}

$$\begin{aligned} E([z_k - E(z_k \mid z^{k-1})] f(z^{k-1})^T) &= E(z_k f(z^{k-1})^T) - E(E(z_k f(z^{k-1})^T \mid z^{k-1})) \\ &= E(z_k f(z^{k-1})^T) - E(z_k f(z^{k-1})^T) = 0. \end{aligned} \tag{5.22}$$

Now, the goal of the Kalman filter is to compute the following quantities:

$$
\begin{aligned}
\hat{x}_{k|k-1} &= E(x_k \mid z^{k-1}), \\
\hat{x}_{k|k} &= E(x_k \mid z^k), \\
P_{k|k-1} &= E([x_k - \hat{x}_{k|k-1}][x_k - \hat{x}_{k|k-1}]^T), \\
P_{k|k} &= E([x_k - \hat{x}_{k|k}][x_k - \hat{x}_{k|k}]^T).
\end{aligned} \tag{5.23}
$$

The first two terms are one step prediction and filtered estimate, respectively. The term $P_{k|k-1}$ and $P_{k|k}$ are the corresponding error covariance estimates. To proceed further, assuming the measurement model in Equation (5.11):

$$
z_k = Cx_k + v_k. \tag{5.24}
$$

In addition, we also assume that initially the two values $P_{k|k-1}$ and $\hat{x}_{k|k-1}$ are given. Then, noting that v_k is zero mean and independent of z^{k-1}, the innovations is given by:

$$
\begin{aligned}
\tilde{z}_k &= z_k - E(z_k \mid z^{k-1}) \\
&= z_k - E(Cx_k + v_k \mid z^{k-1}) \\
&= z_k - CE(x_k \mid z^{k-1}) \\
&= z_k - C\hat{x}_{k|k-1}.
\end{aligned} \tag{5.25}
$$

Before proceeding further, consider for three Gaussian random vectors a, b, c where the innovations is $\tilde{b} = b - E(b \mid c)$. As mentioned above, the innovations $\tilde{b}$ has zero mean and independent from c and thus uncorrelated. Which lead to:

$$
E\left(\begin{bmatrix} \tilde{b} \\ c - m_c \end{bmatrix} \begin{bmatrix} \tilde{b} & c - m_c \end{bmatrix}^T\right)^{-1} = \begin{bmatrix} P_{\tilde{b}}^{-1} & 0 \\ 0 & P_c^{-1} \end{bmatrix}, \tag{5.26}
$$

$$
\begin{aligned}
E(a \mid b, c) &= E(a \mid b - E(b \mid c), c) = E(a \mid \tilde{b}, c) \\
&= m_a + E([a - m_a][\tilde{b}^T, (c - m_c)^T]) \\
&\quad \times E\left(\begin{bmatrix} \tilde{b} \\ c - m_c \end{bmatrix} \begin{bmatrix} \tilde{b} & c - m_c \end{bmatrix}^T\right)^{-1} \left(\begin{bmatrix} \tilde{b} \\ c - m_c \end{bmatrix}\right) \\
&= m_a + E([a - m_a]\tilde{b}^T) P_{\tilde{b}}^{-1} \tilde{b} + E([a - m_a](c - m_c)^T) \\
&\quad \times P_c^{-1}[c - m_c] \\
&= E(a \mid \tilde{b}) + E(a \mid c) - m_a,
\end{aligned}
\tag{5.27}
$$

where m_a, m_b and m_c are the means or expected values of a, b and c and $P_{\tilde{b}}$ and P_c are the covariance of the innovations $\tilde{b}$ and c. In addition, a recursive expression for the filtered estimate $\hat{x}_{k|k} = E(x_k \mid z^k)$ can be obtained by replacing $a \to x_k, b \to z_k, c \to z^{k-1}$. The filtered estimate can be rewritten as:

$$
\begin{aligned}
\hat{x}_{k|k} &= E(x_k \mid z^k) = E(x_k \mid z_k - E(z_k \mid z^{k-1}), z^{k-1}) \\
&= E(x_k \mid \tilde{z}_k, z^{k-1}) \\
&= E(x_k \mid z^{k-1}) + E(x_k \mid \tilde{z}_k) - m_{x_k} \\
\Rightarrow \hat{x}_{k|k} &= \hat{x}_{k|k-1} + E(x_k \mid \tilde{z}_k) - m_{x_k}.
\end{aligned}
\tag{5.28}
$$

The expected value of x_k given the innovations $E(x_k \mid \tilde{z}_k)$ is given as:

$$
E(x_k \mid \tilde{z}_k) = m_{x_k} + P_{x\tilde{z}} P_{\tilde{z}\tilde{z}}^{-1} \tilde{z}_k, \tag{5.29}
$$

where the cross-covariance term is found by adding and subtracting the term $\hat{x}_{k|k-1}$ from x_k:

$$
\begin{aligned}
P_{x\tilde{z}} &= E([x_k - m_{x_k}][z_k - C\hat{x}_{k|k-1}]^T) \\
&= E([x_k - \hat{x}_{k|k-1}][C(x_k - \hat{x}_{k|k-1}) + v_k]^T) + E([\hat{x}_{k|k-1} - m_{x_k}]\tilde{z}_k^T),
\end{aligned} \tag{5.30}
$$

where $E([\hat{x}_{k|k-1} - m_{x_k}]\tilde{z}_k^T) = 0$, however, since $\hat{x}_{k|k-1} - m_{x_k}$ is a deterministic function of z^{k-1}, which is orthogonal to the innovations $z_k - E(z_k \mid z^{k-1})$.

By introducing the one step covariance matrix $P_{k|k-1}$, the cross covariance can be rewritten as:

$$
\begin{aligned}
P_{x\tilde{z}} &= E([x_k - \hat{x}_{k|k-1}][C(x_k - \hat{x}_{k|k-1}) + v_k]^T) \\
&= E([x_k - \hat{x}_{k|k-1}][x_k - \hat{x}_{k|k-1}]^T C^T) \\
&= P_{k|k-1}C^T,
\end{aligned} \tag{5.31}
$$

where one could observe that v_k is zero mean and uncorrelated with both x_k and $\hat{x}_{k|k-1}$.

The covariance matrix of the innovations term can be similarly derived as:

$$
\begin{aligned}
P_{\tilde{z}\tilde{z}} &= E([z_k - C\hat{x}_{k|k-1}][z_k - C\hat{x}_{k|k-1}]^T) \\
&= E([C(x_k - \hat{x}_{k|k-1}) + v_k][C(x_k - \hat{x}_{k|k-1}) + v_k]^T) \\
&= CP_{k|k-1}C^T + R.
\end{aligned} \tag{5.32}
$$

Thus, the filtered estimate or Measurement Update step of the Kalman filter then becomes:

$$\begin{aligned}
\hat{x}_{k|k} &= \hat{x}_{k|k-1} + E(x_k \mid \tilde{z}_k) - m_{x_k} \\
&= \hat{x}_{k|k-1} + P_{x\tilde{z}} P_{\tilde{z}\tilde{z}}^{-1} \tilde{z}_k \\
&= \hat{x}_{k|k-1} + P_{k|k-1} C^T [C P_{k|k-1} C^T + R]^{-1} [z_k - C\hat{x}_{k|k-1}],
\end{aligned} \tag{5.33}$$

which can be rewritten more compactly in terms of the Kalman gain defined as:

$$K = P_{k|k-1} C^T [C P_{k|k-1} C^T + R]^{-1}, \tag{5.34}$$

then, the more compact form of the Measurement Update is:

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K[z_k - C\hat{x}_{k|k-1}]. \tag{5.35}$$

The next one step prediction is obtained directly from the state-space model introduced from before:

$$\begin{aligned}
\hat{x}_{k+1|k} &= E(x_{k+1} \mid z^k) \\
&= E([Ax_k + Bu_k + w_k] \mid z^k) \\
&= AE(x_k \mid z^k) + Bu_k \\
&= A\hat{x}_{k|k} + Bu_k.
\end{aligned} \tag{5.36}$$

The process noise w_k is zero mean and independent of the observation z^k, whereas the input u_k are deterministic. In addition, the input u_k is a deterministic control input to the system.

The last component of the Kalman filter is the filtered or Measurement Update covariance $P_{k|k}$, which can be obtained by:

$$\begin{aligned} P_{k|k} &= E([x_k - \hat{x}_{k|k}][x_k - \hat{x}_{k|k}]^T) \\ &= E([x_k - \hat{x}_{k|k} - K[z_k - C\hat{x}_{k|k-1}]][x_k - \hat{x}_{k|k-1} - K[z_k - C\hat{x}_{k|k-1}]]^T). \end{aligned} \tag{5.37}$$

Let us define the prediction error as:

$$e_{k|k-1} = x_k - \hat{x}_{k|k-1}, \tag{5.38}$$

and note that $P_{k|k-1} = E(e_{k|k-1}e_{k|k-1}^T)$. Thus the measurement covariance update then becomes:

$$\begin{aligned} P_{k|k} &= E([e_{k|k-1} - K[v_k + Ce_{k|k-1}]][e_{k|k-1} - K[v_k + Ce_{k|k-1}]]^T) \\ &= P_{k|k-1} - KCP_{k|k-1} - P_{k|k-1}C^TK^T + K[CP_{k|k-1}C^T + R]K^T. \end{aligned} \tag{5.39}$$

Next, using the definition of the Kalman gain in Equation (5.34) to simplify the above expression:

$$\begin{aligned} K[CP_{k|k-1}C^T + R]K^T &= K[CP_{k|k-1}C^T + R][CP_{k|k-1}C^T + R]^{-1}CP_{k|k-1} \\ &= KCP_{k|k-1} \\ \Rightarrow P_{k|k-1}^TC^TK^T &= P_{k|k-1}^TC^T[CP_{k|k-1}C^T + R]^{-1}CP_{k|k-1} \\ &= KCP_{k|k-1}. \end{aligned} \tag{5.40}$$

The measurement covariance update $P_{k|k}$ then becomes:

$$\begin{aligned} P_{k|k} &= P_{k|k-1} - 2KCP_{k|k-1} + KCP_{k|k-1} \\ &= [I - KC]P_{k|k-1}. \end{aligned} \tag{5.41}$$

We are now ready to show the one step predictor covariance update equals to:

$$\begin{aligned} P_{k+1|k} &= E([x_{k+1} - \hat{x}_{k+1|k}][x_{k+1} - \hat{x}_{k+1|k}]^T) \\ &= E([A[x_k - \hat{x}_{k|k}] + w_k][A[x_k - \hat{x}_{k|k} + w_k]^T]) \\ &= AP_{k|k}A^T + Q. \end{aligned} \tag{5.42}$$

The whole Kalman fitler can now be summarized as:

1. Measurement Update:

$$\begin{aligned} \hat{x}_{k|k} &= \hat{x}_{k|k-1} + P_{k|k-1}C^T[CP_{k|k-1}C^T + R]^{-1}[z_k - C\hat{x}_{k|k-1}] \\ &= \hat{x}_{k|k-1} + K[z_k - C\hat{x}_{k|k-1}]. \end{aligned} \tag{5.43}$$

2. Measurement Covariance Update:

$$\begin{aligned} P_{k|k} &= [I - P_{k|k-1}C^T[CP_{k|k-1}C^T + R]^{-1}C]P_{k|k-1} \\ &= [I - KC]P_{k|k-1}. \end{aligned} \tag{5.44}$$

3. One Step Prediction:

$$\hat{x}_{k+1|k} = A\hat{x}_{k|k} + Bu_k. \tag{5.45}$$

4. Prediction Covariance Update:

$$P_{k+1|k} = AP_{k|k}A^T + Q. \tag{5.46}$$

We can rewrite the Kalman filter in a more familiar and compact form by

reordering step 1. and 2. with step 3. and 4. and by replacing:

$$\begin{aligned} \hat{x}_{k|k} &\rightarrow \hat{x}_k^- \\ \hat{x}_{k|k-1} &\rightarrow \hat{x}_{k-1} \\ P_{k|k} &\rightarrow P_k^- \\ P_{k|k-1} &\rightarrow P_{k-1} \\ \hat{x}_{k+1|k} &\rightarrow \hat{x}_k \\ P_{k+1|k} &\rightarrow P_k. \end{aligned} \tag{5.47}$$

The result two steps Kalman filter is:

1. Time Update ("Predict"):

$$\begin{aligned} \hat{x}_k^- &= A\hat{x}_{k-1} + Bu_k, \\ P_k^- &= AP_{k-1}A^T + Q. \end{aligned} \tag{5.48}$$

2. Measurement Update ("Correct"):

$$\begin{aligned} K &= P_k^- C^T (CP_k^- C^T + R)^{-1}, \\ \hat{x}_k &= \hat{x}_k^- + K(z_k - C\hat{x}_k^-), \\ P_k &= (I - KC)P_k^-, \end{aligned} \tag{5.49}$$

where we denote $\hat{x}_K^-$ and P_k^- as the a-priori state estimate and a-priori covariance estimate and $\hat{x}_k$ and P_k as the a-posteriori state estimate and a-posteriori covariance estimate.

Overall, the Kalman filter estimates a process by using a form of feedback control: the filter estimates the process state at some time and then obtains feedback in the form of (noisy) measurements. As such, the equations for the Kalman filter fall into two groups: Time Update equations and Mea-

surement Update equations. The Time Update equations are responsible for projecting forward (in time) the current state $\hat{x}_k^-$ and error covariance P_k^- estimates to obtain the a-priori estimates for the next time step. The Time Update equations can also be thought of as predictor equations, while the Measurement Update equations can be thought of as corrector equations. The Measurement Update equations are responsible for the feedback, i.e. for incorporating a new measurement into the a-priori estimate to obtain an improved a-posteriori estimate.

The first task during the Measurement Update is to compute the Kalman gain K. It is easy to see that the goal of the Kalman filter is to compute an a-posteriori state estimate $\hat{x}$ as a linear combination of an a-priori estimate $\hat{x}^-$ and a weighted difference between an actual measurement z_k and a measurement prediction $C\hat{x}^-$. The matrix K is chosen to be the gain or blending factor that minimizes the a-posteriori error covariance P_k. Furthermore, we see that as the measurement error covariance R approaches zero, the gain K weights the residual more heavily. Specifically,

$$\lim_{R \to 0} K = C^{-1}. \tag{5.50}$$

On the other hand, as the a-priori estimate error covariance P_k^- approaches zero, the gain K weights the residual less heavily. Specifically,

$$\lim_{P_k^- \to 0} K = 0. \tag{5.51}$$

Another way of thinking about the weighting by K is that as the measurement error covariance R approaches zero, the actual measurement z_k is

"trusted" more and more, while the predicted measurement $C\hat{x}_k^-$ is trusted less and less. On the other hand, as the a-priori estimate error covariance P_k^- approaches zero the actual measurement z_k is trusted less and less, while the predicted measurement $C\hat{x}_k^-$ is trusted more and more.

The next step in the Measure Update is to use the obtained measurement z_k and $C\hat{x}_k^-$ to compute the innovations, and then to generate an a-posteriori state estimate by using that innovations and the Kalman gain K. Lastly the final step is to obtain an a-posteriori error covariance estimate P_k.

After each time and measurement update pair, the process is repeated with the previous a-posteriori estimates used to project or predict the new a-priori estimates. This recursive nature is one of the very appealing features of the Kalman filter and it makes practical implementations much more feasible than (for example) an implementation of a Wiener filter, see [Brown and Hwang, 1996], which is designed to operate on all of the data directly for each estimate. The Kalman filter instead recursively conditions the current estimate on all of the past measurements. This simple implementation is precisely why the Kalman filter was chosen for this project where the application was carried out in real-time.

5.4. Steady State Kalman Filter

It is possible to design a steady-state Kalman filter by computing the steady-state filter gain and covariance. For the scalar case, a steady-state solution for $P_{k|k-1}$ existed as long as the state-space model is time-invariant and

stable; and the measurement model is time-invariant, see [Mendel, 1995].

