

UCLA

UCLA Electronic Theses and Dissertations

Title

A Method of Extremum Seeking Control Based on a Time Varying Kalman Filter and its Application to Formation Flight

Permalink

<https://escholarship.org/uc/item/6pq749v5>

Author

Ryan, John Jens

Publication Date

2012

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA

Los Angeles

**A Method of Extremum Seeking Control Based on a
Time Varying Kalman Filter
and its
Application to Formation Flight**

A dissertation submitted in partial satisfaction
of the requirements for the degree
Doctor of Philosophy in Aerospace Engineering

by

John J. Ryan

2012

ABSTRACT OF THE DISSERTATION

**A Method of Extremum Seeking Control Based on a
Time Varying Kalman Filter
and its
Application to Formation Flight**

by

John J. Ryan

Doctor of Philosophy in Aerospace Engineering

University of California, Los Angeles, 2012

Professor Jason L. Speyer, Chair

This dissertation presents a novel extremum seeking control method which combines a time-varying Kalman filter with a Newton Raphson algorithm. The Kalman filter is used to estimate the gradient and Hessian of a performance function. The resulting estimates are used in the Newton Raphson algorithm to guide the system to a local extremum of the performance function.

Convergence of the method to a local extremum is proven when the system is subject to noisy measurements. This is accomplished by showing that the output of the algorithm is a supermartingale. It is shown that the system will converge to an area around the extremum with a radius defined, in part, by the error covariance of the Kalman filter estimates.

The method is applied to two examples. The first utilizes a single independent parameter performance function. The second applies the method to the problem of formation flight for drag reduction. In the first example, two implementations of the method are examined. The first uses only gradient estimates the second uses both gradient and Hessian estimates. Both implementations show good convergence in the presence of noisy measurements.

The second example is of formation flight for drag reduction. The problem is described in some detail and includes an aerodynamic development of the drag-reduction phenomenon. The problem is explored with two simulations. The first uses coefficient of induced drag as its performance function and estimates the gradient and Hessian of the performance function. It

shows good convergence of the method. The second simulation first uses pitch angle and then aileron deflection as its performance function. It estimates the gradient but not the Hessian of the performance function. It also shows good convergence.

The dissertation of John J. Ryan is approved.

Lieven Vandenberghe

Robert T. M'Closkey,

James S. Gibson

Jason L. Speyer, Committee Chair

University of California, Los Angeles

2012

*This work would not have been possible without the
loving support of my lovely wife Janice.
Her encouragement and unwaivering patience
have carried me through.*

This work is dedicated to her and our beautiful children.

TABLE OF CONTENTS

1	Introduction	1
1.1	The mountain climber problem	1
1.2	Basic Interconnection of Extremum Seeking System	2
1.3	Thesis Overview	3
2	Literature Survey	5
2.1	Categorization of Extremum Seeking Literature	5
2.2	Classical Gradient Methods	6
2.3	Parametric Methods	10
2.4	Other Methods	11
3	Derivation of the Time Varying Kalman Filter	13
3.1	Introduction to the Time Varying Kalman Filter	13
3.2	Derivation of the Estimated Mean and Covariance	14
3.3	The Orthogonal Projection Lemma	15
3.4	Derivation of the Conditional Mean and Covariance	17
3.5	Summary of the Discrete Kalman Filter	22
4	Development of Extremum-Seeking Method	23
4.1	Problem Statement and Basic Notation	23
4.2	Overview of Method	24
4.3	Extremum location estimation	26
4.4	Gradient and Hessian Estimate	28
4.4.1	Persistent Excitation	32
4.4.2	Command Filter	32
4.4.3	System Initialization	33

4.5	Summary of the Extremum Seeking Control Method	33
4.6	Modifications to the Method	34
4.6.1	Sliding Window Modification	35
4.6.2	Gradient Only Modification	37
5	Convergence of Method	39
5.1	Convergence of the Kalman filter	39
5.2	Convergence of the extremum seeking algorithm	42
6	One Dimensional Example	45
6.1	Extremum-Seeking Control Example	45
7	Introduction to Formation Flight for Drag Reduction	56
7.1	Introduction to Formation Flight for Drag Reduction	56
7.2	A Review of Literature Related to Formation Flight for Drag Reduction	57
7.2.1	Biological Studies	58
7.2.2	Aerodynamics Studies	59
7.2.3	Operational Studies	59
7.2.4	Formation Flight with out Drag Reduction	60
7.2.5	Formation Flight for Drag Reduction	60
7.2.6	Extremum Seeking Applied to Formation Flight	61
7.2.7	Flight Experiments	62
7.3	Aerodynamics of Formation Flight	63
8	Application of Extremum Seeking Control to Formation Flight for Drag Reduction	72
8.1	Introduction to Extremum Seeking Control Examples	72
8.2	Vortex and Aircraft models	73
8.3	Inner Loop Control Design	78

8.4	Persistent Excitation Design	90
8.5	Kalman Filter Design	91
8.6	Simulation Results	92
8.7	Extension to a Four Aircraft Formation	94
9	A Second Application of Extremum Seeking Control to Formation Flight for Drag Reduction	107
9.1	Description of Implementation and System Design	107
9.2	Simulation Results	108
9.3	Comparison between the UCLA simulation results and the Boeing simulation results	121
10	Summary and Conclusions	122
Appendices		123
A	Derivation of the Linear Quadratic Regulator	124
A.1	A Dynamic Programming Derivation of the LQR	124
A.1.1	Last Stage	125
A.1.2	Factorization of a Quadratic Matrix Equation	126
A.1.3	Back to the Last Stage	126
A.1.4	Second to Last Stage	127
A.1.5	General Stage	127
A.2	Time Invariant Linear Quadratic Regulator	128
B	Derivation of the Linear Quadratic Tracker	129
C	Extremum Seeking Control Design Tutorial	132
C.1	Step 1: Simulation Development	132
C.2	Step 2: Inner Loop Control Design	133
C.3	Step 3: Persistent Excitation Design	133

C.4	Step 4: Measurement Rate Selection	134
C.5	Step 5: Selection of Number of Measurement	135
C.6	Step 6: Kalman Filter Construction and Tuning	139
C.7	Step 7: Command Conditioning Design	139
C.8	Step 8: Design System Initialization	140
C.9	Step 9: Simulate response	140
C.10	Design Process Summary	141
C.11	Special design considerations	141
C.11.1	Inherent persistent excitation	141
C.11.2	Non-static performance function	142
C.11.3	Slope seeking problems	142
References	144

LIST OF FIGURES

1.1	A Mountain Climber in a Thick Bank of Fog	2
1.2	Basic extremum-seeking method interconnection	3
2.1	Classical gradient method interconnection	7
4.1	Extremum seeking control method interconnection.	25
4.2	Algorithm Interconnection including the Newton-Raphson scheme	27
4.3	Algorithm Interconnection including the Kalman Filter.	31
4.4	Algorithm Interconnection including the command filter	33
4.5	Depiction of sliding window implementation	35
6.1	Frequency response of one closed loop plant	46
6.2	Step response of the closed loop plant	47
6.3	Single parameter system position versus time	49
6.4	Difference between the error covariance and the square of the gradient estimate . .	50
6.5	One parameter performance function and measurements	51
6.6	Estimated and true gradient versus time	52
6.7	Single parameter system position versus time	53
6.8	One parameter performance function and measurements	54
6.9	Estimated and true gradient and Hessian versus time	55
7.1	Airflow around a wing.	64
7.2	Geometry of the Horseshoe Vortex	65
7.3	Geometry of upwash calculation.	67
7.4	Upwash in the plane of a wing at a distance b aft of the wing.	68
7.5	Downwash and lift of a wing	69
7.6	A trailing aircraft wing in the upwash field of a leading aircraft wing.	69

7.7	The upwash induced by a wing-tip vortex impinging on a trailing aircraft wing.	70
8.1	Induced drag coefficient surface	74
8.2	Induced aileron deflection surface	75
8.3	Eigenvalues of the open-loop plant model.	77
8.4	Inner loop control interconnection	83
8.5	Trace of plant eigenvalues	85
8.6	Step response of closed-loop system longitudinal axis ($\delta h, \theta, u, w$)	86
8.7	Step response of closed-loop system longitudinal axis (q, δ_T, δ_E)	87
8.8	Step response of closed-loop system lateral-directional axis ($\delta y, \phi, \psi, w$)	88
8.9	Step response of closed-loop system lateral-directional axis (p, r, δ_R, δ_A)	89
8.10	Geometry of the persistent excitation signal used in the formation flight example.	90
8.11	Aircraft path along induced drag coefficient performance function.	93
8.12	Ratio of induced drag coefficient to induced drag coefficient when not under the influence of a vortex vs. time.	94
8.13	Gradient estimates and true gradient	95
8.14	Hessian estimates and true Hessian	96
8.15	Vertical and lateral position errors.	97
8.16	Aircraft Euler angles versus time.	98
8.17	Aircraft surface deflections versus time.	99
8.18	Induced drag coefficient contour map	100
8.19	Four Aircraft Echelon Formation.	101
8.20	Vertical Response of a Four Aircraft Formation.	102
8.21	Lateral Response of a Four Aircraft Formation.	103
8.22	Path of Second Aircraft along induced drag coefficient performance function.	104
8.23	Path of Third Aircraft along induced drag coefficient performance function.	105
8.24	Path of Fourth Aircraft along induced drag coefficient performance function.	106

9.1	Aircraft path following the pitch angle performance function displayed over the induced drag contour map	109
9.2	Aircraft path over pitch angle performance function near the extremum	110
9.3	Time history of aircraft cross-track, altitude, and pitch angle during formation flight	111
9.4	Kalman filter gradient estimates of the pitch-angle performance function	112
9.5	Standard Deviation of the Kalman filter gradient estimates of the pitch-angle performance function	113
9.6	Aircraft path following the pitch angle performance function on top of the induced drag contour map	114
9.7	Aircraft path following the aileron deflection performance function displayed over the induced drag contour map	115
9.8	Aircraft path over pitch angle performance function near the extremum	116
9.9	Time history of aircraft cross-track, altitude, and pitch angle during formation flight	117
9.10	Kalman filter gradient estimates of the pitch-angle performance function	118
9.11	Standard Deviation of the Kalman filter gradient estimates of the pitch-angle performance function	119
9.12	Aircraft path over pitch angle performance function	120
B.1	Step-Tracking Control Interconnection	129
B.2	Regulator Closed Loop Interconnection	130
C.1	Illustration of various iteration rates for a noisy signal	136
C.2	Illustration of poor persistent excitation and iteration rate combinations	137
C.3	Illustration of good persistent excitation and iteration rate combinations	138

LIST OF TABLES

8.1	Formation Flight Example Open-Loop Plant States	75
8.2	Formation Flight Example Open-Loop Plant Inputs	76
8.3	Formation Flight Example Eigenvalues of the Open-Loop Plant Model.	76
8.4	Formation flight example closed-loop system inputs	79
8.5	Formation flight example closed-Loop system outputs	80
8.6	Innerloop controller gain values	82
8.7	Eigenvalues of the closed-loop plant model.	84

ACKNOWLEDGMENTS

There are many whom I thank for making this work possible and for providing encouragement throughout the process.

I thank Professor Speyer. His guidance and patience made it possible for me to pursue this work while continue work at the NASA Dryden Flight Research Center. I would also like to thank the other members of my committee: Prof. M'Closkey, Prof Gibson, and Prof. Vandenberghe.

I thank the many at the NASA Dryden Flight Research Center who provided encouragement, support, and many helpful discussions about formation flight. Special thanks to Curt Hanson, Joe Pahle, and Patrick Stoliker.

I also thank the NASA Dryden Flight Research Center Graduate Study Leave Program which funded part of the work.

VITA

1972	Born in Minnesota
1995	Bachelor of Arts in Physics, St. John's University, Collegeville, MN.
1999	Master of Science in Mechanical and Aerospace Engineering. University of Minnesota, Minneapolis, MN.
1999-present	Aerospace Engineer at the National Aeronautics and Space Administration, Dryden Flight Research Center.

PUBLICATIONS

Michael J. Allen, Jack Ryan, Curtis E. Hanson. String Stability of a Linear Formation Flight Control System, *AIAA Guidance, Navigation, and Control Conference*, Monterey, California, August 2002. Also published as NASA TM-2002-210733.

Gary Balas, Jack Ryan, JY Shin, W.L. Garrard. A New Technique for Design of Controllers for Turbofan Engines, *34th Joint Propulsion Conference*, OH, July 1998; AIAA-98-3751.

Curtis E. Hanson, Jack Ryan, Michael J. Allen, and Steve R. Jacobson. An Overview of Flight Test Results for a Formation Flight Autopilot, *AIAA Guidance, Navigation, and Control Conference*, Monterey, California, August 2002. Also published as NASA TM-2002-210729.

Jack Ryan, Curtis E. Hanson, Ken A. Norlin, and Michael J. Allen. Data Synchronization Discrepancies in a Formation Flight Control System, *Proceedings of SFTE 32nd Annual International Symposium* Seattle, Washington, Sept 2001. Also published as NASA TM-2001-210720,2001.

John J. Ryan, Jason L. Speyer, Peak-Seeking Control Using Gradient and Hessian Estimates, *Proceedings of the American Control Conference*, Baltimore, MD, June 2010.

CHAPTER 1

Introduction

Extremum-seeking control, also referred to as peak-seeking control, encompasses methods and algorithms which optimize unknown performance functions in real time. The performance functions are defined by parameters of physical dynamic systems which the control-designer does not have detailed a priori knowledge and which can only be measured with physical sensors.

In this dissertation a novel extremum-seeking control scheme is presented. The scheme utilizes a time varying Kalman filter to estimate the gradient and Hessian of the performance function. To the author's knowledge, a time varying Kalman filter has not been applied to extremum-seeking control prior to this work. Use of a time varying Kalman filter provides flexibility to the design approach; the vast library of practical Kalman filter applications and variations can be utilized when designing an extremum-seeking controller. The scheme also provides large flexibility in the design of the associated persistent excitation signal; the only requirement is that it provide sufficient observability of the performance function.

1.1 The mountain climber problem

An effective analogy of the extremum-seeking control problem is as follows: *A mountain climber on a mountain in a thick bank of fog has no map and can only measure his current latitude, longitude, and altitude. The mountain climber is to find the peak of the mountain.* In this analogy, the performance function is the mountain. The independent axes of the performance function are latitude and longitude. The dependent axis of the performance function is altitude. The mountain climber represents a physical plant which must be moved to the mountain peak.

The strategy of the extremum-seeking control method presented herein is to measure the coordinates of the performance function's independent and dependent axes as the plant moves across the performance function. The local gradient and Hessian are estimated using the measurements.



1.1: A Mountain Climber in a Thick Bank of Fog

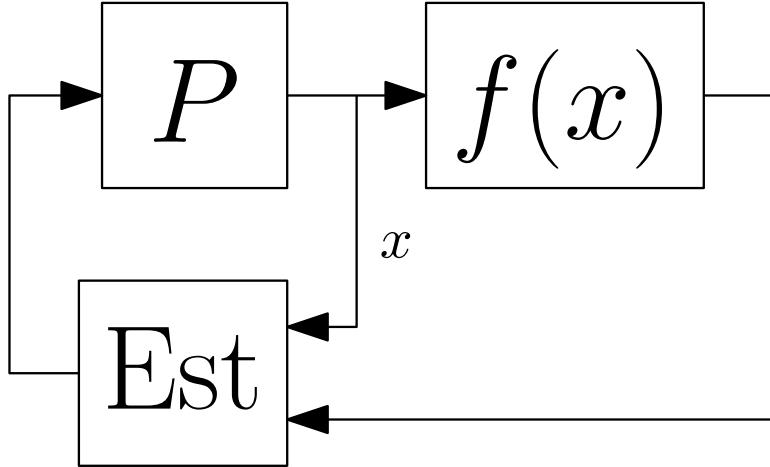
The estimates are then used in a Newton-Raphson scheme to command the plant toward the extremum position.

Applying this strategy to our analogy, the mountain climber will first measure his current latitude, longitude, and altitude. He will then move a short distance and take new measurements. After a short series of movements and measurements he will calculate an estimate of the peak's position defined by this set of measurements and begin to move in that direction. As he travels to the estimated peak location he will continue to take measurements, update his estimate, and change his direction of travel accordingly.

1.2 Basic Interconnection of Extremum Seeking System

A basic interconnection of the extremum seeking system can be constructed from the analogy. Figure 1.2 displays this interconnection; it displays only the most basic components of the system

that will be developed in later chapters. Here the mountain climber (the plant) is denoted by P , his latitude and longitude (the coordinates on the performance function) by x , his altitude (the corresponding value of the performance function dependent parameter) by $f(x)$ and the estimation calculations of the mountain's peak by Est.



1.2: Basic extremum-seeking method interconnection

The mountain climber begins his ascent by moving to a series of different latitudes and longitudes. Measurements of x and $f(x)$ are passed to Est which passes an estimate of the mountain peak's location to P . The mountain climber then moves toward the estimated peak's location while continuing to take measurements. As more measurements are passed to Est, the estimate is refined and P adjusts his path accordingly.

1.3 Thesis Overview

The remainder of this dissertation mathematically develops the extremum-seeking control method, considers convergence of the method in the presence of noise, and provides three example applications of the method. It is organized as follows.

Chapter 2 performs a literature review of extremum-seeking control. It discusses the most prevalent methods and surveys other approaches.

Chapter 3 derives the time varying Kalman filter. The derivation is adapted from other sources and presented for completeness.

Chapter 4 provides a mathematical development of the extremum-seeking control method. It

defines basic notation, presents an overview of the method, and then develops the method from basic assumptions. A summary of the method is then provided and two modifications of the method are presented.

Chapter 5 discusses convergence of the method to the extremum. It first shows convergence of the time-varying Kalman filter when noisy measurements are used to construct the measurement matrix. It then shows that the extremum seeking method will converge to an area around the extremum defined, in part, by the Kalman filter's estimation error-covariance.

Chapter 6 presents the application of the extremum seeking method to a performance function with one independent parameter and one dependent parameter. It then explores the effects varying the sliding-window size and the sample time.

Chapter 7 introduces the problem of formation flight for drag reduction. It provides a literature survey of this problem area and discusses the aerodynamics of formation flight which lead to drag reduction .

Chapter 8 presents the application of the method to the problem of formation flight for drag reduction on a UCLA developed simulation. It discusses the design choices made and presents the simulation results.

Chapter 9 presents a second application of the method to the problem of formation flight for drag reduction. In this chapter the method is employed on a Boeing Company developed simulation. It discusses the design choices made, presents the simulation results, and compares the results with those of the previous chapter.

Chapter 10 summarizes and concludes the dissertation.

Three appendices are also included in this dissertation. Appendix A derives the Linear Quadratic Regulator (LQR). Appendix B derives the LQR-tracker which is used in the second example application. Appendix C provides a design tutorial of the extremum-seeking control scheme.

CHAPTER 2

Literature Survey

In this chapter a review of the extremum seeking control literature is provided. Section 2.1 discusses the categorization of various extremum seeking control methods. Section 2.2 is devoted to the classical gradient method. It describes the method and provides references to some of the applications. Section 2.3 discusses methods which parameterize the performance function and use that parameterization to estimate the location of the extremum. Section 2.4 discuss other approaches which do not fall into the classical gradient method or parameterization method category.

2.1 Categorization of Extremum Seeking Literature

A multitude of approaches to extremum-seeking control are found in published work. Two literature reviews offer a starting point for the discussion. The first, [Lar01], groups extremum seeking methods into three general categories: gradient-method, parametric-method, and artificial-neural-network-method. The first category encompasses techniques which only estimate the gradient to seek the extremum. This category is subdivided into three sub-categories. The first encompasses techniques which use finite differences to calculate the gradient. The second encompasses techniques which use time derivatives to find the gradient. The third sub-category uses continuous perturbations to measure the gradient. Only the last of these sub-categories is prevalent in the literature. In fact, the third sub-category is ubiquitous and likely deserves its own category.

The second general category in [Lar01] encompasses methods which parameterize the performance function and use estimates of the parameters to decide how to drive the system toward the extremum. The method developed in this dissertation will fall into this category because it parameterizes the system with a Taylor series expansion.

The third general category encompasses approaches that utilize artificial-neural-networks to

find the extremum of a performance function. This method is probably not deserving of its own category since the artificial-neural-network-methods parameterize the performance function in a similar fashion to the techniques in the parametric-method category and because there is a relatively small number of published works using artificial-neural-networks.

The second literature review [TMM10] uses two categories to group extremum seeking control methods. The first encompasses methods which use continuous excitation signals to explore the performance function and gain an approximation of the performance function's gradient. This is roughly equivalent to the sub-category in [Lar01] which uses continuous perturbations to measure the gradient.

The second category encompasses methods which use repeated sequences as probes of the performance function and use numerical optimization methods to find the extremum. This category does not translate as well to a single category of [Lar01], but seems to encompass the parameterization-method and the artificial-neural-network-method.

The second literature review focuses on the methods contained in its first category. It provides a history on that method's development, an overview of the method, an analysis of the method, and two simple examples of its use.

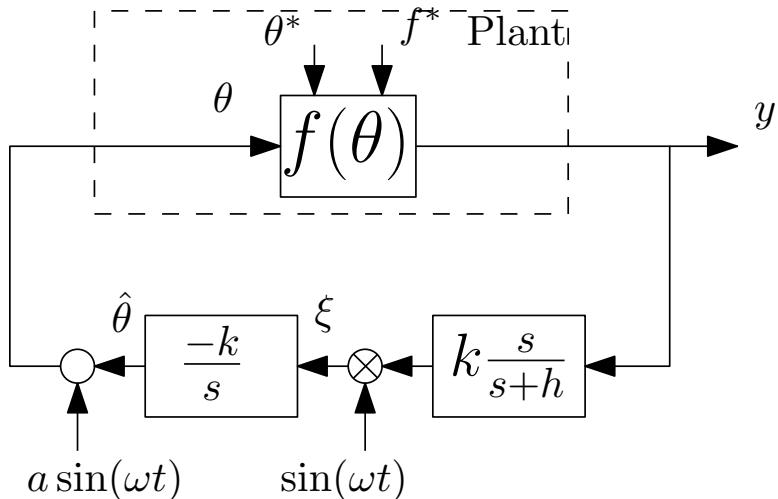
This dissertation chooses a third categorization of the extremum seeking methods. It is similar to the categorization provided in [TMM10] but breaks the second category into two. The first category holds with the first of [TMM10] and is referred to as the classical-gradient-method. It encompasses the extremum seeking methods which use classical control techniques to estimate the gradient of the performance function. The second category is referred to as the parametric-method. It assumes a fixed parameterizable form of the performance function and estimates those parameters. A gradient is then typically calculated and used to command the plant. The third category encompasses all the techniques that do not fit nicely into the first two.

2.2 Classical Gradient Methods

The classical-gradient methods dominate the extremum-seeking control literature. The earliest works all used classical-gradient methods. This is largely due to ease of implementation in analog devices and the lack of inexpensive digital computers. The earliest work was completed in France by Leblanc [Leb22] in 1922. There he describes a mechanism to transfer power from an electrical

transmission line to a tram car. In order to maintain maximum power, he introduces an extremum seeking controller. In 1943 [Kaz43] and 1944 [Kaz44] extremum seeking was studied in Russia by Kazakevich. The first appearance of extremum seeking in English literature was in 1951 by Draper [DL51] who developed an extremum-seeking controller for an internal combustion engine. Extremum-seeking control then gained much attention in the 1950s and 1960s. Examples of this work include [Ost57], [Bla62], and [JS68]. The interest appears to have declined in the 1970s and 1980s but has since had a resurgence of attention in the past 15 years. The classical-gradient methods have recently been further developed in work by Ariyur [AK02], Krstic [KW97], and others who have developed a general design technique and proved local convergence of this approach [AK03].

The ubiquity of the classical-gradient methods make it obligatory to present the approach in some detail. In general, the classical-gradient methods use a modulation-demodulation approach to estimate the gradient of a performance function. The most pervasive approach in the literature is that used by Kristc. We present the basics of that method here with the understanding that the description does not encompass all classical-gradient methods but is well representative. The method is depicted in Figure 2.1. In the figure, a sinusoidal signal is added to the latest estimate



2.1: Classical gradient method interconnection. This figure was adapted from [AK03].

of the extremum coordinates $\hat{\theta}$ and used to command a plant. The plant, which encompasses the performance function $f(\theta)$ responds to the command. Its response y is filtered by a high-pass filter which removes the steady-state portions of y . The result is then multiplied by the same

sinusoidal used before. This results in an estimate of the performance function gradient which is then integrated to generate a new estimate of the extremum coordinates.

The technique is now presented in greater detail and results in demonstrating convergence of the method. The notation is taken from [AK03] in order to be consistent with the references. It differs from the notation used in the rest of the work presented.

The performance function $f(\theta)$ is assumed to be adequately approximated by a quadratic function. To this end a Taylor approximation is used.

$$f(\theta) = f(\theta_0) + f'(\theta - \theta_0(t)) + \frac{f''}{2}(\theta - \theta_0(t))^2 + O((\theta - \theta^*(t)))$$

Here $f'(\cdot)$ denotes the gradient and $f''(\cdot)$ the Hessian. Evaluating equation (4.3.1) at the extremum positions $\theta^*(t)$ provides

$$f(\theta) = f^*(t) + \frac{f''}{2}(\theta - \theta^*(t))^2 \quad (2.2.1)$$

Here $f^*(t) = f(\theta^*(t))$ and $f'(\theta - \theta^*(t)) = 0$. It is assumed that $f'' > 0$ is an unknown constant. The plant indicated in Figure 2.1 consists of the performance function $f(\theta)$, which has inputs of the parameter θ and the optimal parameter θ^* along with the optimal magnitude f^* which occurs at θ^* .

The goal is to move θ to θ^* . Denote the estimation error

$$\tilde{\theta} = \theta^* - \hat{\theta} \quad (2.2.2)$$

The performance function input is the summation of the current estimate of θ^* and a persistent excitation signal

$$\theta = \hat{\theta} + a \sin \omega t \quad (2.2.3)$$

Combining the above equations (2.2.2) and (2.2.3), we arrive at

$$\theta - \theta^* = a \sin \omega t - \tilde{\theta} \quad (2.2.4)$$

which is then substituted into equation (2.2.1) to find

$$y = f^* + \frac{f''}{2} (\tilde{\theta} - a \sin \omega t) \quad (2.2.5)$$

Using the trigonometric identity

$$2 \sin^2(\omega t) = 1 - \cos(2\omega t) \quad (2.2.6)$$

leads to

$$y = f^* + \frac{a^2 f''}{4} + \frac{f''}{2} \tilde{\theta}^2 - a f'' \tilde{\theta} \sin(\omega t) + \frac{a^2 f''}{4} \cos(2\omega t) \quad (2.2.7)$$

The signal y passes through the high pass filter

$$\frac{s}{s+h}$$

which removes the constant terms from the signal leaving

$$\frac{f''}{2} \tilde{\theta}^2 - a f'' \tilde{\theta} \sin(\omega t) + \frac{a^2 f''}{4} \cos(2\omega t) \quad (2.2.8)$$

The resulting signal is then demodulated by multiplying by $\sin(\omega t)$ giving

$$\zeta = \frac{f''}{2} \tilde{\theta} + \frac{a f''}{2} \tilde{\theta} \cos(2\omega t) + \frac{a^2 f''}{8} (\sin(\omega t) - \sin(3\omega t)) + \frac{f''}{2} \tilde{\theta}^2 \sin(\omega t) \quad (2.2.9)$$

Here we used the identity in equation (2.2.6) and the second identity

$$2 \cos(2\omega t) \sin(\omega t) = \sin(2\omega t) - \sin(\omega t)$$

This signal is then passed through the integration block to produce

$$\tilde{\theta} = \frac{k}{s} \left[\frac{f''}{2} \tilde{\theta} + \frac{a f''}{2} \tilde{\theta} \cos(2\omega t) + \frac{a^2 f''}{8} (\sin(\omega t) - \sin(3\omega t)) \right] \quad (2.2.10)$$

where the last term of equation (2.2.9) was dropped because it is quadratic. The last two terms of equation (2.2.10) are high frequency signals. The integration block greatly attenuates these and we are left with

$$\dot{\tilde{\theta}} = -\frac{k a f''}{2} \tilde{\theta}.$$

Since $k f'' > 0$, $\tilde{\theta}$ will go to zero, thus finding the optimal position.

It is worth noting that the approach does not use direct measurements of the independent parameters; instead, the gradient is determined by commanding known sinusoidal perturbation of the independent parameters. This puts strict conditions on the form of the perturbations.

The technique has been shown to have convergence guarantees under certain assumptions. A stability proof of the method is presented in [KW00]. It was the first stability analysis published of this method for a general nonlinear plant. Reference [LK10] employs stochastic persistent excitation signals and analyzes the system with stochastic averaging to show both almost sure stability and stability in probability.

The method has been expanded in a number of ways. In [CKA02] a discrete version of the method is presented. In [Rot00] and [Wal00] the approach is extended to multivariable problems. The same basic approach is used in [SS09] but the amplitudes of the perturbation signals as well as the gain of the integrator block are made to be time varying. In [AK04] the approach is applied to slope-seeking which finds the coordinates on a performance function with a given gradient which is not zero.

The techniques have been used successfully in a large number of applications. In [KW00] an application to a nonlinear aeroengine compressor model is provided. Wang [WYK00] presents an application to axial flow compressor and shows the persistent excitation required is below the noise level of the compressor. In [BK07] the technique is used in separation control on a high-Lift aircraft configuration.

2.3 Parametric Methods

The Parametric Methods category encompasses techniques which parameterize the performance function in some way and use the parameterization to estimate the gradient, Hessian, or extremum of the performance function. It includes techniques which use state-estimators, Kalman filters, adaptive control, and others. There is not a main design framework common in all the techniques as there is with the classical-gradient method. Hence it is not possible to characterize all of the techniques in a general way. Instead, a few examples of techniques which fall into this category are discussed. This will provide a window into the types of techniques found in this category.

In [CSF06] a scheme is developed which uses a Kalman filter to estimate the gradient of the performance function and a steepest descent approach to find the extremum. In contrast to the work presented in this dissertation, the Kalman filter is not time-varying and models the persistent excitation signal in order to estimate the gradient. The scheme is applied to the problem of formation flight for drag reduction.

In [SBC00] the performance function is assumed to be of a known functional type of which the gradient and Hessian are estimated and used to drive the plant toward the extremum. The scheme is noteworthy because the resulting interconnection is near linear and lends itself to stability analysis. In addition, a time-scale separation between the system dynamics and the gradient search need not be assumed as is needed in many other approaches.

There are multiple techniques which take an adaptive control approach. In [MMB08b] and [MMB08a] a state-space observer is used to estimate the gradient and Hessian of a performance function and an adaptation law is used to generate a command to the plant in a Newton-like method. In addition it uses a dither signal amplitude schedule which varies the magnitude of the persistent excitation as a function of performance function gradient. The scheme is applied to minimizing the thermoacoustic oscillations in a model premixed combustor in [Moa10].

In [EL05] a parameter adaptive control approach is used. It uses a recursive least-squares estimator to estimate the parameters of a performance function and then uses a Newton method to find the extremum.

In [GDP03] an adaptive method is proposed and used to estimate the gradient and Hessian of the performance function and to find the extremum of a non-isothermal continuous stirred tank reactors.

In [NS02] and [LHC03b] adaptive control schemes are also used to determine the gradient and find the extremum of a performance function.

There are many other examples not discussed here such as [WKB99], [Yu06], [SXT07], [BAK04], and [FDA66].

2.4 Other Methods

It is more difficult to categorize the techniques which are not clearly in the classical-gradient or parametric category. Detailed descriptions of these techniques are not presented here due to the large amount of space such a discussion would require. Instead the more notable work is referenced with a short description of the key features of the approaches.

In [PO02] a sliding-mode control is used to find the extremum of a performance function. This method proves robust to changes in performance function parameters. The method is used in [PAO02] to find the Nash equilibrium solution of an n -player dynamic uncooperative game.

Popović and Tell [PJM06], [Pop04] do not assume a static performance function but prove convergence of their technique on a non-static performance function when using a waiting-time for plant dynamics to settle to negligible levels. They apply the simultaneous perturbation stochastic approximation algorithm and the Box and Wilson steepest descent method to the problem.

In [ZO05] a numerical optimization scheme is incorporated into an extremum seeking control problem. The method they propose uses line searches and assumes gradient information is available.

Categorization of the various extremum seeking control techniques is not an easy task since there are a plethora of approaches. Here we've captured the two most prevalent approaches in a Classical-Gradient category and a Parametric category. The techniques that do not fall into these categories have been put into a third called Other Methods.

CHAPTER 3

Derivation of the Time Varying Kalman Filter

The extremum-seeking method that is to be developed in this work relies upon the time-varying Kalman filter. It is therefore necessary to devote a chapter to its derivation. Much of the following derivation is taken from [SC08] and [BH97].

3.1 Introduction to the Time Varying Kalman Filter

The Kalman Filter, named after Rudolf E. Kalman, first appeared in papers by Swerling, Kalman, and Bucy in 1958, 1960 [Kal60], and 1961. It is a conditional mean estimator and is constructed in a predictor-corrector form. With the assumption that system noise is Gaussian distributed, it is the minimum variance estimator.

Consider a linear discrete-system

$$\zeta_{k+1} = \Phi_k \zeta_k + \Gamma_k w_k \quad (3.1.1a)$$

$$z_k = H_k \zeta_k + v_k \quad (3.1.1b)$$

where Φ_k is a state-transition matrix, Γ_k is the input matrix, and H_k is the measurement matrix; each of these is assumed to be known. In addition v_k and w_k are zero-mean Gaussian white-noise processes with covariance $W_k = E[w_k w_k^T]$ and $V_k = E[v_k v_k^T]$ respectively. We assume that the initial state $\zeta_0 \sim \mathcal{N}(\bar{\zeta}_0, M_0)$. That is, the initial state is a random value normally distributed around the mean $\bar{\zeta}_0$ with a variance M_0 . In addition, we assume that w_k , v_k , and ζ_0 are independent of each other.

We now state some basic definitions which will be used repeatedly in the development that follows. Given a measurement history of equation (3.1.1b)

$$Z_{k-1} = \left\{ z_0 \ z_1 \ \dots \ z_{k-1} \right\}$$

define the *conditional mean* as

$$\hat{\zeta}_k = \mathbb{E}[\zeta_k | Z_k]$$

This is the estimate of ζ_k given the measurement history Z_k . Define the *estimated mean* as

$$\bar{\zeta}_k = \mathbb{E}[\zeta_k | Z_{k-1}]$$

This is a prediction of what ζ_k will be when the next measurement is taken; note that the subscript of Z is one step behind the subscript of ζ . In addition, define P_{k-1} and M_k as the error covariance generated between measurements and that generated after the measurements respectfully.

3.2 Derivation of the Estimated Mean and Covariance

Let's begin by determining how to calculate the estimated mean given measurements up to time k . From our definition above,

$$\bar{\zeta}_{k+1} = \mathbb{E}[\zeta_{k+1} | Z_k]$$

where we shifted the indexing to better fit with derivations that will follow. Replacing ζ_{k+1} with the model of the dynamics in equation (3.1.1a) we get

$$\bar{\zeta}_{k+1} = \mathbb{E}[\Phi_k \zeta_k + \Gamma_k w_k | Z_k]$$

Noting that expectation is a linear operator and that Φ_k and Γ_k are known,

$$\bar{\zeta}_{k+1} = \Phi_k \mathbb{E}[\zeta_k | Z_k] + \Gamma_k \mathbb{E}[w_k | Z_k]$$

We've stated above that $\hat{\zeta}_k = \mathbb{E}[\zeta_k | Z_k]$ and so get

$$\bar{\zeta}_{k+1} = \Phi_k \hat{\zeta}_k + \Gamma_k \mathbb{E}[w_k | Z_k]$$

We also note that since w_k is independent of v_k and ζ_0 ,

$$\mathbb{E}[w_k | Z_k] = \mathbb{E}[w_k]$$

Since w_k is zero-mean,

$$\bar{\zeta}_{k+1} = \Phi_k \hat{\zeta}_k \tag{3.2.1}$$

This is the estimated mean of ζ at $k + 1$ given the conditional mean at time k . This makes intuitive sense; simply propagate the states through the assumed model of the process. Of course we have not yet determined the conditional mean.

The covariance must also be propagated to a prediction at $k + 1$. To do this, we first define two estimation errors: the error between the conditional mean of ζ and the actual value of ζ ; and the error between the estimated mean of ζ and the actual value of ζ .

$$\bar{e}_k = \zeta_k - \bar{\zeta}_k \quad (3.2.2)$$

$$\hat{e}_k = \zeta_k - \hat{\zeta}_k \quad (3.2.3)$$

The covariance generated between measurements is defined as

$$M_{k+1} = E[\bar{e}_{k+1}\bar{e}_{k+1}|Z_k]$$

Using (3.2.2), (3.2.3), and (3.1.1a) we can write this as

$$M_{k+1} = E\left[\left(\Phi_k \hat{e}_k + \Gamma_k w_k\right) \left(\Phi_k \hat{e}_k + \Gamma_k w_k\right)^T | Z_k\right]$$

Multiplying-out the inside of the expectation we find

$$M_{k+1} = E\left[\Phi_k \hat{e}_k \hat{e}_k^T \Phi_k^T + \Gamma_k w_k w_k^T \Gamma_k^T + 2\Phi_k \hat{e}_k w_k^T \Gamma_k^T | Z_k\right]$$

Since ζ and w are independent processes, so are e and w hence $\Phi_k \hat{e}_k w_k^T \Gamma_k^T = 0$. In addition, as before, Φ_k and Γ_k are known. We can therefore write

$$M_{k+1} = \Phi_k E\left[\hat{e}_k \hat{e}_k^T | Z_k\right] \Phi_k^T + \Gamma_k E\left[w_k w_k^T | Z_k\right] \Gamma_k^T$$

Defining

$$P_k = E\left[\hat{e}_k \hat{e}_k^T | Z_k\right]$$

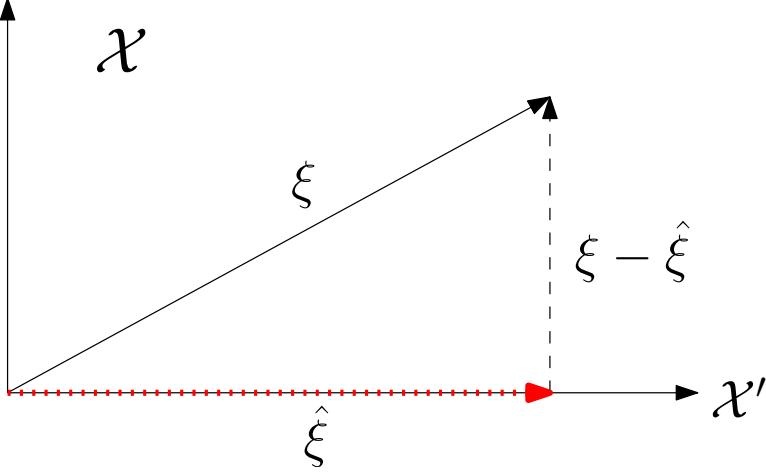
and remembering $W_k = E[w_k w_k^T]$ we can write

$$M_{k+1} = \Phi_k P_k \Phi_k^T + \Gamma_k W_k \Gamma_k^T \quad (3.2.4)$$

We now have equations to compute the estimated mean and its corresponding error covariance. To derive the conditional mean and its corresponding error covariance, we will use a tool known as the Orthogonal Projection Lemma.

3.3 The Orthogonal Projection Lemma

The Orthogonal Projection Lemma is conveniently explained with basic geometric considerations. Consider a vector ξ in a Cartesian vector space \mathcal{X} ; for now assume \mathcal{X} is a two-dimensional plane as shown in Figure 3.3.



3.1: Demonstration of the Orthogonal Projection Lemma. Dotted line denotes $\hat{\xi}$. Dashed line denotes $\xi - \hat{\xi}$

Consider also a subspace of \mathcal{X} denoted as \mathcal{X}' consisting of a family of vectors ξ' ; for now assume \mathcal{X}' is the horizontal axis of the two-dimensional plane. Let $\hat{\xi}$ be the vector in \mathcal{X}' which has the smallest error $(\xi - \hat{\xi})$. It's intuitively obvious that $\hat{\xi}$ is the vector whose error $\xi - \hat{\xi}$ is orthogonal to the set \mathcal{X}' . By extending this reasoning to a Hilbert space we arrive at the Orthogonal Project Lemma.

Theorem 3.3.1 (Orthogonal Projection Lemma). *Let \mathcal{X} be a Hilbert space with $\xi \in \mathcal{X}$ and $\mathcal{X}' \subset \mathcal{X}$ be a subset of \mathcal{X} . Then there exists a unique vector $\hat{\xi}$ such that*

$$\min_{\xi' \in \mathcal{X}} \|\xi - \xi'\| = \|\xi - \hat{\xi}\|$$

if and only if

$$\langle \xi - \hat{\xi}, \xi' \rangle = 0 \quad \forall \xi' \in \mathcal{X}'$$

Proof. (\implies)

Assume that the error $\xi - \hat{\xi}$ is orthogonal to ξ' , i.e. $\langle \xi - \hat{\xi}, \xi' \rangle = 0$ for all $\xi' \in \mathcal{X}'$. Let $\alpha \in \mathcal{X}'$ have any arbitrary magnitude. The magnitude of error vector between ξ and $\hat{\xi} + \alpha$ is

$$\langle \xi - \hat{\xi} + \alpha, \xi - \hat{\xi} + \alpha \rangle = \langle \xi - \hat{\xi}, \xi - \hat{\xi} \rangle + \langle \alpha, \alpha \rangle + 2 \langle \xi - \hat{\xi}, \alpha \rangle$$

Since $\xi - \hat{\xi}$ is orthogonal to $\alpha \in \mathcal{X}'$, $\langle \xi - \hat{\xi}, \alpha \rangle = 0$. Hence

$$\begin{aligned} \langle \xi - \hat{\xi} + \alpha, \xi - \hat{\xi} + \alpha \rangle &= \langle \xi - \hat{\xi}, \xi - \hat{\xi} \rangle + \langle \alpha, \alpha \rangle \\ &\geq \langle \xi - \hat{\xi}, \xi - \hat{\xi} \rangle \end{aligned}$$

This tells us that any $\alpha \neq 0$ results in an error vector of larger magnitude than $\xi - \hat{\xi}$.

(\Leftarrow)

Now assume that $\hat{\xi}$ minimizes the error $\xi - \hat{\xi}$. Further assume that there exists an ξ' which is *not* orthogonal to $\xi - \hat{\xi}$ but

$$\langle \xi - \hat{\xi}, \xi' \rangle = \beta$$

Let λ be a real valued scalar. The magnitude of the error vector between ξ and $\hat{\xi} - \lambda\xi'$

$$\langle \xi - \hat{\xi} + \lambda\xi', \xi - \hat{\xi} + \lambda\xi' \rangle = \langle \xi - \hat{\xi}, \xi - \hat{\xi} \rangle + \lambda^2 \langle \xi', \xi' \rangle + 2\lambda \langle \xi - \hat{\xi}, \xi' \rangle$$

Of course we defined $\beta = \langle \xi - \hat{\xi}, \xi' \rangle$, so we have

$$\langle \xi - \hat{\xi} + \lambda\xi', \xi - \hat{\xi} + \lambda\xi' \rangle = \langle \xi - \hat{\xi}, \xi - \hat{\xi} \rangle + \lambda^2 \langle \xi', \xi' \rangle + 2\lambda\beta$$

Choosing

$$\lambda = -\frac{\beta}{\langle \xi', \xi' \rangle}$$

we arrive at

$$\langle \xi - \hat{\xi} + \lambda\xi', \xi - \hat{\xi} + \lambda\xi' \rangle = \langle \xi - \hat{\xi}, \xi - \hat{\xi} \rangle - \frac{\beta^2}{\langle \xi', \xi' \rangle}$$

But of course

$$\langle \xi - \hat{\xi}, \xi - \hat{\xi} \rangle - \frac{\beta^2}{\langle \xi', \xi' \rangle} < \langle \xi - \hat{\xi}, \xi - \hat{\xi} \rangle$$

contradicting the initial assumption that $\langle \xi - \xi', \xi - \xi' \rangle$ is minimized by $\hat{\xi}$ \square

3.4 Derivation of the Conditional Mean and Covariance

We now present a derivation of the conditional mean and the corresponding error covariance utilized in the Kalman filter. The derivation comes in the form of a mean-square estimation optimization problem stated as: minimize the cost function

$$J = \min_{\{\hat{\zeta}_0(\mathcal{Y}_0), \hat{\zeta}_1(\mathcal{Y}_1), \dots, \hat{\zeta}_N(\mathcal{Y}_N)\}} \mathbb{E} \left[\sum_{k=0}^N \left(\zeta_k - \hat{\zeta}_k(\mathcal{Y}_0) \right)^T Q_k \left(\zeta_k - \hat{\zeta}_k(\mathcal{Y}_0) \right) \right]$$

subject to

$$\zeta_{k+1} = \Phi_k \zeta_k + \Gamma_k w_k \tag{3.4.1}$$

$$y_k = H_k \zeta_k + v_k \tag{3.4.2}$$

and subject to a requirement that the estimates be a *linear* function of the measurements.

In words, we are minimizing, over the N stages, of the state estimate sequence

$$\left\{ \hat{\zeta}_0(\mathcal{Y}_0), \hat{\zeta}_1(\mathcal{Y}_1), \dots, \hat{\zeta}_N(\mathcal{Y}_N) \right\}$$

the variance of the error $\zeta_k - \hat{\zeta}_k$ weighted by the Q_k where $\hat{\zeta}_k$ is a function of the vector space \mathcal{Y}_k generated by the measurement history up to time k ,

$$\mathcal{Y}_k = \text{span} \left\{ \begin{array}{cccc} y_1 & y_2 & \dots & y_k \end{array} \right\}$$

Since the expectation of a summation is equivalent to the sum of the expectations, we can manipulate the cost function to

$$J = \min_{\{\hat{\zeta}_0(\mathcal{Y}_0), \hat{\zeta}_1(\mathcal{Y}_1), \dots, \hat{\zeta}_N(\mathcal{Y}_N)\}} \sum_{k=0}^N \mathbb{E} \left[\left(\zeta_k - \hat{\zeta}_k(\mathcal{Y}_0) \right)^T Q_k \left(\zeta_k - \hat{\zeta}_k(\mathcal{Y}_0) \right) \right]$$

It is also well known that the minimization of a sum is equivalent to the sum of minimizations

$$J = \sum_{k=0}^N \min_{\{\hat{\zeta}_k(\mathcal{Y}_k)\}} \mathbb{E} \left[\left(\zeta_k - \hat{\zeta}_k(\mathcal{Y}_0) \right)^T Q_k \left(\zeta_k - \hat{\zeta}_k(\mathcal{Y}_0) \right) \right]$$

The solution of such a minimization problem can be found via the Orthogonal Projection Theorem discussed in section 3.3. We are now dealing with a stochastic inner product instead of the deterministic vectors discussed earlier. The stochastic inner product states

$$\mathbb{E}[\zeta(i)(y)] = 0$$

Consider an m -vector set of orthonormal basis for \mathcal{Y}_k

$$U_k = \left\{ u_1 \quad u_2 \quad \dots \quad u_k \right\}$$

such that

$$\mathbb{E}[u_i u_j^T] = \begin{cases} I, & i = j \\ 0 & \text{else} \end{cases}$$

This says that two random vectors u_i and u_j are correlated ($i = j$) then the inner product equals one else it is zero. As was shown in section 3.3, the minimum occurs when the error $\zeta_k - \hat{\zeta}_k$ is orthogonal to the basis U_k i.e.

$$\mathbb{E} \left[\left(\zeta_k - \hat{\zeta}_k \right) u_i^T \right] \quad i = 1, \dots, k$$

or

$$\mathbb{E}[\zeta_k u_i^T] = \mathbb{E}[\hat{\zeta}_k u_i^T] \quad \forall i, k$$

This is also true for the sum of the vectors of U_k

$$\sum_{i=1}^k \mathbb{E}[\zeta_k u_i^T] u_i = \sum_{i=1}^k \mathbb{E}[\hat{\zeta}_k u_i^T] u_i$$

where we multiplied each side by u_i . This then becomes

$$\hat{\zeta} = \sum_{i=1}^k \mathbb{E}[\zeta_k u_i^T] u_i \tag{3.4.3}$$

Pulling the $i = k$ expectation out of the summation we get

$$\hat{\zeta} = \sum_{i=1}^{k-1} \mathbb{E}[\zeta_k u_i^T] u_i + \mathbb{E}[\zeta_k u_k^T] u_k$$

Utilizing equation (3.4.1), this is written as

$$\begin{aligned} \hat{\zeta} &= \sum_{i=1}^{k-1} \mathbb{E}[(\Phi_{k-1} \zeta_{k-1} + \Gamma_{k-1} w_{k-1}) u_i^T] u_i + \mathbb{E}[\zeta_k u_k^T] u_k \\ &= \sum_{i=1}^{k-1} \Phi_{k-1} \mathbb{E}[\zeta_{k-1} u_i^T] u_i + \Gamma_{k-1} \mathbb{E}[w_{k-1} u_i^T] u_i + \mathbb{E}[\zeta_k u_k^T] u_k \end{aligned}$$

Recognizing that w_{k-1} is independent of \mathcal{Y}_{k-1} and that it has a zero mean, we can write

$$\hat{\zeta} = \Phi_{k-1} \sum_{i=1}^{k-1} \mathbb{E}[\zeta_{k-1} u_i^T] u_i + \mathbb{E}[\zeta_k u_k^T] u_k$$

Comparing this to equation (3.4.3) we can see a recursion and write

$$\hat{\zeta} = \Phi_{k-1} \hat{\zeta}_{k-1} + \mathbb{E}[\zeta_k u_k^T] u_k$$

Note that at this point we've been able to eliminate the summation over all time steps. We now only need to keep track of the k th and $k - 1$ st time steps. We now determine u_k . We know that u_k must be orthogonal to \mathcal{Y}_{k-1} , which we required above. We choose to use what we found in equation (3.2.1) and multiply it by a gain K_k to get

$$u_k = K_k \left(y_k - H_k \Phi_{k-1} \hat{\zeta}_{k-1} \right)$$

The gain still needs to be determined, but this is indeed orthogonal to \mathcal{Y}_{k-1} . We pause here for a moment to point out that we've required that u_k is the portion of the k th measurement which is orthogonal to all measurements up to time k .

Our equation for $\hat{\zeta}$ has become

$$\hat{\zeta} = \Phi_{k-1}\hat{\zeta}_{k-1} + K_k \left(y_k - H_k \Phi_{k-1}\hat{\zeta}_{k-1} \right)$$

Using equation (3.2.1) we arrive at the recognized form

$$\hat{\zeta} = \Phi_{k-1}\hat{\zeta}_{k-1} + K_k \left(y_k - H_k \bar{\zeta}_k \right) \quad (3.4.4)$$

We now need to determine the gain K_k . To this end we derive the error dynamics. Define

$$\begin{aligned} \hat{e} &= \zeta_k - \hat{\zeta}_k \\ &= (\Phi_{k-1}\zeta_{k-1} + \Gamma_{k-1}w_{k-1}) - \left(\Phi_{k-1}\hat{\zeta}_{k-1} + K_k \left(y_k - H_k \Phi_{k-1}\hat{\zeta}_{k-1} \right) \right) \\ &= (\Phi_{k-1}\zeta_{k-1} + \Gamma_{k-1}w_{k-1}) - \left(\Phi_{k-1}\hat{\zeta}_{k-1} + K_k \left(H_k\zeta_k + v_k - H_k \Phi_{k-1}\hat{\zeta}_{k-1} \right) \right) \end{aligned}$$

where we used equation (3.4.2) to replace y_k . Using equation (3.4.1) to replace ζ_k , we get

$$\begin{aligned} \hat{e} &= (\Phi_{k-1}\zeta_{k-1} + \Gamma_{k-1}w_{k-1}) \\ &\quad - \left(\Phi_{k-1}\hat{\zeta}_{k-1} + K_k \left(H_k(\Phi_{k-1}\zeta_{k-1} + \Gamma_{k-1}w_{k-1}) + v_k - H_k \Phi_{k-1}\hat{\zeta}_{k-1} \right) \right) \\ &= \left(\Phi_{k-1}\zeta_{k-1} - \Phi_{k-1}\hat{\zeta}_{k-1} \right) \\ &\quad - \left(K_k H_k \Phi_{k-1} \zeta_{k-1} + K_k H_k \Phi_{k-1} \hat{\zeta}_{k-1} \right) + (\Gamma_{k-1}w_{k-1} - K_k \Gamma_{k-1}w_{k-1}) + K_k v_k \\ &= (\Phi_{k-1}\hat{e}_{k-1} - K_k H_k \Phi_{k-1}\hat{e}_{k-1}) + (\Gamma_{k-1}w_{k-1} - K_k \Gamma_{k-1}w_{k-1}) - K_k v_k \\ &= (I - K_k H_k) \Phi_{k-1}\hat{e}_{k-1} + (I - K_k H_k) \Gamma_{k-1}w_{k-1} - K_k v_k \end{aligned}$$

The output equation of the dynamics is found by replacing in ζ_k equation (3.4.2) with equation (3.4.1) to get

$$\begin{aligned} y_k &= H_k(\Phi_{k-1}\zeta_{k-1} + \Gamma_{k-1}w_{k-1}) + v_k \\ &= H_k \Phi_{k-1}(\zeta_{k-1} + H_k \Gamma_{k-1}w_{k-1} + v_k) \end{aligned}$$

Using $\zeta_{k-1} = \hat{e}_k + \hat{\zeta}_k$ we get

$$y_k = H_k \Phi_{k-1} \left(\hat{e}_k + \hat{\zeta}_k \right) + H_k \Gamma_{k-1} w_{k-1} + v_k$$

These are then used in the orthogonal projection

$$\mathbb{E}[\hat{e}_k y_k^T] = 0$$

or

$$\begin{aligned} \mathbb{E} \left[(H_k \Phi_{k-1} (\zeta_{k-1} + H_k \Gamma_{k-1} w_{k-1} + v_k) \right. \\ \left. \left(H_k \Phi_{k-1} (\hat{e}_k + \hat{\zeta}_k) + H_k \Gamma_{k-1} w_{k-1} + v_k \right)^T \right] = 0 \end{aligned}$$

Before we multiply-out the left hands side of this equation we note that

$$\mathbb{E}[ev^T] = 0 \quad (3.4.5a)$$

$$\mathbb{E}[ew^T] = 0 \quad (3.4.5b)$$

$$\mathbb{E}[vw^T] = 0. \quad (3.4.5c)$$

We then find

$$\begin{aligned} \mathbb{E}[\hat{e}_k y_k^T] &= \Phi_{k-1} \mathbb{E}[\hat{e}_k \hat{e}_k^T] \Phi_{k-1}^T H_k - K_k H_k \Phi_{k-1} \mathbb{E}[\hat{e}_k \hat{e}_k^T] \Phi_{k-1} H_k \\ &\quad + \Gamma_{k-1} \mathbb{E}[w_{k-1} w_{k-1}^T] \Gamma_{k-1}^T H_k - K_k H_k \Gamma_{k-1} \mathbb{E}[w_{k-1} w_{k-1}^T] \Gamma_{k-1}^T H_k^T \\ &\quad - K_k \mathbb{E}[v_{k-1} v_{k-1}^T] \\ &= \Phi_{k-1} P_{k-1} \Phi_{k-1}^T H_k - K_k H_k \Phi_{k-1} P_{k-1} \Phi_{k-1} H_k \\ &\quad + \Gamma_{k-1} W_{k-1} \Gamma_{k-1}^T H_k - K_k H_k \Gamma_{k-1} W_{k-1} \Gamma_{k-1}^T H_k^T - K_k V_k \\ &= (\Phi_{k-1} P_{k-1} \Phi_{k-1}^T + \Gamma_{k-1} W_{k-1} \Gamma_{k-1}^T) H_k \\ &\quad - K_k [H_k (\Phi_{k-1} P_{k-1} \Phi_{k-1}^T + \Gamma_{k-1} W_{k-1} \Gamma_{k-1}^T) H_k^T + V_k] \end{aligned}$$

Using the definition of M_k from equation (3.2.4), we get

$$M_k H_k^T - K_k [H_k M_k H_k^T + V_k] = 0$$

Hence we find

$$K_k = M_k H_k^T [H_k M_k H_k^T + V_k]^{-1} \quad (3.4.6)$$

We still need the update equation for the error covariance P_k . This is derived via straight forward calculation

$$\begin{aligned} P_k &= \mathbb{E}[\hat{e}_k \hat{e}_k^T] \\ &= \mathbb{E} \left[\left((I - K_k H_k) \Phi_{k-1} \hat{e}_{k-1} + (I - K_k H_k) \Gamma_{k-1} w_{k-1} - K_k v_k \right) \right. \\ &\quad \left. \left((I - K_k H_k) \Phi_{k-1} \hat{e}_{k-1} + (I - K_k H_k) \Gamma_{k-1} w_{k-1} - K_k v_k \right)^T \right] \end{aligned}$$

Multiplying out the equation and dropping all terms who's expectation is zero we get

$$\begin{aligned}
P_k &= \mathbb{E} \left[(I - K_k H_k) \Phi_{k-1} \hat{e}_{k-1} \hat{e}_{k-1}^T \Phi_{k-1}^T (I - K_k H_k)^T \right] \\
&\quad + \mathbb{E} \left[(I - K_k H_k) \Gamma_{k-1} w_{k-1} w_{k-1}^T \Gamma_{k-1}^T (I - K_k H_k)^T \right] + \mathbb{E} \left[K_k v_k v_k^T K_k^T \right] \\
&= (I - K_k H_k) \Phi_{k-1} \mathbb{E} \left[\hat{e}_{k-1} \hat{e}_{k-1}^T \right] \Phi_{k-1}^T (I - K_k H_k)^T \\
&\quad + (I - K_k H_k) \Gamma_{k-1} \mathbb{E} \left[w_{k-1} w_{k-1}^T \right] \Gamma_{k-1}^T (I - K_k H_k)^T + K_k \mathbb{E} \left[v_k v_k^T \right] K_k^T \\
&= (I - K_k H_k) (\Phi_{k-1} P_{k-1} \Phi_{k-1}^T + \Gamma_{k-1} W_{k-1} \Gamma_{k-1}^T) (I - K_k H_k)^T \\
&\quad + K_k V_k K_k^T
\end{aligned}$$

Using equation (3.2.4) again, we arrive at

$$\begin{aligned}
P_k &= (I - K_k H_k) M_k (I - K_k H_k)^T + K_k V_k K_k^T \\
&= M_k - M_k H_k^T K_k^T - K_k H_k M_k + K_k H_k M_k H_k^T K_k^T + K_k V_k K_k^T \\
&= (I - K_k H_k) M_k - (I - K_k H_k) M_k H_k^T K_k^T + K_k V_k K_k^T \\
&= (I - K_k H_k) M_k - M_k H_k^T K_k^T + K_k (M_k H_k^T K_k^T + V_k) K_k^T
\end{aligned}$$

Using equation (3.4.6) we can write

$$\begin{aligned}
P_k &= (I - K_k H_k) M_k - M_k H_k^T K_k^T + M_k H_k^T K_k^T \\
&= (I - K_k H_k) M_k
\end{aligned}$$

3.5 Summary of the Discrete Kalman Filter

The Kalman Filter consists of equations (3.2.1), (3.2.4), (3.4.4), (3.4.7), (3.4.6). To summarize,

$$K_k = M_k H_k^T [H_k M_k H_k^T + V_k]^{-1} \quad (3.5.1a)$$

$$\hat{\zeta}_k = \bar{\zeta}_k + K_k (y_k - H_k \bar{\zeta}_k) \quad (3.5.1b)$$

$$P_k = (I - K_k H_k) M_k \quad (3.5.1c)$$

$$\bar{\zeta}_{k+1} = \Phi_k \hat{\zeta}_k \quad (3.5.1d)$$

$$M_{k+1} = \Phi_k P_k \Phi_k^T + \Gamma_k W_k \Gamma_k^T \quad (3.5.1e)$$

CHAPTER 4

Development of Extremum-Seeking Method

This chapter develops the extremum-seeking method. It begins by defining some basic notation and providing a problem statement. It then sets the stage for a mathematical development by providing a conceptual overview of the method. A mathematical development follows. First the method in which the coordinates of the performance function's extremum are estimated is presented. Then, the method in which the gradient and Hessian of the performance function are estimated is discussed. An extension of the method is then described which utilizes a sliding-window to further improve performance. The section concludes with a summary of the extremum seeking control method.

4.1 Problem Statement and Basic Notation

Consider a unimodal performance function $f(x) : \mathbb{R}^N \rightarrow \mathbb{R}$ with $x \in \mathbb{R}^N$. Components of performance function's independent parameter x will be denoted with subscripts as $x = [x_1, x_2 \dots x_N]$ where N indicates the number of components and the subscripts are used to specify specific components. The dependent parameters of the performance function will be denoted by $f(x)$ or f when the dependence of x is implied. Note that the performance function is not an explicit function of time. While the method presented herein does not prohibit the possibility of a time dependency, it is not considered in the main discussion of the method.

In the extremum-seeking method to be discussed, measurements of x and $f(x)$ are required. It is assumed that these measurements are noisy. For analytical purposes the measurements are represented by a noiseless component added to a noise component. To this end, measurement components representing the noiseless value of the measurement are denoted by a superscript asterisk and noise components by a tilde accent. A measurement of x_i is therefore denoted as

$$x = x_i^* + \tilde{x}_i.$$

A measurement of $f(x)$ is denoted as

$$f(x) = f^*(x) + \tilde{f}(x).$$

Estimates of the gradient and Hessian will also be required. Estimates will be denoted with a hat accent such as \hat{b} .

Consider also a plant $P(u_c(t)) : \mathbb{R}^N \rightarrow \mathbb{R}^N$. The plant is a function of a command $u_c \in \mathbb{R}^N$. The plant is assumed to be Hurwitz.

The algorithm operates in discrete steps. The algorithm iteration will be denoted by k . It will usually appear as a subscript of other variables such as x_k which indicates the k^{th} instance of x .

The problem is to determine a u_c which commands the plant to the x which extremizes $f(x)$. The command u_c is to be determined using a history of measurements of x and $f(x)$ taken along the trajectory of P .

In other words, the information available to discern the u_c which drives P to the coordinates of the performance function extremum include only the history of the plant coordinates and history of the performance function value.

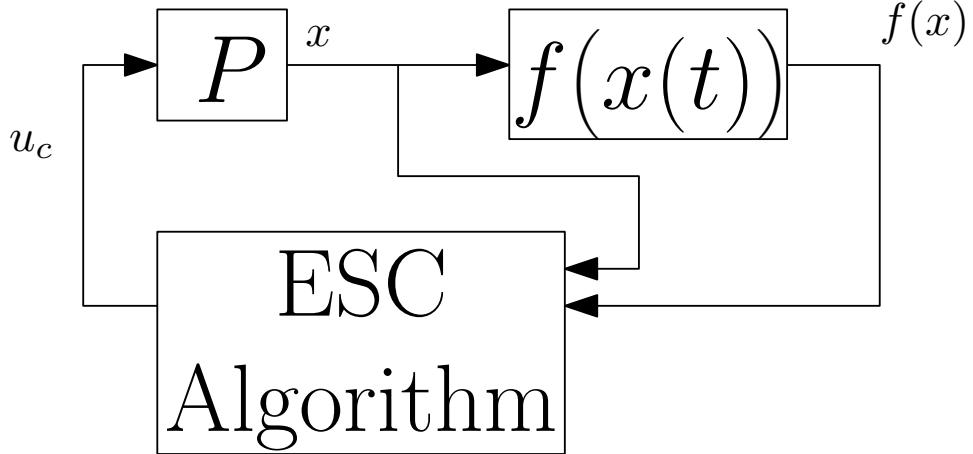
4.2 Overview of Method

The extremum seeking method described herein assumes that measurements of the independent and dependent parameters of the performance function are always available, albeit noisy. The method fits a history of these measurements to an assumed functional form. In what follows, a quadratic form is assumed. The fit is performed by parameterizing the functional form and estimating the best-fit parameters with a time-varying Kalman filter. The time-varying Kalman filter provides robustness to noisy measurements and the ability to use the dependent parameter measurements directly.

The parameter estimates are used to estimate the coordinates of the extremum. In particular, with the quadratic assumption, gradient and Hessian estimates are used in a Newton-Raphson approach to estimate the coordinates of the extremum. The plant is then commanded toward the extremum coordinates.

Figure 4.1 depicts the basic interconnection. The measurements available to the extremum

seeking control algorithm are the value of the independent parameters x and the value of the dependent parameters $f(x)$. The algorithm issues commands u_c to the plant.



4.1: Extremum seeking control method interconnection. The Plant is represented by P , the performance function by $f(x(t))$, the extremum seeking control algorithm by “ESC Algorithm”

For each iteration of the algorithm, the following steps occur. They will be explained in detail in the remainder of this chapter.

1. Measure the current independent and dependent parameter of the performance function. These are the outputs of P and $f(x)$.
2. Estimate the best-fit parameters of the performance function given the assumed functional form and a history of measurements. This occurs in the “ESC Algorithm” block.
3. Calculate an estimate of the performance function’s extremum coordinates. This also occurs in the “ESC Algorithm” block.
4. Command the plant with a command (u_c) formed from the extremum coordinates estimate.
5. Continue at step 1.

In the following sections, Figure 4.1 will be expanded as details of the algorithm are discussed. In particular, the “ESC Algorithm” block will be broken into sub-blocks, and a persistent-excitation input will be added.

4.3 Extremum location estimation

In order to facilitate estimation of the performance function extremum position, the method assumes that the performance function can be approximated by a quadratic function at any point. This is accomplished by parameterizing the performance function using first-order and second-order terms of a Taylor series expansion.

Consider the Taylor series expansion of a performance function $f(x)$ centered at any arbitrary point about a particular point x_k

$$\begin{aligned} f(x) &\approx f(x_k) + b_k^T(x - x_k) \\ &+ \frac{1}{2}(x - x_k)^T M_k(x - x_k) + o(x - x_k) \end{aligned} \quad (4.3.1)$$

where b_k is the gradient at x_k , M_k is the Hessian at x_k , and $o(\cdot)$ represents higher order terms. The origin of the coordinate system can be shifted to the point x_k by rewriting equation 4.3.1 in terms of distance between x_k and any other point as

$$\Delta f_k = b_k^T \Delta x_k + \frac{1}{2} \Delta x_k^T M_k \Delta x_k \quad (4.3.2)$$

where

$$\Delta x_k = x - x_k \quad (4.3.3)$$

and

$$\Delta f_k = f(x) - f(x_k). \quad (4.3.4)$$

For notational simplicity, the dependence of Δf_k on x is implied. The higher-order terms have been dropped by assuming the performance function is adequately modeled as a quadratic function at any particular position.

At the extremum coordinates, the derivative of equation (4.3.2) with respect to Δx_k is equal to zero .

$$\frac{\partial \Delta f_k}{\partial \Delta x_k} = 0 \quad (4.3.5)$$

Performing the derivative equation (4.3.2) becomes

$$\frac{\partial \Delta f_k}{\partial \Delta x_k} = b_k^T + \Delta x_k^T M_k \quad (4.3.6)$$

Setting the derivative equal to zero and solving for Δx_k , we find

$$\Delta x_k = -b_k^T M_k^{-1} \quad (4.3.7)$$

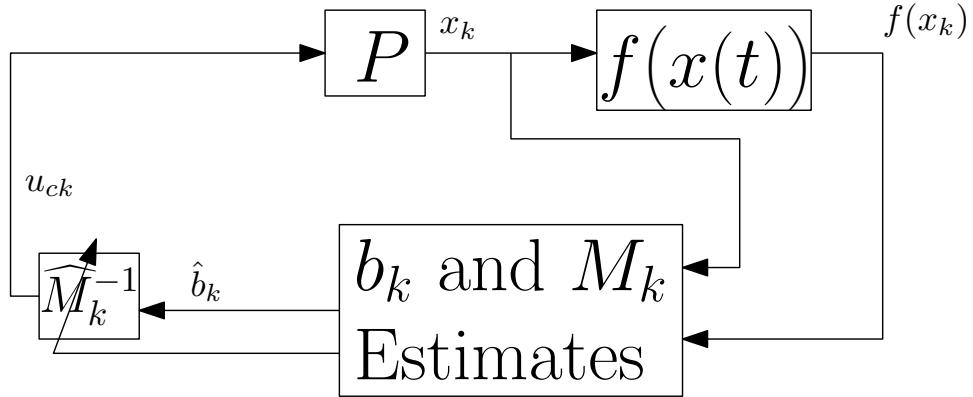
The coordinates of the performance function's extremum are then found, by using equation 4.3.3, to be

$$x_{k+1} = x_k - b_k^T M_k^{-1} \quad (4.3.8)$$

It was no accident that the estimated extremum position is denoted as x_{k+1} . This notation was chosen so that this development would flow into an iterative algorithm. This algorithm is, of course, the Newton-Raphson optimization scheme.

While the Newton Raphson method will form the framework of the extremum seeking control scheme, it can not solve the extremum seeking control problem alone. A complete scheme must have a method to estimate b_k and M_k and deal with finite time response of the plant commands.

Figure 4.2 depicts the extremum seeking scheme interconnection including the Newton-Raphson scheme. It has been expanded from figure 4.1 by replacing the “ESC Algorithm” block with a “ b_k and M_k Estimates” block and a block denoting the M_k estimate.



4.2: Algorithm Interconnection including the Newton-Raphson scheme. The Plant is represented by P , the performance function by $f(x(t))$, the gradient estimate by \hat{b}_k , and the Hessian estimate by \widehat{M}_k

The “ b_k and M_k Estimates” block represent the gradient and Hessian estimate. The command to the plant is given by

$$u_{ck} = -\hat{b}_k^T \widehat{M}_k^{-1} \quad (4.3.9)$$

The $\hat{\cdot}$ has been added to indicate estimates of b_k and M_k are used to form the command. The subscript k has also been added to denote the discrete nature of the measurements and estimate.

4.4 Gradient and Hessian Estimate

The gradient and Hessian of the performance function, in this extremum-seeking method, are estimated from noisy measurements of the performance function independent and dependent parameters. It is desirable to use an estimation scheme which minimizes the estimation variance. Given the iterative nature of extremum-seeking problem, it is also desirable to use a recursive estimation scheme. The Kalman filter meets these desires. In addition, the vast body of work and variations on the Kalman filter provide a wealth of design techniques which this extremum-seeking method can utilize. This makes the method flexible and able to solve a vast array of problems. In this section we design the Kalman filter used to estimate the gradient and Hessian of the performance function. We will determine that a time-varying Kalman filter is a convenient and advantageous choice.

We begin with equation 4.3.2 repeated here

$$\Delta f_k = b_k^T \Delta x_k + \frac{1}{2} \Delta x_k^T M_k \Delta x_k \quad (4.4.1)$$

For the sake of notational simplicity, the following is restricted to the case of two independent parameters x_1 and x_2 . The discussion can be easily extended to n independent parameters; however, the notational requirements become quite burdensome. Denote the measurements of the two parameters at time k as x_{1k} and x_{2k} . Denote the components of the performance function gradient at $(x_{1k} x_{2k})$ as b_{1k} and b_{2k} . Further denote the Hessian of the performance function at $(x_{1k} x_{2k})$ as

$$M_k = \begin{bmatrix} M_{11_k} & M_{12_k} \\ M_{12_k} & M_{22_k} \end{bmatrix}.$$

The difference between measurements of x_1 and x_2 are denoted as

$$\begin{aligned} \Delta x_{1k} &= x_{1k-1} - x_{1k} \\ \Delta x_{2k} &= x_{2k-1} - x_{2k}. \end{aligned}$$

Equation (4.4.1) is then written in a more convenient form as

$$\Delta f_k = \begin{bmatrix} \frac{1}{2}\Delta x_{1k}^2 & \frac{1}{2}\Delta x_{2k}^2 & \Delta x_{1k}\Delta x_{2k} & \Delta x_{1k} & \Delta x_{2k} \end{bmatrix} \begin{bmatrix} M_{11_k} \\ M_{22_k} \\ M_{12_k} \\ b_{1_k} \\ b_{2_k} \end{bmatrix}. \quad (4.4.2)$$

The form of Equation (4.4.2) makes clear that a parameter estimation technique may be used to estimate the gradient and Hessian. In fact, if five measurements are available, the resulting system of equations will be fully determined and a simple matrix inversion will provide a least-squares estimate of the gradient and Hessian. A recursive-least-squares algorithm could be used in a more computationally efficient implementation to provide estimates. Such methods do not take noise into account however. A better approach is to use a Kalman filter to estimate the gradient and Hessian.

Since it is unlikely that the performance function is truly quadratic in form, the quadratic assumption will prove inaccurate. Instead, it is expected that the gradient and Hessian of the performance function will change with the coordinates of the independent axes according to a random-walk. This inaccuracy is modeled by a zero-mean Gaussian white-noise process w_k with variance W_k . The gradient and Hessian are then expected to change with time according to

$$\zeta_{k+1} = I\zeta_k + w_k \quad (4.4.3)$$

where I is a 5×5 identity matrix for this $N = 2$ case. and

$$\zeta_k = \begin{bmatrix} M_{11_k} \\ M_{22_k} \\ M_{12_k} \\ b_{1_k} \\ b_{2_k} \end{bmatrix}.$$

The size of ζ_k is a function of N . It is simple to determine that

$$\text{size}(\zeta_k) = \frac{N^2 + 3N}{2}$$

This sufficiently models the random-walk nature of the gradient and Hessian with changing x .

The measurements of Δx_{1_k} , and Δx_{2_k} and Δf_k are expected to be noisy. The noise associated with the measurement of Δf_k is modeled with a zero-mean Gaussian white-noise process v_k with variance V_k . The noise on the measurements of Δx_{1_k} , and Δx_{2_k} is, for the time being, assumed to be zero. The implications of this assumption will be discussed in the Chapter 5 where convergence is proven for the case of this assumption being false.

The measurement equation is then constructed as

$$\Delta f_k = H_k \zeta_k + v_k \quad (4.4.4)$$

where

$$H_k = \begin{bmatrix} \frac{1}{2}\Delta x_{1_k}^2 & \frac{1}{2}\Delta x_{2_k}^2 & \Delta x_{1_k}\Delta x_{2_k} & \Delta x_{1_k} & \Delta x_{2_k} \end{bmatrix} \quad (4.4.5)$$

If a restriction is imposed that Δx_{1_k} and Δx_{2_k} are constant with every iteration, then a steady-state Kalman filter could be used. This would prove to be a very limiting restriction; when the extremum coordinates are approached, this restriction would cause poor performance. A time-varying Kalman filter is therefore required in which new values of Δx_{1_k} and Δx_{2_k} can be used at every iteration.

Equations (4.4.3) and (4.4.4) constitute the process equation and the measurement equation of the estimation problem. Their components are implemented into the Kalman filter equations (3.5.1) of Chapter 3.