If the state-space model

$$x_k = Ax_{k-1} + Bu_k + w_{k-1}, \tag{5.52}$$

is time-invariant and stable, i.e. all eigenvalues of $[Az - I]$ are inside the unit circle, and the measurement equation is similarly time-invariant, so that:

$$z_k = Cx_k + v_k, \quad v_k \sim N(0, R), k = 1, 2, \ldots, \infty, \tag{5.53}$$

then, for any non-negative definite, symmetric initial contdition $P(0 \mid -1)$, one has:

$$\lim_{k \to \infty} P(k+1 \mid k) = \bar{P}, \tag{5.54}$$

where the steady-state covariance $\bar{P}$ is independent of the initial condition, and satisfies the algebraic Riccati equation:

$$\bar{P} = A\bar{P}[I - C^T(C\bar{P}C^T + R)^{-1}C\bar{P}]A^T + Q, \tag{5.55}$$

where the Riccati equation is derived by forcing $P_{k|k-1}$ to equal a constant matrix P:

$$\begin{aligned} P_{k|k} &= [I - P_{k|k-1}C^T[CP_{k|k-1}C^T + R]^{-1}C]P_{k|k-1} \\ P_{k+1|k} &= AP_{k|k}A + Q \\ \Rightarrow P_{k+1|k} &= AP_{k|k-1}[I - C^T[CP_{k|k-1}C^T + R]^{-1}CP_{k|k-1}]A^T + Q, \end{aligned} \tag{5.56}$$

$$\Rightarrow \bar{P} = A\bar{P}[I - C^T[C\bar{P}C^T + R]^{-1}C\bar{P}]A^T + Q. \tag{5.57}$$

Note that if the prediction covariance has a steady-state value, then the Kalman gain has a similar steady-state value given by:

$$K = P_{k|k-1}C^T[CP_{k|k-1}C^T + R]^{-1} \xrightarrow[k\to\infty]{} \bar{P}C^T[C\bar{P}C^T + R]^{-1} = \bar{K}, \tag{5.58}$$

where $\bar{K}$ is the steady-state Kalman gain and the eigenvalues of the steady-state Kalman filter is given as:

$$[A - \bar{K}CA], \tag{5.59}$$

where the eigenvalues lie within the unit circle. Note that this matrix is obtained by:

$$\begin{aligned} \hat{x}_{k|k} &= A\hat{x}_{k-1|k-1} + \bar{K}[z_k - CA\hat{x}_{k-1|k-1}] \\ &= [A - \bar{K}CA]\hat{x}_{k-1|k-1} + \bar{K}z_k. \end{aligned} \tag{5.60}$$

The requirement that the eigenvalues of $[A - \bar{K}CA]$ lie inside the unit circle is shown from the update for $\hat{x}_{k|k}$ as follows. First recall that the z-transform of causal vector sequence is defined by:

$$X(z) = \sum_{k=0}^{\infty} x_k z^{-k}, \tag{5.61}$$

then taking z-transforms on both sides of the steady-state filter measurement update equation (5.60) would yield:

$$\begin{aligned} \hat{x}_{k|k} &= [A - \bar{K}CA]\hat{x}_{k-1|k-1} + \bar{K}z_k \\ \Rightarrow X(z) &= [A - \bar{K}CA]z^{-1}X(z) + \bar{K}Z(z) \\ \Rightarrow X(z) &= [I - (A - \bar{K}CA)z^{-1}]^{-1}\bar{K}Z(z) \\ &= [zI - (A - \bar{K}CA)]^{-1}z\bar{K}Z(z), \end{aligned} \tag{5.62}$$

where the adjoint of $[zI - (A - \bar{K}CA)]^{-1}$ contains only zeros.

In general, we concluded that the poles of the steady-state Kalman filter are then given by the values of z for which the determinant vanished, i.e. when:

$$det([zI - (A - \bar{K}CA)]) = 0. \tag{5.63}$$

In order for the system to be stable, all poles have to be inside the unit circle. But the poles are at the location where determinant vanishes or when z satisfies:

$$[zI - (A - \bar{K}CA)]\mathbf{v} = 0, \tag{5.64}$$

for some vector $\mathbf{v}$, which in turn implies that z is an eigenvalue of $[A - \bar{K}CA]$. Hence, if all eigenvalues of $[A - \bar{K}CA]$ are inside the unit circle, all pole locations of z are inside the unit circle, and the steady-state filter is stable.

It is important to remember that the Kalman filter gain and error covariance are independent of the actual observations. The steady-state covariance equation (5.57) alone are all that is required for characterizing the performance of a system. Even without any measurement data available, just the

steady-state covariance calculations can be used to obtain the preliminary indications of a filter performance.

In addition, with the steady-state covariance and gain from Equations (5.57 - 5.58), we can avoid computing the covariance and gain for the Kalman filter at each time step. This allow us to avoid the matrix inversion require for computing the Kalman gain and the a-posteriori covance, see Equation (5.49). This would certainly reduce the computation time while we implement a Kalman filter in real-time for this project.

5.5. Extended Kalman Filter (EKF)

As described above in Section 5.3, the Kalman filter addresses the general problem of trying to estimate the state of a discrete-time controlled process that is governed by a linear stochastic difference equation. But what happens if the process to be estimated and (or) the measurement relationship to the process is non-linear? In this section, we will introduce the Extended Kalman Filter (EKF), which is essentially a Kalman filter that linearizes about the current mean and covariance. In something akin to a Taylor series, we can linearize the estimation around the current estimate using the partial derivatives of the process and measurement functions to compute estimates even in the face of non-linear relationships.

Let start by assuming that the system in Equations (5.10-5.11) are now governed by a nonlinear state-space model equation:

$$x_{k+1} = f(x_k, u_k) + w_k, \tag{5.65}$$

and a nonlinear measurement model:

$$z_k = h(x_k) + v_k, \tag{5.66}$$

where the process noise w_k and measurement noise v_k are again assumed to be independent (of each other), white, and with Gaussian (normal) probability distributions. In this scenario, the non-linear function f introduced above relates the state at time step k to the state at the next time step $k+1$. The non-linear function h in the measurement equation relates the state x_k to the measurement z_k.

Next, let us linearize z_k by using the Taylor's series:

$$z_k \approx h(\hat{x}_{k|k-1}) + \sum_{l=1}^{n} \frac{\partial}{\partial x_l}(h(x_k)_l(x_k - x_l)\Big|_{x_l=\hat{x}_{k|k-1}} + v_k \tag{5.67}$$

$$\begin{aligned} \Rightarrow z_k &\approx h(\hat{x}_{k|k-1}) + H(\hat{x}_{k|k-1})(x_k - \hat{x}_{k|k-1}) + v_k \\ \Rightarrow z'_k &\approx z_k - h(\hat{x}_{k|k-1}) + H(\hat{x}_{k|k-1})\hat{x}_{k|k-1} \end{aligned} \tag{5.68}$$

$$\Rightarrow z'_k \approx H(\hat{x}_{k|k-1})x_k + v_k, \tag{5.69}$$

where the pseudo-measurements z'_k are now approximately linearly related to x_k through the Jacobian matrix. The Jacobian matrix is defined by:

$$H(x) = \begin{bmatrix} \frac{\partial h(x)_1}{\partial x_1} & \frac{\partial h(x)_1}{\partial x_2} & \cdots & \frac{\partial h(x)_1}{\partial x_N} \\ \frac{\partial h(x)_2}{\partial x_1} & \frac{\partial h(x)_2}{\partial x_2} & \cdots & \frac{\partial h(x)_2}{\partial x_N} \\ \vdots & & & \vdots \\ \frac{\partial h(x)_M}{\partial x_1} & \cdots & & \frac{\partial h(x)_M}{\partial x_N} \end{bmatrix}. \tag{5.70}$$

The process model can similarly be linearized:

$$x_{k+1} = f(\hat{x}_{k|k}, u_k) + F(\hat{x}_{k|k}, u_k)(x_k - \hat{x}_{k|k}) + w_k, \tag{5.71}$$

where,

$$F(x) = \begin{bmatrix} \frac{\partial f(x)_1}{\partial x_1} & \frac{\partial f(x)_1}{\partial x_2} & \cdots & \frac{\partial f(x)_1}{\partial x_N} \\ \frac{\partial f(x)_2}{\partial x_1} & \frac{\partial f(x)_2}{\partial x_2} & \cdots & \frac{\partial f(x)_2}{\partial x_N} \\ \vdots & & & \vdots \\ \frac{\partial f(x)_M}{\partial x_1} & \cdots & & \frac{\partial f(x)_M}{\partial x_N} \end{bmatrix}. \tag{5.72}$$

The measurement update equations and one-step predictions can be computed by replacing $z_k \to z'_k$, and by using the definition of z'_k:

$$z'_k \approx z_k - h(\hat{x}_{k|k-1}) + H(\hat{x}_{k|k-1})\hat{x}_{k|k-1}. \tag{5.73}$$

The EKF equations become:

1. Measurement Update:

$$\begin{aligned} K &= P_{k|k-1}H(\hat{x}_{k|k-1})^T[H(\hat{x}_{k|k-1})P_{k|k-1}H(\hat{x}_{k|k-1})^T + R]^{-1} \\ \hat{x}_{k|k} &= \hat{x}_{k|k-1} + K[z'_k - H(\hat{x}_{k|k-1})\hat{x}_{k|k-1}] \\ &= \hat{x}_{k|k-1} + K[z_k - h(\hat{x}_{k|k-1})]. \end{aligned} \tag{5.74}$$

2. Measurement Covariance Update:

$$P_{k|k} = [I - KH(\hat{x}_{k|k-1})]P_{k|k-1}. \tag{5.75}$$

3. One Step Prediction:

$$\hat{x}_{k+1|k} = f(\hat{x}_{k|k}, u_k). \tag{5.76}$$

4. Prediction Covariance Update:

$$P_{k+1|k} = F(\hat{x}_{k|k}, u_k)P_{k|k}F(\hat{x}_{k|k}, u_k)^T + Q. \tag{5.77}$$

Again, we can rewrite the EKF in a more familiar form by rearranging the four steps and replacing:

$$\begin{aligned} \hat{x}_{k|k} &\rightarrow \hat{x}_k^- \\ \hat{x}_{k|k-1} &\rightarrow \hat{x}_{k-1} \\ P_{k|k} &\rightarrow P_k^- \\ P_{k|k-1} &\rightarrow P_{k-1} \\ \hat{x}_{k+1|k} &\rightarrow \hat{x}_k \\ P_{k+1|k} &\rightarrow P_k. \end{aligned} \tag{5.78}$$

The complete EKF equations become:

1. Time Update ("Predict"):

$$\begin{aligned} \hat{x}_k^- &= f(\hat{x}_{k-1}, u_k), \\ P_k^- &= FP_{k-1}F^T + Q. \end{aligned} \tag{5.79}$$

2. Measurement Update ("Correct"):

$$\begin{aligned} K &= P_k^- H^T (H P_k^- H^T + R)^{-1}, \\ \hat{x}_k &= \hat{x}_k^- + K(z_k - h(\hat{x}_k^-)), \\ P_k &= (I - KH) P_k^-, \end{aligned} \tag{5.80}$$

where, similar to the Kalman filter, the Time Update equations project the state and covariance estimates from the previous time step $k-1$ to the current time step k. The main difference is the EKF use the non-linear function f and the linearized Jacobian matrix F to estimate the state and covariance. Similarly, the Measurement Update equations use the non-linear measurement function h and the linearized Jacobian matrix H to compute the gain K and make "correction" to the state and covariance estimates.

An important feature of the EKF is that the Jacobian H in the equation for the Kalman gain K serves to correctly propagate or "magnify" only the relevant component of the measurement information. For example, if there is not a one-to-one mapping between the measurement z_k and the state via h, the Jacobian H affects the Kalman gain so that it only magnifies the portion of the residual $z_k - h(\hat{x}_k)$ that does affect the state. It is important to note that a fundamental flaw of the EKF is the distributions (or densities in the continuous case) of the various random variables have changed from normal (Gaussian) distributions after undergoing their respective nonlinear transformations. The EKF is simply an ad hoc state estimator that only approximates the optimality of Bayes rule by linearization.

5.6. Luenberger Observer

The Kalman filter introduced in Section 5.3 is a stochastic observer for a stochastic system containing additive noise processes and with known parameters. The Kalman filter is the optimum estimator when the corrupting noise has a Gaussian probability distribution. Like the Luenberger observer, the Kalman filter also includes a correction factor to insure stability and convergence, but for the Kalman filter it is based on the variances of the noise processes. If accurate estimates of the variance are not available, optimal observer performance is not obtained. In the deterministic case, when small or no random noise is present, the Luenberger observer and its extensions are used for time-invariant systems with known parameters.

In designing feedback control systems, it is very convenient to have the entire state vector be available through direct measurement from the system. However, for the case of the entire state vector cannot be measured, an observer (e.g. Luenberger Observer) can approximately reconstruct the unavailable state-variable for the feedback control to use (see [L. Fagiano and Khammash, 2013] for more details on the automatic control design, which was another part of the main project by Dr. Lorenzo Fagiano)

Let consider a system governed by:

$$x_k = Ax_{k-1}, \tag{5.81}$$

with a measurement equation:

$$y_k = Cx_k, \tag{5.82}$$

and an observer of the system is governed by:

$$z_k = Fz_{k-1} + Ky_k. \tag{5.83}$$

In designing the Luenberger Observer, the F matrix is fixed and the gain matrix K is arbitrary. Thus, a Luenberger Observer is determined uniquely, see [Gopinath, 1971], by the selection of K and takes the form of:

$$\begin{aligned} z_k &= (A - KC)z_{k-1} + Ky_k \\ &= Az_{k-1} + K(y_k - Cz_{k-1}). \end{aligned} \tag{5.84}$$

Next, recall that a system is observable if the matrix:

$$\begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{bmatrix}, \tag{5.85}$$

is a full rank matrix (has rank of n). Thus, corresponding to the real matrices C and A, the set of eigenvalues $A - KC$ can be made to correspond to the set of eigenvalues of any real matrix by suitable choice of the real matrix C. This means the Luenberger Observer can be designed for a linear time-invariant system if and only if the system is observable, see

[Luenberger, 1971]. In addition, the eigenvalues of values of $(A - KC)$ can be placed arbitrarily.

In practice, for continuous case, the eigenvalues of the observer are placed to be negative so that the observer will converge to the state of the actual system. In addition, the eigenvalues are chosen to be somewhat more negative than the eigenvalues of the actual system so that the observed state can converge to the actual state quickly. In theory, the eigenvalues can be placed arbitrarily toward $-\infty$ to allow instant convergence. However, this will turn the observer into a differentiator and thereby become highly sensitive to noise.

Overall, the equation for the Luenberger observer contains a term that corrects the current state estimates by an amount proportional to the prediction error: the estimation of the current output minus the actual measurement. Inclusion of this correction ensures stability and convergence of the observer even when the system being observed is unstable. The simplicity of the Luenberger Observer design and its resolution of the difficulty imposed by missing state-vector measurements make the Luenberger Observer a very attractive tool in feedback design. Thus, that is why we will be using the Luenberger Observer for the kite velocity angle γ (introduced in Section 6.2) as well as approximate the unavailable rate of the kite velocity angle $\dot{\gamma}$.

6. System Description

6.1. Small-scale Prototype Description

Figure 6.1.: Small-scale prototype

For this project, a small-scale prototype was built at University of California, Santa Barbara, shown in Figure 6.1 (see [project EISG, 2013] for a short movie clip). An inflatable power kite is connected to the trailer or Ground Unit (GU) by three cable lines. The outer left and and outer right

lines are used for steering the kite, while the middle line remain as a fixed length of 30m while the kite is flying.

The GU is placed parallel to the downwind direction and fixed to the ground. The GU is equipped with actuators, see Figure 6.2, capable of steering the kite to fly in "figure-eight" crosswind flight paths to generate the maximum forces, as mentioned in Section 2.2 (see [Fagiano and Milanese, 2012]). The kite is placed downwind from the GU at a fixed length of 30m. A human operator or an automatic control algorithm could direct the actuators to steer the kite in "figure-eight" flight path.

Figure 6.2.: Actuators for Kite steering

For this project, the GU does not included any generator thus no electricity is being generated from the flying the kite. The length of the lines

is fixed at 30m and thus not able to generate electricity through the synchronized reeling out of the three lines, as mentioned in Section 2.2 (see [Fagiano and Milanese, 2012]). However by using fixed short line length of 30m, the kite would still operate in an optimal operation of airborne wind energy generation. Ideally, the optimal operation of airborne wind energy generation would occur with a constant line speed equal to one third of the wind speed, see [Loyd, 1980]. With this small-scale prototype, the constant line speed is zero and would still operate similarly to the ideal condition.

In addition, typical airborne wind energy systems (e.g. see Section 2.2) operate with very low line reel-out speed compare to the kite's velocity. For this project, the kite is flying with a fixed line length of 30m which has zero line reel-out speed and similar to low line reel-out speed. Furthermore, by having fixed line length, the forces generated from flying the kite in "figure-eight" pattern would be the largest possible forces for any given wind conditions. This would allow the kite to fly with the highest accelerations and angular velocities, making the estimation problem for this small-scale prototype a lot more challenging.

6.2. Model Coordinate Systems

GU Coordinate System

There are multiple coordinate systems to be considered with the small-scale kite prototype introduced in Section 6.1. All coordinate systems are right-handed reference system. The inertial coordinate system $G \doteq (X, Y, Z)$, see Figure 6.3(a), is set up with the GU at the center. The X axis in the

G frame is parallel to the ground, contained in the symmetry plane of the GU and pointing downwind (an anemometer was used to determined wind direction, see Section 6.4) towards the kite operating area. The Z axis is perpendicular to the ground and pointing upward. The Y axis would be perpendicular to X and Z axis respectively. The kite position vector expressed in the reference system G is denoted as $\vec{p}_G = [p_X, p_Y, p_Z]^T \in \mathbb{R}^3$, where p_X, p_Y and p_Z are the scalar components of $\vec{p}_G$ along the $X, Y,$ and Z.