Since Φ_k of (3.1.1a) is an identity matrix, (3.5.1d) becomes

$$\bar{\zeta}_{k+1} = \hat{\zeta}_k$$

Since Γ_k is also an identity matrix, (3.5.1e) becomes

$$M_{k+1} = P_k + W_k$$

The linear time-varying Kalman filter is then implemented with the following simplified set of equations:

$$\hat{P}_{k+1} = \bar{P}_k + W_k \quad (4.4.6a)$$

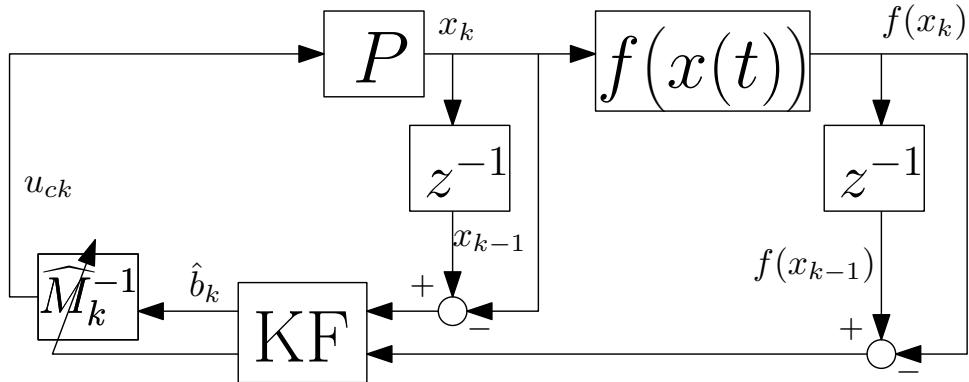
$$\hat{\zeta}_{k+1} = \hat{\zeta}_k + \bar{P}_k H_k^T V_k^{-1} (\Delta f_k - H_k \hat{\zeta}_k) \quad (4.4.6b)$$

$$\bar{P}_k = (\hat{P}_k^{-1} + H_k^T V_k^{-1} H_k)^{-1} \quad (4.4.6c)$$

where \bar{P} is the state covariance matrix, \hat{P} the predicted state covariance matrix, and ζ the state vector.

All of the components in (4.4.6) are either determined by the model or calculated in the Kalman filter except for the noise components W_k and V_k . These can therefore be used as tuning parameters. Typically, initial values of W_k and V_k are based upon previously obtained measurements of the noise in the system or upon a noise model. A trial-and-error process is then used to adjust the values in order to improve the estimates. In general, the values chosen must reflect the accuracies of the process and measurement equations. In the case where the measurements are relatively noise free, V_k should be set to some small value. In the case where the performance function is very close to a true quadratic function, W_k should be set to some small value. A discussion of Kalman filter tuning can be found in [SC08]. Detailed derivations of the linear time-varying Kalman filter can be found in [BH97], [SC08], and [GA01].

Figure 4.3 depicts the algorithm interconnection including the time varying Kalman filter. It



4.3: Algorithm Interconnection including the Kalman Filter. The Plant is represented by P , the performance function by $f(x(t))$, gradient by b_k , Hessian by M_k , and Kalman filter by KF

is updated from Figure 4.2. The “ b_k and M_k Estimates” block has been replaced with a block denoting the Kalman filter and items indicating the Kalman filter inputs. Measurements of Δx_1 , and Δf_k are depicted through the use of subtractions and unit delay operators z^{-1} . The Kalman filter is denoted by “KF”. It estimates the gradient (b_k) and Hessian (M_k). These estimates are combined to form the command

$$u_{ck} = -\hat{b}_k^T \hat{M}_k^{-1} \quad (4.4.7)$$

which is the input to P .

4.4.1 Persistent Excitation

In order to ensure observability of the performance function, persistent excitation of the system is required. The only requirement of the persistent excitation is to provide observability of the performance function. This implies that the persistent excitation must span the independent parameter space. This is an intuitively obvious requirement; if measurements are not taken in the direction of an independent parameter, it would then be impossible to discern a gradient in that direction.

The persistent excitation may take a multitude of forms. It may take the form of separate signals commanding the plant in each dimension of the independent parameter space. Such signals are not required to possess frequency separation from each other, as is common in other extremum seeking schemes. The persistent excitation may be a function of time or a function of the independent parameters. Its form is typically determined by details of the specific application and may take whatever convenient form the designer chooses.

It is typically inserted into the system at the plant input due to convenience; however, this is not a requirement. It may be inserted at any point in the interconnection as long as it will span the independent parameter space. Further discussion of the persistent excitation signal can be found in Appendix C.

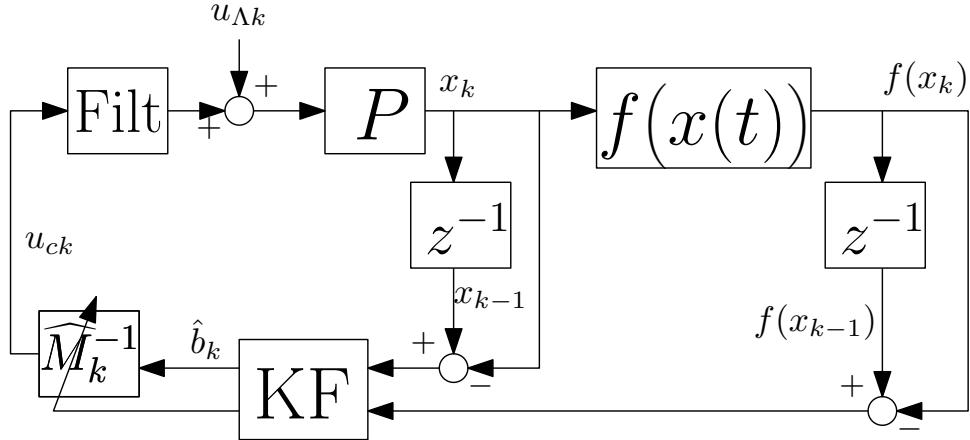
4.4.2 Command Filter

While not necessary for completing a successful design, command filtering is useful to smooth system performance. Due to the discrete nature of the time-varying Kalman filter, the command x_c invariably takes the form of a series of step commands. Depending on the plant command response, it may be advantageous to smooth the step commands. This can help avoid actuator saturation and provide a more desirable plant command response.

The command filter is usually chosen as a low pass filter which will smooth the command to one which the plant can more easily track. Further discussion of the command filter can be found in Appendix C.

Figure 4.4 depicts the algorithm interconnection including the persistent excitation signal and the command filter. It is updated from Figure 4.3 and depicts the full extremum seeking control interconnection. The command filter is represented by “Filt” and the persistent excitation is

depicted by u_{Λ_k} .



4.4: Algorithm Interconnection including the command filter and the persistent excitation signal. The Plant is represented by P , the performance function by $f(x(t))$, gradient by b_k , Hessian by M_k , Kalman filter by KF, the persistent excitation by u_{Λ_k} , and the command filter by Filt.

4.4.3 System Initialization

4.5 Summary of the Extremum Seeking Control Method

The extremum seeking method is now summarized.

1. Initialize the system
 - (a) Take measurements of the current independent and dependent coordinates of the performance function
 - (b) Initialize the time-varying Kalman filter by providing initial values for the time-varying Kalman filter's covariance matrix.
 - (c) Form an initial estimation of the coordinates of the performance function's extremum¹.
2. Command the plant with the estimate of the performance function's extremal coordinates

¹The initial command may be generated by taking a series of measurements of the independent parameters and dependent parameter at a number of coordinates on the performance function and using the time-varying Kalman filter to estimate the extremal position.

3. Set $k = k + 1$
4. Take measurements of the independent and dependent coordinates
5. Calculate gradient and Hessian estimates with the time-varying Kalman filter
 - (a) Form the measurement matrix as in equation 4.4.5

$$H_k = \begin{bmatrix} \frac{1}{2}\Delta x_{1_k}^2 & \frac{1}{2}\Delta x_{2_k}^2 & \Delta x_{1_k}\Delta x_{2_k} & \Delta x_{1_k} & \Delta x_{2_k} \end{bmatrix}$$

- (b) Iterate the Kalman filter with the equations listed in Equation 4.4.6

$$\begin{aligned} \hat{P}_{k+1} &= \bar{P}_k + W_k \\ \hat{\zeta}_{k+1} &= \hat{\zeta}_k + \bar{P}_k H_k^T V_k^{-1} (\Delta f_k - H_k \zeta_k) \\ \bar{P}_k &= (\hat{P}_k^{-1} + H_k^T V_k^{-1} H_k)^{-1} \end{aligned}$$

- (c) Form \hat{b}_k and \widehat{M}_k from ζ_{k+1}
6. Compute the extremum coordinate estimate u_{ck} per equation 4.4.7.

$$u_{ck} = \widehat{M}_k^{-1} \hat{b}_k$$

7. Smooth u_{ck} and combine with a persistent excitation signal
8. Continue with step 2

4.6 Modifications to the Method

There are two primary modifications to the extremum seeking method which are discussed in this section. The first implements a sliding window to expand the number of measurements used in the Kalman filter to estimate the gradient and Hessian. It has the effect of further reducing the influence of noise on the estimations and, in some cases, slows convergence.

The second modification replaces the Newton-Raphson approach with a steepest descent approach. It reduces the complexity of the Kalman filter but can not estimate the coordinates of the extremum; it can only command the plant in the direction pointing towards the coordinates of the extremum.

4.6.1 Sliding Window Modification

A sliding window may be added to the extremum seeking algorithm design such that the difference between \mathcal{N} measurements are used in the estimation at each iteration k . Define the difference between a measure of $f(x)$ at iteration k and iteration $k - n$ as

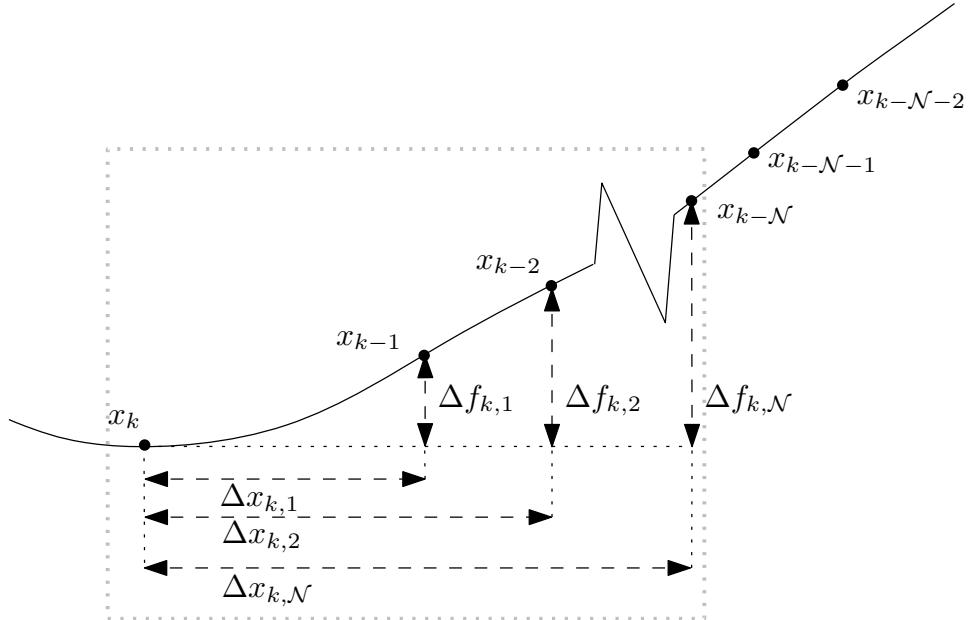
$$\Delta f_{k,n} = f(x_k) - f(x_{k-n})$$

Define the difference between a measure of x at iteration k and iteration $k - n$ as

$$\Delta x_{k,n} = x_k - x_{k-n}$$

Components of $\Delta x_{k,n}$ are denoted with a subscript. For example $\Delta x_{1_{k,1}}$ denotes first component of $\Delta x_{k,n}$ and $\Delta x_{2_{k,1}}$ the second.

The sliding window is easily depicted for the case of one independent parameter ($f(x) : \mathbb{R} \rightarrow \mathbb{R}$) as shown in Figure 4.5. Here the sliding window is of size \mathcal{N} . The performance function is indicated



4.5: Depiction of sliding window implementation. The solid line denotes $f(x) : \mathbb{R} \rightarrow \mathbb{R}$. The dashed lines denote the difference between measurements. The sliding window covers \mathcal{N} measurements. Measurements that occurred before \mathcal{N} are not included in the sliding window.

by a solid line with a zigzag indicating an omitted segment. The history of measurements of x is indicated with subscripts, with x_k denoting the latest. The corresponding measurements of

$f(x)$ are not shown on the figure, but implied by the height of the performance function. The difference between measurements included in the sliding window is depicted as $\Delta f_{k,n}$ and $\Delta x_{k,n}$ with n ranging from 1 to N . The sliding window is depicted by a thick grey dotted square.

In order to facilitate discussion of the incorporation of the sliding window into the Kalman filter, we return to the case of two independent parameters. The difference between measurements for the two parameters is indicated with subscripts. We define

$$\Delta f_{k,n} = f(x_{1_k}, x_{2_k}) - f(x_{1_{k-n}}, x_{2_{k-n}}) \quad (4.6.1a)$$

$$\Delta x_{1_{k,n}} = x_{1_k} - x_{1_{k-n}} \quad (4.6.1b)$$

$$\Delta x_{2_{k,n}} = x_{2_k} - x_{2_{k-n}} \quad (4.6.1c)$$

We also define $v_{k,n}$ as the white noise process associated with each measurement.

The expansion is implemented by modifying the measurement equation (4.4.4) as

$$\begin{bmatrix} \Delta f_{k,1} \\ \Delta f_{k,2} \\ \vdots \\ \Delta f_{k,N} \end{bmatrix} = H_k \zeta_k + \begin{bmatrix} v_{k,1} \\ v_{k,2} \\ \vdots \\ v_{k,N} \end{bmatrix}. \quad (4.6.2)$$

Here

$$H_k = \begin{bmatrix} \frac{1}{2}\Delta x_{1_{k,1}}^2 & \frac{1}{2}\Delta x_{2_{k,1}}^2 & D_{k,1} & \Delta x_{1_{k,1}} & \Delta x_{2_{k,1}} \\ \frac{1}{2}\Delta x_{1_{k,2}}^2 & \frac{1}{2}\Delta x_{2_{k,2}}^2 & D_{k,2} & \Delta x_{1_{k,2}} & \Delta x_{2_{k,2}} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{1}{2}\Delta x_{1_{k,N}}^2 & \frac{1}{2}\Delta x_{2_{k,N}}^2 & D_{k,N} & \Delta x_{1_{k,N}} & \Delta x_{2_{k,N}} \end{bmatrix}$$

and $D_{k,n} = \Delta x_{1_{k,n}} \Delta x_{2_{k,n}}$. The index n takes values between 1 and N . The process equation remains as it is shown in equation (4.4.3). The Kalman filter is implemented as shown in Equation (4.4.6) with

$$\Delta f_k = \begin{bmatrix} \Delta f_{k,1} \\ \Delta f_{k,2} \\ \vdots \\ \Delta f_{k,N} \end{bmatrix}$$

and

$$V_k = \begin{bmatrix} V_{k,1} & 0 & \cdots & 0 \\ 0 & V_{k,2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & V_{k,N} \end{bmatrix}$$

where $V_{k,n}$ is the variance of $v_{k,n}$.

The benefit of increasing the number of measurement differences in the Kalman filter is that it increases the tolerance to noisy measurements; however a too-large N may slow convergence. This is discussed further and demonstrated in chapter 6. The number of measurements is used as a tuning parameter.

4.6.2 Gradient Only Modification

Another modification is to use a time-varying Kalman filter which only estimates the gradient and commands the plant in the steepest descent direction. While this modification is in many ways trivial, it is useful to highlight the most important changes to the algorithm.

The time-varying Kalman filter is designed in the same manner as that described in section 4.4 except the design begins with

$$\Delta f_k = b_k^T \Delta x_k$$

instead of equation 4.3.2. Again restricting the discussion to the case of two independent parameters, the equation analogous to 4.4.2 is quickly found to be

$$\Delta f_k = \begin{bmatrix} \Delta x_{1_k} & \Delta x_{2_k} \end{bmatrix} \zeta_k$$

where

$$\zeta_k = \begin{bmatrix} b_{1_k} \\ b_{2_k} \end{bmatrix}.$$

The obvious implication is that the size of ζ_k

$$\text{size}(\zeta_k) = N$$

Equation 4.4.4 retains the same form

$$\Delta f_k = H_k \zeta_k + v_k$$

but

$$H_k = \begin{bmatrix} \Delta x_{1k} & \Delta x_{2k} \end{bmatrix}$$

The time-varying Kalman filter is again implemented with equation 4.4.6. The gradient estimate is combined with a constant gain K to form the command

$$u_{ck} = -K\hat{b}_k. \quad (4.6.1)$$

CHAPTER 5

Convergence of Method

This chapter shows that the the extremum seeking method will converge while under the influence of noise. It is shown that the extremum seeking method will convergence to an area around the extremum defined by the noise on the system.

The system considered is limited to the case of estimating the performance function gradient, hence to using a steepest descent algorithm. It is assumed that the plant perfectly follows the command such that

$$x_{k+1} = x_k + K\hat{b}_k. \quad (5.0.1)$$

This assumption is easily realized by iterating the Kalman filter at a rate slow enough to allow the plant to reach the command generated by the algorithm.

Convergence is shown by first proving that the time varying Kalman filter's error covariance in bounded from above. A supermartingale is then constructed to show that the steepest descent algorithm utilizing the Kalman filter gradient estimate will converge to an area around the performance function's extremum defined, in part, by the Kalman filter's error covariance bound.

In what follows, a noiseless value of H_k is denoted as H_k^* and the noise included on H_k as \tilde{H}_k such that $H_k = H_k^* + \tilde{H}_k$. Similarly, the true value of the gradient is denoted as b_k^* and error between the estimate \hat{b}_k and b_k^* as e_k .

5.1 Convergence of the Kalman filter

We begin our considerations by discussing convergence properties of the Kalman filter. Jazwinski [Jaz70] shows that the error covariance of the Kalman filter is uniformly bounded from above and below. Jazwinski does not consider however, the case of noise on the observation matrix H_k . The upper-bound theorem presented in Jazwinski is expanded here to include noise on H_k . The proof of the new theorem follows that in Jazwinski quite closely.

Theorem 5.1.1. Consider the dynamical system in equation 3.1.1

$$\zeta_{k+1} = \Phi_k \zeta_k + \Gamma_k w_k \quad (5.1.1a)$$

$$\Delta f_k = H_k \zeta_k + v_k. \quad (5.1.1b)$$

If the system is uniformly completely observable and uniformly completely controllable, and if the initial value of error covariance matrix $P_0 \geq 0$, then P_k is uniformly bounded from above for all $k \geq N$

$$P_k \leq \mathcal{I}^{-1}(k, k-N) \left(I + \Phi^T(i, k) E \left[\tilde{H}_i^T R_i^{-T} \tilde{H}_i \right] \Phi(i, k) \right) + \mathcal{C}(k, k-N), \quad k \geq N$$

where \mathcal{I} is the information matrix and \mathcal{C} is the controllability matrix given by

$$\mathcal{I}(k, 1) = \sum_{i=1}^k \Phi^T(i, k) H_i^{*T} R_i^{-1} H_i^* \Phi(i, k) \quad (5.1.2)$$

$$\mathcal{C} = \sum_{i=0}^k \Phi(k, i+1) \Gamma_i Q_{i+1} \Gamma_i^T \Phi^T(i, k)$$

respectively.

Proof. Consider the least squares estimate of ζ_k based on the most recent N observations, ignoring the system noise w_k and the noise on H_k ,

$$\bar{\zeta}_k^N = \mathcal{I}^{-1}(k, k-N) \sum_{i=k-N}^k \Phi^T(i, k) H_i^T R_i^{-1} y_i, \quad k \geq N$$

This estimate is suboptimal; it is *not* the minimum variance estimate. As a consequence,

$$P_k \leq E \left[(\zeta_k - \bar{\zeta}_k^N) (\zeta_k - \bar{\zeta}_k^N)^T \right] \quad (5.1.3)$$

We compute the covariance in 5.1.3. We can easily show

$$\zeta_i = \Phi(i, k) \zeta_k - \Phi(i, k) \sum_{j=1}^{k-1} \Phi(k, j+1) \Gamma(j) w_{j+1}$$

We can then write

$$\Delta f_i = H_i \Phi(i, k) \zeta_k + v_i - H_i \Phi(i, k) \sum_{j=i}^{k-1} \Phi(k, j+1) \Gamma(j) w_{j+1}$$

Therefore

$$\begin{aligned}\zeta_k - \bar{\zeta}_k^N &= -\mathcal{I}^{-1}(k, k-N) \sum_{i=k-N}^k \Phi^T(i, k) H_i^T R_i^{-1} v_i \\ &\quad + \mathcal{I}^{-1}(k, k-N) \sum_{i=k-N}^k \Phi^T(i, k) H_i^T R_i^{-1} H_i \Phi(i, k) \\ &\quad \times \sum_{j=1}^{k-1} \Phi(k, j+1) \Gamma(j) w_{j+1}\end{aligned}$$

The covariance

$$E\left[\left(\zeta_k - \bar{\zeta}_k^N\right) \left(\zeta_k - \bar{\zeta}_k^N\right)^T\right]$$

is now computed. Completing the multiplication inside the expectation, we write

$$\begin{aligned}E\left[&\mathcal{I}^{-1}(k, k-N) \left(\sum_{i=k-N}^k \Phi^T(i, k) H_i^T R_i^{-1} v_i v_i^T R_i^{-T} H_i \Phi(i, k)\right) \mathcal{I}^{-T}(k, k-N)\right. \\ &- \left(\mathcal{I}^{-1}(k, k-N) \sum_{i=k-N}^k \Phi^T(i, k) H_i^T R_i^{-1} v_i\right) \\ &\times \left(\mathcal{I}^{-1}(k, k-N) \sum_{i=k-N}^k \Phi^T(i, k) H_i^T R_i^{-1} H_i \Phi(i, k) \sum_{j=1}^{k-1} \Phi(k, j+1) \Gamma(j) w_{j+1}\right)^T \\ &- \left(\mathcal{I}^{-1}(k, k-N) \sum_{i=k-N}^k \Phi^T(i, k) H_i^T R_i^{-1} H_i \Phi(i, k) \sum_{j=1}^{k-1} \Phi(k, j+1) \Gamma(j) w_{j+1}\right) \\ &\times \left(\mathcal{I}^{-1}(k, k-N) \sum_{i=k-N}^k \Phi^T(i, k) H_i^T R_i^{-1} v_i\right)^T \\ &+ \left(\mathcal{I}^{-1}(k, k-N) \sum_{i=k-N}^k \Phi^T(i, k) H_i^T R_i^{-1} H_i \Phi(i, k) \sum_{j=1}^{k-1} \Phi(k, j+1) \Gamma(j) w_{j+1}\right) \\ &\left.\times \left(\mathcal{I}^{-1}(k, k-N) \sum_{i=k-N}^k \Phi^T(i, k) H_i^T R_i^{-1} H_i \Phi(i, k) \sum_{j=1}^{k-1} \Phi(k, j+1) \Gamma(j) w_{j+1}\right)^T\right] \quad (5.1.4)\end{aligned}$$

Consider the first term of Equation 5.1.4. Using $H_i = H_i^* + \tilde{H}_i$ it is expanded to

$$\begin{aligned}\mathcal{I}^{-1}(k, k-N) \left(&\sum_{i=k-N}^k \Phi^T(i, k) H_i^{*T} R_i^{-T} H_i^* \Phi(i, k)\right. \\ &\left.+ \Phi^T(i, k) E\left[\tilde{H}_i^T R_i^{-T} \tilde{H}_i\right] \Phi(i, k)\right) \mathcal{I}^{-T}(k, k-N) \quad (5.1.5)\end{aligned}$$

where we used the fact that $E\left[\tilde{H}_i\right] = 0$. Using Equation 5.1.2 this reduces to

$$\mathcal{I}^{-1}(k, k-N) \left(I + \Phi^T(i, k) E\left[\tilde{H}_i^T R_i^{-T} \tilde{H}_i\right] \Phi(i, k) \mathcal{I}^{-T}(k, k-N)\right)$$

The second and third terms of Equation 5.1.4 are zero since v_i and w_{j+1} are independent and $\text{E}[v_i] = \text{E}[w_{j+1}] = 0$. We are then left with

$$\begin{aligned}
& \text{E}\left[\left(\zeta_k - \bar{\zeta}_k^N\right) \left(\zeta_k - \bar{\zeta}_k^N\right)^T\right] \\
&= \mathcal{I}^{-1}(k, k - N) \left(I + \Phi^T(i, k) \text{E}\left[\tilde{H}_i^T R_i^{-T} \tilde{H}_i\right] \Phi(i, k) \mathcal{I}^{-T}(k, k - N) \right) \\
&\quad + \text{cov} \left[\left(\mathcal{I}^{-1}(k, k - N) \sum_{i=k-N}^k \Phi^T(i, k) H_i^T R_i^{-1} H_i \Phi(i, k) \sum_{j=1}^{k-1} \Phi(k, j+1) \Gamma(j) w_{j+1} \right) \right] \\
&\leq \mathcal{I}^{-1}(k, k - N) \left(I + \Phi^T(i, k) \text{E}\left[\tilde{H}_i^T R_i^{-T} \tilde{H}_i\right] \Phi(i, k) \mathcal{I}^{-T}(k, k - N) \right) \\
&\quad + \text{cov} \left[\left(\mathcal{I}^{-1}(k, k - N) \sum_{i=k-N}^k \Phi^T(i, k) H_i^T R_i^{-1} H_i \Phi(i, k) \sum_{j=k-N}^{k-1} \Phi(k, j+1) \Gamma(j) w_{j+1} \right) \right] \\
&= \mathcal{I}^{-1}(k, k - N) \left(I + \Phi^T(i, k) \text{E}\left[\tilde{H}_i^T R_i^{-T} \tilde{H}_i\right] \Phi(i, k) \mathcal{I}^{-T}(k, k - N) \right) + \mathcal{C}(k, k - N)
\end{aligned}$$

Hence

$$\begin{aligned}
P_k &\leq \mathcal{I}^{-1}(k, k - N) \left(I + \Phi^T(i, k) \text{E}\left[\tilde{H}_i^T R_i^{-T} \tilde{H}_i\right] \Phi(i, k) \mathcal{I}^{-T}(k, k - N) \right) \\
&\quad + \mathcal{C}(k, k - N), \quad k \geq N
\end{aligned}$$

□

This result differs from that provided in [Jaz70] only in the addition of the term

$$\mathcal{I}(k, k - N) \Phi^T(i, k) \text{E}\left[\tilde{H}_i^T R_i^{-T} \tilde{H}_i\right] \Phi(i, k) \mathcal{I}^{-T}(k, k - N).$$

5.2 Convergence of the extremum seeking algorithm

We are now in a position to make a statement regarding the convergence of the extremum seeking algorithm. We limit ourselves to only estimating the gradients in the Kalman filter, thus we are limiting consideration to the algorithms use of a steepest descent algorithm.

To show convergence, we show that the sequence $f(x_k)$ is a supermartingale. We further show that convergence will no longer occur once the system reaches an area around the performance function's extremum defined, in part, by the Kalman filter's error covariance bound.

Theorem 5.2.1. *Consider the extremum seeking control algorithm developed in Chapter 4. The algorithm will converge to a radius around the extremum position if*

$$K^T M K + 2K^T < 0$$

and

$$\hat{b}_k^T (K^T M K + 2K) \hat{b}_k + \text{trace } D V K P_k K^T V^T D < 0$$

where P_k is the error covariance of the gradient estimate discussed in Theorem 5.1.1, V is the eigenvector matrix of M and D is the diagonal matrix of the square root of the eigenvalues of M .

Proof. The proof is completed by showing that the sequence generated by $f(x_k)$ is a supermartingale. That is, we show that the stochastic process $f(x_k)$ satisfies

- $\mathbb{E}[f(x_k)] < \infty$, $k \geq 0$ and
- $\mathbb{E}[f(x_{k+1})|\mathcal{F}] < f(x_k)$, $k \geq 0$

where \mathcal{F}_k is the completion of the σ -field in probability space generated by the history of measurements of $f(x)$ and x up to step k . The first item is true by definition of the performance function. The second item is shown in the following.

Consider the equation used to generate x_{k+1} displayed in equation 5.0.1

$$x_{k+1} = x_k + K\hat{b}_k.$$

Here \hat{b}_k is the Kalman filter estimate of b_k^* . The estimate will always differ from the true value by some amount. To account for this difference, define $e_k = \hat{b}_k - b_k^*$. Then

$$x_{k+1} = x_k + K(\hat{b}_k + e_k).$$

The algorithm assumes a quadratic form of the performance function as defined in 4.3.2

$$f(x) - f(x_k) \approx \frac{1}{2} (x - x_k)^T M (x - x_k) + (x - x_k)^T \hat{b}_k$$

By setting $x = x_{k+1}$ we find

$$\begin{aligned} f(x_{k+1}) - f(x_k) &= \frac{1}{2} (\hat{b}_k + e_k)^T K^T M K (\hat{b}_k + e_k) + (\hat{b}_k + e_k)^T K^T \hat{b}_k \\ &= \frac{1}{2} (\hat{b}_k + e_k)^T K^T M K (\hat{b}_k + e_k) + \hat{b}_k K^T \hat{b}_k + e_k^T K^T \hat{b}_k \\ &= \frac{1}{2} \hat{b}_k^T K^T M K \hat{b}_k + \frac{1}{2} e_k^T K^T M K e_k + \hat{b}_k^T K \hat{b}_k + e_k^T K^T \hat{b}_k \\ &= \frac{1}{2} \hat{b}_k^T (K^T M K + 2K) \hat{b}_k + \frac{1}{2} e_k^T K^T M K e_k + e_k^T K^T \hat{b}_k \end{aligned}$$

For the case of $e_k = 0$, the algorithm will converge if

$$K^T M K + 2K^T < 0 \quad (5.2.1)$$

In this case the condition for convergence does not involve \hat{b}_k . When $e_k \neq 0$, \hat{b}_k becomes entangled in the question of convergence. Applying the conditional expectation we find

$$\mathbb{E}[f(x_{k+1}) - f(x_k) | \mathcal{F}] = \frac{1}{2} \hat{b}_k^T (K^T M K + 2K) \hat{b}_k + \frac{1}{2} \text{trace } DV K P_k K^T V^T D$$

Here V is the eigenvector matrix of M , and D is the diagonal matrix of the square root of the eigenvalues of M . We used the fact that $\mathbb{E}[\hat{b}_k e_k | \mathcal{F}] = 0$, and the error covariance of \hat{b}_k , $P_k = \mathbb{E}[e_k e_k^T | \mathcal{F}]$ as discussed in Theorem 5.1.1. A second condition for convergence is then

$$\hat{b}_k^T (K^T M K + 2K) \hat{b}_k + \text{trace } DV K P_k K^T V^T D < 0$$

This completes the proof. \square

The significance of this condition is not readily seen in this form. By assuming M is positive definite and choosing the gain $K = -M^{-1}$ we can more easily recognize the significance. We find

$$\mathbb{E}[f(x_{k+1}) - f(x_k) | \mathcal{F}] = \frac{1}{2} \text{trace } DV \left(-\hat{b} \hat{b}^T + P_k \right) V^T D.$$

This condition indicates that the magnitude of the gradient must be greater than the magnitude of the error covariance for convergence to continue. This makes intuitive sense; if the algorithm can not determine the direction of the gradient, it can not command the system in a direction to continue convergence. This defines a boundary around the minimum point around which the algorithm will no longer converge.

CHAPTER 6

One Dimensional Example

This chapter provides a simple example of extremum seeking control system design using the extremum-seeking control method discussed in the previous chapters. It begins with a description of the example extremum-seeking control problem. It then describes the design of an inner loop controller and an extremum-seeking control loop. Further discussion about the various design trade-offs associated with this extremum seeking method can be found in Appendix C.

6.1 Extremum-Seeking Control Example

To demonstrate the extremum-seeking control method, an example is presented in which the performance function $f(x) : \mathbb{R} \rightarrow \mathbb{R}$ has one independent and one dependent parameter. The performance function is chosen to be $f(x) = (\cos(x/8.4) + 1.5)(x/6 - 0.4)^2$. This performance function provides a gradient and Hessian which change as a function of the performance functions independent parameter. This performance function can be seen in Figure 6.8. At $x = 20$ the performance function has a small gradient and negative curvature. At $x = 10$ the gradient is much larger and the curvature switches to a positive value. In the simulations that follow, the system is initialized at $x = 22$. The Kalman filter is then forced to estimate both positive and negative Hessian values as the plant moves toward the minimum. Measurements of the performance function magnitude are corrupted with Gaussian distributed noise $\mathcal{N}(0, 0.01)$. There is no noise imposed upon the measurements of the performance function's independent parameter.

The plant model is taken directly from [NS02]. A continuous linear plant, it is given by

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -5 & -9 & -5 \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad C = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$$

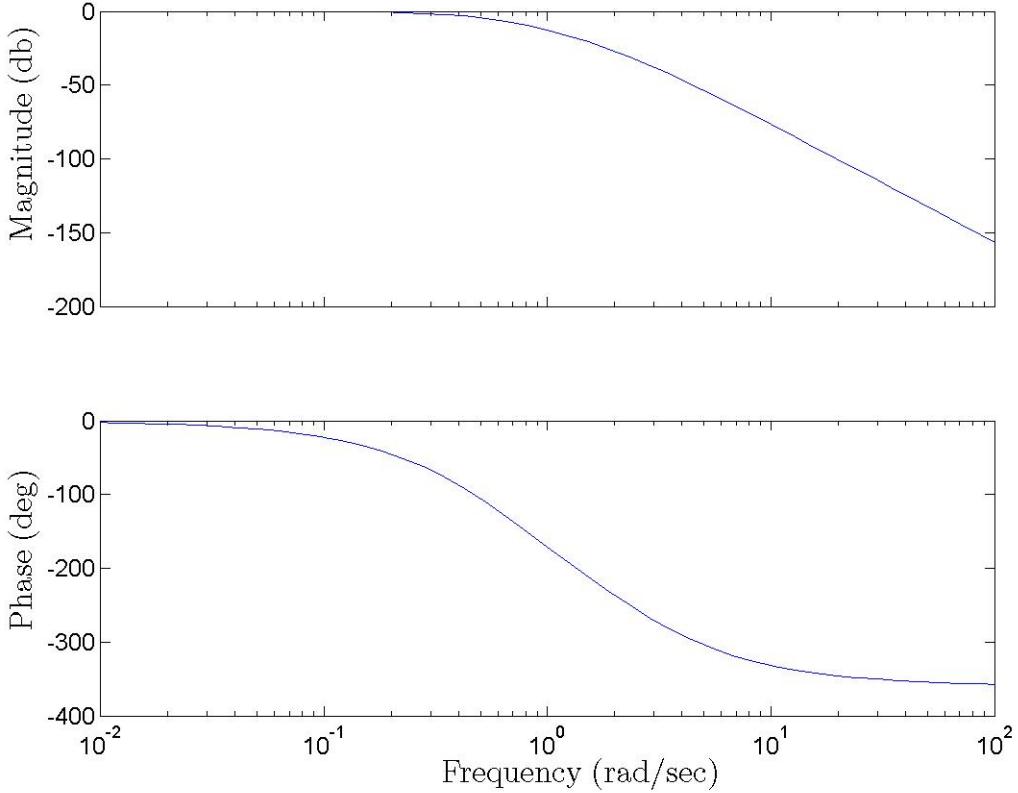
It is stabilized with an inner-loop control system given by

$$u(t) = K_1 x(t) + K_2 \int_0^t (x(\tau) - r(\tau)) d\tau$$

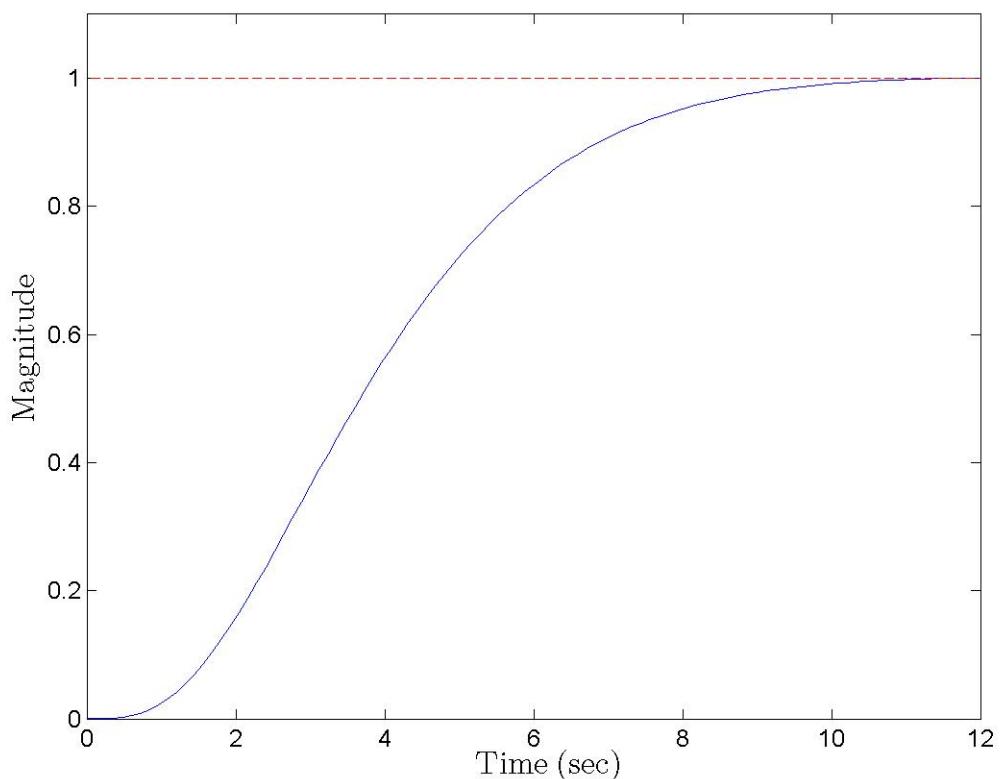
where $K_1 = 0$, $K_2 = -1.5$. The resulting closed loop system is given by

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -6 & -9 & -5 & -1.5 \\ 1 & 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -1 \end{bmatrix} \quad C = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}$$

Figure 6.1 displays a Bode plot of the closed loop system and Figure 6.2 displays a step response of the closed loop system. Figure 6.1 indicates the closed loop system possesses 14 dB of gain margin and 180 degrees of phase margin. Figure 6.2 indicates the closed loop system response to a step command exhibits no overshoot and takes approximately 12 seconds to reach the commanded value.