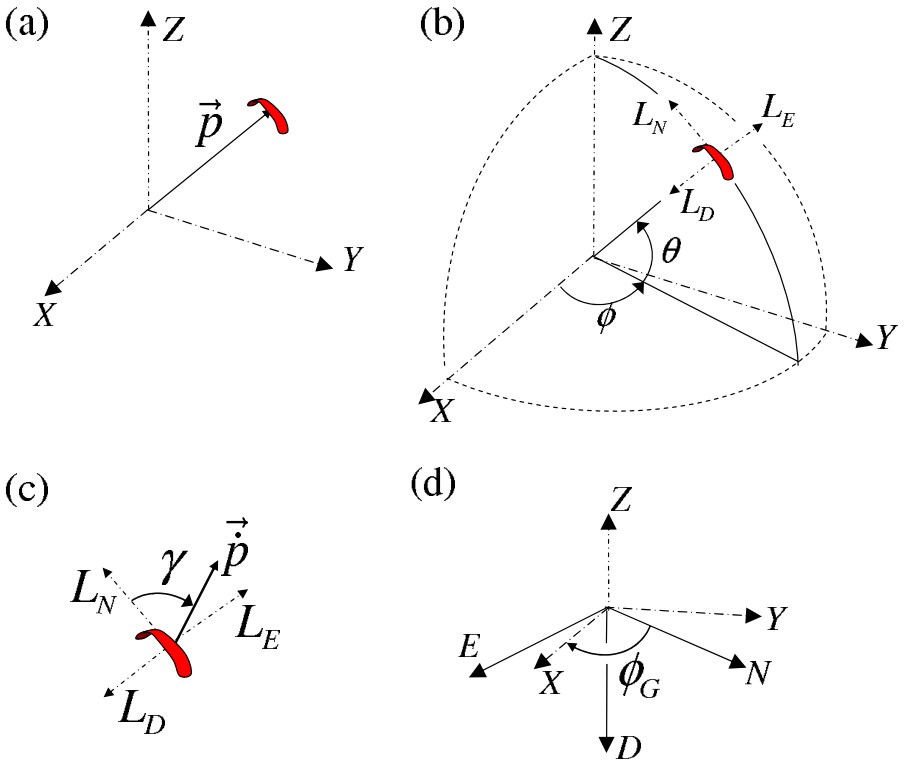


Figure 6.3.: Coordinate Systems: (a) GU coordinate system G and position of the kite $\vec{p}$; (b) local coordinate system $L = (L_N, L_E, L_D)$ and spherical coordinate θ, ϕ; (c) kite velocity angle γ; (d) GU coordinate system G, NED and angle ϕ_G

Spherical Coordinate System and "Local" Coordinate System

The kite's position can also be expressed in the spherical coordinate of θ and ϕ, see Figure 6.3(b). The angle between the (X, Y) plane and vector $\vec{p}_G$ is θ, where $\theta \in [0, \frac{\pi}{2}]$. The angle between the X axis and the projection of $\vec{p}_G$ onto the (X, Y) plane is ϕ, where ϕ is a positive rotation around the Z axis. In addition, we can convert from spherical coordinate (θ, ϕ) to G coordinate via:

$$\vec{p}_G = \begin{bmatrix} p_X \\ p_Y \\ p_Z \end{bmatrix} = r \begin{bmatrix} \cos(\theta)\cos(\phi) \\ \cos(\theta)\sin(\phi) \\ \sin(\theta) \end{bmatrix}, \tag{6.1}$$

where r is the distance between the GU and the kite (30m). In addition, the angles θ and ϕ could be used to defined the non-inertial Cartesian coordinate system $L \doteq (L_N, L_E, L_D)$. This coordinate will be called "local" coordinate where any vector from the G coordinate (i.e. $\vec{p}_G$), can be expressed in L coordinate by using the rotation matrix $R_{G\to L}$:

$$\vec{p}_L = R_{G\to L}\vec{p}_G, \tag{6.2}$$

where

$$R_{G\to L} = \begin{bmatrix} -\sin(\theta)\cos(\phi) & -\sin(\theta)\sin(\phi) & \cos(\theta) \\ -\sin(\phi) & \cos(\phi) & 0 \\ -\cos(\phi)\cos(\theta) & -\sin(\phi)\cos(\theta) & -\sin(\theta) \end{bmatrix}, \tag{6.3}$$

and the inverse transformation is given by the rotation mtarix $R_{L\to G} = R_{G\to L}^{-1} = R_{G\to L}^{T}$. The axes L_N and L_E are defined as the tangent plane at

point $\vec{p}_G$ to the sphere of radius r, see Figure 6.3(b). The kite's trajectory is confined on that sphere (due to the fixed line length $r = 30$m) and the axes L_N and L_E can be interpreted as the local north and local east direction relative to that sphere. Hence, L_E is always parallel to the the ground (the (X, Y) plane) and L_D will be the local down, pointing from the kite to the center of the sphere (where the GU is).

Kite Coordinate System

Next, let us consider the kite's coordinate system, $K \doteq (K_x, K_y, K_z)$, and the standard geographical North East Down coordinate system, $NED \doteq (N, E, D)$. The coordinate system K is centered at $\vec{p}_G$, and it is a non-inertial and is fixed with respect to the kite. K_x corresponds to the kite's longitudinal symmetry axis and pointing from the trailing edge to the leading edge of the kite. K_y is aligned with the transversal axis of the kite and pointing from the left kite tip to the right kite tip. Lastly, K_z will complete a right handed system and is perpendicular to K_x and K_y respectively. We can convert a vector from NED coordinate to G coordinate by using the rotation matrix $R_{NED \to G}$:

$$R_{NED \to G} = \begin{bmatrix} \cos(\phi_G) & \sin(\phi_G) & 0 \\ \sin(\phi_G) & -\cos(\phi_G) & 0 \\ 0 & 0 & -1 \end{bmatrix}, \tag{6.4}$$

where $\phi_G \in [0, 2\pi]$ is the angle between the N axis and the X axis , see Figure 6.3(d). The angle ϕ_G is measured as a positive rotation around the D axis (provided by a GPS installed on the GU, see Section 6.4). In addition, a vector could be converted from the K coordinate system to

the NED coordinate system by the rotation matrix $R_{K\to NED}(q)$, where $q = [q_1, q_2, q_3, q_4]^T \in \mathbb{R}^4$ is the quaternion, described in Chapter 4, defining the relative orientation between K and NED (see e.g. [LaValle, 2006] and [Coutsias and Romero, 2004]):

$$R_{K\to NED}(q) = \begin{pmatrix} 2(q_1^2+q_2^2)-1 & 2(q_2q_3-q_1q_4) & 2(q_2q_4+q_1q_3) \\ 2(q_2q_3+q_1q_4) & 2(q_1^2+q_3^2)-1 & 2(q_3q_4-q_1q_2) \\ 2(q_2q_4-q_1q_3) & 2(q_3q_4+q_1q_2) & 2(q_1^2+q_4^2)-1 \end{pmatrix}, \quad (6.5)$$

and the transformation from K coordinate to G coordinate is given by the rotation matrix $R_{K\to G} \doteq R_{NED\to G}R_{K\to NED}$. As described in Section 6.4, the quaternion q is provided by the IMU fixed onto the kite. In other word, the IMU provides the measurements of the kite (e.g. acceleration) in the K frame and also provides the quaternions which allow us to translate the kite measurements to other coordinate frame (e.g. G frame).

Kite's velocity angle

Let define the kite velocity angle as the angle between the kite velocity vector and the local north axis, L_N, see Figure 6.3(c). The angle γ is measured positive for a positive rotation around the local down axis L_D and defined as $\gamma \in [-\pi, \pi]$ and:

$$\gamma \doteq \arctan_2(\dot{p}_{L_E}, \dot{p}_{L_N}), \quad (6.6)$$

where $\arctan_2(\dot{p}_{L_E}, \dot{p}_{L_N}) \in [-\pi, \pi]$ is the 4-quadrant arc tangent function and $\dot{p}_{L_E}$and $\dot{p}_{L_N}$ are proportional to the sine and cosine component of γ

respectively.

In addition, since the line length is fixed at 30m, the vector $\vec{p}$ is always contained in the (L_N, L_E) plane. However, the definition (6.6) is more general and also hold for the case of varies line length. Furthermore, the angle γ is very important as the feedback variable for automatic control algorithm because it can be easily related to the kite's position, see e.g. [L. Fagiano and Khammash, 2013]. For example, $\gamma = 0, \gamma = \frac{\pi}{2}$and $\gamma = \pi$ would indicate that the kite is moving toward the local north, parallel to the ground toward the local east, and moving toward the ground.

6.3. Kinematic Model Equations

The model equations to be considered for the kite are based entirely on the kinematics of the system. They are basically obtained by differentiating the acceleration vector $\ddot{\vec{p}}_G(t)$ twice with respect to the continuous time variable t. Using only the kinematics equations bring two important advantages:

1. They provide an exact model (there are no neglected dynamic even for flexible kite).

2. They do not depend on any of the system's characteristics such as mass, shape, moment of inertia, aerodynamics of the kite.

These features do have influence on the motion of the kite through complex, infinite dimensional nonlinear dynamics, whose inputs are the steering command given by the GU and the (unmeasured) wind; and yet such dynamics are irrelevant for our scope if a measurement or estimate of $\ddot{\vec{p}}_G(t)$ is

available. However, while the kinematics of the inertial G frame are given by linear operator (i.e. derivatives), we still end up with nonlinear model equations due to the fact that different variables are measured in different reference frames.

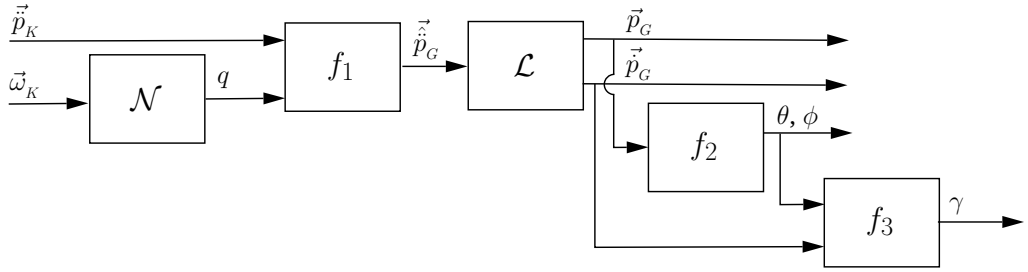


Figure 6.4.: Scheme of the considered kinematic model

Next, we will split the model into five inter connected subsystems, see Figure 6.4. The considered variables are the kite position $\vec{p}_G(t)$, velocity $\dot{\vec{p}}_G(t)$ and acceleration $\ddot{\vec{p}}_G(t)$ in the inertial reference G frame; the kite acceleration $\ddot{\vec{p}}_K(t)$ and the quaternion q from the non-inertial reference K frame; the kite angular velocity $\vec{\omega}_K(t)$ in the reference frame K. Finally, the overall model equation shown in Figure 6.4 are:

$$\mathcal{L}: \begin{bmatrix} \frac{d\vec{p}_G(t)}{dt} \\ \frac{d\dot{\vec{p}}_G(t)}{dt} \end{bmatrix} = \begin{bmatrix} 0_3 & I_3 \\ 0_3 & 0_3 \end{bmatrix} \begin{bmatrix} \vec{p}_G(t) \\ \dot{\vec{p}}_G(t) \end{bmatrix} + \begin{bmatrix} 0_3 \\ I_3 \end{bmatrix} \ddot{\vec{p}}_G(t) \tag{6.7a}$$

$$\mathcal{N}: \begin{bmatrix} 0 & -\omega_{K_x} & -\omega_{K_y} & -\omega_{K_z} \\ \omega_{K_x} & 0 & -\omega_{K_z} & \omega_{K_y} \\ \omega_{K_y} & \omega_{K_z} & 0 & -\omega_{K_x} \\ \omega_{K_z} & -\omega_{K_y} & \omega_{K_x} & 0 \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} \tag{6.7b}$$

$$f_1: \ddot{\vec{p}}_G(t) = R_{NED\to G}(\phi_G) R_{K\to NED}(q(t)) \ddot{\vec{p}}_K(t) \tag{6.7c}$$

$$f_2: \begin{cases} \theta(t) = \arcsin(\frac{p_Z(t)}{r}) \\ \phi(t) = \arctan_2(\frac{p_Y(t)}{p_X(t)}) \end{cases} \tag{6.7d}$$

$$f_3: \begin{cases} \ddot{\vec{p}}_L = R_{G\to L} \ddot{\vec{p}}_G \\ \gamma(t) = \arctan_2(\dot{p}_{L_E}, \dot{p}_{L_N}(t)) \end{cases} \tag{6.7e}$$

where 0_3 is a 3×3 matrix of zeros and I_3 is a 3×3 identity matrix. Overall, the inputs to the system would be $\ddot{\vec{p}}_K$ and $\vec{\omega}_K(t)$. The states of the system are $\vec{p}_G(t), \dot{\vec{p}}$ and $q(t)$. The system is primarily composed of a linear dynamic system $\mathcal{L}$, a nonliear dynamic system $\mathcal{N}$ and three static nonlinear functions f_1, f_2 and f_3. The only parameters in the system are the constant angle ϕ_G, which gives the orientation of the GU with respect to the geographical north (see Figure 6.3), and the line length r (30m). Both of these parameters can be measured very accurately and are assumed to be known exactly.

6.4. Sensors Setup

On-board Kite Sensor

Figure 6.5.: IMU (SBG:500N)

There are several sensors installed on both the GU and the kite. For the on-board kite sensors, an Inertial Measurement Unit (IMU) manufactured by SBG System®(see [SBG Systems S.A.S., IG-500N, 2009]), shown in Figure 6.5, was used for this project. Due to its small size and light weight of 44 grams, the IG-500N IMU could be placed directly on the kite, see Figure 6.6. The IMU contains three accelerometers, three gyroscopes, three magnetometers, a barometer and embedded with a GPS, see Appendix A. While being attached to the kite, the IMU can transmit its measurements through a 900 MHz radio to a receiver on the GU at a rate of 115200 Baud.

The accelerometers have a bandwidth of 50 Hz, measurement range of ± 5g (where g is the gravity constant), a bias of $\pm 4 * 10^{-3}$ g, with nonlinearity $< 10^{-2}$ g, and noise density of $2.5 * 10^{-4}$ g/$\sqrt{\text{Hz}}$.

The gyroscopes have a bandwidth of 40 Hz, measurement range of $\pm 300°/s$, a bias of $\pm 10^{-1\circ}/s$, nonlinearity $< 310^{-1\circ}/s$, and noise density of $5 * 10^{-2\circ}/s\sqrt{\text{Hz}}$.

Figure 6.6.: IMU attached onto kite

The magnetometers have a bandwidth of 500 Hz, measurement range of ± 1.2 Gauss, a bias of $\pm 5 * 10^{-4}$ Gauss, nonlinearity $< 2.4 * 10^{-3}$ Gauss, and noise density of 10^{-5} Gauss/$\sqrt{\text{Hz}}$.

The barometers have a bandwidth of 9 Hz with a resolution of 0.2 m. All sensors mentioned so far are suitable for this particular project as the kite angular velocities are expected to be in the rage of $\pm 200°/\text{s}$, and the kite accelerations lie in the range of ± 4.5g. In addition, all sensors mentioned so far are acquired with sampling time of $T_s = 0.02$s.

Lastly, the IMU is embedded with a GPS which has a nominal horizontal accuracy (i.e. the (N, E) planes) of ± 2.5m and a bandwidth of 4 Hz (sampling time of 0.25 s). The GPS also give the measurement in the vertical direction but with extreme inaccuracy. Thus we would be using the barometers for the measurement of the kite vertical position.

GU Sensors

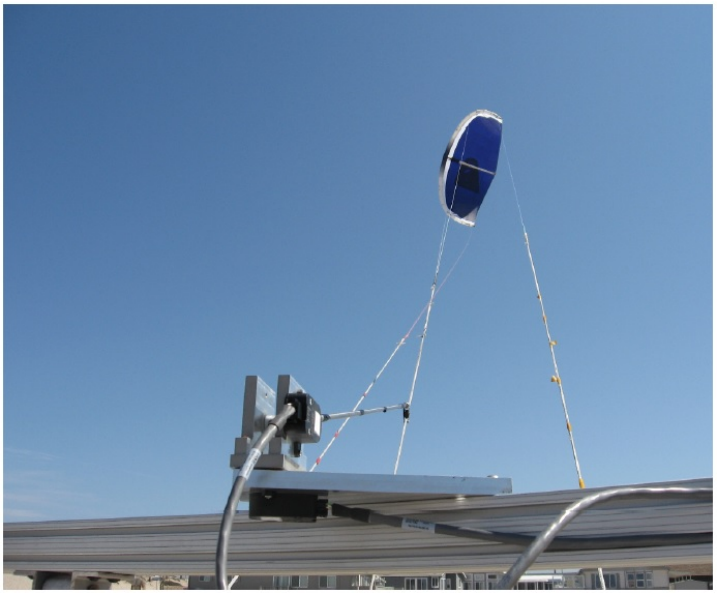


Figure 6.7.: Line angle measurement unit

A GPS and three magnetometers are installed on the GU to provide the GU's geographical location in the NED coordinate as well as the angle ϕ_G. In addition, the GU is equipped with a "line angle" measurement unit, see Figure 6.7, which provides a direct measurement of the angle of the main line connecting the kite to the GU.

The main component of the line angle measurement unit is two incremental encoders with the measurement rate of 50 Hz. The encoders are connected to the center line via a metal rod and a small pulley which allows the center line to slide along with low friction with respect to the rod while the kite is flying. One of the encoders measures θ_B, the angle between the metal rod and the (X, Y) plane, see Figure 6.8(a). While the other encoder measures

ϕ_B, the angle between the X axis and the projection of the metal rod onto the X, Y plane, see Figure 6.8(b).

The incremental encoders provide accurate absolute angle measurements through a standard integration algorithm that sums the encoder's counts and compares the signal given by the two encoder tracks, which are shifted by 90° in phase to compute the direction of motion (i.e. clockwise or counter-clockwise). The obtained signals from the encoders are very robust against noise and drift since they are based on discrete impulses.

In addition, an additional index impulse was used to re-set a standard position each time the encoders went through it. Each encoder has a resolution of $\frac{2\pi}{400}$ rad (i.e. 400 counts for each full revolution). With the fixed center line length of 30m, this translates in about 0.2m of resolution in each direction. While the center line is moving, so would the metal rod and by measuring θ_B and ϕ_B, we can convert the two angles into measurements of θ and ϕ via:

$$L = \sqrt{L_1^2 + L_2^2} \tag{6.8a}$$

$$\theta'_B = \theta_B - \arctan(\frac{L_1}{L_2}) \tag{6.8b}$$

$$l'_1 = L\sin(\theta'_B) \tag{6.8c}$$

$$l'_2 = L\cos(\theta'_B)\cos(\phi_B) - l_2 \tag{6.8d}$$

$$l'_3 = L\cos(\phi'_B)\sin(\phi_B) \tag{6.8e}$$

$$\theta = \arctan(\frac{l'_1 + l_1}{\sqrt{l'^2_2 + l'^2_3}}) \tag{6.8f}$$

$$\phi = \arctan(l'_3, l'_2) \tag{6.8g}$$

where L_1 and L_2 are the length of the metal rod and the pulley, see Figure 6.8. The lengths l_1 and l_2 are the distances between the positions of the encoders P_2 and the attachment point of the center line on the GU, P_1. All four lengths L_1, L_2, l_1, l_2 are fixed and exactly known.