6.1: Frequency response of the closed loop system.



6.2: Step response of the closed loop system. The command is indicated by a dashed line.

Due to the slow response of the plant the extremum seeking control method's measurement rate was chosen to be 0.25 Hz. This provided the collection of three measurements in the time period of the closed loop system responding to a command. This slow rate helps alleviate the chance of multiple noisy measurements being taken in close proximity of each other. In such an instance, noise could dominate the change in measurement and deteriorate the gradient and Hessian estimates. Further discussion on the selection of measurement rates is found in section C.4. This particular problem proved not to be sensitive to the chosen rate. Faster and slower rates of measurement still proved successful.

The matrix H_k in equation (4.6.2) was selected to have 4 rows. The other elements of equation (4.6.2) were selected to be of compatible size. The size of H_k , as discussed in section 4.4, sets the size of the moving window used by the method. This particular problem also proved to be mostly insensitive to the number of rows in H_k . Only a small number of design iterations was necessary to settle on 4 rows.

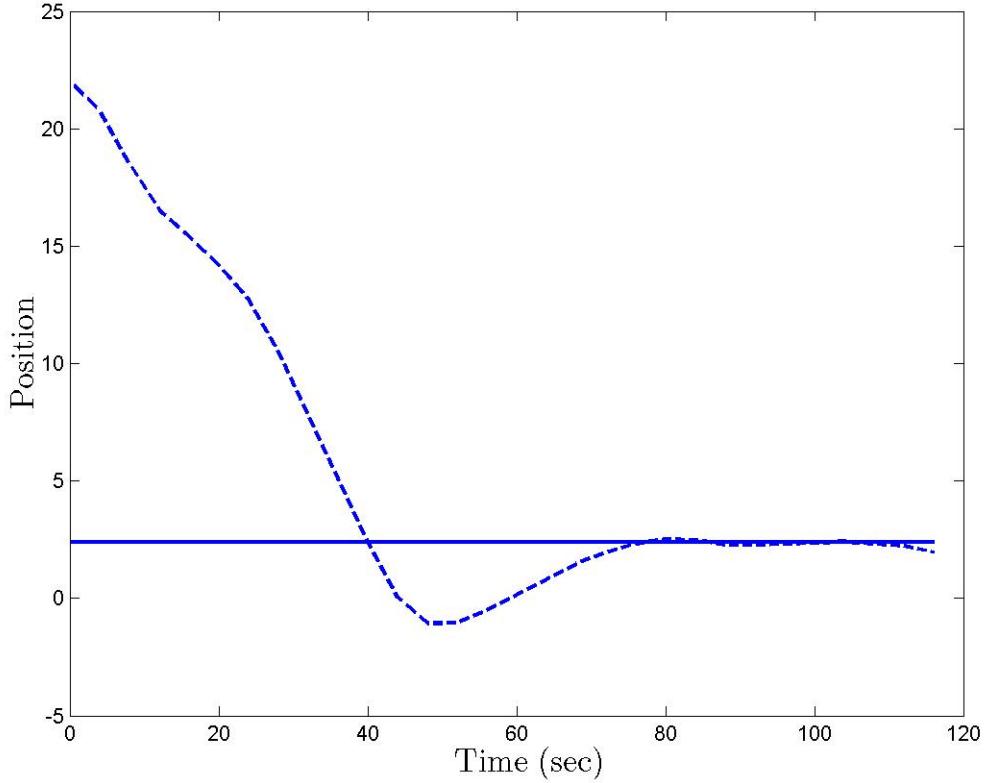
The command filter was set to 1 for simplicity. There were no requirements imposed upon plant rates nor response characteristics so no smoothing of the step commands issued by the method were necessary.

In this particular example, the persistent excitation, was set to 0. In most problems this is not a good choice. As discussed in section 4.4, the persistent excitation ensures observability of the performance function which is required to guarantee convergence. In this single-independent-parameter example however, an initial command issued to the plant provides the required observability at the initiation of the simulation. As the simulation progresses in time, the movement of the plant due to the algorithm commands continue to provide observability. When the plant reaches the extremum however, noise dominates the measurements so the Kalman filter can not continue to provide accurate estimates.

The system was implemented in a 1.0 Hz fixed-step discrete simulation. An initial command was provided to the plant. As the system responded to the command, measurements of the performance function's independent parameter were provided to the Kalman filter.

In the first example the Kalman filter generated an estimate of the gradient and commanded the plant according to equation 4.6.1 as $u_{ck} = -K\hat{b}_k$. Figures 6.3, 6.5, and 6.6 present the results of the simulation. The coordinate of the plant on the performance function is presented in Figure

6.3 as a function of time. The coordinates of the extremum is depicted by a solid line and is reached by the system in approximately 80 seconds. The system moves slightly away from the

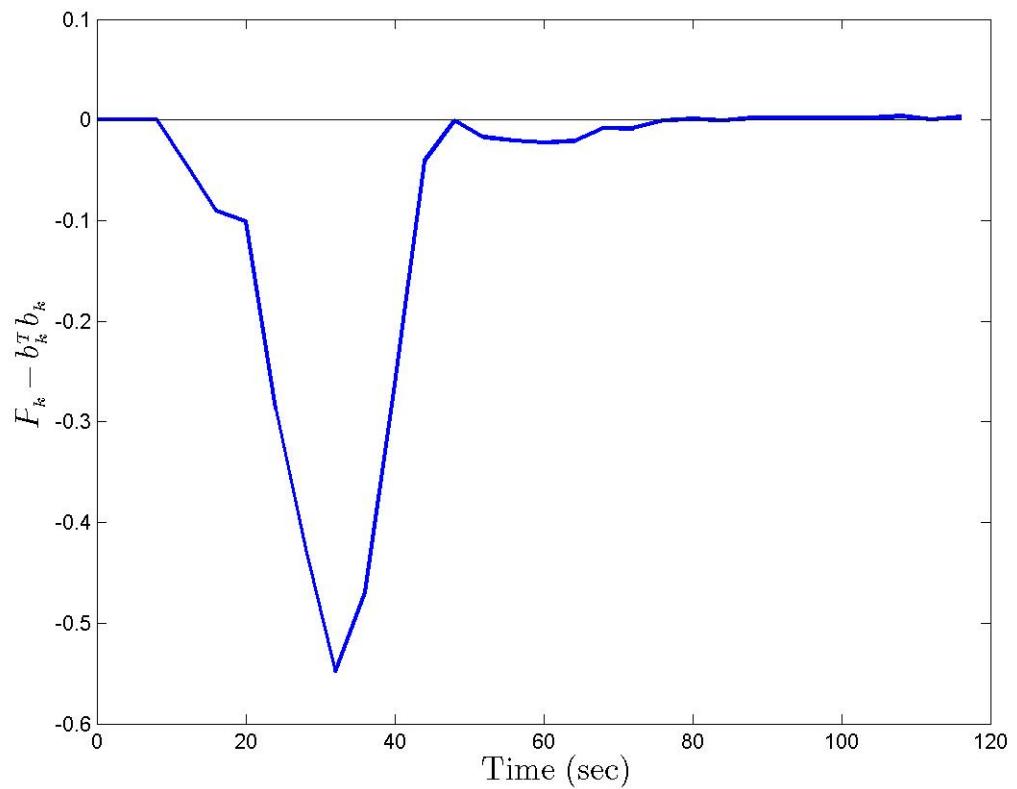


6.3: Plant position versus time. Dashed line: Plant position. Solid line: Performance function minimum.

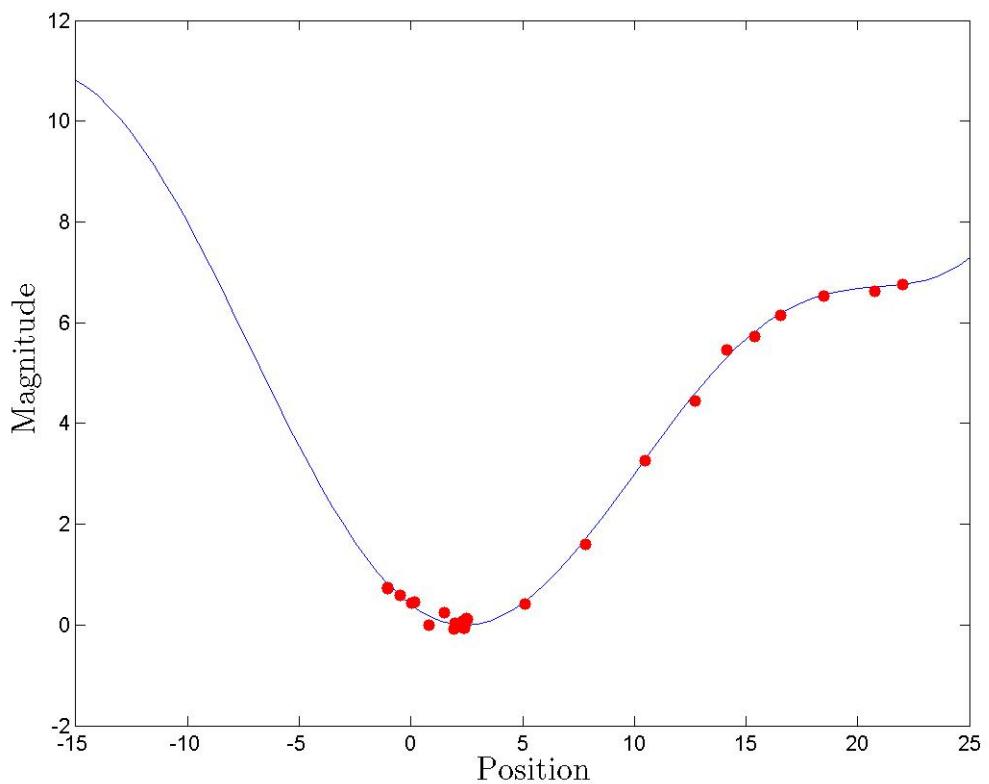
extremum at 90 seconds. This is due to the the difference between the error covariance and the square of the gradient estimate becoming positive as seen in figure 6.4. Chapter 5 explains that when $P_k > \hat{b}_k^T \hat{b}_k$, further convergence is not assured.

Figure 6.5 presents the history of measurements superimposed atop the performance function. The noise in the measurements is apparent where the red dots do not lie atop the performance function. The system started on the right hand side of the figure and moved to the left. When the system reached the minimum, many measurements were taken very close together.

The gradient estimate is shown in Figure 6.6. The Kalman filter estimates are represented by dashed lines. The true gradient and Hessian are represented by solid lines. A system initialization stage took three measurements before beginning estimation of the gradient and Hessian, thus, the figures show the first non-zero estimate at 12 seconds. It is apparent that the estimations



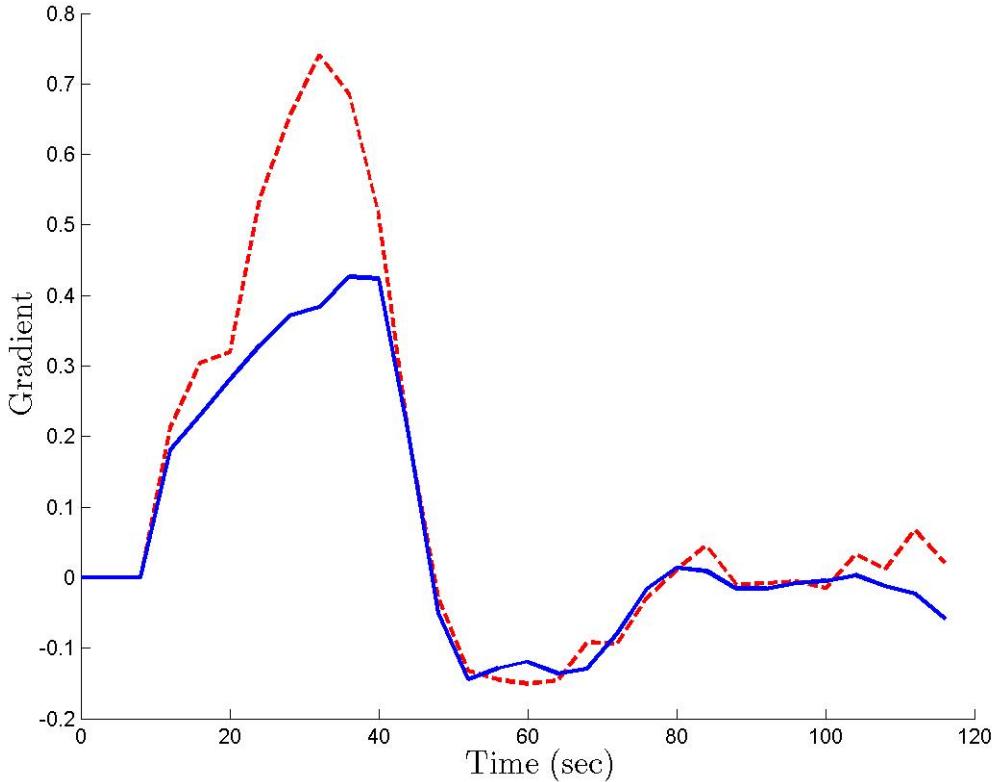
6.4: Difference between the error covariance and the square of the gradient estimate.



6.5: Performance function and measurement history. The performance function is represented by the solid line. Measurements are represented by dots. At time 0, the system was located at $x = 23$ and moved to the left.

began to suffer once the system neared the minimum. This is because as the system approached the extremum, Δf_k became small and was buried in noise, leading to poor estimates.

If the simulation were allowed to execute for a longer time period, the coordinates of the plant on the performance function would have randomly moved about the minimum.



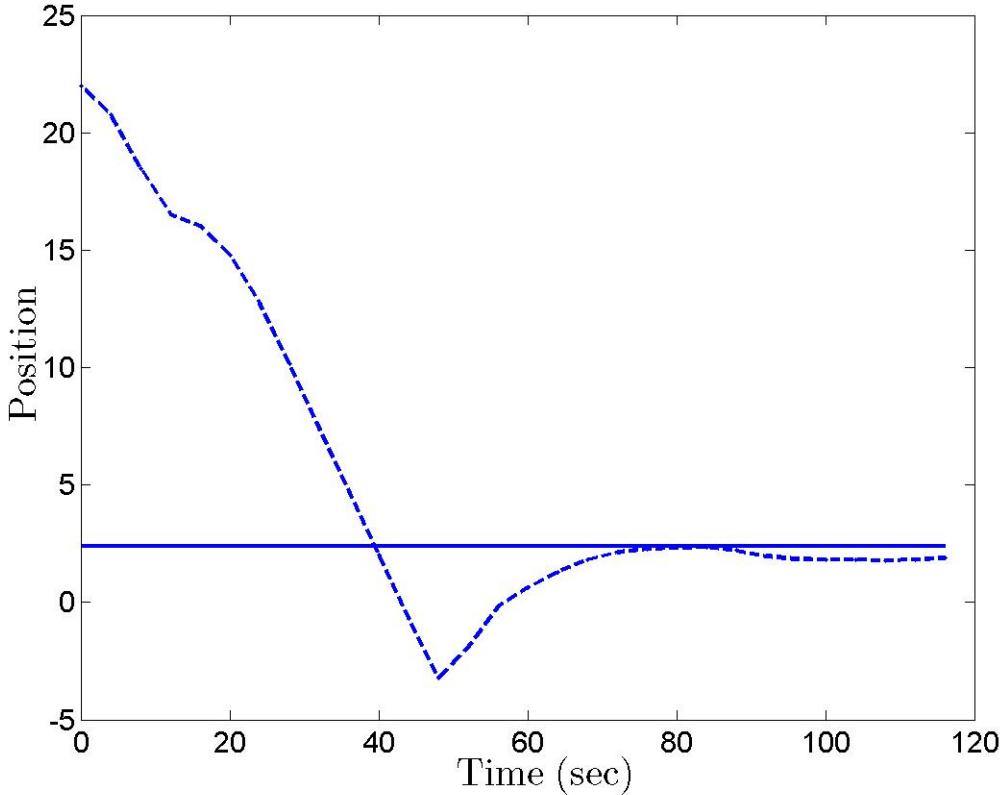
6.6: Estimated and true gradient versus time. Dashed line: Kalman filter estimate. Solid line: True gradient.

In a second implementation, the Kalman filter was modified to generate estimates of the gradient and Hessian which were combined, according to equation 4.4.7 as $u_{ck} = b_k M_k^{-1}$, to provide plant commands.

Typically, the Hessian estimate does not perform as well as the gradient estimate. To partially compensate for this, the system switched between a steepest-descent approach and a Newton approach. When the smallest singular value of the Kalman filter's error covariance, $\underline{\sigma}(P_k) \leq 0.005$, a Newton approach was used. When $\underline{\sigma}(P_k) > 0.005$, a steepest-descent approach was used. The switching threshold was used as a tuning parameter and selected by trial and error.

Figures 6.7, 6.8, and 6.9 present the results of the simulation. The coordinate of the plant on

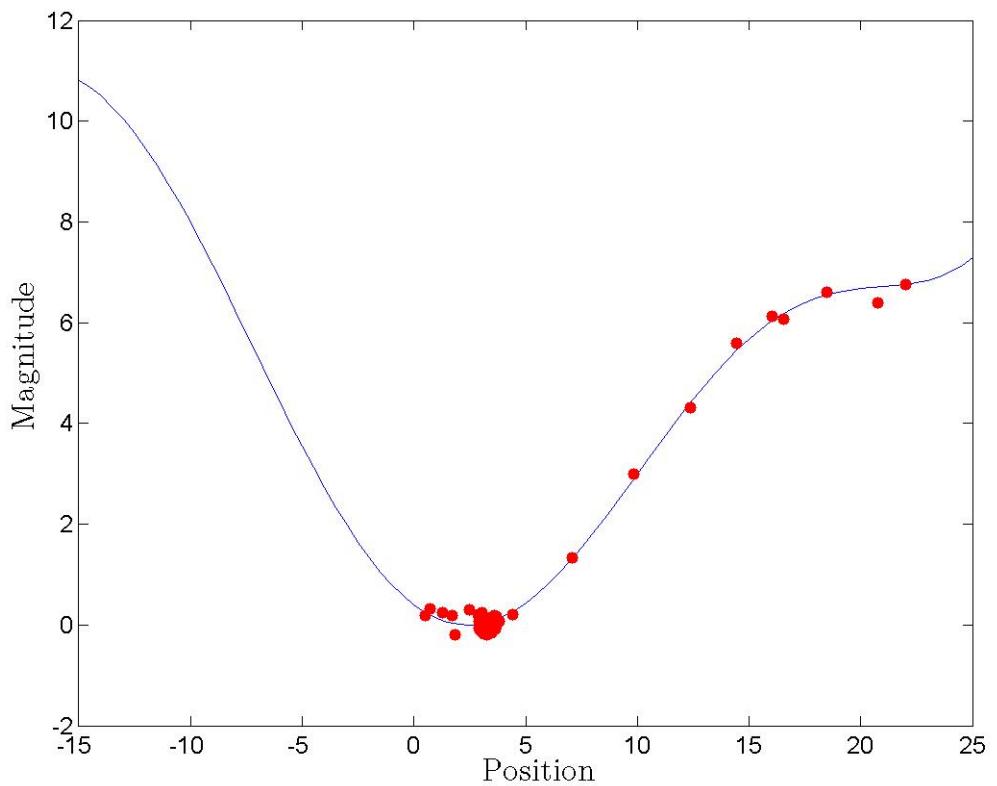
the performance function is presented in Figure 6.7 as a function of time. The coordinates of the extremum is depicted by a solid line and is reached by the system in approximately 75 seconds. The system moves slightly away from the extremum at 90 seconds due to noisy measurements.



6.7: Plant position versus time. Dashed line: Plant position. Solid line: Performance function minimum.

Figure 6.8 presents the history of measurements superimposed atop the performance function. The noise in the measurements is apparent where the red dots do not lie atop the performance function. The system started on the right hand side of the figure and moved to the left. When the system reached the minimum, many measurements were taken very close together which led to the large mass of points at the minimum.

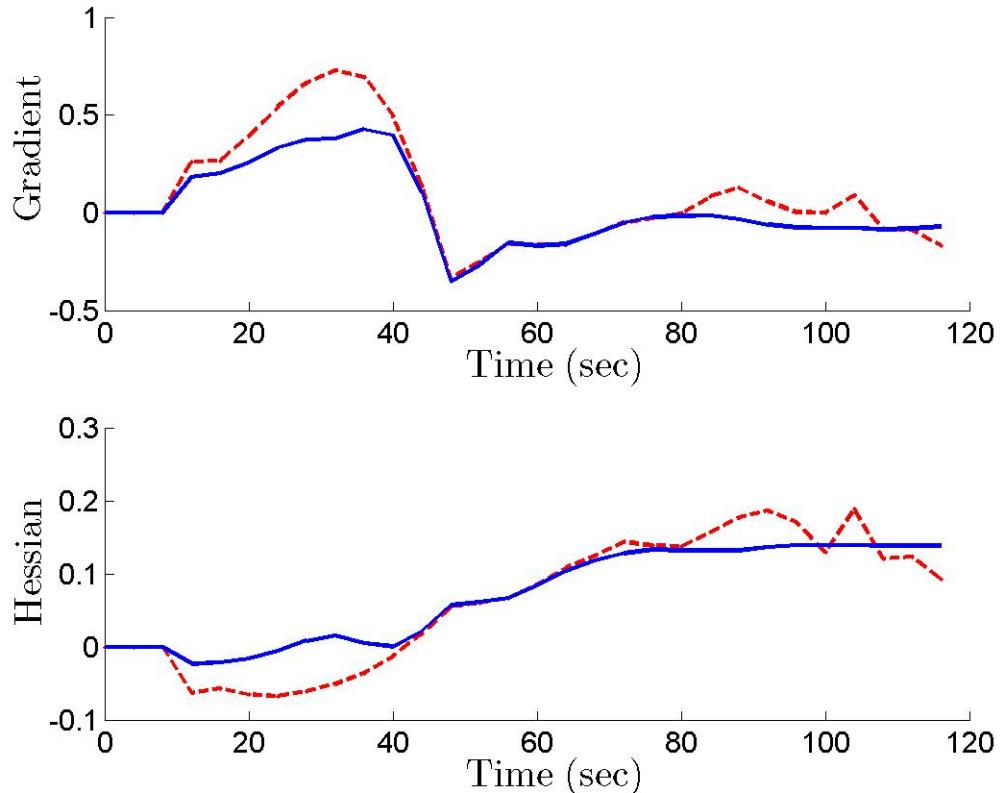
The gradient and Hessian estimates are shown in Figure 6.9. The Kalman filter estimates are represented by dashed lines. The true gradient and Hessian are represented by solid lines. A system initialization stage took three measurements before beginning estimation of the gradient and Hessian, thus, the figures show the first non-zero estimate at 12 seconds. It is apparent that the estimations began to suffer once the system neared the minimum. This is because as the



6.8: Performance function and measurement history. The performance function is represented by the solid line. Measurements are represented by dots. At time 0, the system was located at $x = 23$ and moved to the left.

system approached the extremum, Δf_k became small and was buried in noise, leading to poor estimates.

If the simulation were allowed to execute for a longer time period, the coordinates of the plant on the performance function would have randomly moved about the minimum.



6.9: Estimated and true gradient and Hessian versus time. Dashed line: Kalman filter estimates. Solid line: True gradient and Hessian.

This example illustrates application of the method to a simple one-input one-output problem. The gradient and Hessian estimates track the true values well and the system quickly reaches the extremum. The example in Chapter 8 illustrates the method applied to a multiple-independent parameter problem.

CHAPTER 7

Introduction to Formation Flight for Drag Reduction

The chapter introduces the problem of formation flight for drag reduction. It provides a review of previous work completed in this area and a discussion of the aerodynamics behind the drag reduction phenomenon. Chapters 8 and 9 provide two examples of extremum seeking control applied to formation flight for drag reduction.

7.1 Introduction to Formation Flight for Drag Reduction

Consider the problem of formation flight for drag reduction. In this problem, an automatic control system must reduce the induced drag on a trailing aircraft by positioning the wingtip of the trailing aircraft in the wingtip vortex of a leading aircraft. Substantial drag reduction and fuel savings can be achieved by employing this technique [Hum96]. This task is achievable by an unassisted pilot but it incurs high pilot effort and hence pilot fatigue restricts practical use.

An improvement to a piloted system is to employ an automatic control system which positions the trailing aircraft at a fixed, predetermined position behind the leading aircraft. The predetermined position could be chosen by conducting a survey of the trailing vortex position with a LIDAR or similar system. Such a system is simple to employ and would most likely provide some measure of drag reduction; however, uncertainty in the vortex position, and changing vortex position with flight and atmospheric conditions precludes obtainment of the maximum drag reduction possible.

Extremum-seeking control, applied to this problem, will provide the maximum drag reduction possible at all flight conditions. The system must be engaged at some predetermined position but will then continually seek the optimal position. A difficulty in this approach is the selection of a performance function; drag reduction is not easily measured directly. Instead an analog performance function must be used. Such analogs could be rolling moment, pitch angle, fuel

flow, or some other measurement. Each has advantages and disadvantages. For instance, fuel flow reduction is directly related to drag reduction and may seem to be a good choice; however engine response is typically slow and fuel flow is hence a latent measurement of drag. The best solution of this difficulty may be to estimate the drag reduction via a combination of various measurements.

7.2 A Review of Literature Related to Formation Flight for Drag Reduction

Formation flight for drag reduction has attracted much attention over the last 20 years. Research has been made into a number of areas. Biologists and engineers have studied the use of formation flight by migratory birds to increase their flight range and decrease energy consumption. Researchers have examined the structure of wingtip vortices and their effect on aircraft. Multiple tactics to exploit the energy in wingtip vortices to reduce drag and fuel flow of a trailing aircraft have been developed. They all employ the same strategy of positioning a trailing aircraft's wing in the wing tip vortex of a leading aircraft. The tactics range from manually controlling the trailing aircraft position to automatically controlling the trailing aircraft position with varying extremum-seeking control schemes. There have also been a hand full of flight tests which verify the fuel flow savings achievable with formation flight. This section provides a sampling of the vast body of research conducted in this area. It provides an overview of some of these studies and tests and provides references for the interested reader to find more details. It is not a complete bibliography but provides a sampling of the breadth of work and highlights some of the more important work.

The research conducted can be broken into seven categories. While overlap between the categories exist, the division is useful for organizing the works and emphasizing the wide breadth of work in this area. The categories are *Biological Studies*, *Aerodynamics Studies*, *Operational Studies*, *Formation Flight without Drag Reduction*, *Formation Flight for Drag Reduction*, *Extremum Seeking Applied to Formation Flight*, and *Flight Experiments*.

The *Biological Studies* category includes studies of migratory birds which exploit aerodynamic influences of the flock to save energy. The *Aerodynamics Studies* category includes mathematical models of the wing-tip vortex and experimental measurements of the vortex strength and struc-

ture. Ironically, much of the work in this category comes from researchers seeking to minimize the vortex influence to prevent upset of aircraft near airports. The *Operational Studies* category includes investigations into such things as the logistics of aircraft joining in long distance routes and the optimal position of aircraft in multiple-aircraft formations. This category probably contains the most diversity of studies. The *Formation Flight without Drag Reduction* category includes formation autopilot design descriptions and analyses of systems which automatically maintain aircraft formations but do not have drag reduction as a goal. The *Formation Flight for Drag Reduction* category includes papers which design autopilots in order to maintain formation and realize drag reduction but do not utilize an extremum-seeking control scheme. The *Extremum Seeking Applied to Formation Flight* category includes descriptions of systems which employ an extremum-seeking scheme in order to maximize the drag reduction of a formation of aircraft. The *Flight Experiments* category includes descriptions of flight experiments which strived to verify the theoretical predictions of drag reduction obtainable from formation flight. There are only a handful of works in this category.

7.2.1 Biological Studies

Formation flight of migratory birds has been a source of interest to biologist and engineers for over 100 years. One of the first investigations was performed by Wieselberger in 1914 [Wie14]. One year after he collaborated with Prandtl on the lifting-line theory, he explained why birds flying in V-formations use less energy to flap their wings than those on solo flights. He used constant strength horseshoe vortices to model three birds in a staggered formation (i.e. a diagonal-line formation). He showed a 15.8% drag saving for the middle bird and that the formation is more efficient than a single bird traveling alone.

Since the early investigations much more work has been completed. Investigations into biological formation flight have ranged from measuring the heart rate on migrating birds [WMC01], to exploring formation flight of birds as nonlinear self-organized phenomenon [Sug03]. Cutts, [CS94] studied the spacing in formations of pink-footed geese and calculated the energy savings gained by these birds through formation flight.

Sugimoto [Sug03] takes a rigorous mathematical approach to analyzing formation flight of birds. He derives basic equations by considering a dynamical system of multiple interacting

wings through induced flow fields. Through this, he shows that formation flight of birds is stable and self-organized. He points out that the implication is that formation flight emerges from nonlinear dynamical interactions and does not exhibit chaotic behavior.

Filipino [Fil96] also examines the formation flight of birds. He uses a horse-shoe vortex method to calculate that a parabola-like, inverted-U formation provides a flock with equally distributed induced drag reduction.

7.2.2 Aerodynamics Studies

The aerodynamics behind formation flight for drag reduction has been extensively studied. Much of the work in this area had the goal of reducing the impact of wing-tip vortices on aircraft near airports. Much of it is focused on the structure and strength of the wing-tip vortices. An extensive bibliography of the vortex work can be found in [Hal91]. Further examples of some of the work can be found in the papers presented at the AGARD conferences such as, [AGA91], [AGA96]. The exploitation of the wing-tip vortices to save energy was first investigate by Shlitchting in 1942. In two papers [Sch42] [Sch44], Schlichting investigated energy savings achievable by flying two aircraft in formation. He determined values of energy savings for varying numbers of aircraft and spacing. His studies utilized horseshoe vortices and examined V, transverse-line, and inverted-V formations. He also investigate how variations in altitude affect the energy saving potential of formation flight. In [Bla00] the aerodynamics for a simulation of a large number of vehicles in close formation flight under aerodynamic coupling is presented. The benefits of formation flight for a large number of vehicles is also discussed.

7.2.3 Operational Studies

Many researchers have investigated logistics and practical implementation issues of formation flight for drag reduction. Aspects of this category range from basic calculations of energy savings to calculating the impact of changing flight schedules to accommodate formation flight of aircraft.

Nehrbass investigated the application of formation flight to point-to-point commercial aircraft service design [NFG04]. He found, in part, that formation flight enables airlines to feasibly use a point-to-point route structure in place of a hub and spoke system when formation flight is used to save on fuel costs. Iglesias [IM02] developed a method to find the optimum load distribution

of a V-formation that gives minimum induced drag for the formation. System studies reported in Hansen [HC02] have shown that a 10-percent drag reduction for a commercial airliner conducting a daily Los Angeles-to-New York roundtrip results in reduced fuel usage in excess of \$500,000 for each airplane each year, as well as emission reductions of carbon dioxide and oxides of nitrogen of 10 percent and 15 percent, respectively. Reference [BFK09] investigates efficacy of formation flight for drag reduction for existing commercial aircraft schedules. It considers aircraft originating from different cities, joining in formation, and landing at a common destination.

Reference [NFK10] looks at practical issues with formation flight and states that extended formations are likely only practical in low to moderately low turbulence levels and for streamwise spacing less than about 50 spans. They also state that tracking error and turbulent gusts cause the largest variations in induced drag savings but that technology in precision navigation and remote sensing can be useful to help minimize these errors

In [JCR95] the cost savings for 747-400 aircraft flying in formation for the maximum 7200nm range is investigated. It shows that 15000kg less fuel would be used which translates into a direct operating cost saving of about 2% for both aircraft.

7.2.4 Formation Flight with out Drag Reduction

Multiple papers discuss control systems designed to track a leading aircraft while rejecting disturbances from the vortex but do not attempt to achieve drag reduction. Among these is [GSC06]. It develops and flight tests a formation flight control system. The flight testing is completed on a radio-controlled aircraft. It does not attempt to achieve drag reduction in its flight tests. [SMT70] outlines the history of formation flight development up to 1970. It looks only at formation flight as a way to provide precision air drop and to reduce pilot work load. It does not consider formation flight as a means to achieve drag reduction.

7.2.5 Formation Flight for Drag Reduction

Multiple papers attempt to achieve drag reduction with various schemes. Some do not use an extremum-seeking control systems but rely upon a priori knowledge of the vortex. In [Pro99] a formation flight control system is developed to maintain formation geometry when the leading aircraft maneuvers. The controller is evaluated in a simulation which includes aerodynamic

effects resulting from the trailing aircraft flying in the wing-tip vortex of the leading aircraft. The system does not include any extremum seeking control but assumes an a priori knowledge of the vortex position. Reference [Lav02] uses an indirect adaptive control system which counters the influences of the leading aircraft vortex to maintain the position of a trailing aircraft in the vortex of a leading aircraft. In [SPC00] a sliding model control system is developed to maintain position inside of a vortex. It also evaluates the system in a simulation and assumes an a priori knowledge of the vortex. In [OPJ04], a classical-control design approach is taken and does not employ an extremum-seeking control scheme. The system developed adequately follows the leading aircraft and rejects vortex disturbances. In [PDP01], a control system is designed which simply ignores the external disturbances of the leading aircraft vortex but finds this system yields satisfactory performance via simulation.

While it is clearly possible to realize drag reduction without an extremum-seeking control scheme, this approach will inevitably not provide the maximal drag reduction possible. The attraction of such an approach is the simplicity with which it can be implemented. There is no need to measure vortex strength in real time nor provide persistent excitation into the system. The draw back is that the maximum drag reduction will not be realized and the system can not automatically adjust for changes in the vortex size or position.

7.2.6 Extremum Seeking Applied to Formation Flight

Multiple papers provide a means to locate and maintain a relative position which optimizes the fuel savings garnered from formation flight. All of these employ some form of extremum seeking control. Binetti [BAK03] uses a classical gradient extremum seeking method to position a trailing aircraft in the optimal position behind a leading aircraft. The system is designed using experimental vortex data of a C-5 Galaxy aircraft. It uses the pitch angle of the trailing aircraft as the performance function.

Lavretsky [LHC03a] uses direct adaptive neural feed forward network based control, and on-line extremum seeking command generation. The trailing aircraft throttle position is used as the performance function. This provides a smooth function to minimize but is subject to the slow response of the propulsion system. The scheme is shown to provide bounded output tracking and to minimize the effect of uncertainty.

Chichka [CSF06] approaches the problem of formation flight with a modern estimation technique. He uses a steady-state Kalman filter to estimate the gradient of the rolling-moment of a trailing aircraft. An extremum-seeking loop then maximizes the rolling-moment to drive the trailing aircraft to the optimal formation location.

Hanson [Han03] uses a fuzzy logic technique to estimate the lateral position of a leading aircraft vortex and drives the trailing aircraft to that position with a classical control system. This paper only considers lateral movement of the trailing aircraft.

7.2.7 Flight Experiments

A number of flight experiments have been conducted which confirm the drag reduction predictions. In these experiments fuel flow reduction and power reduction were measured by special instrumentation installed on the aircraft.

The experiments demonstrated large gains in fuel efficiency can be had. Only one flight experiment implemented an automatic method to maintain the optimal aircraft position for drag reduction. The others used piloted systems and were only intended to quantify the fuel savings and drag reduction achievable. One flight test used a small aircraft in the wing-tip vortex of a larger aircraft. It demonstrated the larger fuel flow saving possible. There have been no flight experiments utilizing two large aircraft.

The first reported flight-demonstration of formation flight for drag reduction was by [BH90]. It employed an extremum-seeking control scheme to position a trailing aircraft in the wing tip vortex of a leading aircraft. There have not been extremum seeking control flight tests since this initial test. In this test the extremum seeking controller measured the aileron deflection of the trailing aircraft and the upwash on the trailing aircraft at subsequent relative positions. These measurements were used to determine a direction and distance for the trailing aircraft to move. This approach proved effective although loss of power reduction was experienced due to the inherent persistent excitation. When flying under manual control, the experiment realized 15% power reduction. When under automatic control the experiment yielded 10% power reduction.

In 2001 the NASA Dryden Flight Research Center flew an F/A-18 in formation and measured the fuel flow reduction achieved [HC02], [RCV02],[VRW02]. The technique employed was to measure the fuel flow at gridded relative positions. The tests resulted in the creation of a map of

the fuel flow reduction at various relative positions. The maximum fuel flow reduction measured was 14%.

Also in 2001, the United States Air Force Test Pilot School measured fuel flow reduction of T-38 aircraft flying the wingtip vortex of another T-38 [WJB02], [Eug01]. These experiments examined two-ship and three-ship formations. It measured a maximum of 8.8% fuel flow savings in the two-ship formation and minimal savings in the three-ship formation. Theoretical calculations and the flight test technique employed in this study cast doubt on the generality of the results. This project also experimented with an automatic control system designed to maintain relative positions between aircraft. The system did not track the vortex nor employ an extremum-seeking control system.

In 2003 the NASA Dryden Flight Research Center flew a F/A-18 in the wingtip vortex of a DC-8. The experiment measured a 29% fuel flow reduction in the F/A-18. This experiment was conducted with manual flying.