In addition to the line angle measurement unit, an anemometer was also installed on the GU at 4m above the ground, to provide the wind speed and direction. Although the wind measurements are not directly used for the filtering algorithms, they are very useful to analyze the obtained results in Chapter 8.

In summary, we implicitly assume that:

- The line angle measurement unit provides a measure of the θ and ϕ angle.

- The GPS provides a measure of the position in the G frame, p_X and p_Y.
- The barometer provides a measure of the position in the G frame, p_Z.
- The IMU provides the measurements in the K frame: $\vec{\ddot{p}}_K, \vec{\omega}_K$ and q.

These assumptions imply that the kite lines are parallel to the kite position vector, $\vec{p}_G$. In addition, the IMU coincides with $\vec{p}_G$ and because the IMU is fixed to the kite, the measured accelerations and angular velocities are corresponding to those of the kite.

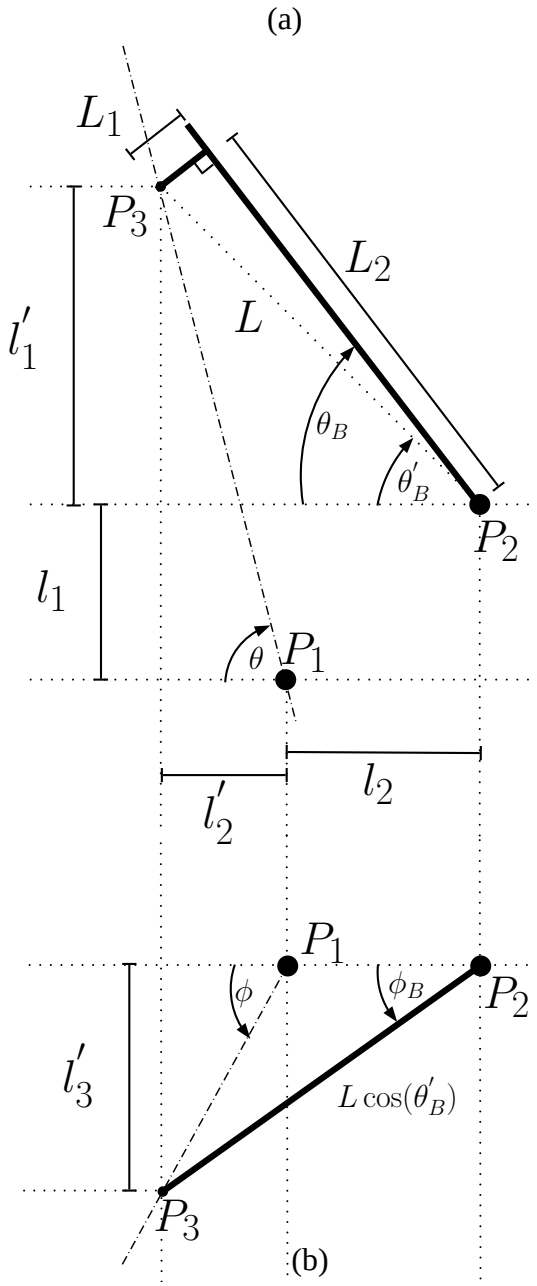


Figure 6.8.: (a) side view and (b) top view. Dash-dotted line: center line of the kite; thick solid lines: metal rod and pulley linking the encoders to the center line.

7. Sensor Fusion Algorithms

7.1. Kite Model & Kalman Filter

In this section we present different algorithms to estimate the kite position $\vec{p}_G(t)$ and velocity angle $\gamma(t)$ (described in Section 6.2). In the following chapter, the notation '$\sim$' is used to indicate a noise-corrupted measurement and/or unfiltered estimate of a given variable, e.g. $\vec{\tilde{p}}_G$ is the measurement of the position vector seen from system G, $\vec{p}_G$. In addition, the notation '$\wedge$' is used to indicate the estimate variables, i.e. the outputs of the Kalman filter.

By exploiting the structure of the considered kinematic model (introduced in Section 6.3) to separate the problem of estimating the kite orientation (i.e. the quaternion $q(t)$) from the problem of estimating the kite position and velocity. In fact, since the nonlinear system $\mathcal{N}$ is known and the IMU provides measurements for both system $\mathcal{N}$ input and state, i.e. $\vec{\tilde{\omega}}_K(t)$ and $\tilde{q}(t)$; an Extended Kalman Filter (EKF) (see Section 5.5) could be used to obtain the estimate quaternions $\hat{q}(t)$. Thankfully, most commercial IMU already have an EKF embedded and in particular, the IMU employed in this project is already included its own EKF, indicated as $\mathcal{N}_{EKF}$ (see Figure

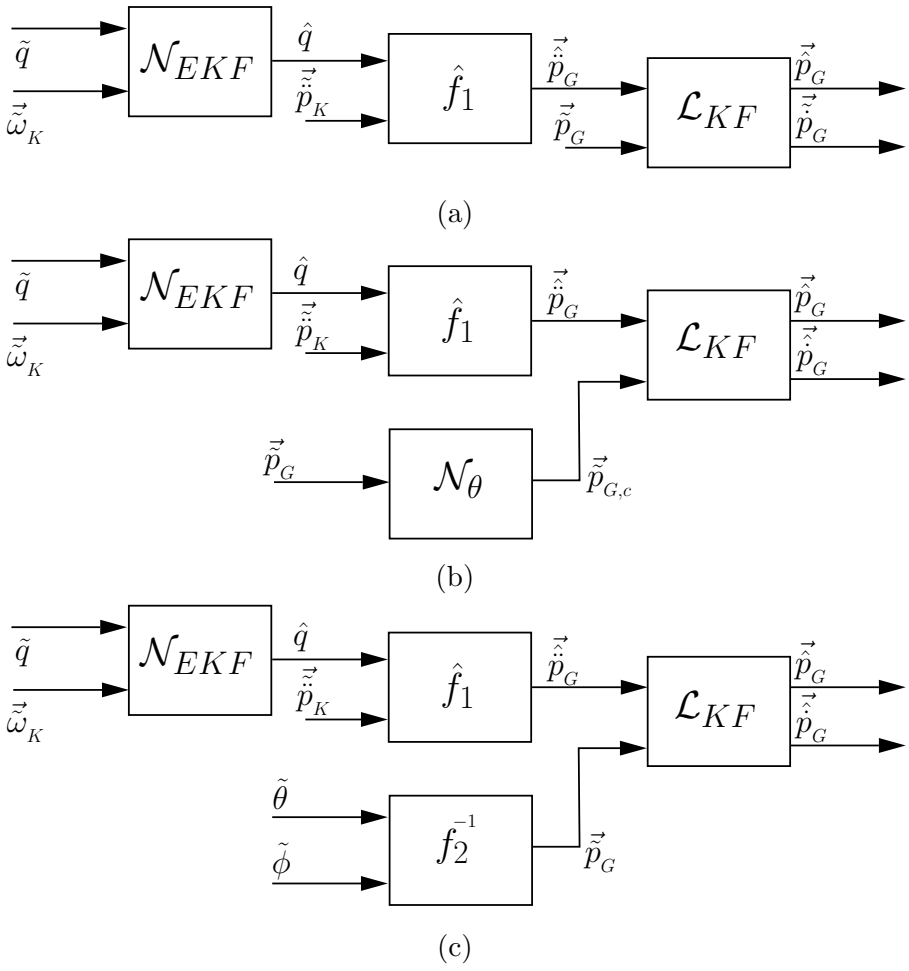


Figure 7.1.: Schemes of the designed observers for: (a) First Approach (b) Second Approach (c) Third Approach

7.1), for the purpose of estimating the quaternion, see e.g. [Sabatini, 2011].

The obtained attitude estimate have a range of 360° around all three axes of the K frame (i.e. K_x, K_y, K_z) with static accuracies of $\pm 0.5°$ around K_x, K_y and $\pm 1°$ around K_z; a dynamic accuracy of $\pm 1°$ RMS, resolution $< 510^{-2\circ}$ and repeatability error of $< 210^{-1\circ}$. Such performances are sufficient to capture the rotational motion of the wing during crosswind flight with high accuracy.

Using the estimate $\hat{q}(t)$ provided by $\mathcal{N}_{EKF}$ via the IMU, the rotation matrix $R_{K\to NED}(\hat{q}(t))$ can be computed by using Equation (6.5). An estimate $\hat{\vec{\ddot{p}}}_G(t)$ of $\vec{\ddot{p}}_G(t)$ could then be computed through the static nonlinear fucntion $\hat{f}_1$, given as:

$$\hat{f}_1 : \hat{\vec{\ddot{p}}}_G(t) = R_{NED\to G}(\phi_G) R_{K\to NED}(\hat{q}(t)) \hat{\vec{\ddot{p}}}_K(t) + \begin{bmatrix} 0 \\ 0 \\ \mathrm{g} \end{bmatrix}, \tag{7.1}$$

where $\hat{f}_1$ is obtained from f_1, introduced in Section 6.3 (see Equation (6.7c)), using the estimated rotation matrix $R_{K\to NED}(\hat{q}(t))$ instead of the "true" one and then remove the bias given by the gravity acceleration, g. The gravity was taken into account because of the characteristics of the employed MEMS accelerometers embedded in the IMU.

Furthermore, consider that the estimate $\hat{\vec{\ddot{p}}}_G(t)$ can be re-written as:

$$\hat{\vec{\ddot{p}}}_G(t) = \vec{\ddot{p}}_G(t) + \vec{\eta}_p(t), \tag{7.2}$$

where $\vec{\eta}_p(t) \in \mathbb{R}^3$ is some estimation error that we consider as the process noise. In addition, the estimate $\vec{\hat{\ddot{p}}}_G(t)$ can be regarded as a noise-corrupted input for the linear system $\mathcal{L}$, see Figure 6.4. Furthermore, we will use a steady-state Kalman Filter (introduced in Section 5.4), based on $\mathcal{L}$ (indicated as $\mathcal{L}_{KF}$ in Figure 7.1), to compute the estimations $\vec{\hat{p}}_G(k)$ and $\vec{\hat{\dot{p}}}_G(k)$. In order to do that, we need to find the state-space model by discretize $\mathcal{L}$ by forward difference with the employed sampling time T_s:

$$\begin{bmatrix} \vec{p}_G(k+1) \\ \vec{\dot{p}}_G(k+1) \end{bmatrix} = \underbrace{\begin{bmatrix} I_3 & T_s I_3 \\ 0_3 & I_3 \end{bmatrix}}_{A} \begin{bmatrix} \vec{p}_G(k) \\ \vec{\dot{p}}_G(k) \end{bmatrix} + \underbrace{\begin{bmatrix} 0_3 \\ T_s I_3 \end{bmatrix}}_{B} (\vec{\ddot{p}}_G(k) + \vec{\eta}_p(k)), \tag{7.3}$$

$$\vec{\hat{p}}_G(k) = \underbrace{\begin{bmatrix} I_3 & 0_3 \end{bmatrix}}_{C} \begin{bmatrix} \vec{p}_G(k) \\ \vec{\dot{p}}_G(k) \end{bmatrix} + \vec{\eta}_m(k)), \tag{7.4}$$

where $k \in \mathbb{Z}$ is the discrete time variable and $\vec{\eta}_m(k) \in \mathbb{R}^3$ represents the output measurement noise. In classical Kalman filtering theory, $\vec{\eta}_p(k)$ and $\vec{\eta}_m(k)$ are assumed to be independent white Gaussian noises with covariance matrices Q and R, see e.g. [Welch and Bishop, 2001]. For this project, the measurement and process noises can be reasonably assumed to be independent since they pertain to completely different sensors, (i.e. accelerometers, gyroscopes and magnetometers for the input $\vec{\hat{\ddot{p}}}_G(k)$ and either GPS and barometer or line angle sensor for the output measurement $\vec{\hat{p}}_G(k)$).

However, these independent measurements are not white Gaussian processes since they are the result of algorithm like $\mathcal{N}_{EKF}$ or the GPS. Nevertheless,

the matrices Q and R can still be used as tuning knobs in the design of the linear observer (see Section 8.1 for details), i.e. a larger Q with respect to R will yield an observer that relies more on the measured output and carries out more aggressive corrections of the estimated states and vice versa for a larger R will yield an observer that relies more on the state-space model and input measurements to predict the next states.

Using the state-space model from Equations (7.3-7.4), input and measurement $\vec{\ddot{p}}_G(k)$ and $\vec{p}_G(k)$, the steady-state Kalman filter is then obtained by the following (see [Welch and Bishop, 2001], [Bar-Shalom et al., 1993]):

Time Update:

$$\begin{bmatrix} \vec{\hat{p}}_G^-(k) \\ \vec{\dot{\hat{p}}}_G^-(k) \end{bmatrix} = A \begin{bmatrix} \vec{\hat{p}}_G(k-1) \\ \vec{\dot{\hat{p}}}_G(k-1) \end{bmatrix} + B\vec{\ddot{p}}_G(k). \tag{7.5}$$

Measurement Update:

$$\begin{bmatrix} \vec{\hat{p}}_G(k) \\ \vec{\dot{\hat{p}}}_G(k) \end{bmatrix} = \begin{bmatrix} \vec{\hat{p}}_G^-(k) \\ \vec{\dot{\hat{p}}}_G^-(k) \end{bmatrix} + K \left(\vec{p}_G(k) - C \begin{bmatrix} \vec{\hat{p}}_G^-(k) \\ \vec{\dot{\hat{p}}}_G^-(k) \end{bmatrix} \right), \tag{7.6}$$

where $\vec{\hat{p}}_G^-(k)$ and $\vec{\dot{\hat{p}}}_G^-(k)$ are the a-priori state estimates at time step k given the knowledge of the process prior to step k; while $\vec{\hat{p}}_G(k)$ and $\vec{\dot{\hat{p}}}_G(k)$ are the a-posteriori state estimates at time step k given the knowledge of the measurement $\vec{p}_G(k)$. The steady-state Kalman gain K could be computed off-line as:

$$K = AP_\infty C^T (CP_\infty C^T + R)^{-1}, \tag{7.7}$$

where P_∞ satisfies the following Algebraic Riccati Equation:

$$P_\infty = AP_\infty A^T - AP_\infty C^T(CP_\infty C^T + R)^{-1}CP_\infty A^T + Q. \qquad (7.8)$$

Both the steady-state gain K and the steady-state covariance matrix P_∞ were computed using the Matlab function kalmd() before the Kalman filter was implemented in real-time via an xPC Real-Time Target Machine (see Appendix B).

Next, we will introduce three different approaches considered in this project, see Figure 7.1. All three of them use the above mentioned procedure, $\hat{f}_1$ Equation (7.2), to obtain $\vec{\hat{p}}_G(k)$. In addition, all three approaches also use the described Kalman Filer, Equation (7.5) - (7.6), to estimate the kite position and velocity. However, each approach uses different strategies and sensors to obtain the position measurement $\vec{\tilde{p}}(k)$.

7.2. First approach: GPS and barometer

We design a first observer, (see Figure 7.1(a)), by using the GPS to obtain the measurements $\tilde{p}_X, \tilde{p}_Y$ and the barometer for the measurement $\tilde{p}_Z$. Hence as mentioned before in Section 6.4, the two sensor together provide the position measurement $\vec{\tilde{p}}_G(k)$ to be used with the Kalman filter $\mathcal{L}_{KF}$. While the GPS can provide the measurement for the altitude, the related error ($\pm$50 m in our setup) is too large to be used in the considered application. The barometer, on the other hand, is quite accurate provided that an initial tuning procedure is carried out for each test to remove the initial bias induced by the changing weather conditions.

Since the accelerometers have a sampling frequency of 50 Hz, while the GPS provides measurements at a rate of 4 Hz (as mentioned in Section 6.4), we cannot use the Kalman filter directly due to the usage of different sensor in this application. Thus the multi-rate Kalman filter will be used to solve this sensors fusion problem. The multi-rate Kalman filter consists in performing the Time Update (Equation (7.5)) at each sampling time of $T_s = 0.02$ s, and performs the Measurement Update (Equation (7.6)) at every 0.25 s, i.e. when the new GPS measurement is available. Moreover, since the GPS receiver requires some time to perform its computations, a time-varying delay $\tau(t)$, is introduced, typically 100-200 ms, up and up to 1 s for few samples. However, this delay can be measured by using the time stamps of the GPS signal, provided by the employed IMU, and of the other sensors. Using the measurement of the delay, the GPS signal can be synchronized with the other measurements.

7.3. Second approach: GPS and barometer with geometric correction

In the second approach the filter design employs the same sensors as the first approach but it is also carries out a correction of the (X, Y) components of the measured position given by the GPS, The correction is done by projecting the measured position back onto the sphere of radius r, see Figure 6.3(b). Such a correction is based on two observations:

1. The kite motion is constrained on such a sphere of known radius r

because of the fixed line length.

2. The measurements $\tilde{p}_X$ and $\tilde{p}_Y$ given by the GPS are less reliable than the barometer measurement $\tilde{p}_Z$.

Given $\tilde{Z}$, by using Equation 6.7(d), we can then compute a measure $\tilde{\theta}$ of angle θ. Moreover, we can correct $\tilde{p}_X$ and $\tilde{p}_Y$ by re-scaling them in order to match with the line length r projected onto (X, Y), i.e. $r\cos(\tilde{\theta})$ (compare Equation (6.1) and Figure 6.3(b)):

$$\tilde{p}_{X,c} = \tilde{p}_X \frac{r\cos(\tilde{\theta})}{\sqrt{\tilde{p}_X^2 + \tilde{p}_Y^2}}, \tag{7.9a}$$

$$\tilde{p}_{Y,c} = \tilde{p}_Y \frac{r\cos(\tilde{\theta})}{\sqrt{\tilde{p}_X^2 + \tilde{p}_Y^2}}. \tag{7.9b}$$

By using the vector $\vec{\tilde{p}}_{G,c} = [\tilde{p}_{X,c}, \tilde{p}_{Y,c}, \tilde{p}_Z]^T$, it will be used as position measurement for the multi-rate Kalman filter. This second approach is shown in Figure 7.1(b), where the correction Equations (7.9a-b) is indicated as $\mathcal{N}_\theta$.

7.4. Third approach: Line angle sensor

The third observer we consider, see Figure 7.1(c), employs the line angle measurement unit mention in Section 6.4 to obtain the position measurement. Hence, in this approach we move away from using the GPS and

barometer measurements. By using Equation (6.7d) :

$$\tilde{p}_X(k) = r\cos(\tilde{\theta}(k))\cos(\tilde{\phi}(k)), \tag{7.10a}$$

$$\tilde{p}_Y(k) = r\cos(\tilde{\theta}(k))\sin(\tilde{\phi}(k)), \tag{7.10b}$$

$$\tilde{p}_Z(k) = r\sin(\tilde{\theta}(k)), \tag{7.10c}$$

where $\tilde{\theta}(k)$ and $\tilde{\phi}(k)$ are the measurements provided at 50 Hz by the line angle encoders. Due to the use of incremental encoders, the measurement noise on $\tilde{\theta}(k)$ and $\tilde{\phi}(k)$ is basically given just by the quantization error resulting from the encoders' resolution.

7.5. Remark on Three Approaches

The three approaches presented above differ by two main aspects:

1. The use of different sensors for the position measurements, which differentiate approach 1 and approach 2 (both use GPS and barometer) from approach 3 (use line angle sensors).

2. The inclusion of the know kinematic constraint given by the cable lines, that forces the kite to move on a sphere of radius r.

The latter aspect is not considered in approach 1, while it is accounted for in approach 2, where we re-project the measured kite position via Equation (7.9) on the sphere of radius r before performing the Measurement Update in $\mathcal{L}_{KF}$.

Furthermore, approach 3 directly computes the kite position from the spherical coordinates via Equations (7.10a-c). Hence, approach 3 automatically

obtains a measured position that lies on the sphere. It can be noted that we do not include the tether constraint directly in the Kinematic Model Equations (6.7), rather we enforce it on the measured output $\vec{\tilde{p}}_G$. Adding the constraint to the model would still yield an exact model (i.e. without model uncertainty), as least as long as the straight line assumption holds. However, the filtering algorithm would then have to take such constraint into account, hence the standard linear techniques like the Kalman filter could not be used anymore as nonlinear elements would be introduced into the model.

In addition, the ad-hoc modifications would have to be implemented. We tried one such approach by projecting the state on the constraint for the Time Update before performing the Measurement Update. This strategy did not yield any advantage with respect to the algorithms presented above for approaches 2 and 3, nor did a combination of the two techniques, i.e. projecting the state on the constraint in the Time Update step and then use projected measurements for the Measurement Update.

In summary, the inclusion of the tether constraint in the model used in $\mathcal{L}_{KF}$ does not yield any advantage w.r.t. enforcing the constraint on the position measurements and using a "free particle" linear model in $\mathcal{L}_{KF}$. The latter approach, which we used in this project, has the advantage of simplicity and well-understood theory for the stability of the estimation error dynamics.

7.6. Kite velocity angle filter

In order to obtain an estimate of the kite velocity angle γ, see Section 6.2 (Equation (6.6)), we first used Equation (6.7d) to obtain the angles $\hat{\theta}$ and $\hat{\phi}$ from the filtered position $\vec{\hat{p}}_G$. The latter is obtained, together with the estimated velocity $\vec{\dot{\hat{p}}}_G$, using one of the observers presented in the previous sections. The estimate of rotation matrix $R_{G\to L}(\hat{\theta}, \hat{\phi})$ could then be computed by using Equation (6.3), and an estimate of the velocity vector in the L frame $\vec{\dot{\hat{p}}}_L$ can be computed as:

$$\vec{\dot{\hat{p}}}_L = R_{G\to L}(\hat{\theta}, \hat{\phi})\vec{\dot{\hat{p}}}_G. \tag{7.11}$$

Using $\vec{\dot{\hat{p}}}_L$, an unfiltered variable $\tilde{\gamma} \simeq \gamma$ can be computed using definition (6.6) as:

$$\tilde{\gamma} = \arctan_2(\dot{\hat{p}}_{L_E}, \dot{\hat{p}}_{L_N}). \tag{7.12}$$

Next, we employ a standard Luenberger observer (introduced in Section 5.6) based on the following state-space model:

$$\begin{bmatrix} \gamma(k+1) \\ \dot{\gamma}(k+1) \end{bmatrix} = \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \gamma(k) \\ \dot{\gamma}(k) \end{bmatrix}. \tag{7.13}$$

In particular, the state-space equation of the Luenberger observer are:

$$\begin{bmatrix} \hat{\gamma}(k+1) \\ \dot{\hat{\gamma}}(k+1) \end{bmatrix} = \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \hat{\gamma}(k) \\ \dot{\hat{\gamma}}(k) \end{bmatrix} + K_\gamma(\tilde{\gamma}(k) - \hat{\gamma}(k)). \tag{7.14}$$

Hence, the observer yields a filtered estimate of the velocity angle $\hat{\gamma}$, as well as the estimate of the rate of the velocity angle $\dot{\hat{\gamma}}$. The estimate of the

velocity angle $\hat{\gamma}$ is required in order to have a smooth signal, suitable to be used as feedback control; while $\hat{\dot{\gamma}}$ can be also used as a feedback variable and to study the turning dynamics of the wing in response to a steering deviation, (see [L. Fagiano and Khammash, 2013] for more details on the automatic control design, which was another part of the main project by Dr. Lorenzo Fagiano). The only tuning parameter for the Luenberger observer is its static gain $K_\gamma \in \mathbb{R}^2$.

Clearly, the three different approaches presented above to estimate the kite position and velocity yield estimates of γ with different accuracy. In the following Section, we discuss the performance obtained with the considered filters in experimental tests.

8. Tuning Guidelines and Experimental Results

We designed three different observers for $\vec{\hat{p}}_G$ as described in Section 7.2-7.4, and consequently three observers for $\hat{\gamma}$. Throughout this chapter, we use the superscripts ^{1},^{2} ,^{3} to indicate the filtered position obtained with the approaches of Section 7.2-7.4 respectively. We employed different kites in the experiments in order to demonstrate that the proposed approach is independent from system's features like wing area, mass, efficiency, etc. In particular, we used three different Airush One® inflatable power kites, see [AIRUSH, 2013], with 12 m^2, 9 m^2 and 6 m^2 area. The length of the lines still remain as $r = 30$ m while we tested all three different kites.

Before presenting the results of the experimental tests, we provide some considerations on the tuning of the algorithms described in Chapter 7.

8.1. Tuning guidelines

For a chosen sampling time (i.e. $T_s = 0.02$ s in our case), the three parameters to be tuned in the presented approaches are: the matrices Q and R

that define the gain K of $\mathcal{L}_{KF}$ (see Equation (7.5 - 7.6)); and the gain K_γ of the Luenberger observer (see Equation (7.14)).

Tuning Kalman Filter

For the choice of Q and R, a rather simple analysis can be made in this specific case. In fact, in the inertial frame the position components p_X, p_y and p_Z are the outputs of three double integrators that are decoupled, see Equation (6.7a). Hence if Q and R are chosen to be diagonal, i.e.:

$$R = \begin{bmatrix} R_{11} & 0 & 0 \\ 0 & R_{22} & 0 \\ 0 & 0 & R_{33} \end{bmatrix} \quad Q = \begin{bmatrix} Q_{11} & 0 & 0 \\ 0 & Q_{22} & 0 \\ 0 & 0 & Q_{33} \end{bmatrix}, \tag{8.1}$$

the filter $\mathcal{L}_{KF}$ enjoys a similar structure, so that one can carry out the analysis by considering just one of the three directions, and the considerations hold also for the other two components. Let us consider for example the X direction, the related system has the acceleration $\ddot{p}_X$ as output, and the gain of the observer is determined by the ratio $\lambda = Q_{11}/R_{11}$, which is the only tuning parameter. We can then analyze the transfer function of $\mathcal{L}_{KF}$ from each of its two inputs, i.e. the acceleration $\hat{\ddot{p}}_X$ and the position measurement $\tilde{p}_X$, to its output, for example the filtered position $\hat{p}_X$:

$$\hat{p}_X(z) = F_u^{KF}(z)\hat{\ddot{p}}_X(z) + F_Y^{KF}(z)\tilde{p}_X(z), \tag{8.2}$$

where $x(z)$ is the z-transform of the discrete-time signal $x(k)$.

Figure 8.1 shows the Bode plot of the magnitude of the discrete-time transfer functions $F_u^{KF}(z)$ and $F_Y^{KF}(z)$ as a function of λ. Let us consider function

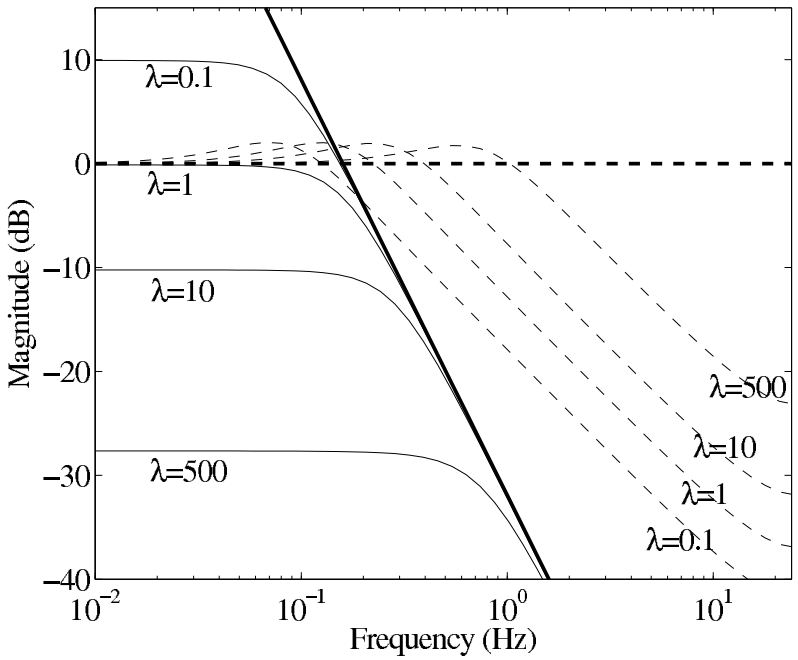


Figure 8.1.: Bode plot of the magnitude of the discrete-time transfer functions $F_u^{KF}(z) \doteq \hat{p}_X(z)/\hat{\tilde{p}}_X(z)$ (solid lines) and $F_Y^{KF}(z) \doteq \hat{p}_X(z)/\tilde{p}_X(z)$ (dashed lines) for different values of weighting ration $\lambda \doteq Q/R$, together with their respective asymptotic behaviors (thick lines).

F_y^{KF} first. For "low" frequencies, this function is close to 0 dB (showed as thick dashed line in Figure 8.1) irrespective of λ, as one would expect since the filter output $\hat{p}_X$ and the measurement $\tilde{p}_X$ correspond to the same physical quantity.

As for function F_u^{KF}, it can be noted that for "high" frequencies its magnitude is the same as that of a transfer function $\frac{T_s^2}{(z-1)^2}$ (showed as thick solid line in Figure 8.1), irrespective of λ. This is also expected, since the latter function corresponds to a double integrator and indeed the position p_X is obtained (less the initial conditions) by integrating twice the acceleration (i.e. the input of F_u^{KF}). The frequency ranges where these equivalences

hold true depend on λ: the smaller this value is, the larger the frequency range where F_u^{KF} corresponds to a double integrator and the smaller the range where $F_u^{KF} \approx 1$, and vice-versa.

A similar analysis can be done for the transfer functions from the two inputs to the estimated velocity $\hat{\dot{p}}_X$. The considerations above, combined with the characteristics of the employed sensors, provide quite intuitive guidelines on how to tune λ (hence Q and R). If the sensors used to measure the position have good bandwidth and low high-frequency noise, a higher value of λ should be chosen, hence relying on the position sensors for a larger range of low frequencies, and on the integration of the acceleration for higher frequencies. This is the case of the line angle sensor in approach 3, for which we set $\lambda = 500$ (see Figure 8.1).

If, on the contrary, the employed position sensor has poor dynamic performance, a lower value of λ can be used, in order to try to rely on the integration of the acceleration also for low and mid-range frequencies. This is the case of the GPS of approaches 1 and 2, for which we chose $\lambda = 10$. Clearly, these guidelines do not have an absolute validity, since they have to be applied by considering the relative quality of the employed sensors.

Tuning Luenberger observer

The observer from Equation (7.14) is a dynamical system with one input, i.e. the unfiltered velocity angle $\tilde{\gamma}$ from Equation (7.13), and two outputs, i.e. the filtered velocity angle $\hat{\gamma}$ and it derivative $\hat{\dot{\gamma}}$. By changing the gain K_γ, different bandwidths and zero-pole locations of the transfer functions

between the input and each of the outputs can be obtained, see Figure 8.2.

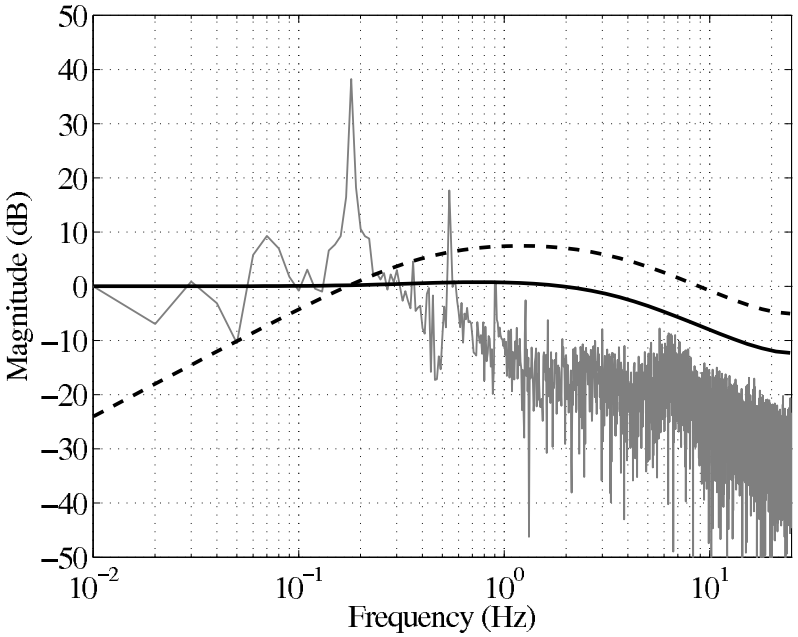


Figure 8.2.: Bode plot of the magnitude of the discrete-time transfer functions $F_{y_1}^{LO}(z) \doteq \hat{\gamma}(z)/\tilde{\gamma}(z)$ (solid lines) and $F_{y_2}^{LO}(z) \doteq \hat{\dot{\gamma}}(z)/\tilde{\gamma}(z)$ (dashed lines) obtained with the observer gain $K_\gamma = [0.40.9]^T$, and magnitude of the fast Fourier Transform of the input signal $\tilde{\gamma}(z)$ (gray line).

In addition, Figure 8.2 shows the Bode plot of the magnitude of the transfer functions $F_{y_1}^{LO}(z) \doteq \hat{\gamma}(z)/\tilde{\gamma}(z)$ and $F_{y_2}^{LO}(z) \doteq \hat{\dot{\gamma}}(z)/\tilde{\gamma}(z)$, obtained with the gain $K_\gamma = [0.40.9]^T$ that we choose for our tests. Furthermore, Figure 8.2 also shows the magnitude of the Fast Fourier Transform of the signal $\tilde{\gamma}(z)$ (i.e. the input to the observer), computed from the experimental data collected during more than four hours of autonomous flight in crosswind conditions. From the latter, the main frequency component can be easily identified, at around 0.16 Hz, corresponding to the frequency of a full figure-eight path (whose period is typically around 5- 6 s in our experimental setup. Another peak at around 0.54 Hz can be also noted, as well as a third one at 0.9 Hz.

As a general guideline, the gain K_γ shall be chosen such that the important components of the input signal are not distorted in magnitude nor in phase. With the chosen tuning of the observer, the transfer function $F_{y_1}^{LO}(z)$ is such that the frequency components up to around 1.5 Hz pass the filter with little magnitude distortion and phase lag (see Figure 8.2, solid line). In other word, the Luenberger observer we designed is a low-pass filter with the cut-off frequency of 1.5 Hz.

As for function $F_{y_2}^{LO}(z)$, it can be noted that in a similar range of frequencies this filter behaves like the discrete-time derivative $\frac{z-1}{T_s}$, hence providing an estimate of the turning rate $\dot{\gamma}$ (Figure 8.2, dashed line). Both filter then attenuate signal components with frequencies $>$ 2-3 Hz.