7.3 Aerodynamics of Formation Flight

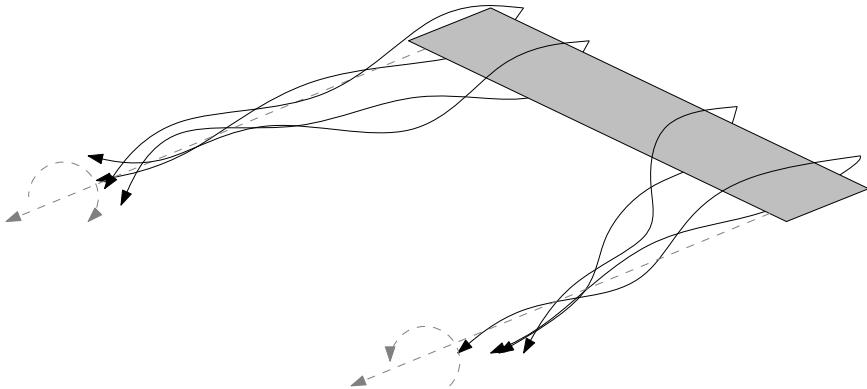
In this section we discuss the aerodynamics of close formation flight which provides drag reduction. The discussion is intended to provide a simple mathematical description of the phenomenon in order to build an understanding of the aerodynamic modeling used in following chapters.

Close Formation flight of aircraft creates complex aerodynamic interactions between the aircraft in formation. The wake generated by each aircraft induces forces and moments on all other aircraft in the formation; however, aircraft in trailing positions experiences far larger forces and moments than aircraft in leading positions. When the formation configures itself properly, a reduction in drag is experienced by members of the formation. It is well known that this drag reduction phenomenon is utilized in nature by flocks of birds [CS94, Hum83, WPB07]. There have been many mathematical models developed which capture the effect to varying degrees of accuracy such as [BAK03, McG61, Hum83, HK82, RV96, SSL77, Nel74, CSF06, Bla00, BI72].

The purpose of this section is to develop a simple mathematical description of the phenomenon in order to facilitate a basic understanding and some engineering intuition that can be applied to extremum-seeking design. No attempt is made to describe subtle details of the aerodynamic interactions between aircraft nor the airflow around the vehicles; instead, only the general inter-

actions of the wings of two aircraft are considered. This is done because the primary effects of the phenomenon are due to changes to the local angle of attack which alters the lift distribution along the wing. These primary effects can therefore be modeled by considering only the wing. To this end, we begin with a narrative of how the wing-tip vortices form on a leading wing and how they influence a trailing wing. We then develop a mathematical model of the circulation around a leading wing and the resulting wing-tip vortices. Finally, we consider the effect of the wing-tip vortices on a trailing wing and show why it leads to drag reduction.

Consider two aircraft in formation at coincident and constant altitude in straight and level flight. The leading aircraft wing creates lift by generating a pressure difference between the upper and lower wing surfaces. This pressure difference creates circulation around the wing as the high pressure air below the wing moves to the lower pressure area above the wing. In finite span wings, much of this air flows around the wing tips. The air on the bottom surface tends to move toward the wing-tip while the air above the wing tends to move away from the wing-tip. This circulation is continuously shed by the wing at the wings trailing edge creating a wake which rolls up into vortices further aft of the wing in near line with the wingtips. Figure 7.1 displays the general airflow above, below and aft of the wing. The solid lines indicate the paths of particles of air. The dashed lines indicate the general aft movement and general circulation of the air. The two vortices rotate such that a downwash is created in between the center lines of the vortices and an upwash is created outside of the center lines of the vortices.

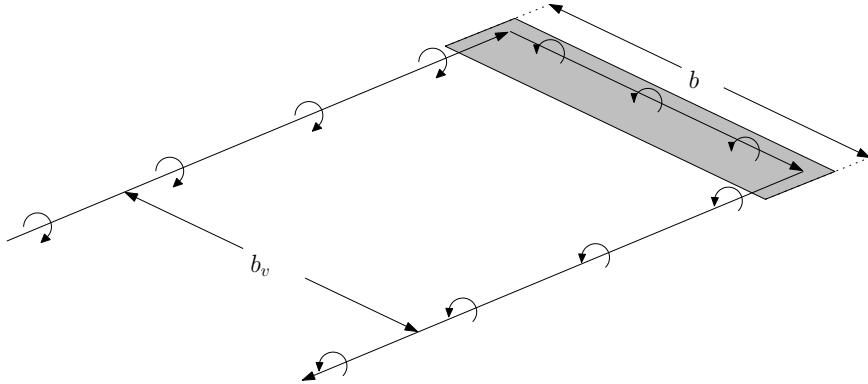


7.1: Airflow around a wing.

The trailing aircraft is positioned such that one of its wing tips is near a vortex core and the entire span of the that wing is in the upwash field of that vortex. The vortex upwash increases the local angle of attack along the trailing wing, thereby increasing the wing's total generated lift. In

order to maintain trim lift, the free-stream angle of attack must be reduced. This is accomplished by reducing its lift via reduced angle of attack. As a consequence, the induced drag of the trailing aircraft is also reduced as the downwash behind the wing is diminished.

In order to develop a mathematical model we limit the discussion to that of two nonmaneuvering aircraft in the subsonic regime. We assume inviscid and incompressible flow, and that the trailing aircraft has no effect on the leading aircraft nor its vortices. Only the wingtip vortices of the leading aircraft are therefore considered and are treated as invariant with respect to time. This implies that the vortices do not move nor decay. We approximate the leading and trailing aircraft by straight constant chord wings of span b and surface area A . The wing is modeled by a bound vortex with semi-infinite trailing vortices connected to the bound vortex near the wingtips. The bound vortex is placed at the quarter-chord location of the wing. Figure 7.2 illustrates the geometry. Note that the vortices result in an upwash outside of the horseshoe shape and a downwash inside of the horseshoe shape.



7.2: Geometry of the Horseshoe Vortex. The shaded parallelogram represents the wing. The vortex is represented by the straight line segments with semicircles indicated the direction or vorticity.

The lift generated by the bound vortex portion of the horseshoe vortex can be found via the Kutta-Joukowski theorem which states

$$L' = \rho V \Gamma(y) \quad (7.3.1)$$

where L' is the lift per unit span, ρ the fluid density, V the free steam velocity, Γ the circulation around the wing, and y the coordinate along the span of the wing. Integrating L' across the wing

provides the lift due to the bound vortex

$$L = 2\rho V \int_0^{b/2} \Gamma(y) dy \quad (7.3.2)$$

Assuming an elliptical lift distribution, the circulation

$$\Gamma(y) = \Gamma_0 \sqrt{1 - \left(\frac{2y}{b}\right)^2} \quad (7.3.3)$$

where Γ_0 is the circulation at the centerline of the wing. The lift is therefore given by

$$L = 2\rho V \int_0^{b/2} \Gamma_0 \sqrt{1 - \left(\frac{2y}{b}\right)^2} dy \quad (7.3.4)$$

The integral in equation (7.3.4) is of the form

$$\int \sqrt{\alpha^2 - u^2} du$$

which has the solution

$$\frac{1}{2}u\sqrt{\alpha^2 - u^2} + \frac{\alpha^2}{2} \sin^{-1}\left(\frac{u}{\alpha}\right)$$

The integral in equation 7.3.4 is evaluated as

$$\int_0^{b/2} \Gamma_0 \sqrt{1 - \left(\frac{2y}{b}\right)^2} dy = \frac{\pi b}{8} \Gamma_0$$

Substituting this results into equation 7.3.2 we arrive at

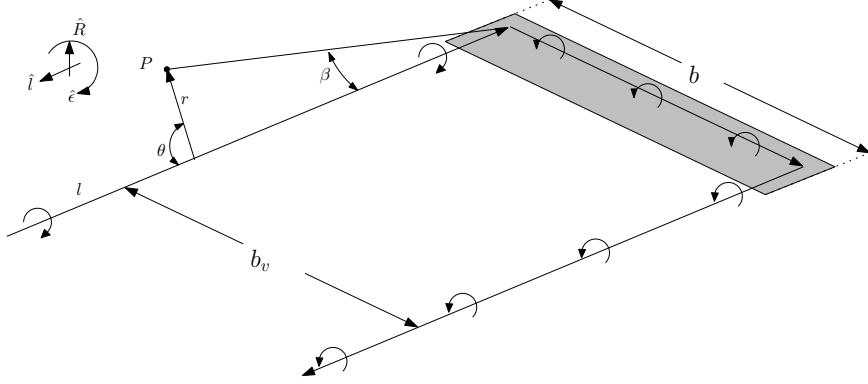
$$L = \frac{\pi}{4} \rho V b \Gamma_0$$

Comparing this to equation (7.3.1) it is clear that the lift from a wing of elliptical lift distribution is equivalent to that of a wing of uniform lift distribution with effective span $b_v = \frac{\pi}{4}b$

This analysis assumes that the semi-infinite vortices attach to the bound vortex at the effective span b_v . It therefore ignores the complicated aerodynamics involved with wake roll-up. This approximation, while poor near the trailing edge of the leading wing, proves sufficient at a few wingspans distance behind the leading wing.

The bound and semi-infinite vortices are assumed to satisfy the second Helmholtz theorem, which states that an irrotational fluid cannot produce or dissipate vorticity [Bra06](p12). This implies that the semi-infinite vortices possess the same circulation as the bound vortex.

The induced upwash w due to one semi-infinite vortex at an arbitrary point P is calculated through the Biot-Savart law by integrating over the semi-infinite vortex filament l as shown in



7.3: Geometry of upwash calculation.

figure 7.3. Here we use a cylindrical coordinate system centered on the vortex filament with unit vectors \hat{l} , \hat{R} , and $\hat{\epsilon}$ which point, respectively, in the direction of the vortex filament, in the radial direction orthogonal to the vortex filament, and the direction orthogonal to the previous two. The upwash is given by

$$w = \int \frac{\Gamma dl \times \hat{R}}{4\pi |r^2|} \quad (7.3.5)$$

where the integral is taken over the length of l and dl is an infinitesimal segment of l . It is convenient to write the integral in terms of the angle θ between the radial vector r and the vortex filament l . To this end we note that

$$dl = -r \sin(\theta) d\theta.$$

to get

$$w = \int \frac{\Gamma r \sin(\theta) d\theta \hat{l} \times \hat{R}}{4\pi |r^2|} \quad (7.3.6)$$

The integral then becomes

$$w = \frac{\Gamma}{4\pi r} \int_{\pi}^{\beta} \sin \theta \, d\theta \hat{\epsilon} \quad (7.3.7)$$

It is evaluated from $\theta = \beta$ to $\theta = \pi$ to give

$$w = \frac{\Gamma}{4\pi r} (\cos \beta - 1) \hat{\epsilon} \quad (7.3.8)$$

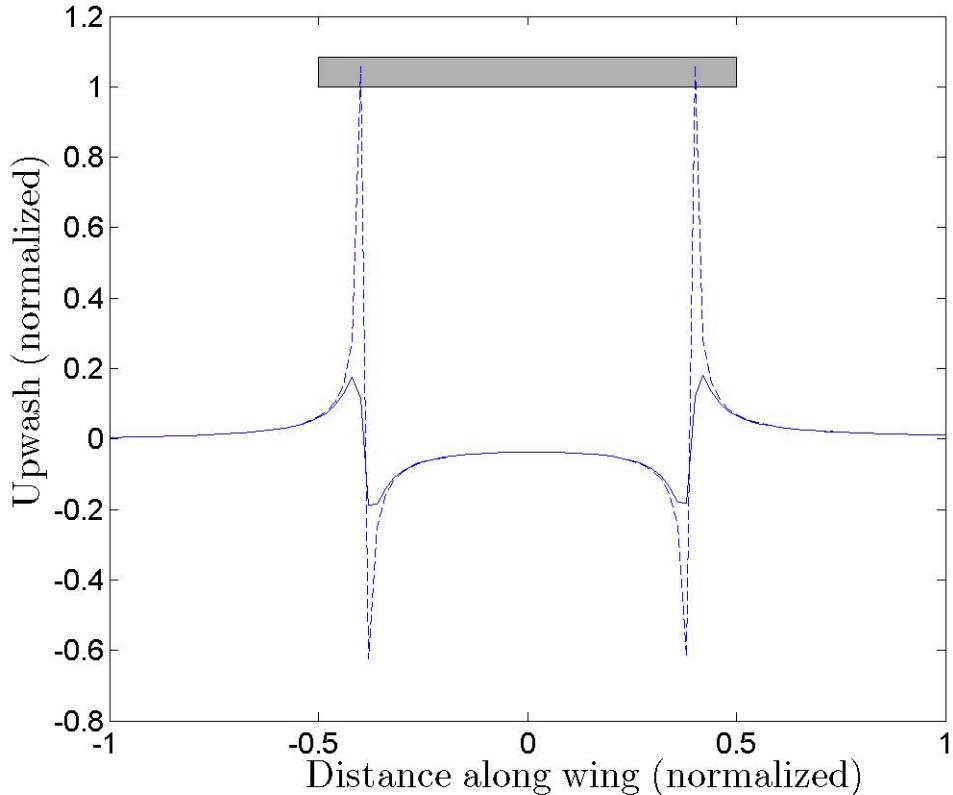
This derivation has the limitation that at $r = 0$ the upwash goes to infinity. This limitation can be overcome by using the correction for γ provided by [BH82] (p 2-3).

$$\Gamma_c(r) = \frac{\Gamma}{1 + (r_c/r)^2}$$

where r_c is the radius of the core of the vortex. The upwash is then

$$w_c = \frac{\Gamma}{4\pi r (1 + (r_c/r)^2)} (\cos \beta - 1) \hat{e} \quad (7.3.9)$$

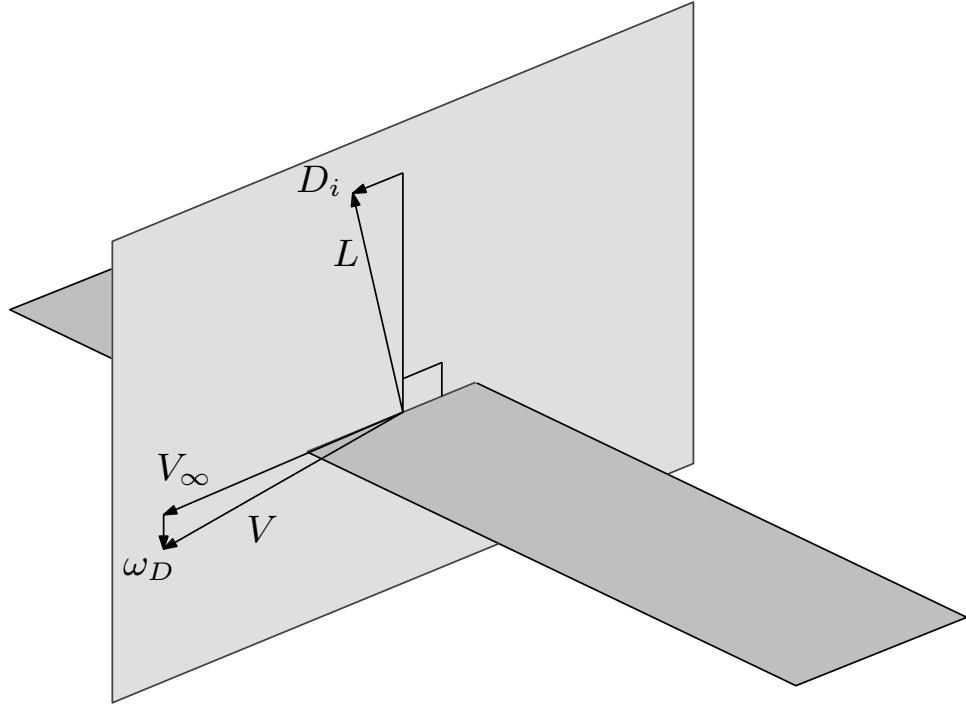
Superposition of the upwash from both semi-infinite vortices and the bound vortex provides the total upwash at any point behind the leading aircraft. With an inverse proportional relationship to r , it is clear that downstream of the wing, the downwash due to the bound vortex is negligible. It is also clear that when near a semi-infinite vortex, the upwash is dominated by the nearest vortex. Figure 7.4 depicts the upwash caused by both semi-infinite vortices in the plane of, and at a distance b behind the leading wing. It depicts both w and w_c . The upwash is greatest at the vortex center and goes to zero away from the vortices. Between the vortices, a downwash is present.



7.4: Upwash in the plane of a wing at a distance b aft of the wing. Dashed line: uncorrected upwash mode. Solid Line: correct upwash model. A grey rectangular wing is also included in the figure to reference the upwash wing position.

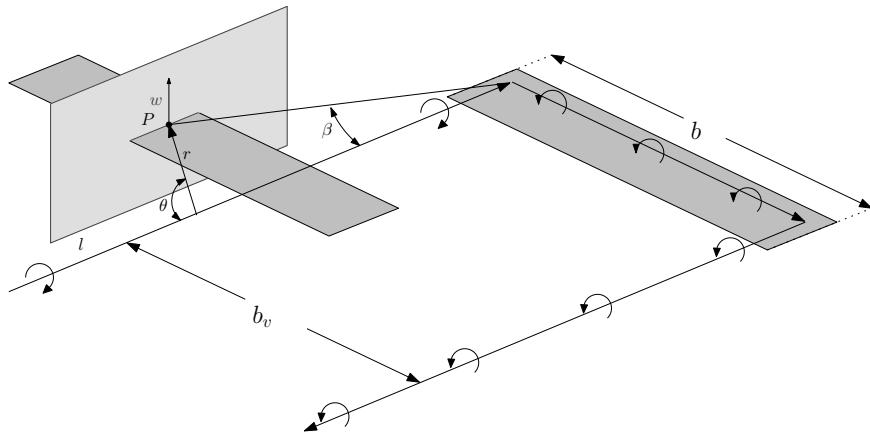
Now consider the effect downwash has on the lift of a wing. For our purposes it will suffice

to consider a single slice of the wing orthogonal to the plane of the wing intersecting point P as shown in Figure 7.5. The downwash adds to the freestream velocity V_∞ creating a local air flow velocity V . Since the lift generated by a wing is orthogonal to the local air flow, the lift vector L is rotated aft. The component of the lift vector pointing aft is the induced drag. Figure 7.5 shows the local air flow vector, the lift vector, and the induced drag vector D_i .

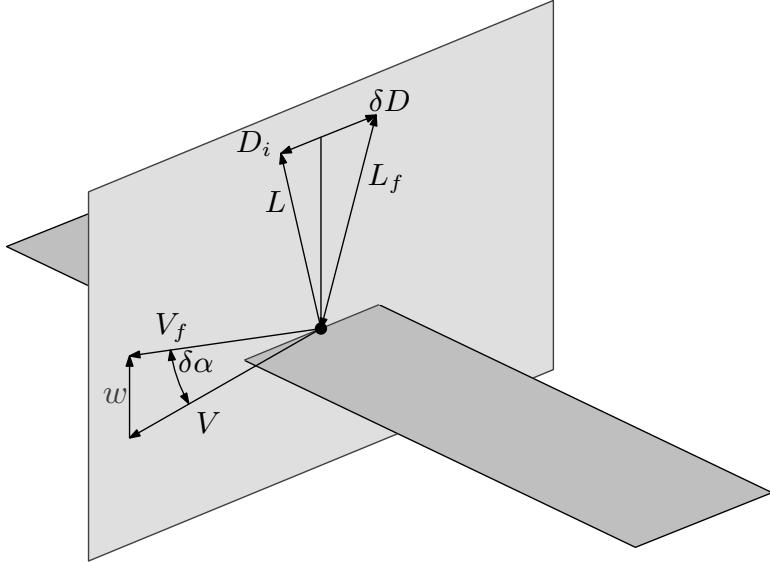


7.5: Downwash and lift of a wing

Also consider the effects of a trailing wing in the upwash field of the leading wing. Figure 7.6 displays the trailing wing in the upwash field. Figure 7.7 displays a more detailed view.



7.6: A trailing aircraft wing in the upwash field of a leading aircraft wing.



7.7: The upwash induced by a wing-tip vortex impinging on a trailing aircraft wing.

The upwash on the wing causes the effective air velocity V_f over the trailing aircraft's wing to be turned at an angle $\delta\alpha$. The upwash is small in magnitude as compared to that of the free stream velocity, V . This implies that the resulting free stream velocity due to the vortex V_f has approximately the same magnitude as V . Since lift on a wing is defined as perpendicular to the free-stream velocity, the trailing aircraft's effective lift vector L_f is rotated at the same angle $\delta\alpha$. This vector can be decomposed into components parallel and orthogonal to the free stream velocity V . The component parallel to V is given by

$$\delta D = L_f \sin(\delta\alpha)$$

and points in the direction opposite that of V . Using the small angle approximation,

$$\delta D = L_f \frac{w}{V}$$

The vector δD effectively reduces the induced drag of the wing. The vector orthogonal to V has magnitude $L_f \cos \delta\alpha$. Assuming $L_f \approx L$, the orthogonal vector has a smaller magnitude than L such that the change in lift required of the wing is

$$\delta L = L - L_f \cos(\delta\alpha)$$

The model provides insight into the phenomenon which can be used for the design of an automatic formation flight control system. It indicates that a control system must strive to keep

the trailing aircraft wingtip near the vortex core but outside of the downwash field. It also shows that roll angle of the trailing aircraft must be kept to a minimum; small roll angles will move the wing outside of the area of largest upwash. The reduction of drag also implies that the trailing aircraft must reduce its throttle in order to maintain its relative longitudinal position behind the leading aircraft.

The model indicates that the upwash field falls off as a function of $1/r$. Therefore, when near the vortex core, small changes in relative position can mean large changes in the magnitude of the upwash. This suggests that a extremum-seeking control system should return a larger drag reduction than a system which only uses a priori estimates of the vortex core position.

CHAPTER 8

Application of Extremum Seeking Control to Formation Flight for Drag Reduction

8.1 Introduction to Extremum Seeking Control Examples

This and the next chapter present the application of the extremum seeking control method to the problem of formation flight for drag reduction. Specifically, they consider the problem of maximizing the drag reduction from a two-ship formation of transport class aircraft.

This Chapter uses UCLA-developed models of a wingtip vortex and an aircraft flying within the vortex. A static map of induced drag coefficient on the trailing aircraft is used for the performance function. A linear time invariant model is used to model the trailing aircraft dynamics. The implementation uses a single sinusoidal persistent excitation signal which always acts in the direction orthogonal to the extremum seeking command.

Chapter 9 uses Boeing-developed models of the wingtip vortex and the trailing aircraft. Pitch attitude and aileron deflection are used as performance functions in separate simulations. An extensive nonlinear model of trailing aircraft dynamics is used. The implementation uses two persistent excitation signals, each perturbing one direction of the independent parameters.

Both implementations assume that the leading aircraft flies in a straight-and-level path such that the wingtip vortex is time invariant. They are both successful at guiding the trailing aircraft to a position where the maximum drag reduction is achieved.

This Chapter begins with a discussion of the implementation of the extremum seeking control method to formation flight models developed by UCLA. The models of the performance function and the aircraft model are first presented. The design of the inner loop control including the choices of feedback and the resulting controller gains are then discussed. This Chapter then discusses the extremum seeking control design and the choices made during the design process.

Simulation results are then presented with a discussion of the various trade-offs made during the design.

8.2 Vortex and Aircraft models

This section discusses the model of the leading aircraft vortex and of the trailing aircraft dynamics. The wingtip vortex generated by the leading aircraft in the formation is modeled by interpolating between values contained in a set of look-up-tables. The look-up-tables include values of the induced drag coefficient of the trailing aircraft as a function of lateral and vertical relative position between the leading and trailing aircraft. It also includes values of the aileron deflection required to maintain trim of the trailing aircraft as a function of lateral and vertical relative position. The look-up-table of induced drag coefficient is used as the performance function. The aileron deflection look-up-table for any given position is used as a disturbance input to the trailing aircraft model.

The look-up-tables were generated using a vortex-lattice method. The vortex-lattice method uses discrete horseshoe vortices to approximate the circulation at any point on a surface. The surface is broken into quadrilateral panels with a horseshoe vortex attached to each at the one-quarter chord line and extending into the wake in two semi-infinite filaments parallel to the free stream. Each panel is subject to the onset flow from the free-stream velocity. The strength of the vortex of each panel is determined by the flow-tangency condition at the midpoint of the three-quarter chord of each panel. The flow tangency condition at a control point (x_p, y_p) is

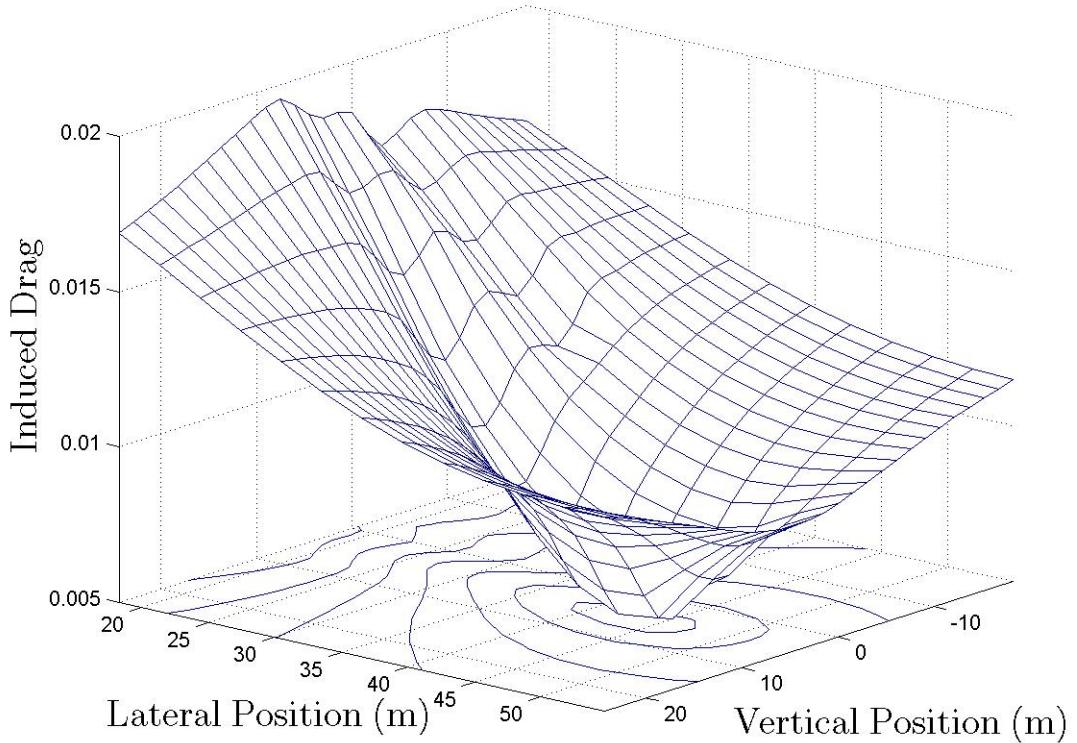
$$V_\infty \sin \alpha - \sum_{i,j} \Gamma_{i,j} w_{i,j}(x_p, y_p) = 0$$

where the subscripts i and j are used denote the individual panels. This forms a system of equations allowing the solution of $\Gamma_{i,j}$. The forces and moments on each panel are then calculated with the Kutta-Joukowski theorem displayed in Equation 7.3.1.

In the example which follows, all of the lifting surfaces of the leading and trailing aircraft were modeled with a vortex-lattice method. The control surfaces of the trailing aircraft were also modeled. Entries of induced drag coefficient and aileron deflection look-up-tables were calculated by positioning the trailing aircraft at a multitude of locations behind the leading aircraft. At each location the trailing aircraft was trimmed to counter the forces and moments induced by

the leading aircraft vortex. The process was repeated at multiple vertical and lateral relative positions.

Figures 8.1 and 8.2 display the resulting surfaces of induced drag coefficient and aileron deflection respectively. Figure 8.1 indicates that the induced drag is smallest at approximately



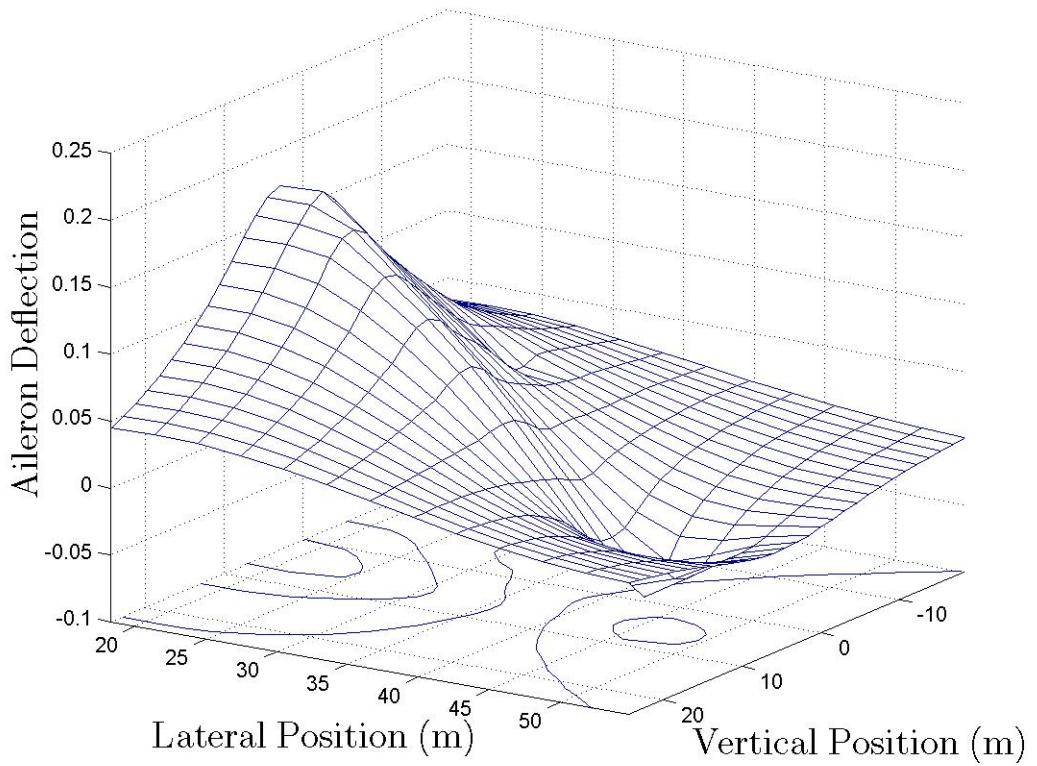
8.1: Induced drag coefficient surface

45 meters of relative lateral separation and 5 feet vertical separation. Near this extremum, the function appears to be near quadratic. Further away, the curvature changes sign and the gradient becomes much smaller.

Figure 8.2 indicates the maximum aileron deflection occurs at approximately 47 meters of relative lateral separation and 5 feet vertical separation, very near to the induced drag coefficient extremum. It appears quadratic near the extremum but flattens further away.

The trailing aircraft is modeled with an 11-state, 4-input, 10 Hz discrete state-space model. The states of the linear model are displayed in Table 8.1.

The inputs to the linear model are displayed in Table 8.2. The model outputs were identical



8.2: Induced aileron deflection surface

Symbol	Description
u	velocity out the nose of the aircraft
v	velocity out the right wing of the aircraft
w	velocity out the belly of the aircraft:
ϕ	roll angle
θ	pitch angle
ψ	yaw angle
p	roll rate
q	pitch rate
r	yaw rate
h	relative vertical separation distance
y	relative vertical separation distance.

8.1: Formation Flight Example Open-Loop Plant States

Symbol	Description
δ_E	change in elevator deflection
δ_A	change in aileron deflection
δ_R	change in rudder deflection
δ_T	change in thrust

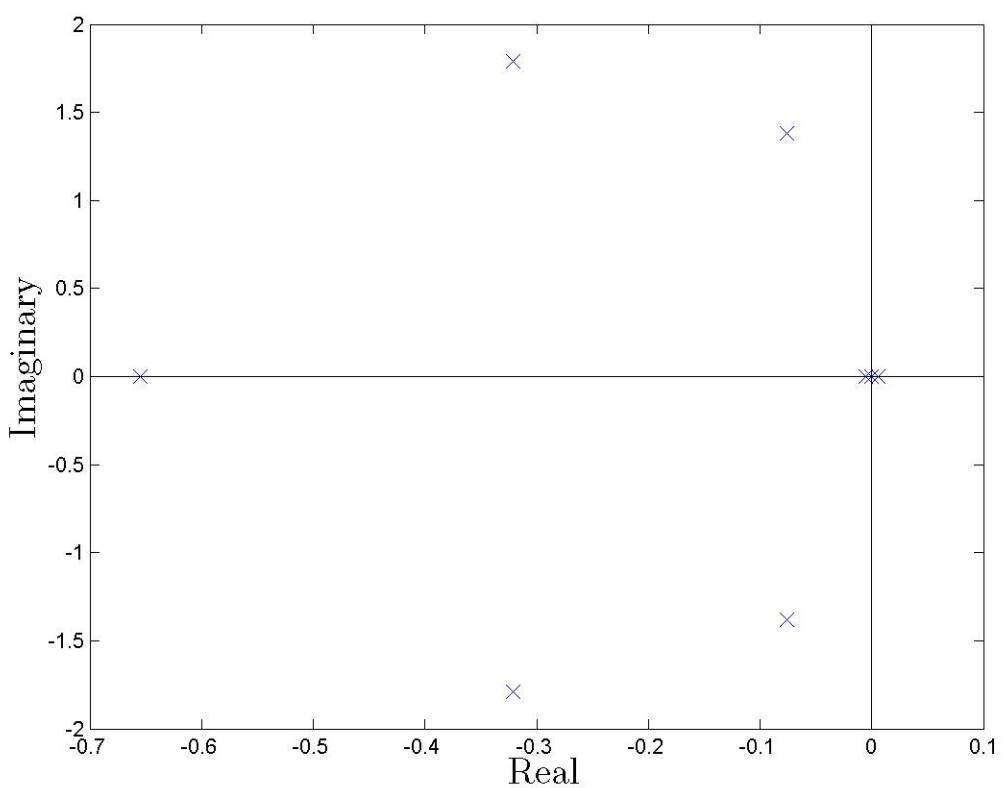
8.2: Formation Flight Example Open-Loop Plant Inputs

to the states displayed in Table 8.1. The eigenvalues of the model along with their associated frequency (ω) and damping ratio (damp) are listed in Table 8.3 and displayed in Figure 8.3

λ	ω	damp
-0.32+1.79i	1.82	0.18
-0.32-1.79i	1.82	0.18
-0.076+1.38i	1.38	0.055
-0.076-1.38i	1.38	0.055
-0.65	0.65	1
-0.0057	0.0057	1
0.0057	0.0057	-1
0.00019	0.00019	-1
2.43e-007	2.43e-007	-1
0	0	-1
0	0	-1

8.3: Formation Flight Example Eigenvalues of the Open-Loop Plant Model.

The model is unstable with three eigenvalues in the right half plane. Two are from the longitudinal states and one from the lateral states. The lateral states also have two eigenvalues at zero.



8.3: Eigenvalues of the open-loop plant model.

8.3 Inner Loop Control Design

This section describes the design of a stabilizing inner-loop control of the aircraft used in the formation flight example. It is designed independently of the extremum-seeking control; the extremum seeking control methodology assumes that the system plant is stable and is able to track commands issued by the extremum-seeking loop.

An inner-loop control system was built around the aircraft model to ensure stability and sufficient tracking of relative position commands. The inner-loop control design objectives were to

1. Stabilize the system.
2. Provide zero steady-state tracking error of lateral and vertical relative position commands issued by the extremum seeking control.
3. Maintain longitudinal relative position to prevent the trailing aircraft from slowly drifting out of formation.
4. Minimize roll angle used during maneuvering
5. Maintain low angular rates in order to minimize potential passenger discomfort.

Some of these objectives are contradictory and so not always fully realizable. For instance to meet the goal of swift tracking, the goal of minimizing roll angle can not be fully met. As with most control designs, a balance of the design objectives had to be achieved.

A linear-quadratic-regular (LQR) design technique was selected for construction of the control system. More specifically an LQR-Tracker design was chosen. This technique was selected due to its ease of implementation and guaranteed robustness properties. In the technique, the plant is augmented with integral states of values which the designer desires approach zero. The augmented plant is then used in a standard LQR design which minimizes the cost function

$$\int_0^{\infty} x^T Q x + u^T R u \, dt$$

where Q and R are weighting matrices on the states, x , and inputs, u , respectively. It is assumed that all states of the augmented plant are available for feedback. If all states are not available, a Kalman filter can be used to estimate the states. The combination of an LQR controller with a

Kalman filter is called a Linear Quadratic Gaussian controller (LQG controller). A derivation of the LQR and the extension to LQR-tracker is supplied in appendices A, and B.

The system was designed to have inputs listed in table 8.4. These inputs allow the extremum seeking control loop to drive the aircraft to relative lateral and vertical positions.

Symbol	Description
δh	lateral relative position command
δy	vertical relative position command

8.4: Formation flight example closed-loop system inputs

The plant was augmented with integral states of the lateral position error, vertical position error, longitudinal velocity command, and roll angle. These integral states were selected to assist meeting the control design objectives of zero steady-state tracking error of the relative position commands, maintaining longitudinal relative position, and minimizing roll angle.

The outputs of the system were chosen to be those shown in table 8.5. This choice allows analysis of all of the initial aircraft states, the augmented states, and the actuator commands. Each of these are used to assess the control system performance.