8.2. Experimental results: first and second approaches

The third approach result turned out to be the most accurate one thanks to the use of the line angle sensor, which practically yields very accurate position readings. Hence, in this section we take the estimates $\vec{\hat{p}}_G^3$ and $\tilde{\gamma}^3$, obtained with the third approach, as "true values" and we evaluate the performance of the first two approaches with respect to the third one. Referring to Section 8.1, we employed $\lambda = 10$ for all three motion directions, and $K_\gamma = [0.4, 0.9]^T$. An example of the courses of the position components along the X and Z axes are shown in Figures 8.3-8.4.

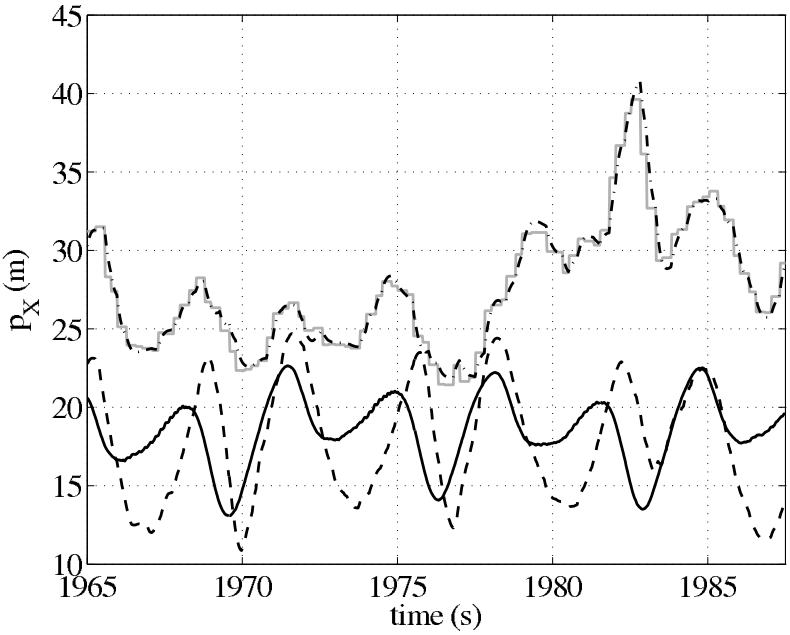


Figure 8.3.: Estimates of the position along the X axis, $p_X(t)$, obtained with the first (dashed-dotted), second (dashed) and third (solid) approaches, and raw GPS data (gray)

It can be noted that the error between $\hat{p}_X^1, \hat{p}_X^2$ and $\hat{p}_X^3$ can be quite high (of the order of 10 m), with $\hat{p}_X^2$ being generally more accurate than $\hat{p}_X^1$. This shows that the correction using Equation (7.9) introduced in Section 7.3 indeed improves the estimate with respect to using the plain GPS reading (compare the dash-dotted and dashed lines in Figure 8.3). Moreover, both $\hat{p}_X^1$ and $\hat{p}_X^2$ are affected by some delay with respect to $\hat{p}_X^3$. Such a delay, together with the quite poor accuracy, give rise to significant errors in the γ estimates.

Similar results are obtained for the Z axis, while the estimates $\hat{p}_X^1$ and $\hat{p}_X^2$ result to be more accurate and they are not affected by delays, thanks to the use of the barometric altitude instead of the GPS, see Figure 8.4. We note that for the vertical position p_Z, the first two approaches are equivalent,

hence resulting in the same filtered values.

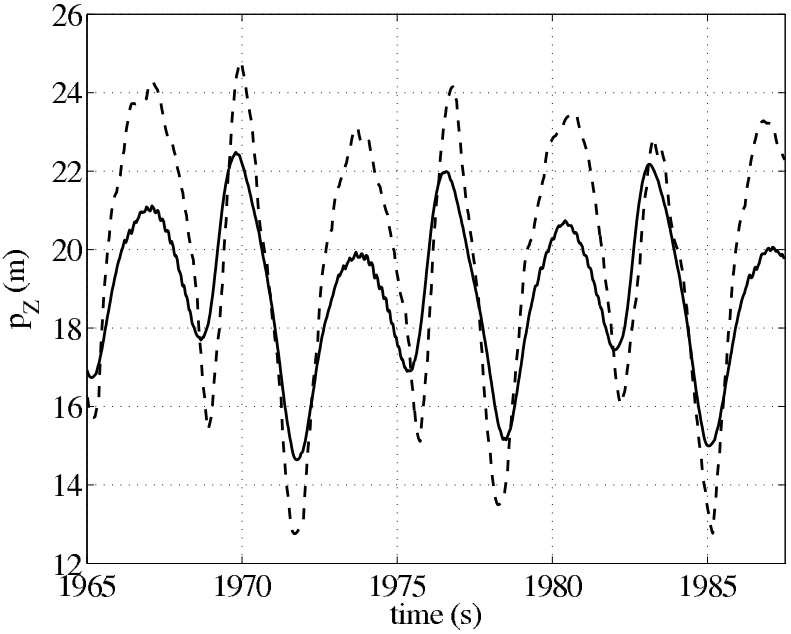


Figure 8.4.: Estimates of the position along the Z axis, $p_Z(t)$, obtained with either the first and second (dashed) or the third (solid) approach.

The considerations above are confirmed by the analysis of extensive experimental data. Table 8.1 shows the root mean square errors (RMSE) between the first two approaches and the third one, for all of the estimated variables, grouped in different ranges of wind speeds measured at 4m above the ground with the anemometer, as mentioned in Section 6.4. The results of Table 8.1 account for about 5 hours of operation at different wind speeds. Considering that a single "figure-eight" loop has a period of 5-6 seconds, the data in the table correspond to about 3300 complete "figure-eight" paths.

The results in the Table 8.1 also confirm that the second approach is generally better than the first one in estimating the wings position, especially at larger wind speeds (see Table 8.1 for 3 m/s and above), while the two ap-

Table 8.1.: Root mean square errors between the estimates obtained with either the first or second approach and the third one, for different wind speeds measured at 4 m above the ground.

Wind speed (m/s)	< 2	2-3	3-4	> 4
p_X^1 (m)	7.73	14.11	16.36	46.23
p_X^2 (m)	5.05	7.03	9.12	13.63
p_Y^1 (m)	10.32	12.81	15.40	24.53
p_Y^2 (m)	10.10	10.15	10.33	10.65
p_Z^1 (m)	1.68	2.98	4.27	7.70
p_Z^2 (m)	1.68	2.98	4.27	7.70
γ^1 (rad)	1.35	1.44	1.57	2.24
γ^2 (rad)	1.44	1.47	1.79	2.34

proaches give similar errors on the γ estimates, which deviate significantly from the third approach. Moreover, it can be noted that the accuracy of the first two approaches gets worse as the wind speed is higher. Since the kite velocites depends linearly on the wind speed, this results indicate that the GPS performance get worse as the wing movements get faster and changes of direction are more frequent. Indeed the provided GPS accuracy ($\pm$ 2.5 m in the (N, E) plane) is valid for steady state conditions only.

Finally, it can be noted that also the estimates of p_Z get generally worse with larger wind speed. The reason for this result is that the barometric altitude reading is influenced by the change of air pressure due to the increased wing speed and hence increased airflow on the IMU, which is tuned in static conditions. This phenomenon can be compensated by implementing a correction (e.g. based on the wings speed or line force), which is related to the airflow.

8.3. Experimental results: third approach

We now focus on the results obtained with the third approach. Referring to Section 8.1, we choose the tuning parameters as $\lambda = 500$ and $K_\gamma = [0.4, 0.9]^T$. We first show an example of the time courses of the filtered velocity angle $\hat{\gamma}$ and its rate $\hat{\dot{\gamma}}$, see Figure 8.5. It can be noted that the estimates are quite smooth, hence they are suitable to be used for feedback control, and that the derivative $\hat{\dot{\gamma}}$ has a rather small lag with respect to $\hat{\gamma}$. In order to assess the contribution provided by the inertial sensors to the estimated quantities, we show next, in Figure 8.6, the course of $\hat{\gamma}$ obtained either with the IMU (solid line) or without, i.e. considering $\vec{\hat{\dot{p}}}_G = 0$.

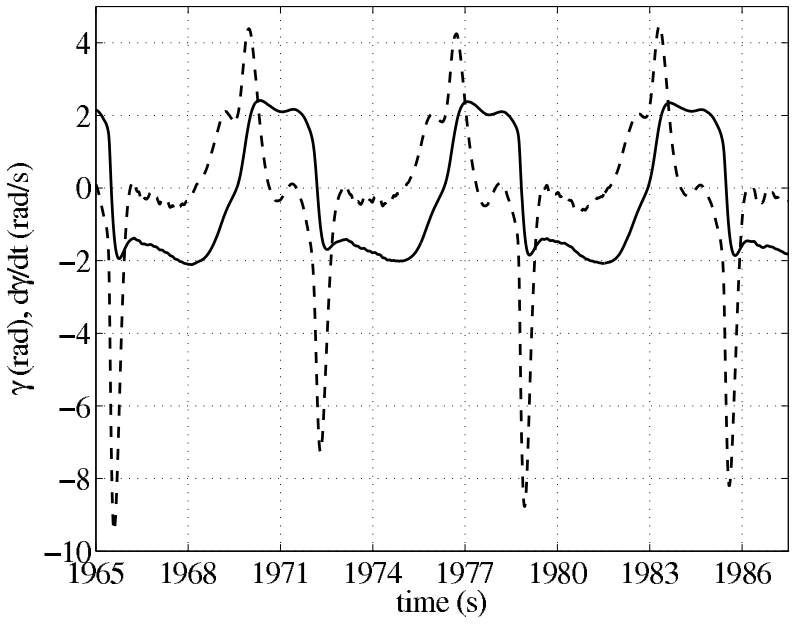


Figure 8.5.: Time courses of the filtered velocity angle $\hat{\gamma}$ (solid), and its derivative $\hat{\dot{\gamma}}$ (dash). Design parameters: $\lambda = 500, K_\gamma = [0.4, 0.9]^T$. Kite size: 12 m^2, wind speed: 2.3 m/s measured at 4 m above the ground

It can be noted that in the second case, a quite significant lag is present in the estimate. Although the control system can still operate satisfactorily in

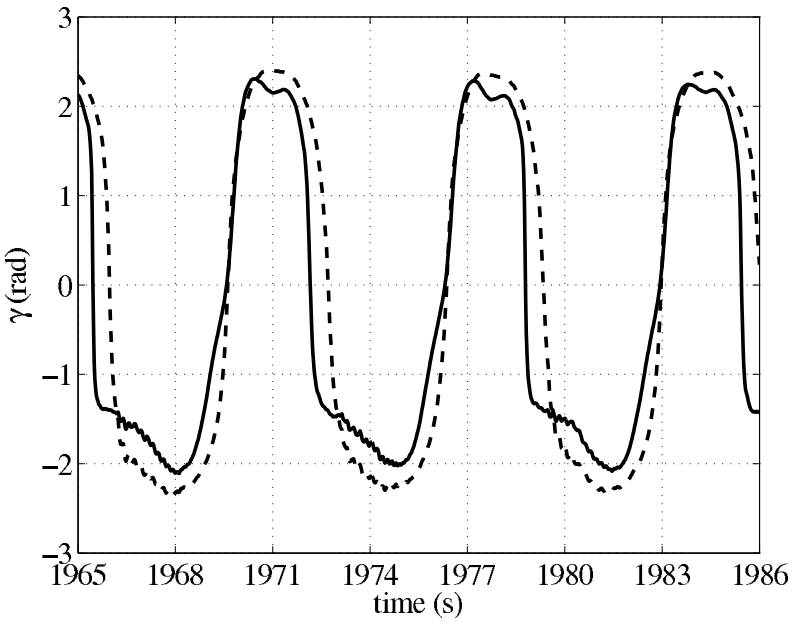


Figure 8.6.: Time courses of the filtered velocity angle $\hat{\gamma}$ obtained with the third approach with the IMU (solid), and without (dash). Design parameters: $\lambda = 500, K_\gamma = [0.4, 0.9]^T$. Kite size: 12 m^2, wind speed: 2.3 m/s measured at 4 m above the ground.

this case, its performance get worse. The same conclusion can be drawn also by analyzing Figures 8.7 - 8.8, which show the same flown path, together with its filtered position in the (ϕ, θ) plane and the estimated velocity angles, again obtained either with or without the IMU. The raw position data given by the line angle sensor are also shown in both figures.

By comparing Figures 8.7 - 8.8, it can be noted that the use of the IMU yields a better accuracy not only of the velocity angle estimate (compare the two figures e.g. for $\phi \simeq -0.75, \theta \simeq 0.7$), but also of the filtered position, as shown by the larger errors between the raw data and the filtered ones in Figure 8.7. The reason of such a difference lies in the fact that the inertial sensors of the IMU help to anticipate the kite movement, hence reducing the lag in the filtered variables.

In summary, while the third approach provides estimates that are good enough for feedback control without using the IMU, the fusion of the inertial sensors of the IMU with the line angle measurement system yields the best performance. In Figure 8.7, it can be noted that the estimated velocity angle is quite accurate, with the gradient being almost always tangent to the estimated wings flying path. The length of the arrows in Figure 8.7 is proportional to the wings speed with respect to the ground. Similar good results have been obtained with all the three different wings, and in different wind conditions (see Figures 8.9 - 8.11 for some examples), hence showing the robustness of the approach, deriving from the use of kinematic equations to design the filters. These features make the use of the proposed approach and of $\hat{\gamma}^3$ most suited for feedback control, see [project EISG, 2013] for a short movie clip concerned with autonomous flights carried out using such estimation algorithm.

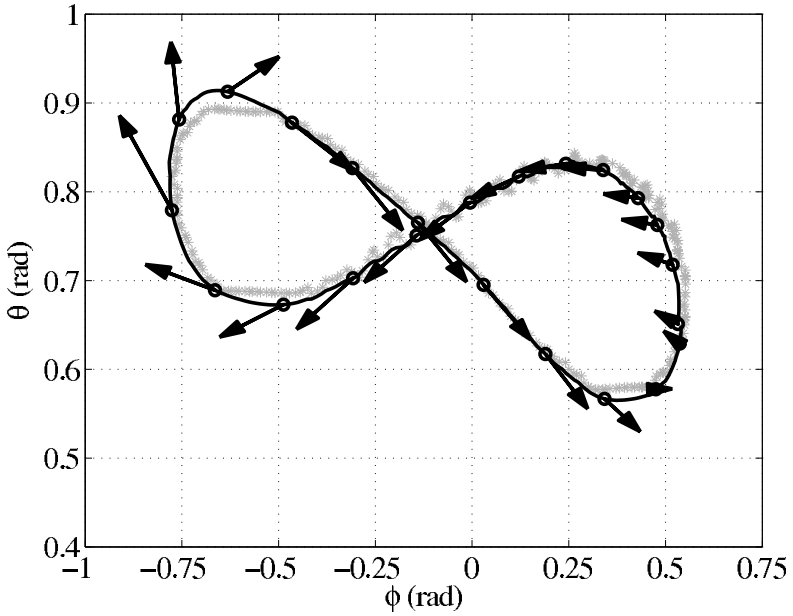


Figure 8.7.: Filtered position in the (ϕ, θ) plane obtained from the third approach with the inclusion of the IMU (solid line and '$\circ$') and related velocity angle estimates (arrows). The raw position data is shown with gray '$*$'. Design parameters: $\lambda = 500, K_\gamma = [0.4, 0.9]^T$. Kite size: 12 m^2, wind speed: 2.3 m/s measured at 4 m above the ground.

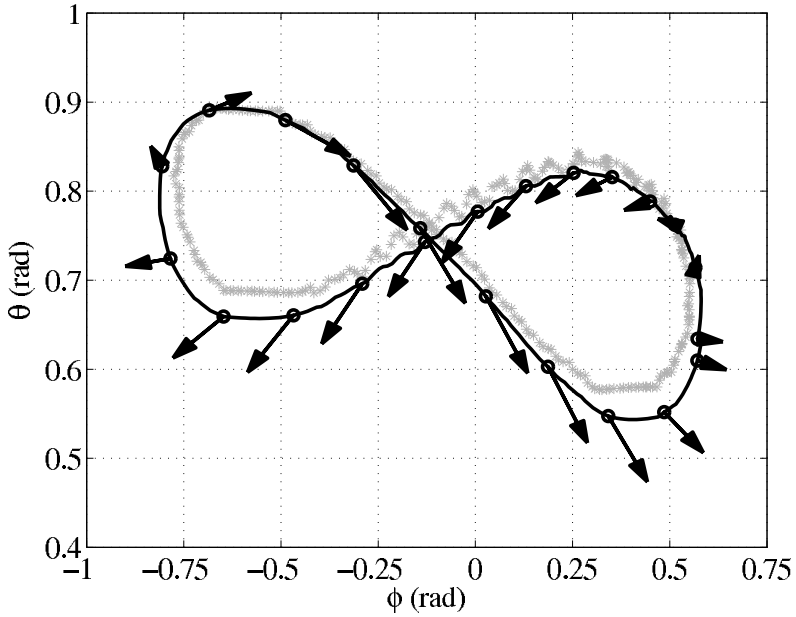
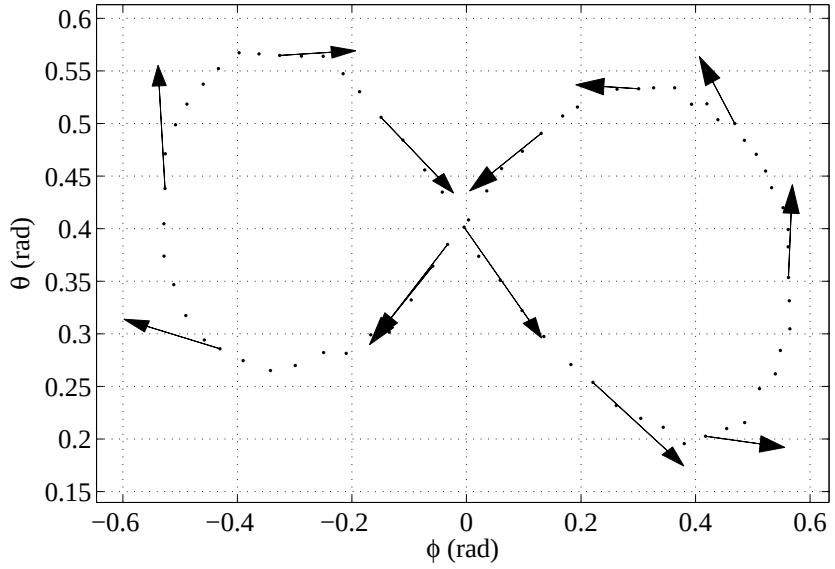
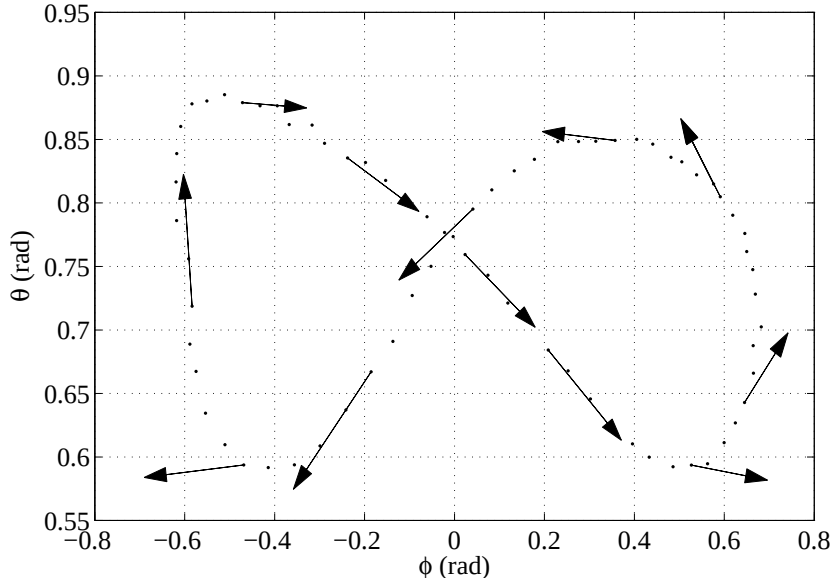


Figure 8.8.: Filtered position in the (ϕ, θ) plane obtained from the third approach without the inclusion of the IMU (solid line and '$\circ$') and related velocity angle estimates (arrows). The raw position data is shown with gray '$*$'. Design parameters: $\lambda = 500, K_\gamma = [0.4, 0.9]^T$. Kite size: 12 m^2, wind speed: 2.3 m/s measured at 4 m above the ground.

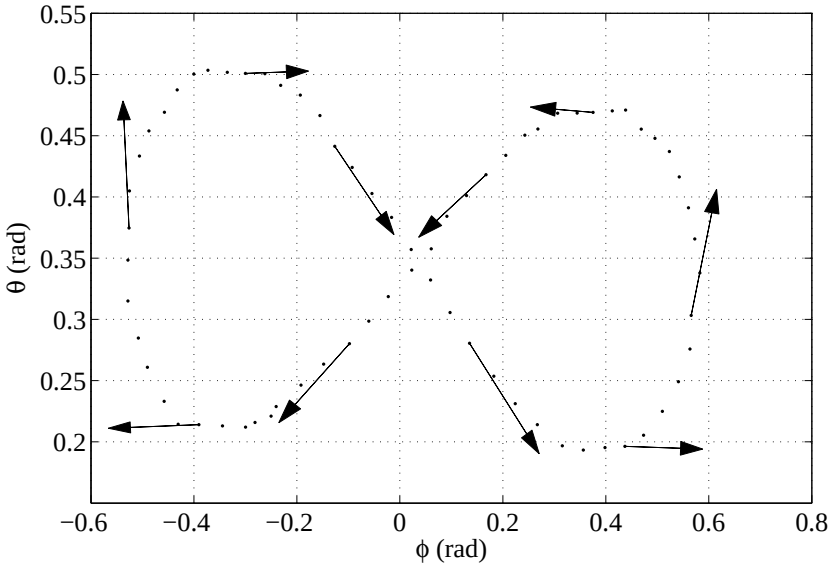


(a)

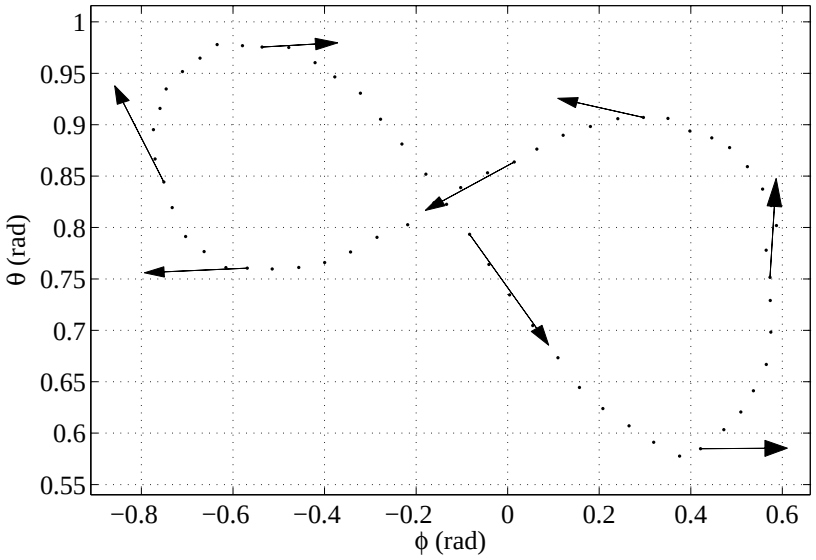


(b)

Figure 8.9.: Tracking of the kite paths (dotted) and related gradients (arrows) obtained with a 6 m^2 kite with about (a) 2 m/s and (b) 4 m/s wind speed meausred at 4 m above the ground.

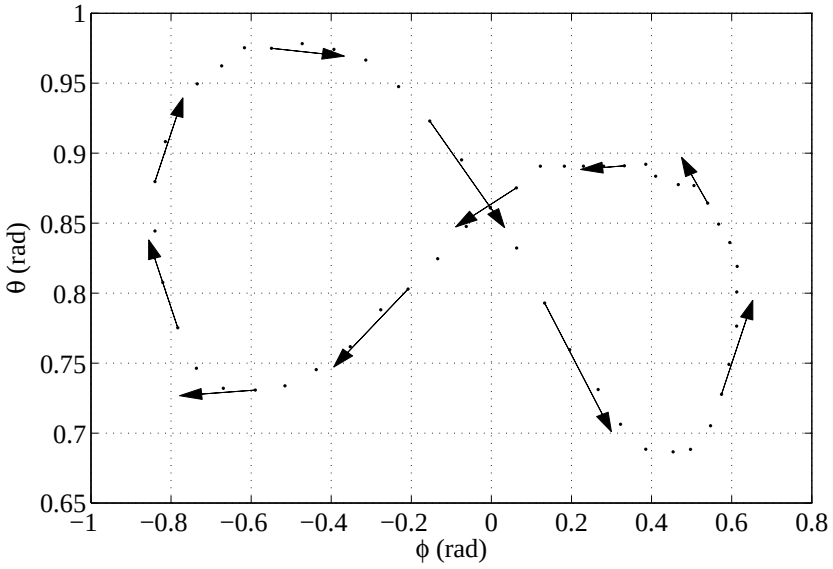


(a)

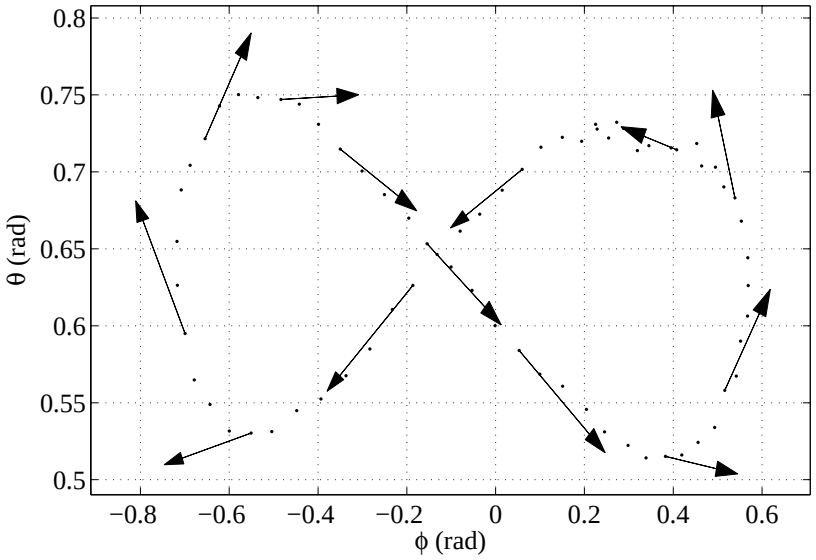


(b)

Figure 8.10.: Tracking of the kite paths (dotted) and related gradients (arrows) obtained with a 9 m^2 kite with about (a) 2 m/s and (b) 4 m/s wind speed meausred at 4 m above the ground.



(a)



(b)

Figure 8.11.: Tracking of the kite paths (dotted) and related gradients (arrows) obtained with a 12 m^2 kite with about (a) 2 m/s and (b) 4 m/s wind speed meausred at 4 m above the ground.

9. Discussion And Conclusions

We presented a study on the problem of sensor fusion for tethered kite to be used in airborne wind energy systems, focusing in particular on the crosswind flight conditions. We designed three algorithms to estimate the kite position and velocity angle, using different sensors, and we applied them in experimental tests.

The approaches and the results presented in this paper are directly significant for small-scale airborne wind energy systems when operating with relatively short lines. Moreover, as already mentioned, the use of a kinematic model renders the approaches suitable for virtually any size and type of wing. The main underlying assumption for the validity of the approaches is the presence of a straight tether linking the kite to the ground. As long as the tether lines remain straight and rigid, the main assumption would remain valid. In our experimental setup, with 30m long lines and force values ranging typically from 500N to 3000N in crosswind conditions, this was always the case. In such conditions, our results indicate that the approaches based on GPS are not usable in practice for the purpose of feedback control in crosswind motion. This holds in particular for the first approach for the position estimate, and for both the first and second approaches for the γ

estimate.

Hence, our results suggest that the use of accurate and fast position measurement devices, like the line angle sensor, are essential to obtain high accuracy when the lines are relatively short w.r.t. the wing speed. More specifically, the GPS is affected by poor accuracy (which can be partially improved by exploiting the kinematic constraint given by the tether) and by a (time-varying) delay, typically about 0.2s but up to 1s in some cases. In addition, a kite with only 30m of lines can change direction very quickly while flying in the "figure-eight" pattern. This ensure the GPS employed in this project, with its poor accuracy and delay, can never provide a reasonable measurement of the kite position. Finally, we also showed with experimental results that the fusion of inertial sensors with the position measurements yields the best performance in estimating the position, velocity and velocity angle of the wing, and also when an accurate line angle sensor is used.

This situation is likely to change when longer lines are used, so that the additional line drag and weight might reduce the traction force exerted by the wing, giving rise to line sagging. Clearly, the line sagging could occur with longer length of the lines, this might happen depends on the wing size, efficiency and wind speed. In such conditions, the accuracy of the line angle sensor decreases (perhaps becoming too inaccurate), hence making the third approach less effective, however on the other hand we expect the performance of the first and second approach to improve. In fact, it is reasonable to assume that the accuracy of the GPS would be the same in

absolute values also with longer tether length, hence yielding smaller errors in terms of angular positions, which are the ones that matter for feedback control in several approaches. This would hold particularly if the span of the flying path increases with the length of the lines (e.g. if the same trajectory in the (ϕ, θ) plane is kept), and indeed the capability of tracking the kite with sufficiently good accuracy might be a design constraint for the size of the flown trajectories.

With a longer flight path, the kite would certainly change its direction more slowly and gradually while flying in the "figure-eight" pattern. Thus allowing the GPS receiver more sufficient time to recover from its poor accuracy and delay. Moreover, the use of GPS sensors with higher performance (e.g. differential GPS) can certainly improve the accuracy obtained with the first two approaches, which then could be mixed with the third one. This represents an interesting line of research, that can be pursued only with a larger testing setup with full reel-out capabilities and electricity generation as described in Section 2.2.

Focusing again on the third approach, eventual line sag could be also detected and corrected by exploiting the barometric altitude reading and the measure of the forces acting on the lines, as well as other additional measurements like onboard airspeed, which is also related to the generated force: from our own experience and from discussions with several researchers working in the field of airborne wind energy, line sag is almost absent during crosswind flight if the line forces are large enough. This aspect provides further design and operation criteria for HAWP systems. The chosen length of

the tethers and reel-out speed should be matched with the characteristics of the wing (like size and efficiency). In addition, with the wind conditions, in order to make sure that the load on the lines is large enough (relative to their length) to allow good position measurements from the ground with a line angle sensor. This point provides a further reason to investigate the phenomenon of line sag, not only for the sake of deriving more accurate models for numerical simulations, but also for the purpose of estimation.

Lastly, we think that the use of larger and heavier wings would also yield better accuracy with the first two approaches, due to larger flight paths (whose radius increases approximately linearly with the wing span) and higher inertia that should lead to an improvement in the relative GPS accuracy.

A. IG-500N-Leaflet

IG-500N

IG-500 SERIES When size and performance matter, professionals trust the proven MEMS-based IG-500 series. After many years of success, the IG-500 Series is recognized for its high performance and reliability.

SBG SYSTEMS

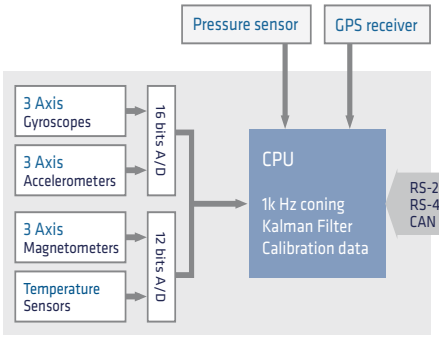
IG-500N - The Most Reliable Miniature INS/GPS

KEY FEATURES

- Accurate attitude even in high G maneuvers
- Automatic magnetic declination and local gravity
- Precise UTC referenced output and SyncOut signal
- NMEA / ASCII output for GPS drop in replacement

MINIATURE AND ROBUST

- Compact and lightweight 44 grams
- Low power design down to 800 mW
- Robust Aluminum enclosure



3D Orientation
- Euler
- Matrix
- Quaternion

3D Navigation
- 3D velocity
- 3D Position

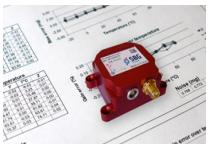
Sensors
- 3D Angular Rate
- 3D Acceleration
- 3D Magnetic field

Heave

The IG-500N is a miniature inertial navigation system with an embedded GPS. It is composed of:

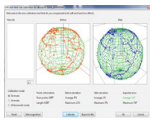
- a MEMS-based Inertial Measurement Unit (IMU) integrating 3 gyroscopes, 3 accelerometers, and 3 magnetometers,
- a 4 Hz, 50 channels GPS receiver,
- a barometric pressure sensor (altitude),
- and an on board Extended Kalman Filter (EKF).

THE MOST ADVANCED CALIBRATION TECHNIQUES



To ensure high data integrity, the IG-500N is calibrated from -40 to 85°C for bias, gain, linearity, misalignments, cross-axis and gyro-g. Every sensor is then intensively tested and shipped with its own calibration report.

COMPENSATION OF MAGNETIC DISTURBANCES



The magnetometer calibration tool compensates both soft and hard iron effects using a powerful algorithm. A calibration library is provided to be integrated in your system. You can easily calibrate your sensor in real conditions to obtain the most efficient compensation.

HIGH IMMUNITY TO VIBRATIONS

The IG-500N is especially reliable in vibrating environments. Each accelerometer is calibrated, and powerful algorithms have been designed to filter vibrations such as a 1 k Hz coning and a FIR filtering.

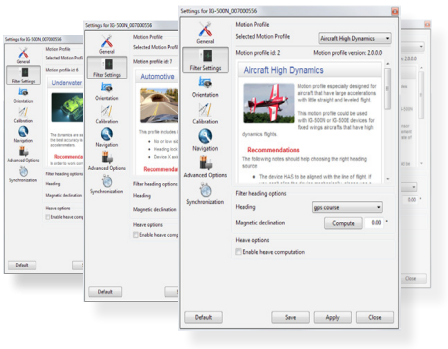
Performance adjusted to your application

MOTION PROFILES

Motion Profiles adjust automatically to:

- Kalman Filter parameters,
- Vibration level,
- Dynamic model,
- Magnetic disturbance immunity, etc.

In a few clicks, Motion Profiles tune your sensor to your application constraints.



Example: If you want to install the IG-500 sensor in an airplane, select the "Aircraft High Dynamics" motion profile and all parameters will be automatically adjusted.

UAV & PAYLOADS

This INS/GPS comes with a barometric pressure sensor, making the IG-500N a perfect all-in-one solution for UAV navigation and stabilization, as well as payload orientation and positioning.