The control design weightings Q and R were selected to primarily penalize roll angle and change in longitudinal velocity. They were selected by an iterative process while attempting to meet a balance between all of the competing design goals. The final design used the weighting

Symbol	Description
u	velocity out the nose of the aircraft
v	velocity out the right wing of the aircraft
w	velocity out the belly of the aircraft:
ϕ	roll angle
θ	pitch angle
ψ	yaw angle
p	roll rate
q	pitch rate
r	yaw rate
h	relative vertical separation distance
y	relative vertical separation distance.
δ_E	change in elevator deflection
δ_A	change in aileron deflection
δ_R	change in rudder deflection
δ_T	change in thrust

8.5: Formation flight example closed-Loop system outputs

matrices

$$Q = \begin{bmatrix} 0.1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.001 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.001 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 100 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 30 \end{bmatrix} \quad (8.3.1)$$

and

$$R = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 100 \end{bmatrix} \quad (8.3.2)$$

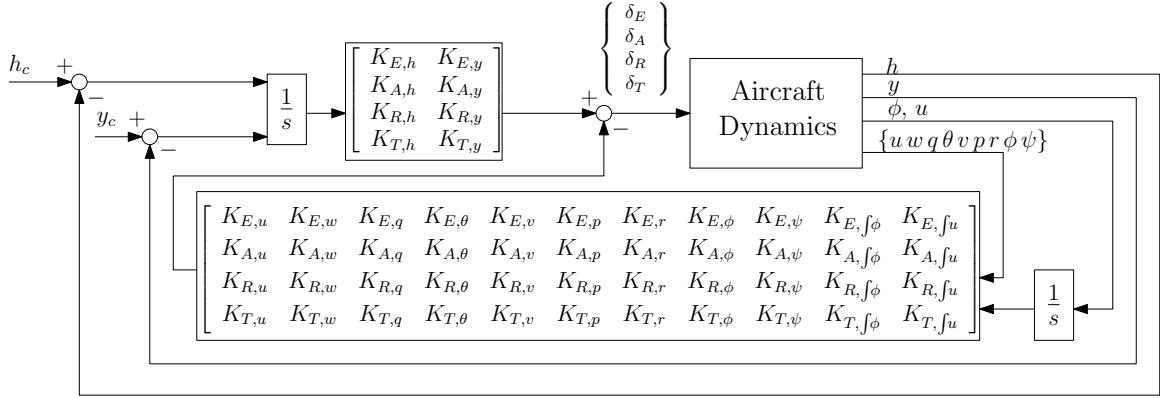
In the Q matrix, the largest weighting was placed on the integral of roll angle. This was necessary to minimize the roll angle used for lateral positioning. The integral states of vertical positioning, horizontal positioning, and longitudinal positioning also had relatively high weightings. In the R matrix, the largest weighting was placed on rudder deflection. This also proved necessary to meet the control design objective of minimizing roll angle.

The resulting control gains are displayed in Table 8.6 where the gains are indexed by the signal which the gain is multiplying, and the command which the multiplied signal is affecting.

An interconnection of inner-loop control system is shown in figure 8.4. In the figure h_c and y_c denote the vertical and lateral relative position commands. The blocks marked with $\frac{1}{s}$ indicate integration of the signals entering the block. Controller gains are indicated by $K_{\cdot,\cdot}$, where the second subscript denotes the signal which the gain is multiplying, and the first subscript denotes

	δE	δT	δA	δR
u	0.71	2.27	-1.39e-010	5.32e-012
v	3.34	-11.33	8.75e-010	-2.64e-011
w	-7.068	4.29	-3.54e-010	1.07e-011
ϕ	-2586.40	8799.10	-6.78e-007	2.05e-008
θ	-2.97	8.19	-5.60e-010	1.27e-011
$f h$	-1.26	2.30	-1.74e-010	3.89e-012
ψ	2.48e-009	-1.18e-008	18.52	-0.96
p	-1.54e-010	7.44e-010	1.76	0.178
q	1.81e-009	-8.66e-009	13.42	-2.11
r	-1.34e-007	6.38e-007	-989.92	51.97
h	1.91e-006	-9.07e-006	14224.93	-739.31
y	4.12e-010	-1.69e-009	10.30	0.044
$f \phi$	-2.20e-009	1.064e-008	-1.68	0.986
$f y$	1.03e-010	-3.84e-010	3.12	0.053
$f u$	5.02	2.19	-1.12e-010	3.46e-012

8.6: Innerloop controller gain values



8.4: Inner loop control interconnection

the command which the multiplied signal is affecting.

The resulting closed loop system proved stable with the eigenvalues displayed in Table 8.7.

Figure 8.5 displays the closed loop eigenvalues along with their open-loop counterparts. The open-loop and closed-loop eigenvalues are connected by a line indicating the trajectories which the eigenvalues follow as the closed loop gains vary from zero to their final values.

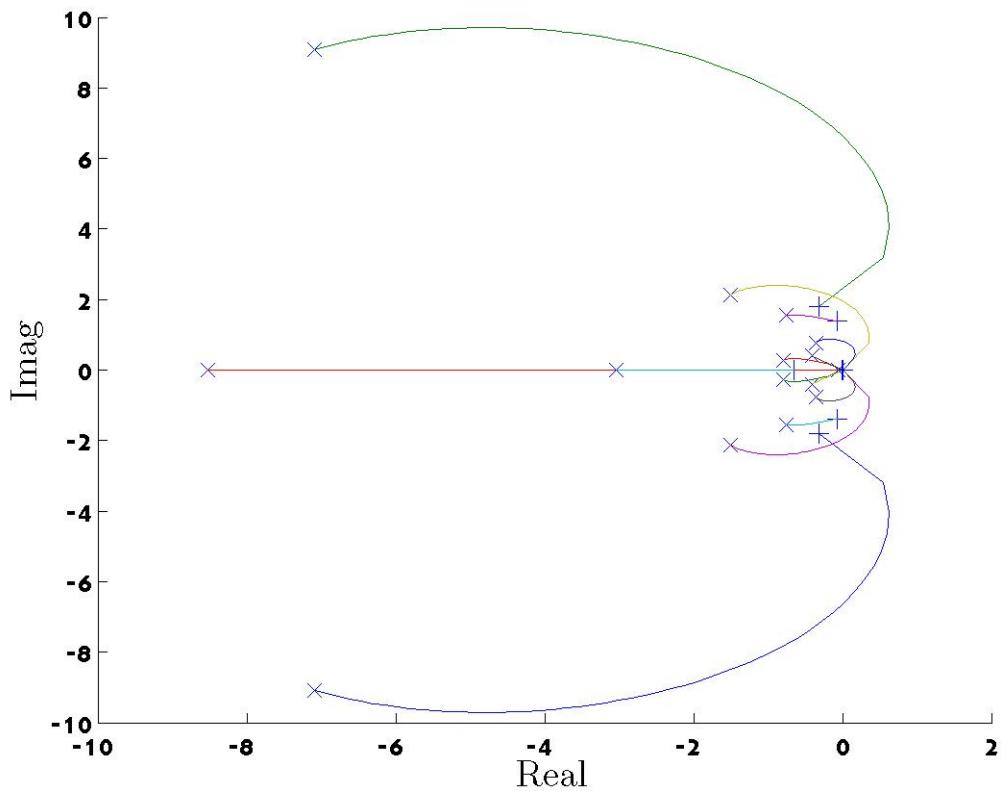
Figures 8.6 and 8.7 display the closed loop system response to a δh step command. The aircraft changes altitude quickly and reaches the commanded altitude in approximately 6 seconds with barely-noticeable over-shoot. Zero steady-state tracking is achieved in approximately 8 seconds. Small changes in pitch angle and pitch rate result during the transition. The elevator movement is small. The change in velocity of the aircraft initially decreases but a thrust command brings the velocity quickly back to zero thus sufficiently maintaining relative longitudinal position.

Figures 8.8 and 8.9 display the closed loop system response to a δy step command. The aircraft moves laterally quickly and reaches the commanded altitude in 8 seconds with very small overshoot. Zero steady-state tracking is achieve in approximately 12 seconds. The aircraft initially rolls to approximately 0.2 radians and then returns close to 0 radians. The ailerons deflect to approximately 0.4 radians to achieve this roll angle. Side-slip reaches -0.05 radians as the rudder is used to assist with lateral movement in an attempt to minimize the necessary roll angle. Larger rudder commands could be used to further assist in minimizing roll angle, but the side-slip on the aircraft would then grow larger.

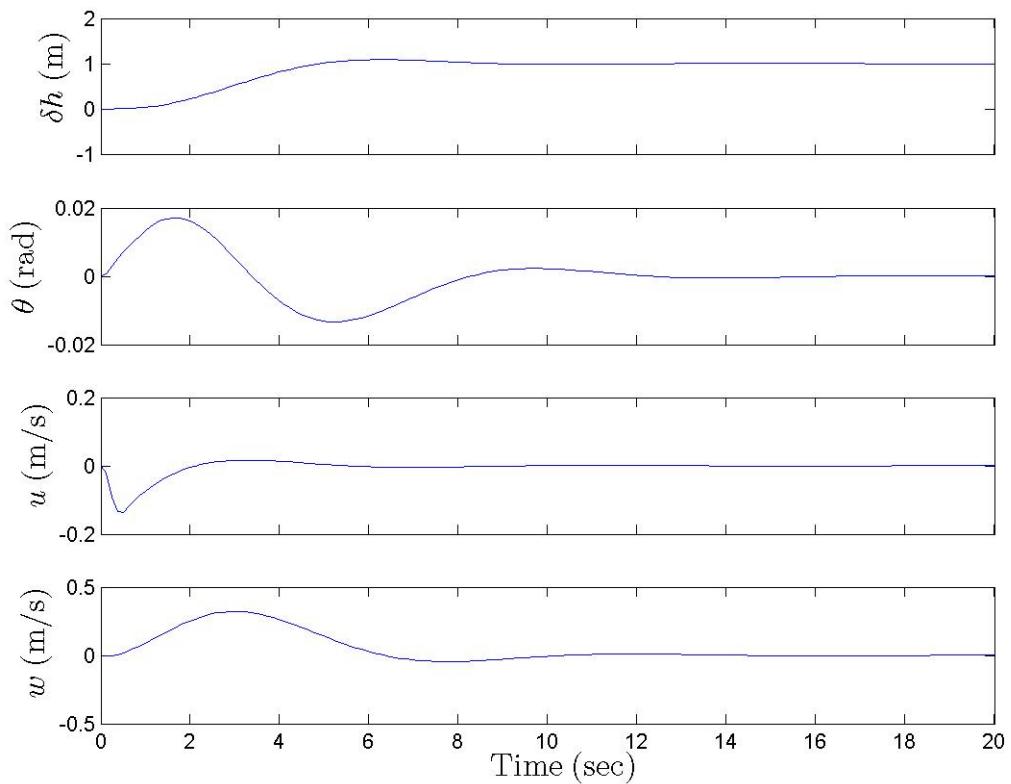
The figures indicate that the inner-loop control design objectives were met with the control design. The closed loop system is stable; zero steady-state tracking error is achieved of the lateral

λ	ω	damp
-7.09077+9.07893i	11.5198	0.61553
-7.09077-9.07893i	11.5198	0.61553
-8.5259	8.5259	1
-3.0385	3.0385	1
-1.5029+2.1339i	2.61	0.57584
-1.5029-2.1339i	2.61	0.57584
-0.7609+1.5688i	1.7436	0.4364
-0.7609-1.5688i	1.7436	0.4364
-0.79723+0.28409i	0.84633	0.94198
-0.79723-0.28409i	0.84633	0.94198
-0.36068+0.76365i	0.84454	0.42707
-0.36068-0.76365i	0.84454	0.42707
-0.40231+0.39702i	0.56523	0.71177
-0.40231-0.39702i	0.56523	0.71177
-0.056864	0.056864	1

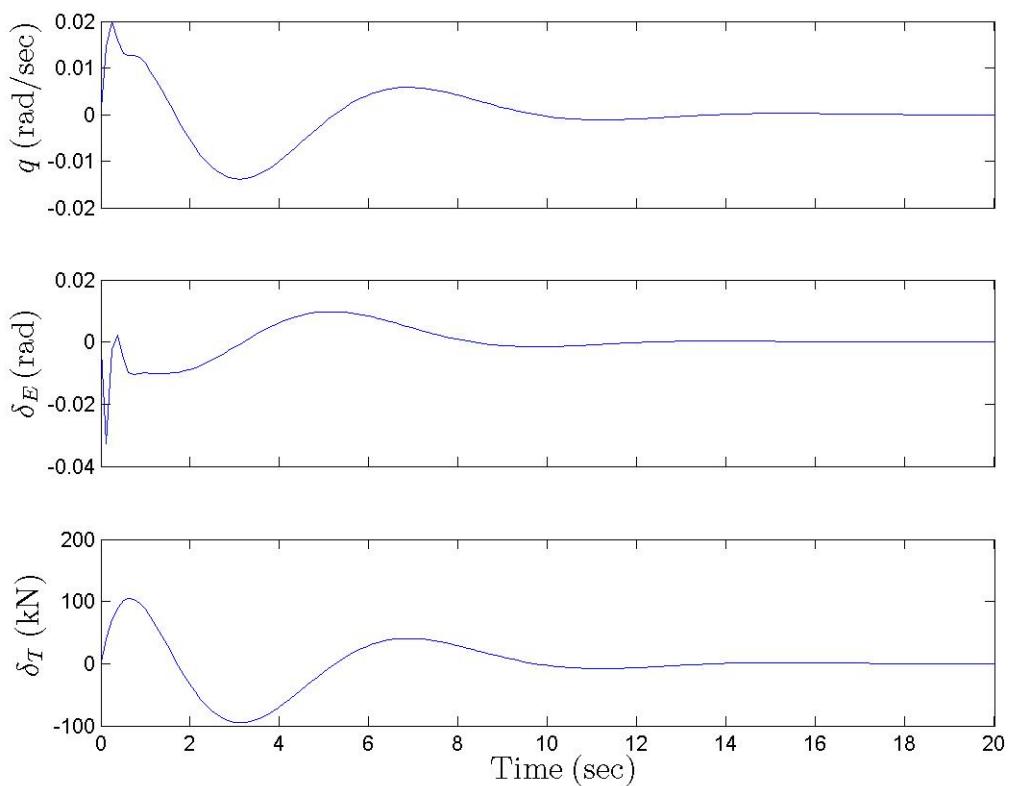
8.7: Eigenvalues of the closed-loop plant model.



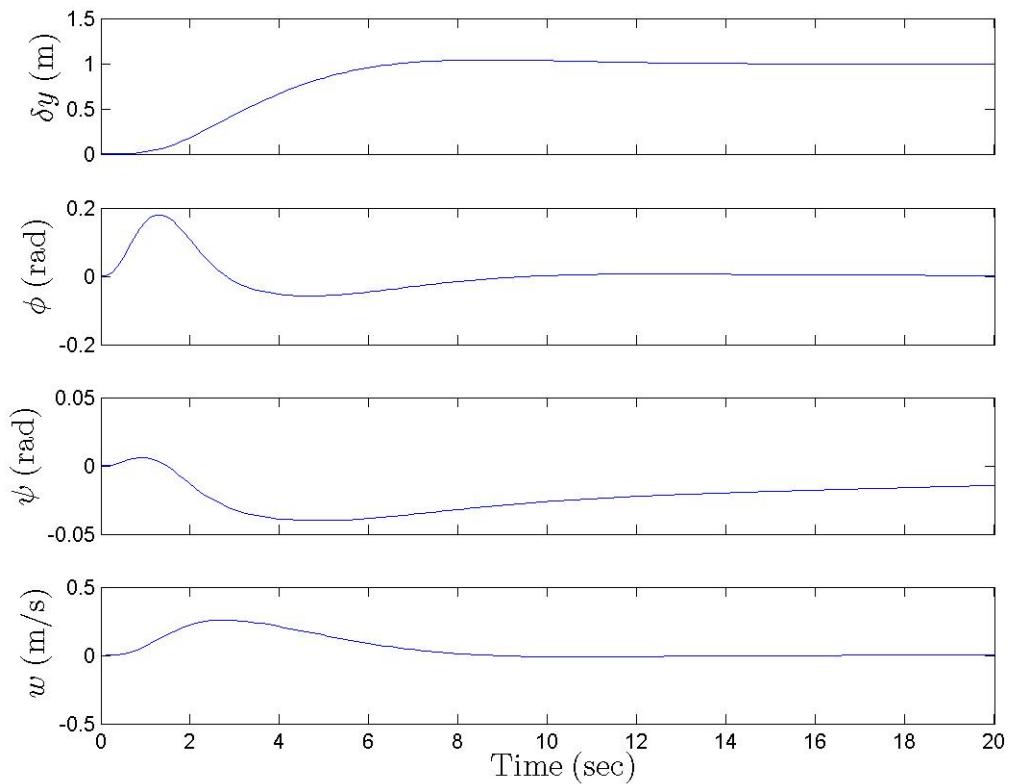
8.5: Open loop and close loop eigenvalues of the plant model. The open loop eigenvalues are denoted by +; closed loop eigenvalues by \times . The solid lines denote the trajectory the open loop poles follow to reach their closed loop positions



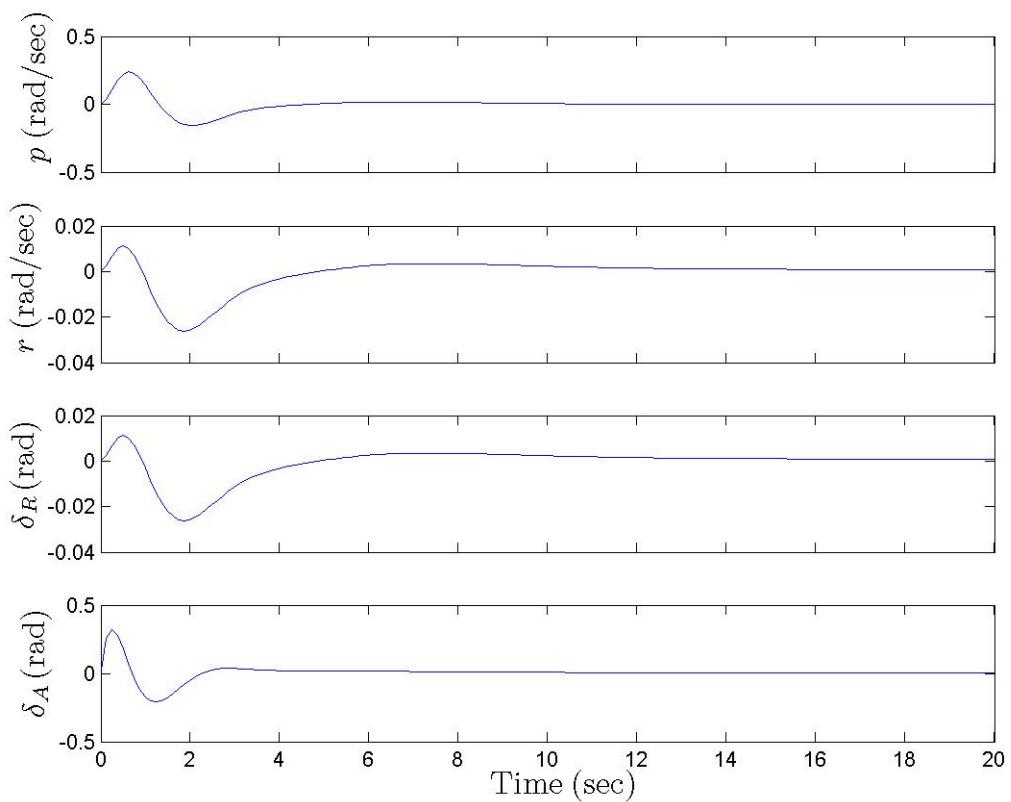
8.6: Step response of the closed loop system longitudinal axis. δh is the change in relative vertical position; θ incremental pitch angle; u incremental longitudinal velocity; w incremental vertical velocity



8.7: Step response of the closed loop system longitudinal axis. q is the pitch rate of change δ_T is the incremental thrust; δ_E the incremental elevator angle



8.8: Step response of the closed loop system lateral-directional axis. δy is the change in relative lateral position; ϕ is the incremental roll angle; ψ the incremental sideslip angle; w the incremental lateral velocity



8.9: Step response of the closed loop system lateral-directional axis. p is the roll rate; r is the slideslip rate; δ_R the incremental rudder angle; δ_A the incremental aileron angle

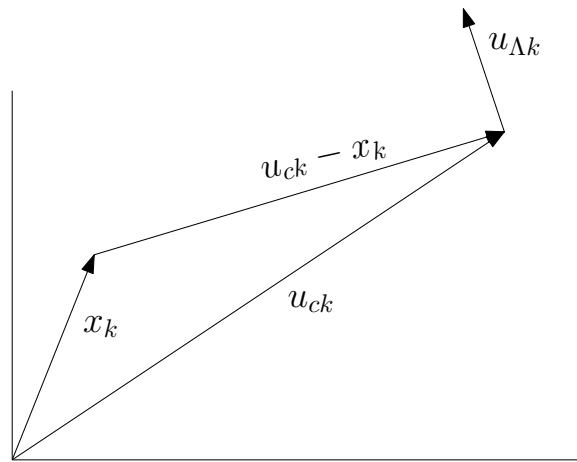
and vertical relative position commands; longitudinal relative position is maintained; and the aircraft experiences small roll angles and low angular rates.

8.4 Persistent Excitation Design

The persistent excitation vector $x_{\Lambda k}$ for the formation flight example was chosen by considering expected characteristics of the performance function, characteristics of aircraft response, and a desire to minimize passenger discomfort. Since the performance function was expected to be noisy, it was desirable to build the persistent excitation such that its magnitude would be large enough to reduce the effects of noise on the Kalman filter estimates. Since the closed loop plant takes 8 seconds to reach its command in its slowest axis, the persistent excitation was chosen to not command any faster than a rate of 0.125 Hz. In order to minimize passenger discomfort it was desired to avoid the seemingly random movement which results when separate sinusoidal signals command each axis.

With all of these considerations in mind, and after a few iterations of trial and error, the persistent excitation was chosen to be a 0.6 rad/s, 0.7 meter sinusoidal acting orthogonal to the direction commanded by the extremum seeking controller. Further discussion on the choice of persistent excitation can be found in appendix C.

The chosen persistent excitation signal is constructed as illustrated in figure 8.10. In the



8.10: Geometry of the persistent excitation signal used in the formation flight example.

figure, $x_{\Lambda k}$ is the persistent excitation signal at iteration k . The vector x_k is the current coordinate

measurement, x_{ck} is the current command issued by the extremum seeking loop. The persistent excitation is defined as

$$x_{\Lambda k} = \Lambda \sin(\omega t + \phi_k) \hat{\epsilon}_k$$

such that

$$\langle \hat{\epsilon}_k, x_{ck} - x_k \rangle = 0.$$

Here Λ is the persistent excitation signal magnitude and $\hat{\epsilon}_k$ is the unit vector orthogonal to $x_{ck} - x_k$. The phase angel ϕ_k is defined as

$$\phi_k = \phi_{k-1} + \omega (x_{ck} - x_k)$$

The persistent excitation vector is added to the extremum position estimate to create the command to the plant

$$x_{ck} + x_{\Lambda k}$$

This choice avoids the complicated and unintuitive path which results from separate persistent excitation signals applied to each axis.

8.5 Kalman Filter Design

Since the persistent excitation was chosen to iterate at 0.6 rad/s, the Kalman filter was chosen to iterate at 0.1 Hz. This allowed 10 measurement points to be taken for each cycle of persistent excitation. Other rates were also tried, but 0.1 proved to be effective. To reduce the effect of measurement noise, measurements were taken at 10 Hz in between the Kalman filter iterations and averaged to form $f_{k,n}$, $x_{1_{k,n}}$, and $x_{2_{k,n}}$ of (4.6.1). This was reasonable due to the small change in performance function coordinates and magnitude between measurements.

The covariance matrices were chosen as

$$V_k = \begin{bmatrix} 1/3 & 0 & 0 & 0 & 0 \\ 0 & 1/3 & 0 & 0 & 0 \\ 0 & 0 & 1/3 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

and

$$W_k = 1.$$

The matrix H_k in equation (4.6.2) was selected to have 15 rows ($N = 15$). This too was ultimately selected by trial and error. The other elements of equation (4.6.2) were selected to be of compatible size. The command filter was chosen to be an 10 Hz integrator, $\text{Filt} = 0.1/(z - 1)$. This resulted in a ramping position command in place of the step command generated by the 0.1 Hz Kalman filter estimates.

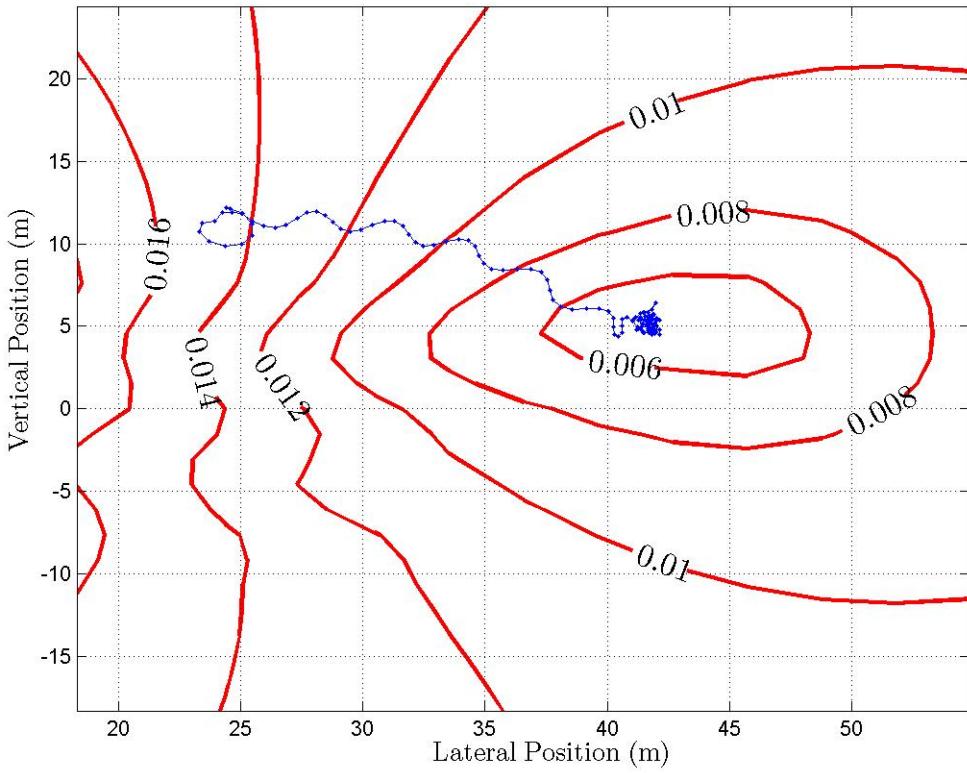
The system was implemented to switch between a steepest-descent and a Newton approach. When the Hessian estimate was positive definite and the minimum singular value of the Kalman filter error covariance $\underline{\sigma}(P_k) < 6$ a Newton approach was used. Gradient descent was used when this was not the case. The switching threshold was used as a tuning parameter and selected by trial and error.

8.6 Simulation Results

The system was tested by simulation. In the simulation, the aircraft was initially positioned to the left and above the leading aircraft right wingtip vortex core. Figure 8.11 depicts the path the aircraft followed during the peak-seeking simulation. The contours represent induced drag coefficient. The system was initially commanded to trace a 0.7 meter radius circle to generate initial gradient and Hessian estimates. It was then allowed to move toward the minimum. The system primarily moved orthogonally through the contours of the plot as it moved to the minimal location. The system reached the local minimum in 300 seconds.

Figure 8.12 depicts the change in induced drag coefficient as the trailing aircraft moved to the optimal position. The ratio of the induced drag coefficient to induced drag coefficient when not under the influence of a vortex (C_{Di}/C_{Di_0}) is shown. The figure indicates that the aircraft started at a location where the vortex negatively affected the induced drag. The minimum induced drag coefficient reached is 40 percent of the value obtained when not under the influence of the vortex.

Figures 8.13 and 8.14 show the gradient and Hessian estimates as a function of time. The solid lines in the figures represent the true gradient and Hessian while the dashed lines represent the estimates at each Kalman filter iteration. The gradient estimate approximates the true gradient well over the length of the entire simulation, as shown in Figure 8.13; however, the Hessian estimate illustrated in Figure 8.14 is less accurate.

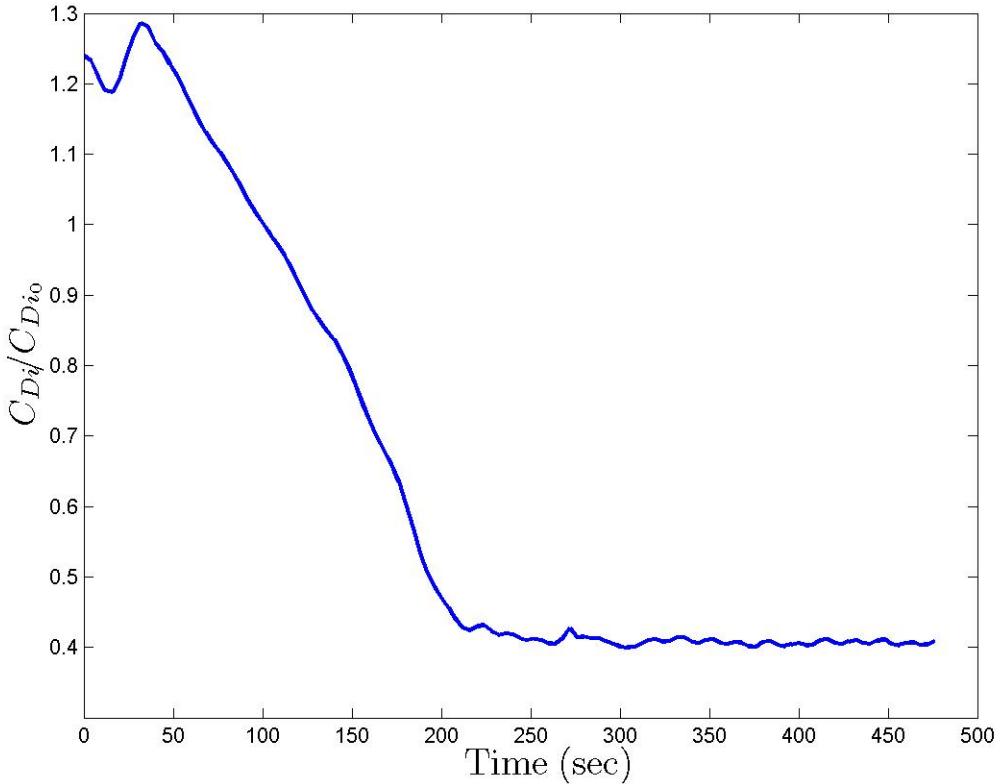


8.11: Aircraft path along induced drag coefficient performance function.

The error between position commands and the aircraft response is depicted in Figure 8.15. The error never exceeds 0.8 meters. The oscillatory behavior is due to the excitation command.

The aircraft Euler angles are depicted in Figure 8.16. The aircraft angles stay within reasonable values, never exceeding 5 degrees. The high-frequency oscillatory appearance of the angles is due to non-smooth commands being provided to the aircraft. Improvement to the command filter could minimize these oscillations. The slower-period oscillations are due to the excitation. The roll and yaw angles share the task of moving the aircraft laterally. By changing the weightings contained in R of the inner-loop control design, surface movements can be tuned to use more roll or yaw angle.

The aircraft surface deflections are displayed in Figure 8.17. As with the Euler angles, the high-frequency oscillation is due to the non-smooth commands to the aircraft and the slower oscillation is due to the excitation. Aileron deflection goes to 10 degrees and rudder deflection goes to 5 degrees when the aircraft is tracing the initially-commanded circle. The simulation ends with all surface deflections except aileron near 0. Aileron deflection remains at 3 degrees because



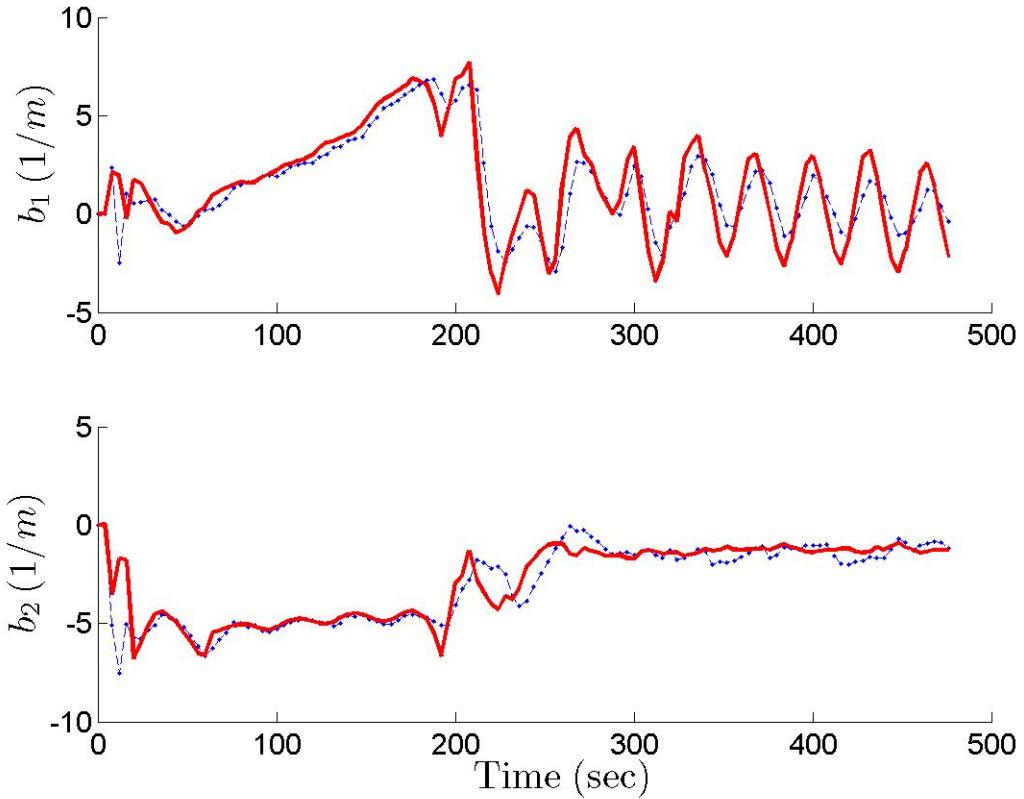
8.12: Ratio of induced drag coefficient to induced drag coefficient when not under the influence of a vortex vs. time.

the aileron continues to counteract the vortex-induced rolling moment.

The system was tested with many starting positions. Figure 8.18 displays 6 separate simulation runs performed with identically tuned systems. Each performed an initialization circle and then used the estimates from the Kalman filter to move toward the extremum. Each was successful in reaching the extremum. The simulation-runs which moved down shallower slopes took longer to reach the extremum. This is as expected since the gradient magnitude is directly proportional to the magnitude of the command to the plant.

8.7 Extension to a Four Aircraft Formation

The formation flight example is now extended to four aircraft in an echelon formation as depicted in figure 8.19. Each aircraft in the formation is impacted by the wingtip vortex of its preceding aircraft; it is assumed that the wingtip vortices of the other aircraft have negligible impact. As

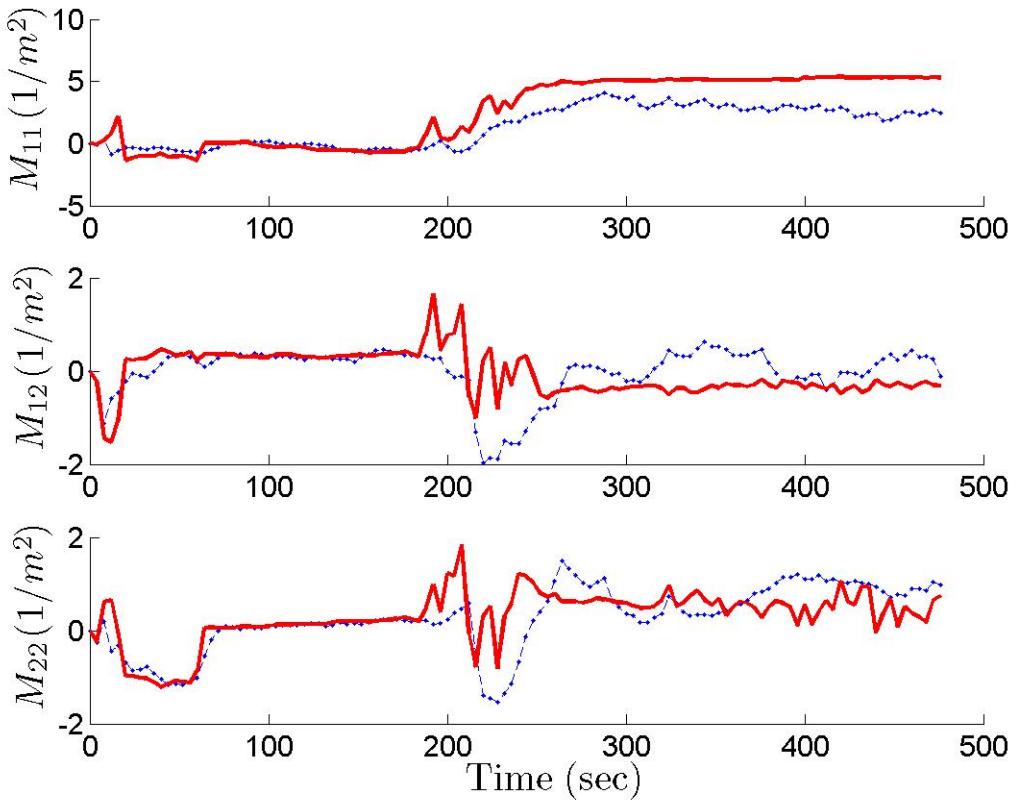


8.13: Gradient estimates and true gradient. Solid line: truth. Dots: Estimate.

in the previous example, each vortex is assumed static and fixed in the reference frame of its preceding aircraft. The goal is for each trailing aircraft to minimize drag by moving to the optimal relative position.

The first aircraft in the formation is assumed to be in straight and level flight. The second aircraft positions itself behind the first to minimize its drag. The wingtip vortex generated by the second aircraft is modeled with a static map fixed in the second aircraft's reference frame. The third aircraft positions itself behind the second aircraft to minimize its drag. Since the second aircraft maneuvers to reach its optimal position, the third aircraft must track a moving performance function. Since the extremum seeking algorithm inputs relative positions, no modifications were required to perform this task. The fourth aircraft in the formation positions itself behind the third to minimize drag. As with the third aircraft, it must track a moving performance function.

The control system and extremum seeking algorithm developed for the two aircraft formation are employed for this extended example without modification. Each aircraft attempts to minimize its drag by positioning itself within the wingtip vortex of the preceding aircraft. No information

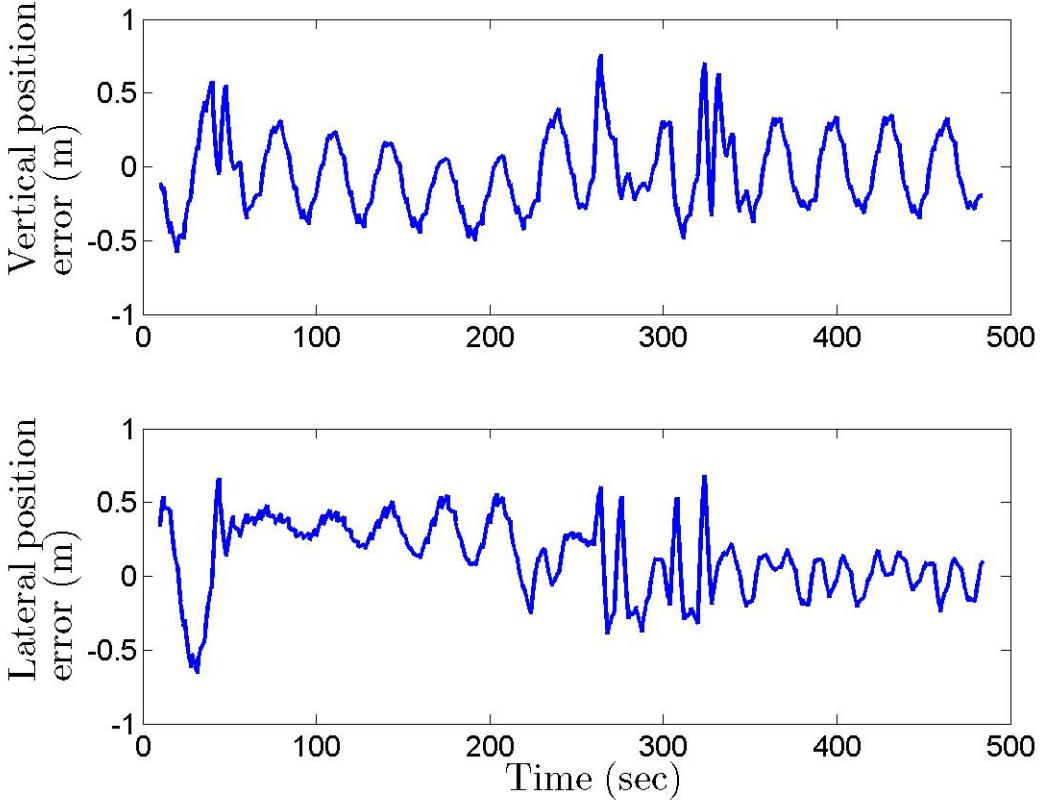


8.14: Hessian estimates and true Hessian. Solid line: Truth. Dots: Estimate.

of the preceding aircraft's states are used. This strategy is best characterized as a predecessor-following strategy.

The question of string stability must be considered with a system consisting of more than two aircraft. The term string stability refers to the response of a string of vehicles to disturbances experienced by the leading vehicle. There has been extensive recent interest in string stability primarily from the viewpoint of intelligent vehicle highway system. See for example, [XG11], [TS10], and [MB10]. The control design presented above does not take string stability into consideration, and so guaranteed string stability is not expected. In fact, it has been shown in [SPH04] and [SD90] that string stability is not possible with a predecessor-follower formation strategy using any linear time-invariant controller.

Simulation confirms that string stability is not achieved by the LQR control systems used in this example. Figures 8.20 and 8.21 display the response of the four-aircraft formation to a step command issued to the leading aircraft in the vertical and lateral directions respectively. A line has been added to the plot to mark the leading aircraft's maximum magnitude. It is clear that

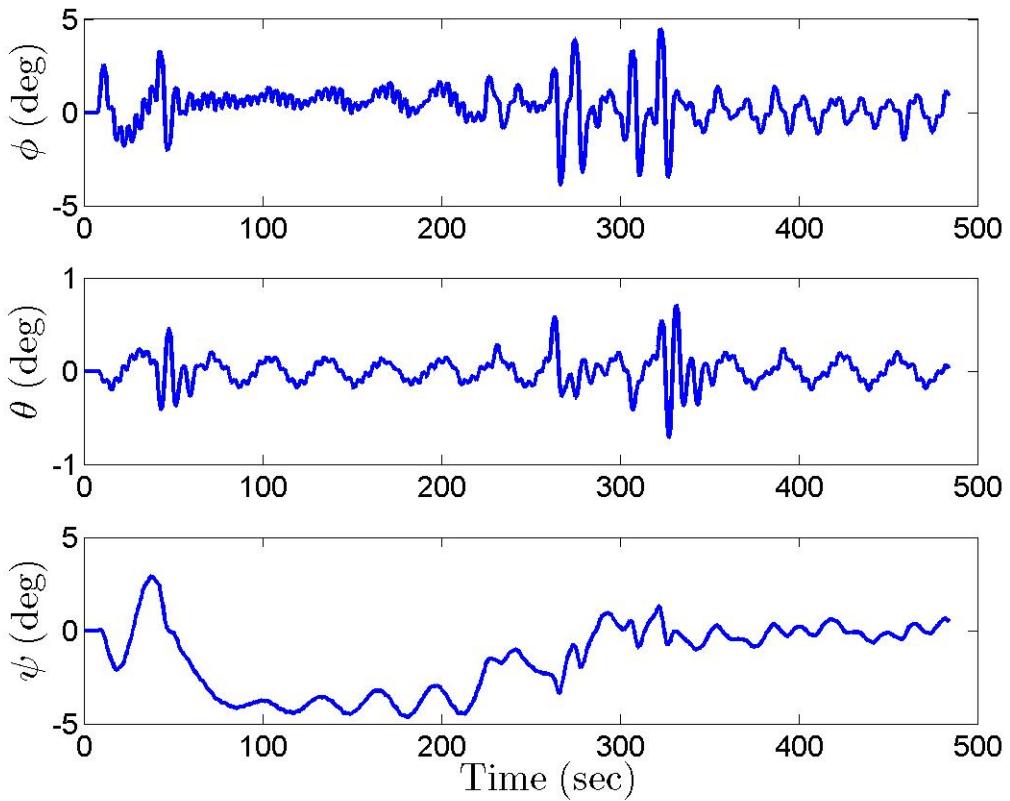


8.15: Vertical and lateral position errors.

the response of the second, third, and fourth aircraft amplify the overshoot of the first. Here, the extremum seeking algorithm was not included in the loop. Instead, the position of the n^{th} aircraft is used as the reference command to the $n + 1^{st}$ aircraft.

String stability is achievable by other formation strategies. One strategy uses bidirectional control which entails each vehicle to move according to information about both preceding and succeeding vehicles. A second strategy to assure string stability is to provide all vehicles in the formation information about the leading vehicle. A third strategy is to use heterogeneous controllers [KD04] [SH07] such that each system in the string will not have identical frequencies at which magnitude peaks above 0 dB occur. Another strategy uses nonlinear controllers.

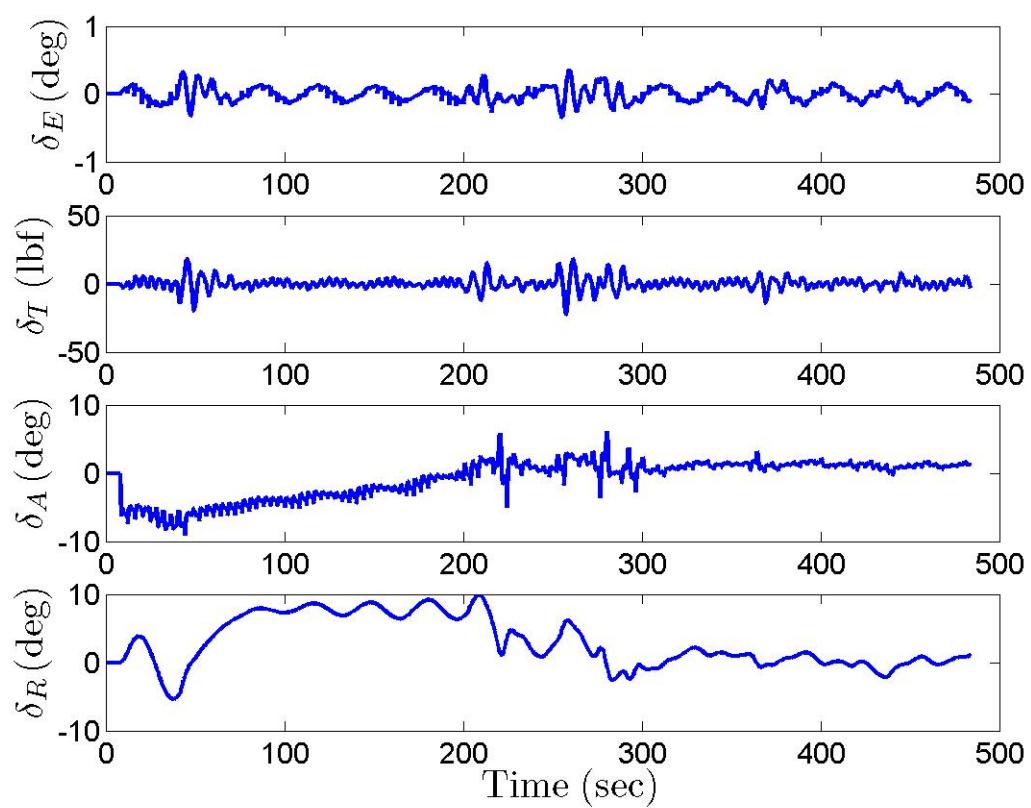
When the extremum seeking algorithm is included in the loop, the string instability of the system does not prevent convergence of each aircraft to the optimal locations. Figures 8.22, 8.23, and 8.24 show the path of the second, third, and fourth aircraft of the four aircraft formation when trying to obtain minimal drag. Each figure is in the reference frame of the preceding aircraft. The performance functions are therefore fixed in each figure even though they are not fixed in the



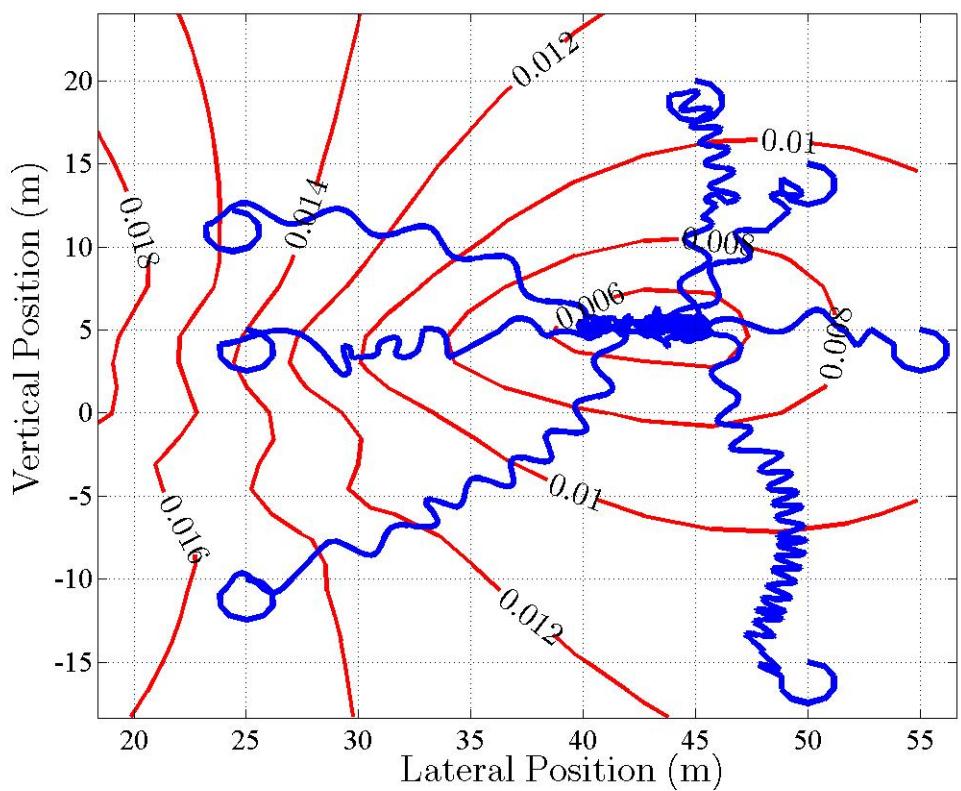
8.16: Aircraft Euler angles versus time.

leading aircraft's reference frame.

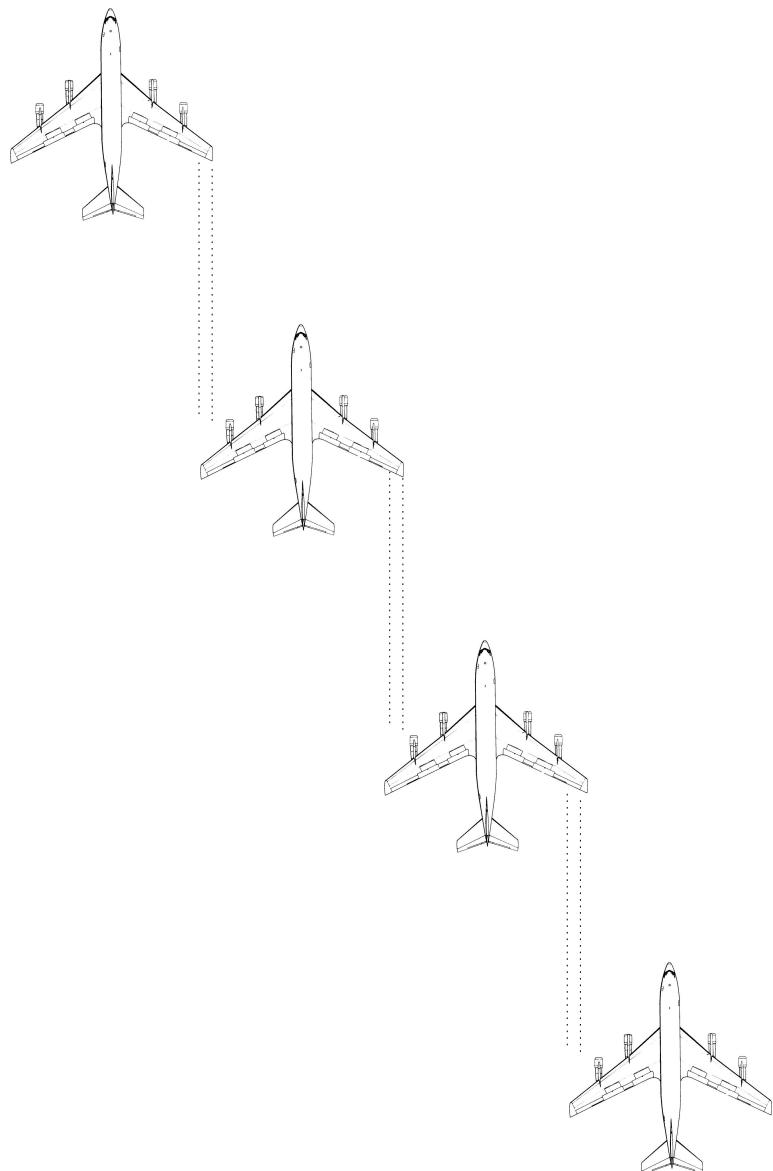
In this formation, the first aircraft does not maneuver. The second aircraft uses the extremum seeking algorithm to find the optimal location behind the first aircraft. It utilizes a persistent excitation signal and the extremum seeking algorithm as described above. The third aircraft also uses the extremum seeking algorithm. It finds the optimal location behind the second aircraft. It does not use any additional persistent excitation but relies upon the movement of the second aircraft to provide observability. It does not use any information about the leading aircraft. The fourth aircraft uses the extremum seeking algorithm to find the optimal location behind the third aircraft. As with the third aircraft, it does not insert any additional persistent excitation into the system. It does not use any information about the leading or second aircraft. The figures clearly show that each aircraft successfully finds the optimal position.



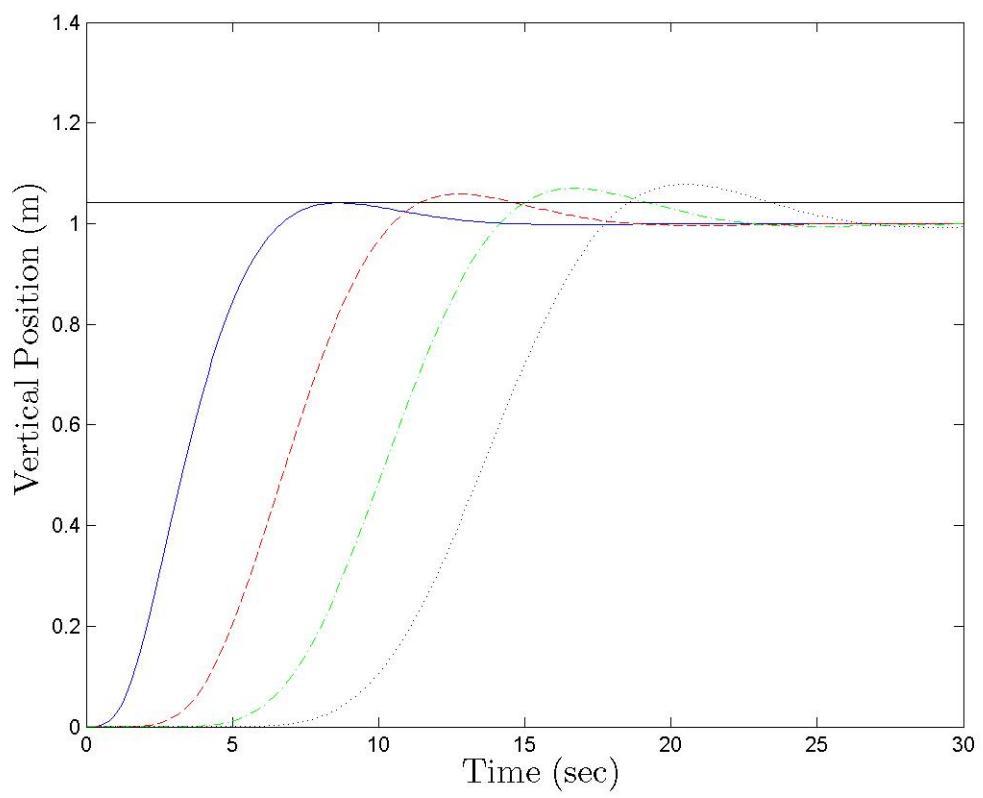
8.17: Aircraft surface deflections versus time.



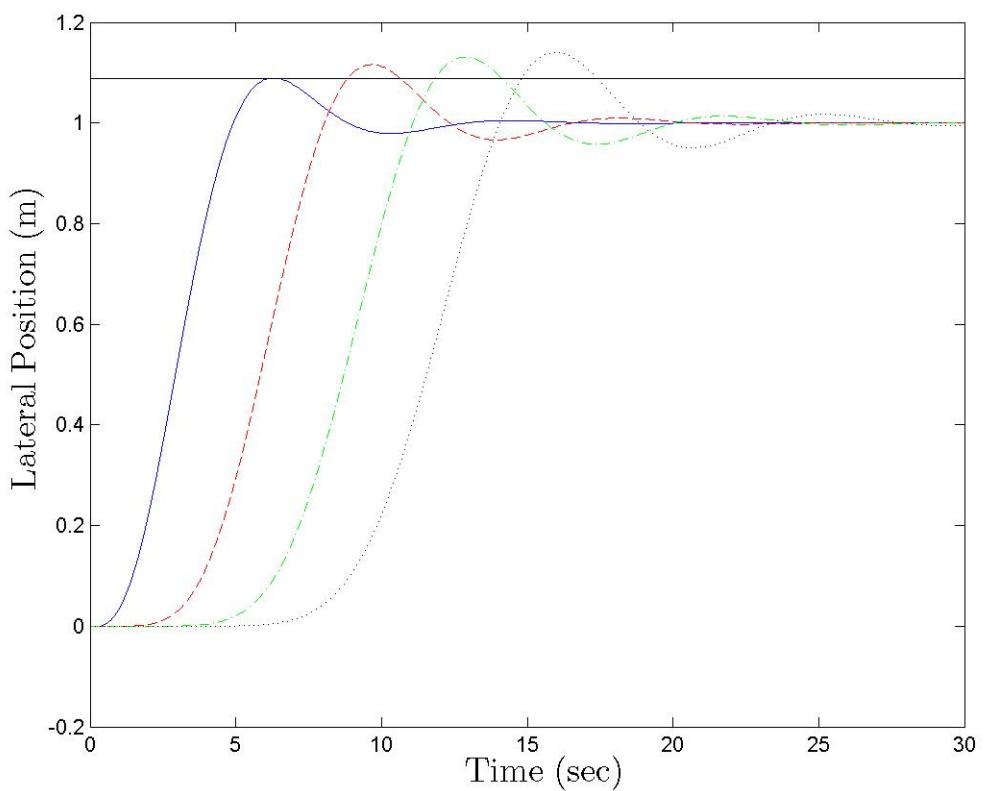
8.18: Induced drag coefficient contour map



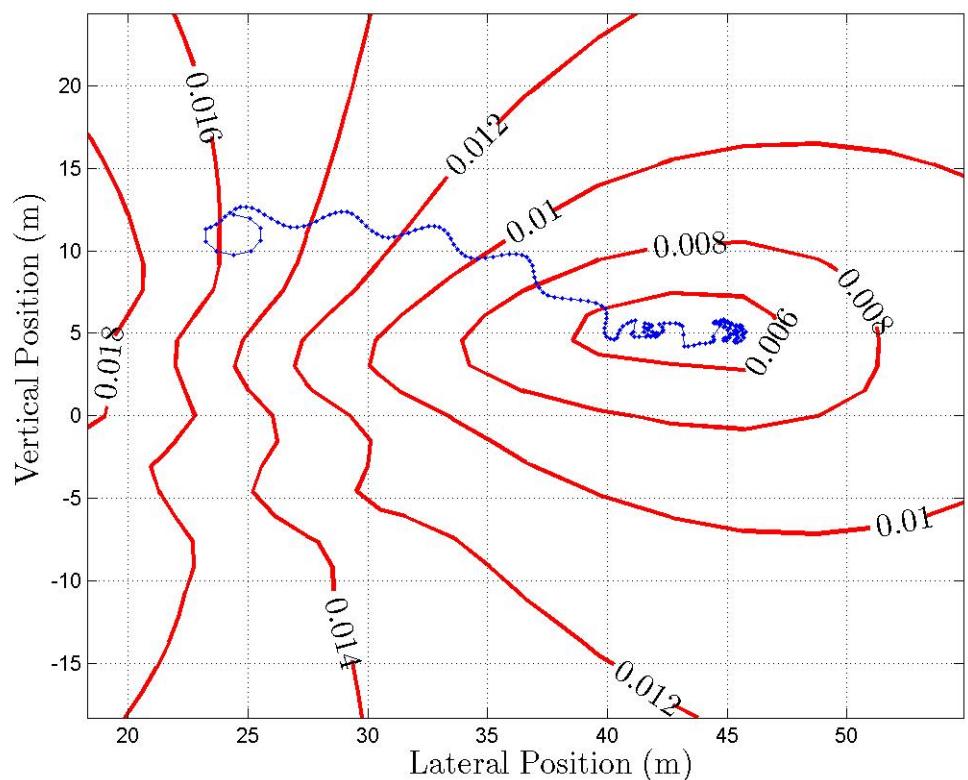
8.19: Four Aircraft in an Echelon Formation. Each aircraft is impacted by the wing tip vortex of its preceding aircraft



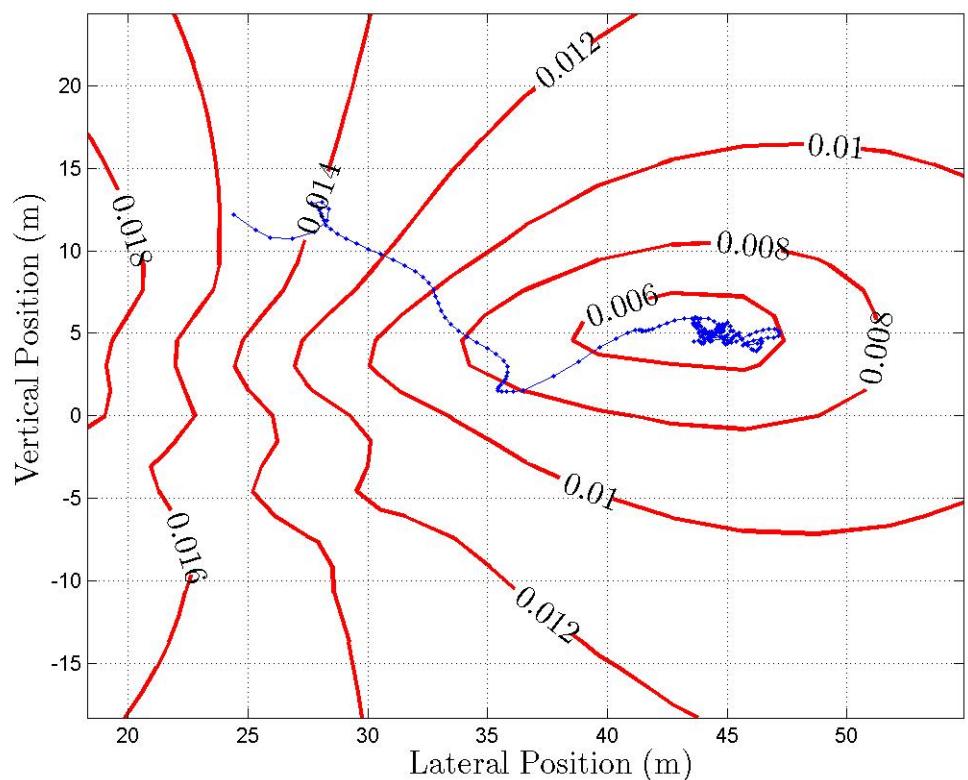
8.20: Vertical Step Response of a Four Aircraft Formation.



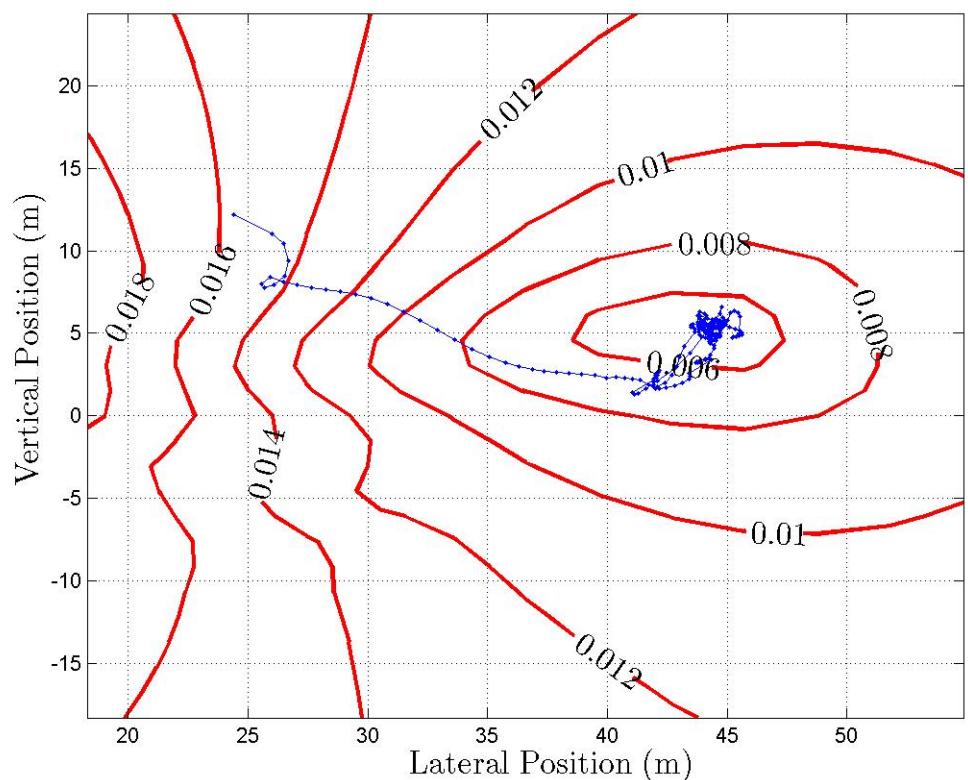
8.21: Lateral Step Response of a Four Aircraft Formation.



8.22: Path of Second Aircraft along induced drag coefficient performance function.



8.23: Path of Third Aircraft along induced drag coefficient performance function.



8.24: Path of Fourth Aircraft along induced drag coefficient performance function.

CHAPTER 9

A Second Application of Extremum Seeking Control to Formation Flight for Drag Reduction

This chapter discusses the implementation of the extremum seeking control method to a Boeing Company developed simulation. It differs from the implementation in the previous chapter in simulation fidelity and in design choices made. The simulation implementation is first discussed, simulation results are then presented. The chapter concludes with a discussion comparing the results of the previous chapter with those of this chapter.

9.1 Description of Implementation and System Design

The extremum seeking algorithm was implemented in a simulation provided by the Boeing Company. As with the simulation discussed in the previous chapter, it too simulated two transport class aircraft in close formation flight. The Boeing simulation was more detailed than the one developed by UCLA. It included an engine model, a dynamic wake model, a Dryden turbulence model, and an inner loop control system designed to counter the influence of the vortex. The aerodynamic models were based off of incremental aerodynamic force look-up tables generated with computational fluid dynamic methods. No measurement noise is included in the simulation.

Two different performance function were selected for two studies. In the first, pitch angle was chosen as the performance function. This was chosen because the wing tip vortex of the leading aircraft induces lift on the trailing aircraft. In order to maintain trim, the trailing aircraft must pitch down. This induced-pitch is then a good analogue of the induced-drag.

The second dependent parameter used to form a performance function was the aileron deflection. This is a good analogue of drag reduction because the aileron deflection is used to retain trim of the trailing aircraft as it is influenced by the wing-tip vortex.

The extremum seeking control system was designed in a similar manner as described in the previous example but did not estimate the performance function Hessian. It therefore used a steepest descent approach to find the extremum. The persistent excitation, in this case, was selected to be two separate signals, one acting in the vertical direction, the other acting in the lateral direction. The vertical signal was chosen as a 0.05 Hz sinusoid with 1 ft magnitude. The lateral signal was chosen as a 0.01 Hz sinusoid with 1 ft magnitude.

The Kalman filter was chosen to operate at 10 Hz. It used 4 rows in the H_k matrix of Equation 4.4.6 discussed in Chapter 4. The initial state covariance matrix was chosen to be

$$W = 10^{-10} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

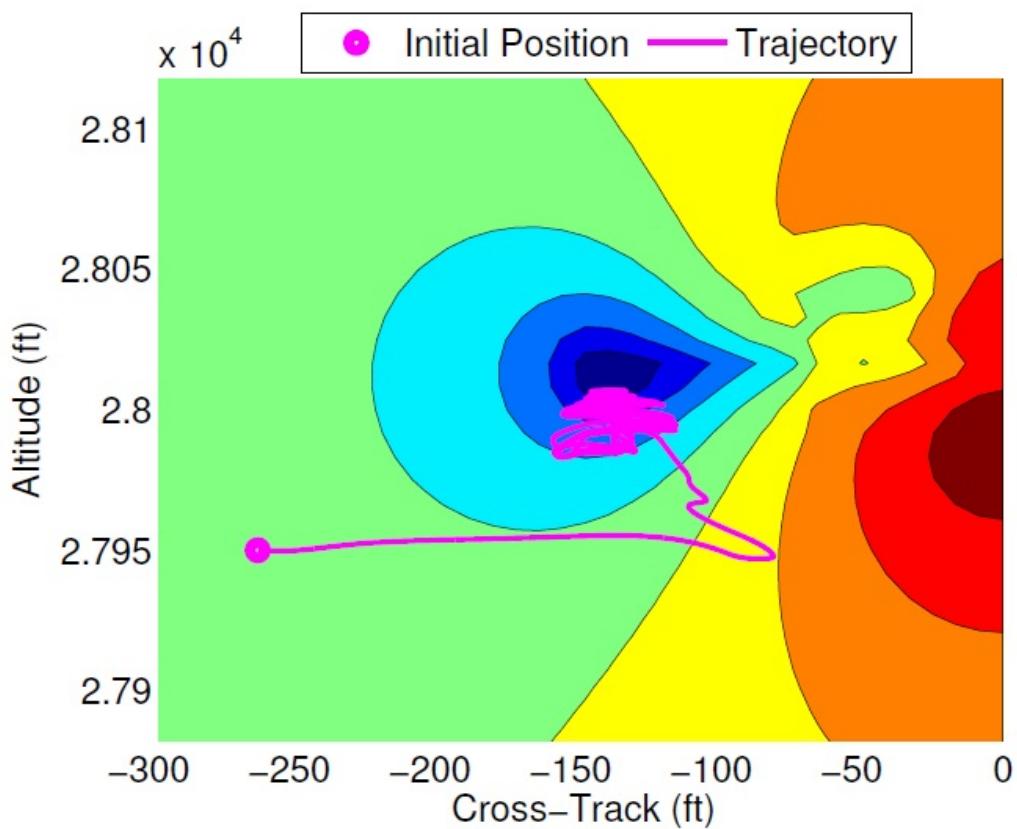
and the initial measurement covariance matrix was chosen to be

$$V = 10^{-8} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & 0 & 9 & 0 \\ 0 & 0 & 0 & 16 \end{bmatrix}$$

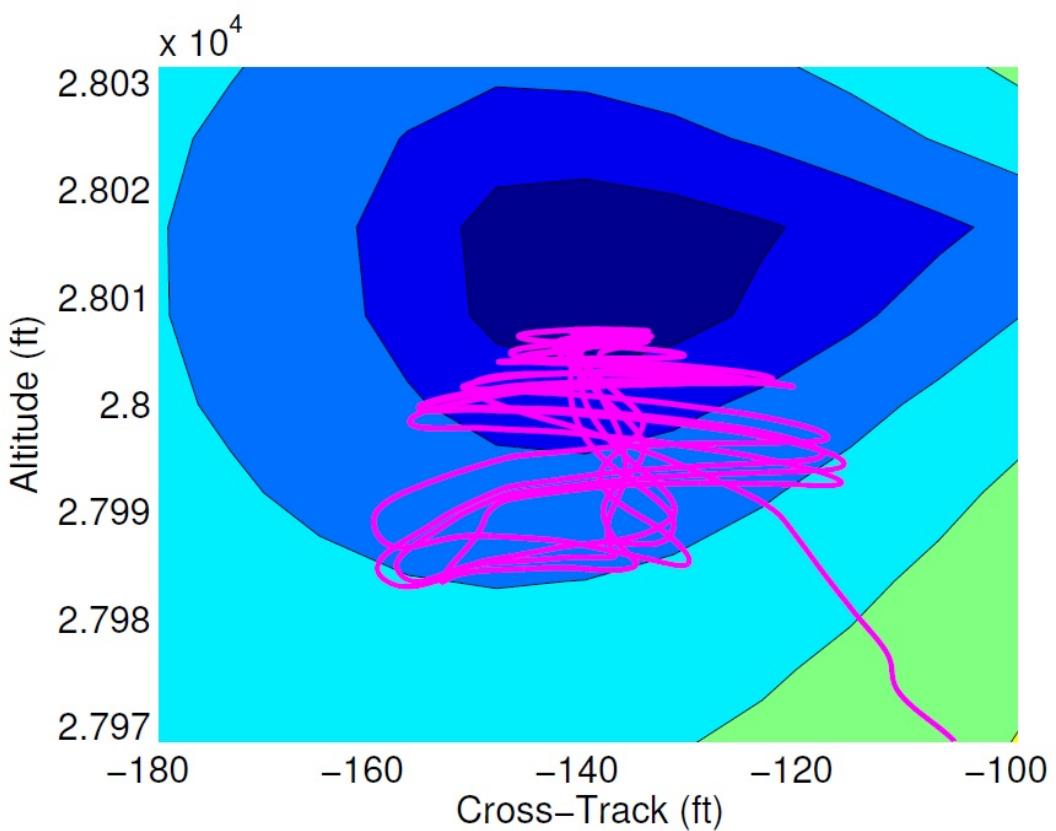
9.2 Simulation Results

The design was tested in the Boeing-developed simulation. The results are displayed in Figures 9.1 through 9.12. Figure 9.1 displays the path of the aircraft following the pitch angle performance function displayed over the induced drag contour map. It takes approximately 500 seconds for the system to reach a point near the pitch angle extremum. The aircraft then slowly oscillate near the extremum. Figure 9.2 more closely displays the oscillation near the extremum. The extremum pitch angle does not quite coincide with the extremum of the induced drag. This is not a surprising result; Section 7.1 of Chapter 7 indicated that analog measurements of induced drag will not necessarily have extremum coincident with the induced drag extremum.