PERFORMANCE SAILING

The IG-500N provides robust heading, 360° attitude, and GPS position. It distinguishes itself by automatically computing the magnetic declination, and the local gravity.

GYRO-STABILIZED CAMERA

Payload orientation & stabilization is more efficient thanks to IG-500N high update rate, low latency, and the provided true heading based on GPS and accelerations.

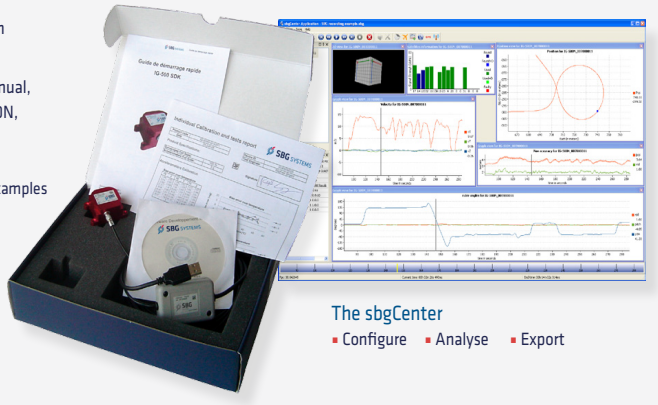
CAR MOTION ANALYSIS

The IG-500N is ready-to-use for chassis roll over detection, optimal trajectory determination, over and under steering characterization in high dynamics, and extreme temperature.

Development kit

The Development Kit which comes with your IG-500N sensors contains:

- A quick start guide and the user manual,
- The calibration report of your IG-500N,
- A USB converter cable,
- Useful software and tools:
  - A C library and some code sources examples
  - The Magnetometer Calibration Tool
  - LabView & Matlab plugins
  - The sbgCenter configuration & analysis software
  - The sbgUpdater that automatically alerts you and install the new software version.



The sbgCenter

- Configure
- Analyse
- Export

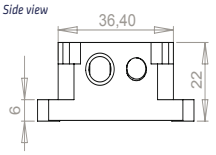
IG-500N - Specifications

PARAMETER	SPECIFICATIONS			REMARKS
Performance	**Aerospace**	**Ground**	**Marine / Subsea**	
Roll/Pitch (Dynamic)	1.0° RMS	0.8° RMS	0.35° RMS	Under good GPS availability
Heading (Dynamic)	1.0° RMS	0.5° RMS	1.0° RMS	Depends on heading aiding source
Resolution	< 0.05°	< 0.05°	< 0.05°	
Velocity (RMS)	< 0.1 m/s	< 0.1 m/s	< 0.1 m/s	Under good GPS availability
Position (SEP)	< 2 m	< 2 m	< 2 m	Under good GPS availability
Heave	-	-	10 cms or 10 %	Whichever is greater
Sensing range	360° in all axes, no mounting limitation			Solid state sensors
Inertial Sensors	**Accelerometers**	**Gyroscopes**	**Magnetometers**	
Measurement range	± 5 g	± 300 °/s	± 1.2 Gauss	Refer to sensors options table
Non-linearity	< 0.2 %	< 0.05 %	< 0.2 %	% of full scale
Initial bias error	± 5 mg	± 0.5 °/s	± 0.5 mGauss	Over temperature range
Bias in-run stability	± 0.06 mg	20 °/hr	-	Allan variance - constant temperature
Scale factor stability	< 0.1 %	< 0.05 %	< 0.5 %	Over temperature range
Noise density	0.25 mg/√ Hz	0.05 °/s/√ Hz	0.01 mG/√ Hz	
Alignment error	< 0.05°	< 0.05°	< 0.1°	
Bandwidth	250 Hz	240 Hz	500 Hz	1 k Hz gyroscopes coning integrals
Sampling rate	10,000 Hz	10,000 Hz	1,000 Hz	Advanced anti-aliasing FIR filters
GPS Receiver				
Receiver type	L1 frequency, C/A Code, 50 Channels, SBAS, 4 Hz			
Acquisition time	< 1.0 s / 29 s			Hot start / Cold start
Tracking sensitivity	-160 dB			
Pressure Sensor				
Resolution	2.5 Pa / 20 cms / < 1 feet			
Pressure accuracy	± 50 Pa / ± 150 Pa			Relative / Absolute
Sampling rate	50 Hz			
Communication				
Available data	Euler angles, quaternion, rotation matrix, velocity, position, heave, calibrated sensor data, delta angles, barometric data, device status, raw GPS data, UTC time reference, etc.			Each output can be enabled or disabled by the user. Output rate is user selectable
Output rate	100 Hz for orientation, velocity and position			500 Hz in IMU mode only
Serial interface	RS-232, RS-422, TTL 3.3V or USB Binary proprietary protocol and NMEA/ASCII			RS-422 only for S and O packages USB using an external adapter
CAN interface	CAN 2.0A/B up to 1 Mbit/s			Only available for S and O packages
Physical				
Dimensions OEM	27 x 30 x 14 mm, 1.1 x 1.2 x 0.6"			
Dimensions box	36 x 49 x 22 mm, 1.4 x 1.9 x 0.9"			B package
	36 x 49 x 25 mm, 1.4 x 1.9 x 1"			S package
Weight OEM	10 grams, 0.02 pounds			
Weight box	44 grams, 0.1 pounds			B package
	48 grams, 0.1 pounds			S package
Specified temperature	-40 to 85°C, -40 to 185°F			Non-condensing environment
Shock limit	1,000 *g* (Powered); 2,000 *g* (Unpowered)			Shocks can affect performance
Operating vibration	3 g RMS (20 Hz to 2 k Hz per MIL-STD 810G)			Valid for 18g accelerometers
Electrical				
Operating voltage	3.3 V to 30 V			
Power consumption	800 mW @ 5.0 V			High efficiency DC/DC converter
SyncOut, Trigger	Open drain pull-up voltage -0.3 to 25 V			Open drain, use a pull-up resistor
Start-up time	< 1 s			Valid data

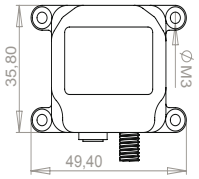
MECHANICAL DRAWING

All dimensions are in millimeters

Side view



Top view



PRODUCT CODE

▪: standard product options

IG-500N-G#A#P#-#

GYROSCOPES
2: 75 °/s
3: 150 °/s
4: 300 °/s ▪
5: 600 °/s
6: 1200 °/s

ACCELEROMETERS
1: 2 g 2: 5 g ▪ 3: 18 g

PROTOCOL MODE
1: RS-232 ▪
2: Serial TTL
3: CAN 2.0A/B
4: RS-422

PACKAGING
B: Box small size ▪
S: Box with SyncOut
O: OEM version

OEM VERSION

SBG SYSTEMS
www.sbg-systems.com

SBG Systems EMEA (Headquarters)
Phone: +33 1 80 88 45 00
E-mail: contact@sbg-systems.com

SBG Systems North America
Phone: +1 (773) 754 3272
E-mail: contact.usa@sbg-systems.com

B. xPC Real-Time Target Machine Spec

Performance Real-Time Target Machine

Hardware Specifications

Summary

Complete, assembled, and tested Speedgoat Performance real-time target machine including chassis, industrial mainboard, CPU, RAM, industrial-grade SATA flash device (hard disk options for data logging recommended), dedicated Intel® Gigabit Ethernet card, handles, rack mounting brackets, USB memory device, external USB floppy drive, and boot environment for xPC Target™ installed on main drive.

Includes Speedgoat tools and drivers software for xPC Target for current release and one release prior. Twelve months of free e-mail delivery of Speedgoat tools and drivers software for future releases of MathWorks software included. Support for older releases available upon request.

Chassis	
Enclosure	4U 19"-compatible aluminium chassis, front-accessible
Color	Silver powder-coated, natural aluminum
Size	Height: 177.8mm (4U), 193mm including stand-up feet, width: 440mm, 480mm including rack handles, depth: 360mm, 400mm including handles
Weight	12kg (excluding I/O modules, cables, and terminal boards)
Power supply	400W, 100-240V, 50-60Hz, fan-less, zero-noise
Fans	Two high-quality, low-noise fans at rear
Handles	Two for desktop use, two for rack installation

Main Board and CPU	for CPU 2.13 or 2.66	for CPU 3.16 or CPU 3.33
Processor	Intel Core 2 Duo 2.13GHz or Intel Core 2 Duo 2.66GHz (option)	Intel Core 2 Duo 3.16GHz (option) or Intel Core 2 Duo 3.33GHz (option)
Form factor	ATX	ATX
Chipset	Intel 82945G and ICH7, 1066 MHz front side bus	Intel Q35 and ICH9, 1333MHz front side bus
Bus	PCI, 32-bit/33MHz	PCI, 32-bit/33MHz
Memory	2048MB DDR2 RAM	2048MB DDR2 RAM
Drive support	IDE/SATA (flash or hard disk device)	SATA (flash or hard disk device)
Graphics	VGA, integrated in chipset	VGA, integrated in chipset
USB	6 x USB 2.0	8 x USB 2.0
Parallel Ports	AT bi-directional, EPP, ECP	AT bi-directional, EPP, ECP
Serial Ports	8 x RS232	4 x RS232
Keyboard and mouse	2 x PS/2	2 x PS/2
BIOS	American Megatrends® (AMI)	Phoenix® Award
No. of PCI slots	6 + 1 x PCI Express	4 + 2 x PCI Express + 1 x ISA

Drives	
Main drive	1 x 1GB flash device. Option: 4GB flash or 160GB hard disk device.
Floppy drive	3 ½“ 1.44 external USB floppy drive

Accessible components	
PCI slots for I/O modules	Five (three if ordered with the Intel Core 2 Duo 3.16GHz or 3.33GHz CPU)
Power inlet	AC 100-240V, 50/60Hz, at rear
Power switch	Rear panel
Seconday power switch	Front panel
Reset button	None (use secondary power switch)
Power LED	Front panel, combined with secondary power switch
VGA video connector	Front panel
USB	Five sockets on front panel. Intended use: xPC Target kernel and file transfer between host and target.
3 ½“ 1.44 floppy drive	External, USB. Intended use: classic method for xPC Target kernel and file transfer.
Ethernet	Front panel. Dedicated network card in one of the six PCI slots, single 10/100/1000GB/s. Intended use: primary host-target communication.
RS232	Front panel. Intended use: back-up host-target communication.
Keyboard and mouse	Front panel

Environment	
Temperature	0º to +50ºC (operating)
Humidity	15-80%, non-condensing

Software	
OS / RTOS	FreeDOS / xPC Target kernel, preinstalled on IDE CompactFlash Disc
Host PC	Utilities for kernel transfer, I/O drivers and Simulink® test models for your selected I/O modules

Bibliography

[A. Ilzhfer and Diehl, 2007] A. Ilzhfer, B. H. and Diehl, M. (2007). Nonlinear mpc of kites under varying wind conditions for a new class of large-scale wind power generators. *International Journal of Robust and Nonlinear Control*, 17:1590–1599.

[AIRUSH, 2013] AIRUSH (2013). http://www.airush.com/2013/.

[Ampyx, 2013] Ampyx (2013). http://www.ampyxpower.com/.

[Archer and Jacobson, 2005] Archer, C. L. and Jacobson, M. Z. (2005). Evaluation of global wind power. *J. Geophys*, 110(D12110).

[Bar-Shalom et al., 1993] Bar-Shalom, Y., Li, X. R., and Kirubarajan, T. (1993). *Estimation and Tracking: Principles, Techniques, and Software*. Artec House, Inc., New York, NY.

[Brown and Hwang, 1996] Brown, R. G. and Hwang, P. Y. C. (1996). *Introduction to Random Signals and Applied Kalman Filtering: with MATLAB Exercises and Solutions*. Wiley & Sons, New York USA.

[Coutsias and Romero, 2004] Coutsias, E. A. and Romero, L. (2004). The quaternions with an application to rigid body dynamics. Technical Report SAND2004-0153, Sandia National Laboraties.

[(EIA), 2011] (EIA), U. S. E. I. A. (2011). International energy outlook 2011. Technical report, WASHINGTON DC, USA.

[Erhard and Strauch, 2013] Erhard, M. and Strauch, H. (2013). Sensors and navigation algorithms for flight control of tethered kites. In *European Control Conference 2013*, pages 3132–3143, Zurich, Switzerland.

[Fagiano and Milanese, 2012] Fagiano, L. and Milanese, M. (2012). Airborne wind energy: an overview. In *American Control Conference 2012*, pages 3132–3143, Montreal, Canada.

[Fagiano et al., 2011] Fagiano, L., Milanese, M., and Piga, D. (2011). Optimization of airborne wind energy generators. *International Journal of Robust and Nonlinear Control*. In press, early view available. doi: 10.1002/rnc.1808.

[Fletcher and Roberts, 1979] Fletcher, C. A. J. and Roberts, B. W. (1979). Electricity generation from jetstreams winds. *Journal of Energy*, 3:241–249.

[Gopinath, 1971] Gopinath, B. (1971). *On the control of linear multiple input-output systems*. Bell Syst. Tech. J.

[Gurria, 2008] Gurria, A. (2008). Climate change mitigation what do we do. Technical report, OECD.

[Houska and Diehl, 2007] Houska, B. and Diehl, M. (2007). Optimal control for power generating kites. In 9^{th} *European Control Conference*, Kos, GR.

[(IEA), 2008] (IEA), I. E. A. (2008). World energy outlook 2008. Technical report, Paris, France: IEA PUBLICATIONS.

[Kailath and Hassibi, 2000] Kailath, T., S. A. H. and Hassibi, B. (2000). *Linear Estimation*. Prentice Hall, Upper Saddle River,NJ USA.

[Kalman, 1960] Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Transactions of the ASME–Journal of Basic Engineering*, 82(Series D):35–45.

[Kelly, 1994] Kelly, D. G. (1994). *Introduction to Probability*. Macmillan Publishing Company.

[Kitenergy, 2009] Kitenergy (2009). http://www.kitenergy.net/.

[Klumpp, 1976] Klumpp, A. (1976). Singularity-free extraction of a quaternion from a direction-cosine matrix. *Journal of Spacecraft*, 13(12):754–755.

[L. Fagiano and Khammash, 2013] L. Fagiano, A. Zgraggen, M. M. and Khammash, M. (2013). Automatic crosswind flight of tethered wings for airborne wind energy: modeling, control design and experimental results,. *arXiv*, 1301.1064. submitted to IEEE Trans. on Control Syst. Technology.

[LaValle, 2006] LaValle, S. M. (2006). *Planning Algorithms*. Cambridge University Press, Illinois, US.

[Lefferts and Shuster, 1982] Lefferts, E.J., F. M. and Shuster, M. (1982). Kalman filtering for spacecraft attitude estimation. *Journal of guidance, control, and dynamic*, 5(5):417–429.

[Loyd, 1980] Loyd, M. L. (1980). Crosswind kite power. *Journal of Energy*, 4(3):106–111.

[Luenberger, 1971] Luenberger, D. G. (1971). An introduction to observers. *IEEE Transactions on Automatic Control*, 16(6).

[M. Canale and Milanese, 2010] M. Canale, L. F. and Milanese, M. (2010). High altitude wind energy generation using controlled power kites. *IEEE Transactions on Control Systems Technology*, 18(2):279–293.

[Makani, 2013] Makani (2013). http://www.makanipower.com.

[Manalis, 1976] Manalis, M. S. (1976). Airborne windmills and communication aerostats. *Journal of Aircraft*, 13(7):543–544.

[Markley, 1994] Markley, F. (1994). New quaternion attitude estimation method. *Journal of guidance, control, and dynamic*, 17(2):311–317.

[Maybeck, 1974] Maybeck, P. S. (1974). *Applied Optimal EstimationKalman Filter Design and Implementation*. notes for a continuing education course offered by the Air Force Institute of Technology, Wright-Patterson AFB, Ohio, semiannually since December 1974.

[Mendel, 1995] Mendel, J. (1995). *Lessons in Estimation Theory and Signal Processing, Communications, and Control*. Prentice Hall, NJ USA.

[P. Williams and Ockels, 2008] P. Williams, B. L. and Ockels, W. (2008). Optimal crosswind towing and power generation with tethered kites. *Journal of guidance, control, and dynamics*, 31:81–93.

[project EISG, 2013] project EISG (2013). Autonomous flexible wings for high-altitude wind energy generation. *Santa Barbara USA*. experimental test movie, August 2012. Available online: http://lorenzofagiano.altervista.org/.

[R. Thresher and Veers, 2007] R. Thresher, M. R. and Veers, P. (2007). To capture the wind. *IEEE Power & Energy Magazine*, 5(6):34–46.

[Sabatini, 2011] Sabatini, A. M. (2011). Kalman-filter-based orientation determination using inertial/magnetic sensors: Observability analysis and performance evaluation. *Sensors*, 11(10):9182–9206.

[SBG Systems S.A.S., IG-500N, 2009] SBG Systems S.A.S., IG-500N (2009). http://www.sbg-systems.com/products/ig-500n.

[SkySails, 2013] SkySails (2013). http://www.skysails.info.

[T. Burton and Bossanyi, 2001] T. Burton, D. Sharpe, N. J. and Bossanyi, E. (2001). *Wind Energy: Handbook*. John Wiley and Sons.

[Welch and Bishop, 2001] Welch, G. and Bishop, G. (2001). *An Introduction to the Kalman Filter*. ACM, Inc., Chapel Hill, NC.

[Windlift, 2010] Windlift (2010). http://www.windlift.com/.

[Wittenburg, 1977] Wittenburg, J. (1977). *Dynamics of Systems of Rigid Bodies*. B.G. Teubner, Wiley, Stuttgart.