Figure 9.3 displays the time history of the aircraft cross track, altitude, and pitch angle. The time history of the pitch angle corresponds with the time histories discussed above. The pitch angle experiences a relatively quick reduction from 4 degrees to 3.5 degrees in the first 500 seconds and then more slowly moves about the extremum. The figure also indicates that in the first 500 seconds, the aircraft moves laterally from -250 feet to -150 feet of cross track and climbs 50

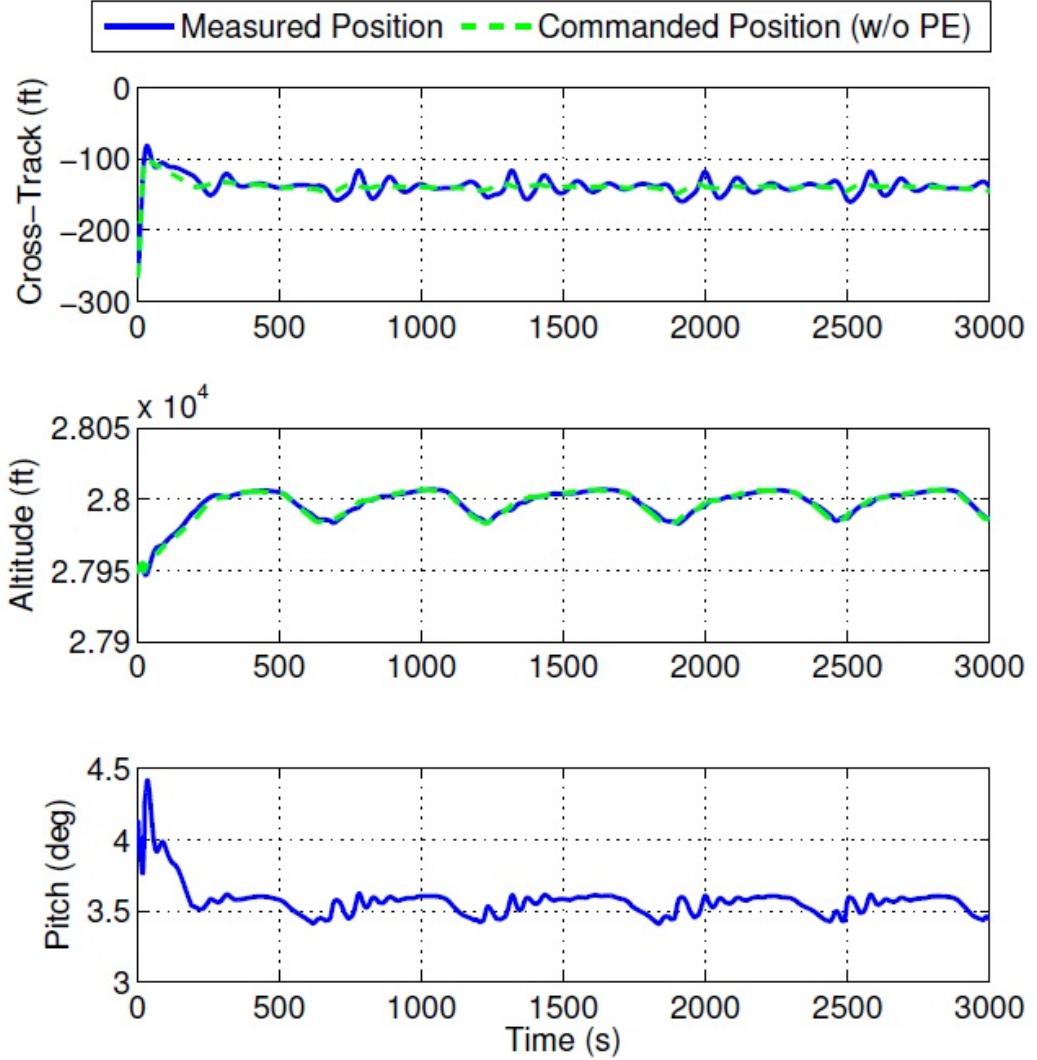


9.1: Aircraft path following the pitch angle performance function displayed over the induced drag contour map. The aircraft initial position is marked with a filled circle. The aircraft path is shown by a solid line. The contour of the pitch angle performance function is indicated by the multicolored contour map; dark blue indicates the smallest performance function magnitude, deep red indicates the highest performance function magnitude.



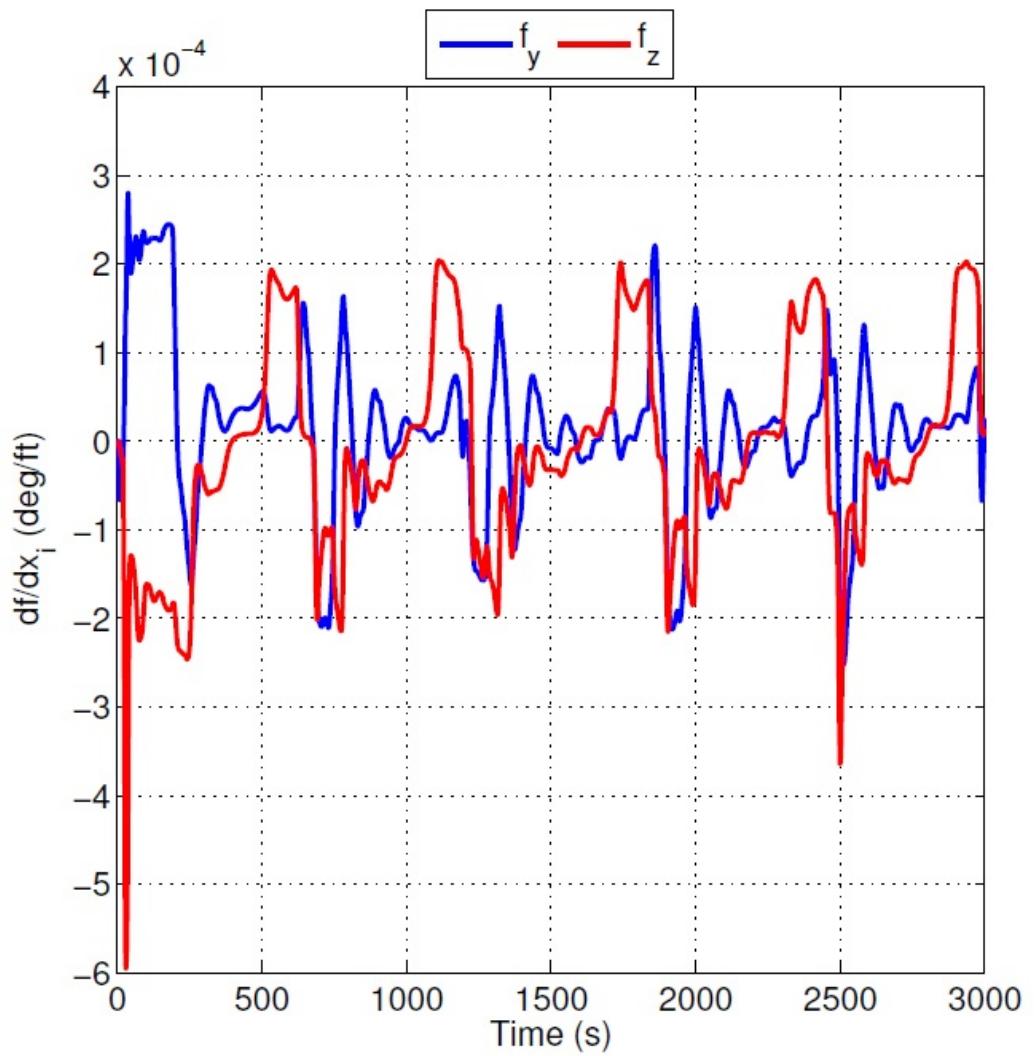
9.2: Aircraft path over the pitch angle performance function near the extremum. The aircraft initial position is marked with a filled circle. The aircraft path is shown by a solid line. The contour of the pitch angle performance function is indicated by the multicolored contour map; dark blue indicates the smallest performance function magnitude, deep red indicates the highest performance function magnitude.

feet in altitude. After the first 500 seconds, the aircraft oscillates in altitude at 0.002 Hz and in cross-track at 0.01 Hz.

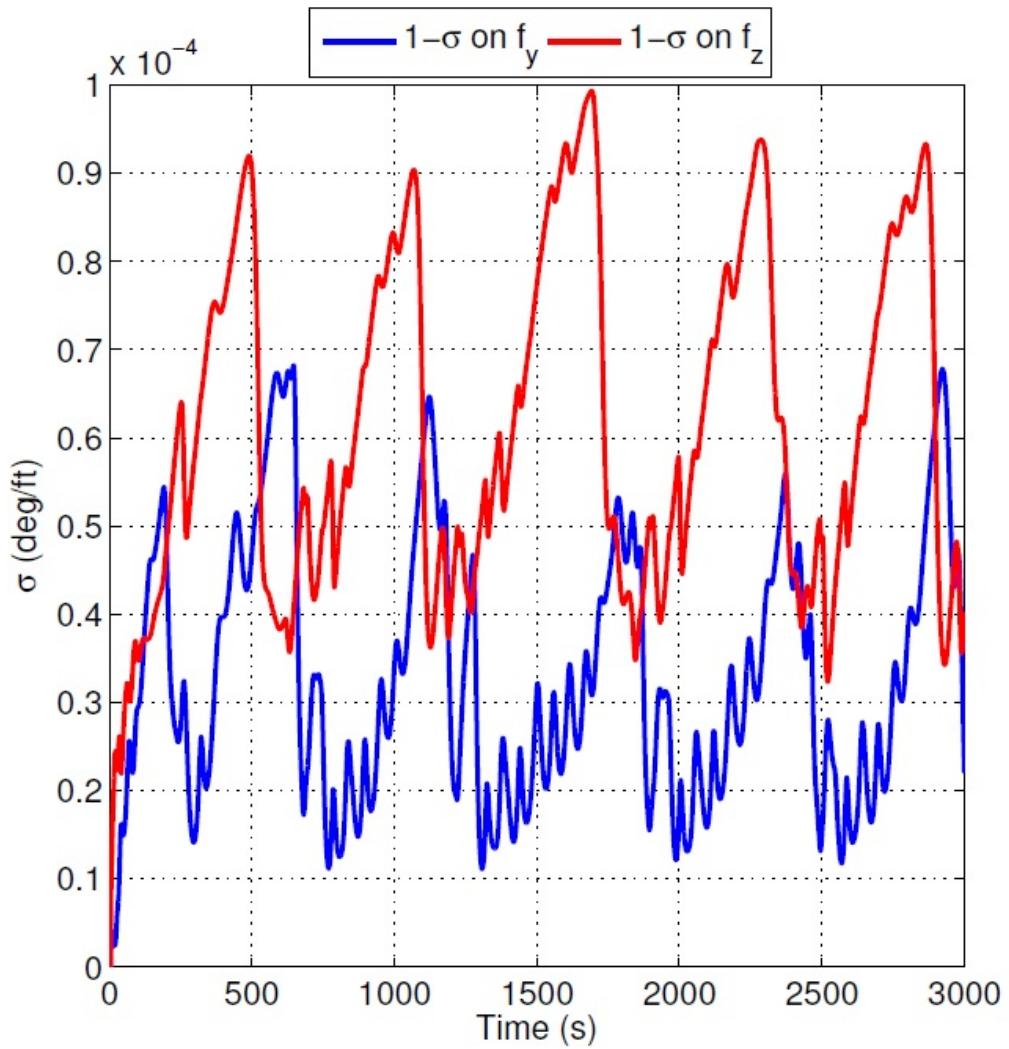


9.3: Time history of Aircraft cross-track, altitude, and pitch angle during formation flight. The commanded values are indicated by green dashed lines; they do not include the persistent excitation command. The aircraft states are indicated by solid blue lines.

Figure 9.4 displays the gradient estimates made by the Kalman filter during the simulation run. Figure 9.5 displays the standard deviation of the gradient estimates made by the Kalman filter during the simulation run. The standard deviation corresponds with the oscillations experienced by the altitude and cross-track. As the standard deviations get large, the estimates get worse and the aircraft moves away from the extremum. As the standard deviations get smaller, the estimates improve and the system moves toward the extremum.

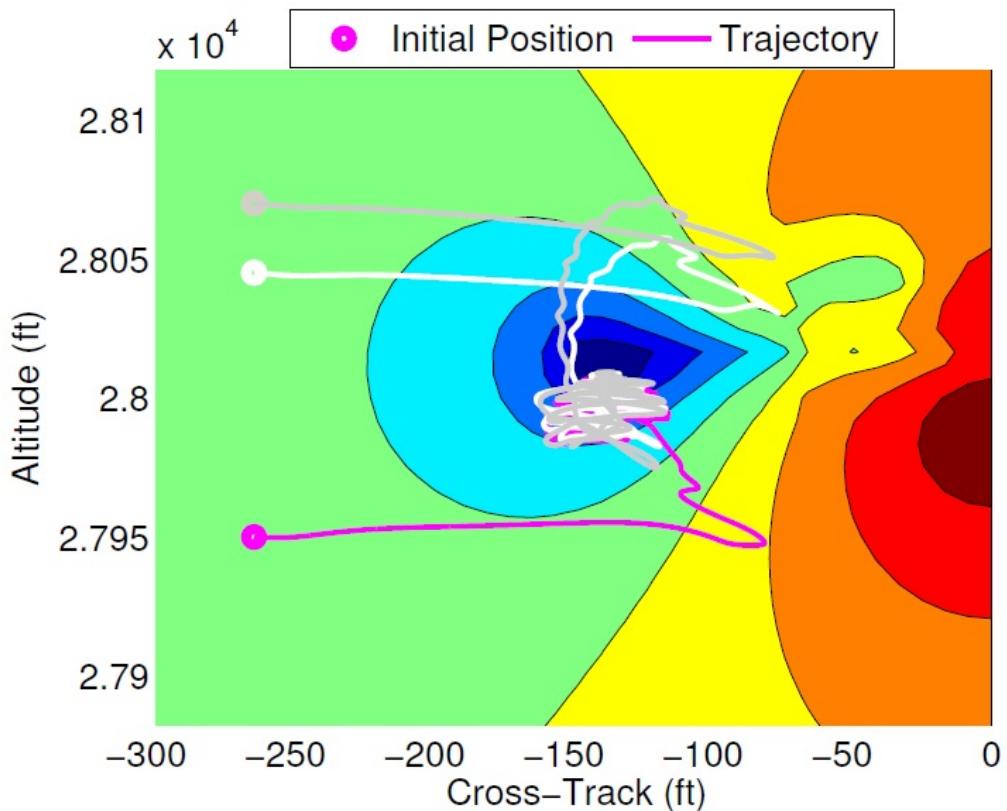


9.4: Time history of estimates made by the Kalman filter of the pitch-angle performance function gradient. The blue line indicates the gradient in the cross-track direction. The red-line indicates the gradient in the vertical direction.



9.5: Time history of standard deviation of the estimates made by the Kalman filter of the pitch-angle performance function gradient. The blue line indicates the gradient in the cross-track direction. The red-line indicates the gradient in the vertical direction.

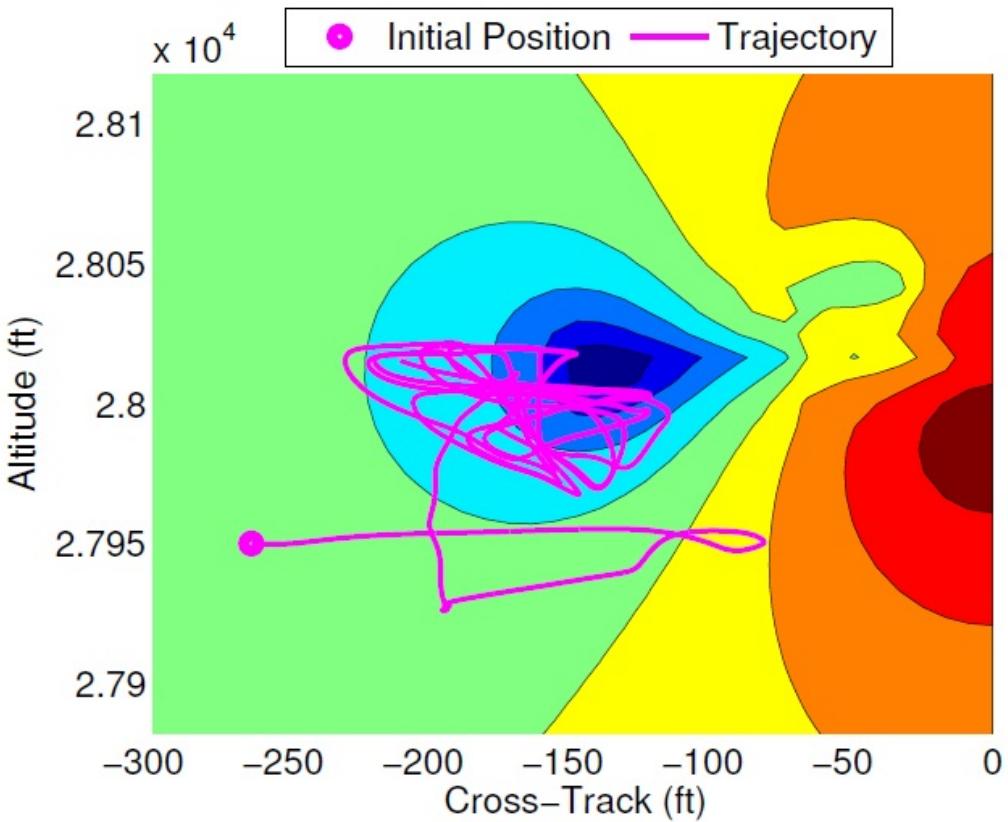
Figure 9.6 displays three separate simulation runs of the aircraft following the pitch angle performance function on top of the induced drag contour map. The system was initiated at a different location in each simulation run. The system performed similarly in each simulation-run. It quickly moved to the area of the extremum and then oscillated around in its vicinity. Each simulation-run converges to the same area a bit below the extremum of the contour map. This is because the extremum of the contour map is not coincident with the extremum of the pitch angle performance function.



9.6: Aircraft path following the pitch angle performance function on top of the induced drag contour map. The aircraft initial position is marked with a filled circle. The aircraft path is shown by a solid line. The contour of the pitch angle performance function is indicated by the multicolored contour map; dark blue indicates the smallest performance function magnitude, deep red indicates the highest performance function magnitude.

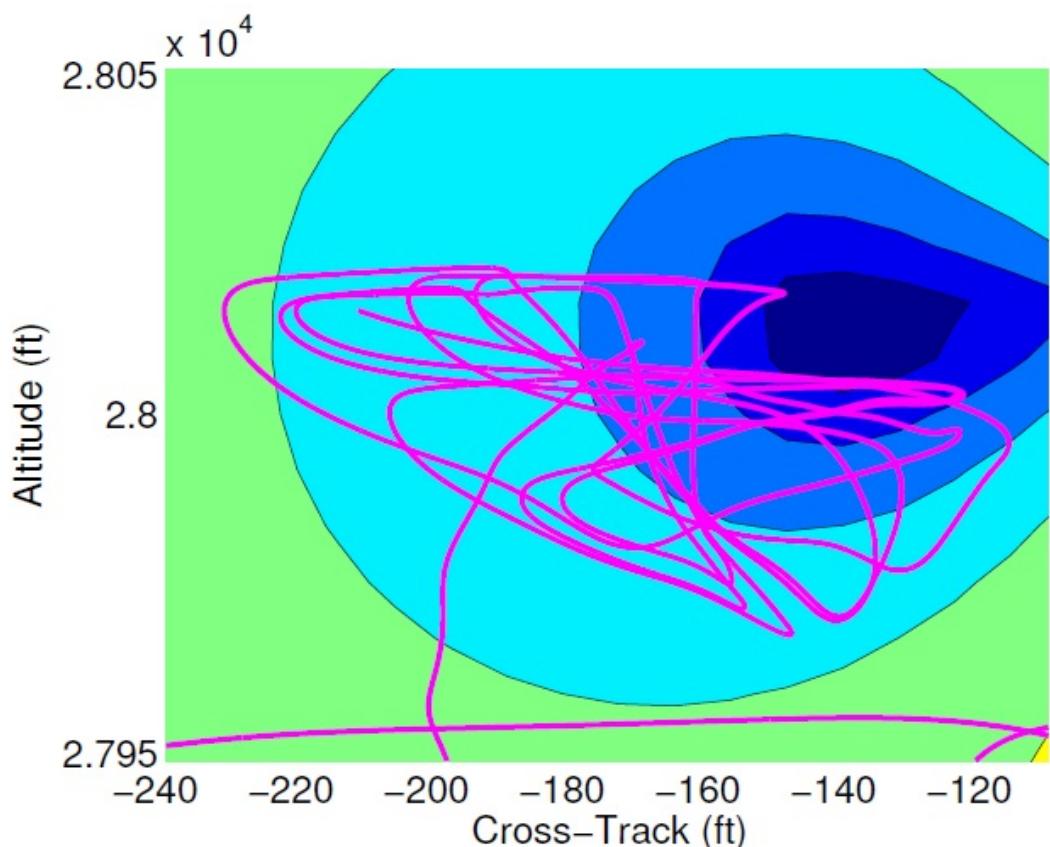
The extremum seeking control system design was also examined using aileron deflection for the performance function. The persistent excitation signal and Kalman filter used were identical to that used with the pitch angle performance function. The system did not perform as well

with aileron deflection performance function as it did with the pitch angle performance function. Figure 9.7 displays the aircraft path on the performance function contour. The aircraft initially takes a path very similar to the pitch-angle example but then moves away from the induced drag contour map extremum. After a short while it then moves in the direction of the induced drag contour map extremum. Once near the extremum it oscillates around in proximity of the extremum. The oscillations are much larger than those seen with the pitch angle performance function. Figure 9.8 displays a closer look of this oscillation.



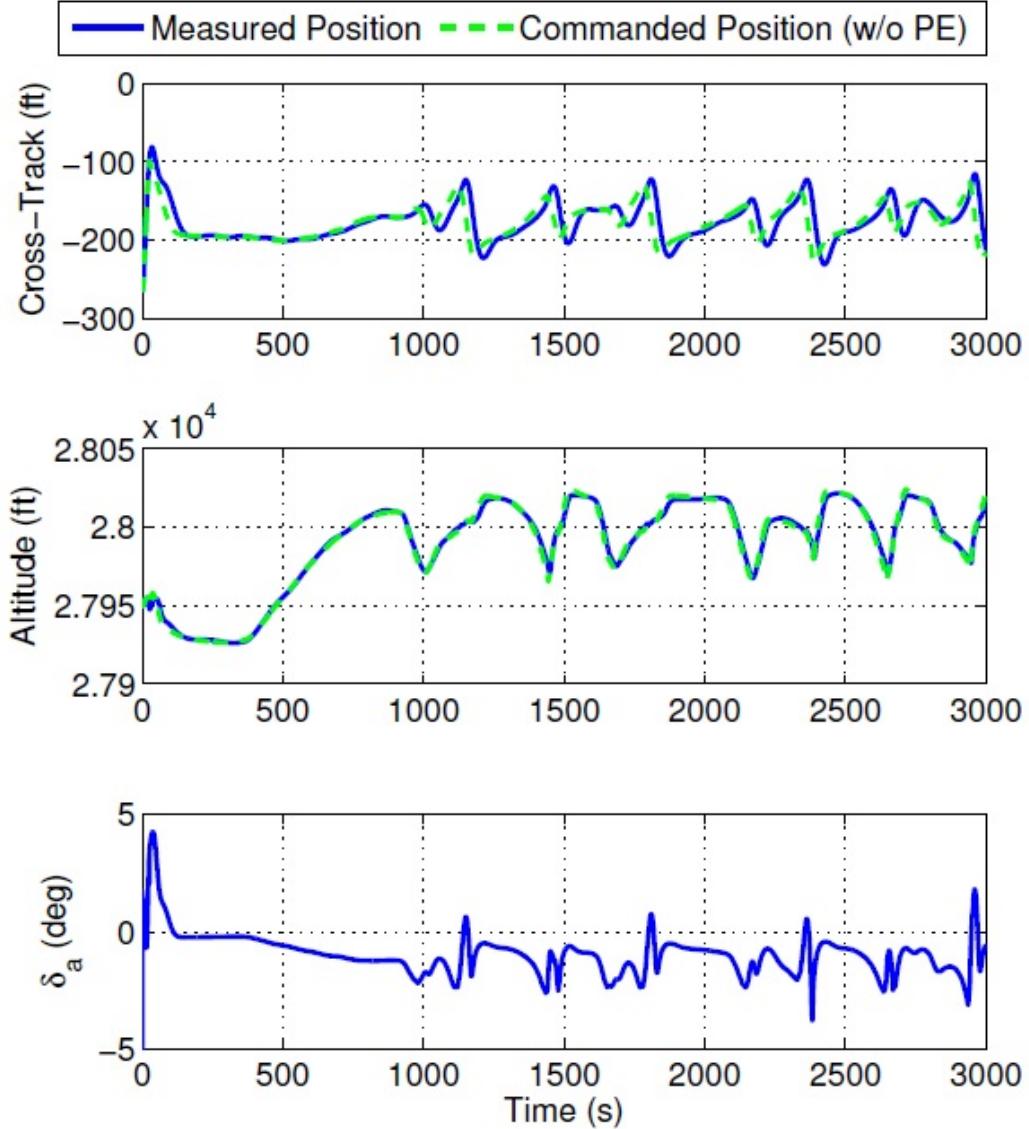
9.7: Aircraft path following the aileron deflection performance function displayed over the induced drag contour map. The aircraft initial position is marked with a filled circle. The aircraft path is shown by a solid line. The contour of the pitch angle performance function is indicated by the multi-colored contour map; dark blue indicates the smallest performance function magnitude, deep red indicates the highest performance function magnitude.

Figure 9.9 displays the cross-track, altitude, and aileron deflection of the aircraft. Initially the aileron deflection goes to 5 degrees as the cross track goes to -100 ft and the altitude remains



9.8: Aircraft path over the aileron deflection performance function near the extremum. The aircraft initial position is marked with a filled circle. The aircraft path is shown by a solid line. The contour of the pitch angle performance function is indicated by the multicolored contour map; dark blue indicates the smallest performance function magnitude, deep red indicates the highest performance function magnitude.

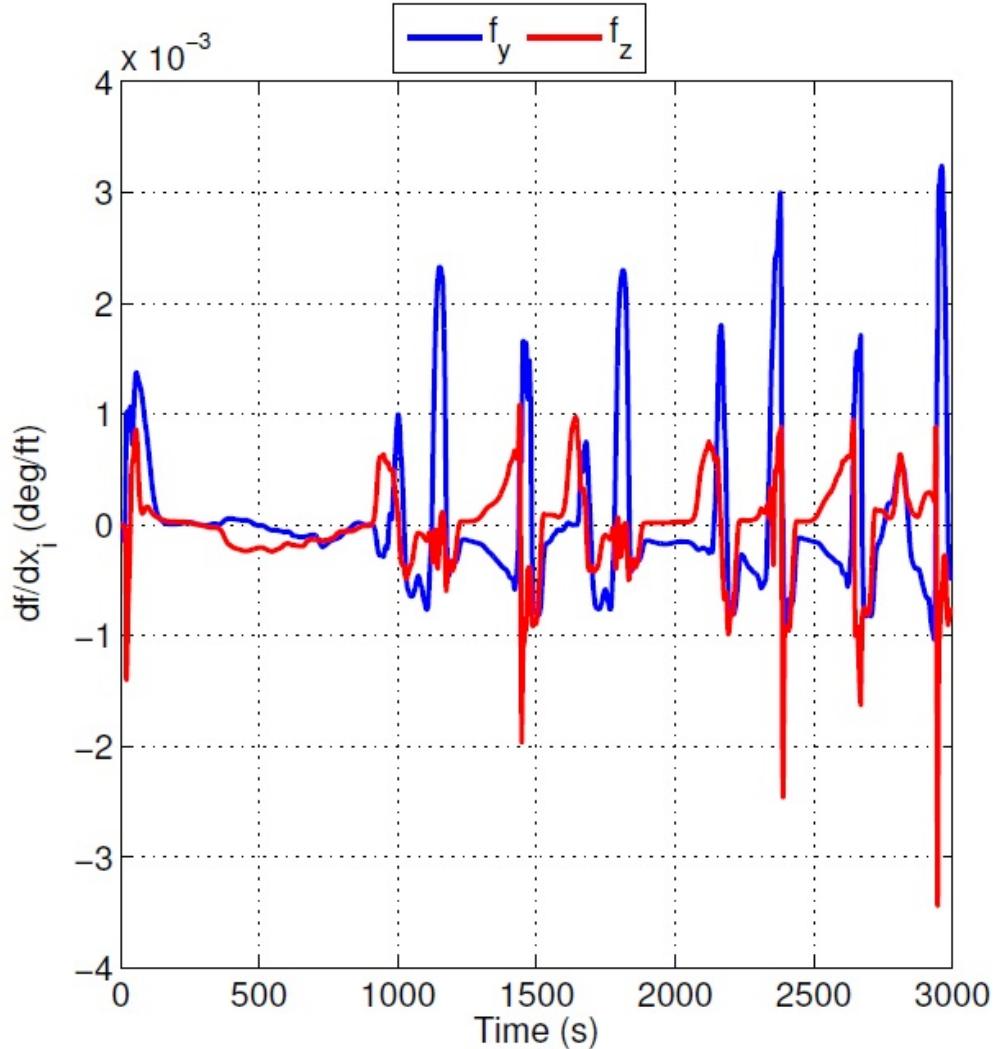
relatively constant. The aileron deflection then decreases to -3 degrees at 1000 seconds while the aircraft climbs approximately 604 feet in altitude. The aileron deflection then oscillates about -3 degrees as the aircraft oscillates in altitude and cross-track.



9.9: Time history of Aircraft cross-track, altitude, and pitch angle during formation flight. The commanded values are indicated by green dashed lines; they do not include the persistent excitation command. The aircraft states are indicated by solid blue lines.

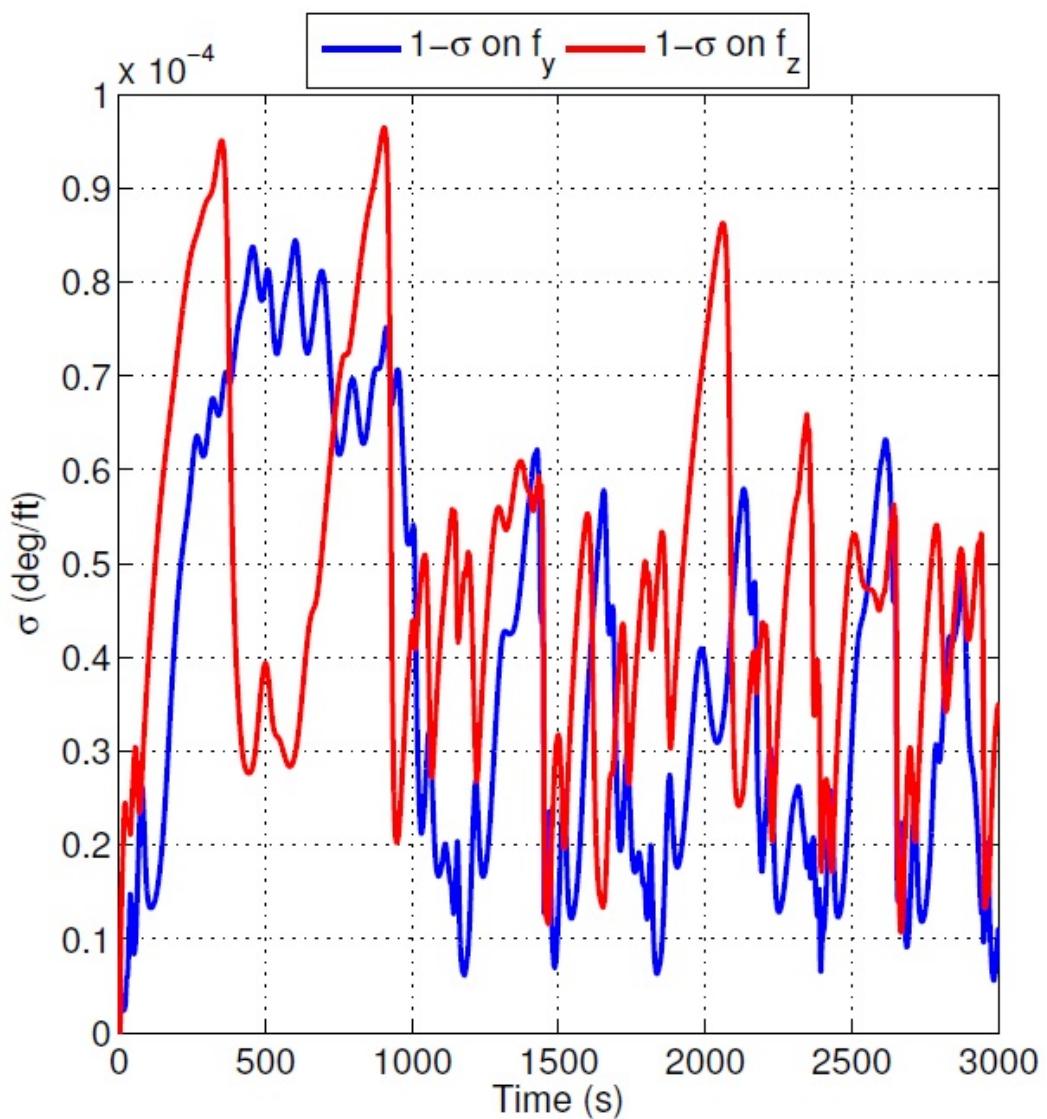
Figure 9.10 displays the gradient estimates of the system. The oscillations in the altitude and cross-track measurements correspond with the gradient estimates. This is as expected since the magnitude of the gradient estimates are directly proportional to the aircraft relative position

commands. Figure 9.11 displays the standard deviation of the gradient commands. They, on average, are smaller than the standard deviation seen in the pitch-angle performance function simulation run. This indicates a shallower performance function near the extremum then that of the pitch-angle performance function.



9.10: Time history of estimates made by the Kalman filter of the pitch-angle performance function gradient. The blue line indicates the gradient in the cross-track direction. The red-line indicates the gradient in the vertical direction.

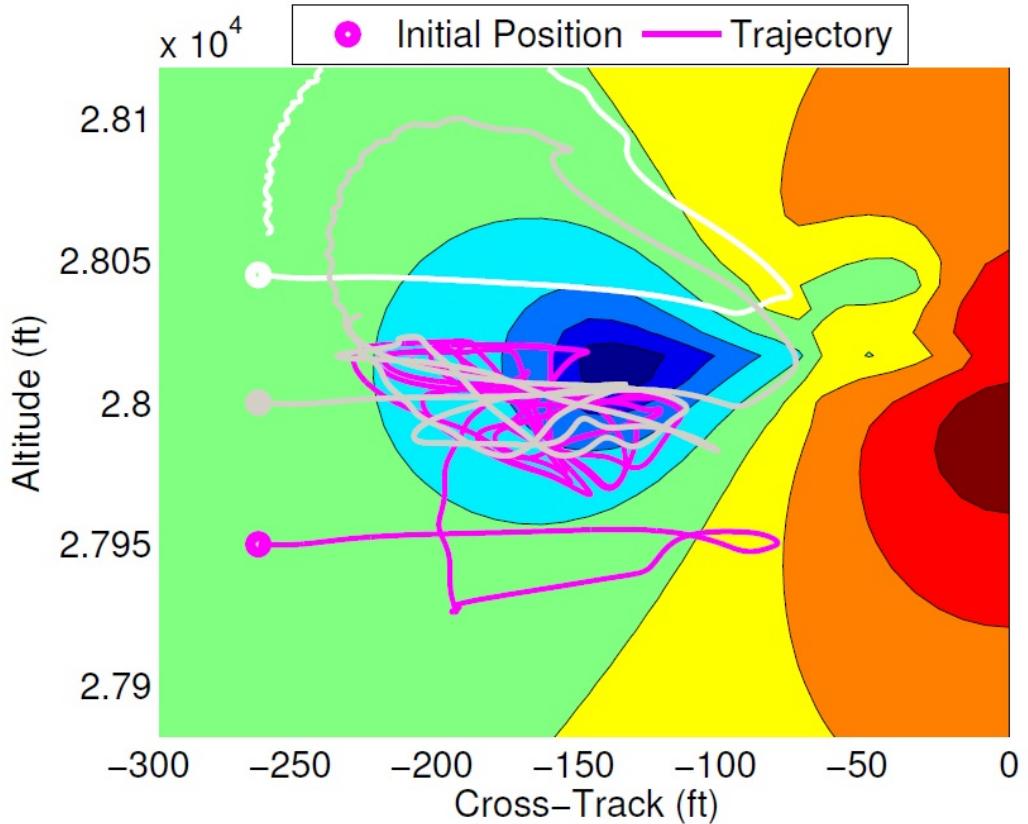
Figure 9.12 displays the aircraft position of three simulations runs on to of the induced drag contour map. Each run performed in a similar fashion. They all begin by moving in the cross track direction with little change in altitude. They then move the opposite direction with two of the three arching up and around the extremum. All three then head toward the extremum.



9.11: Time history of standard deviation of the estimates made by the Kalman filter of the pitch-angle performance function gradient. The blue line indicates the gradient in the cross-track direction. The red-line indicates the gradient in the vertical direction.

One of the simulation runs did not reach the extremum in the time allocated for the simulation run but is clearly heading in that direction. The other two reach and then oscillate around the extremum.

As with the pitch angle performance function's extremum, the aileron deflection performance function's extremum does not coincide with the induced drag extremum. It appears to be approximately 30 feet in cross track and 20 feet in altitude away from the induced drag extremum.



9.12: Aircraft path over the pitch angle performance function. The aircraft initial position is marked with a filled circle. The aircraft path is shown by a solid line. The contour of the pitch angle performance function is indicated by the multicolored contour map; dark blue indicates the smallest performance function magnitude, deep red indicates the highest performance function magnitude.

9.3 Comparison between the UCLA simulation results and the Boeing simulation results

The implementation of the extremum seeking method in the UCLA simulation performed better than the implementation in the Boeing simulation. This is partially due to the persistent excitation signal used. The UCLA implementation used a larger magnitude sinusoidal signal which was always orthogonal to the direction of the aircraft position command. The Boeing implementation used two smaller magnitude sinusoidal signals acting in the direction of the independent axis. In addition, while the UCLA persistent excitation and measurement rate could be selected to ensure that the measurements used in each iteration were distributed well, the nature of the Boeing simulation prohibited a similar construction. Instead, the measurement rate of the Boeing implementation was subject to the rate of the overall simulation.

CHAPTER 10

Summary and Conclusions

This dissertation has presented a new method of extremum seeking control. The method utilizes a time varying Kalman filter to estimate the gradient and Hessian of a performance function. The estimates are then used to command the plant toward the extremum of the performance function.

The extremum seeking method was mathematically developed and a proof of the method's convergence in the presence of noise was provided. Three examples of implementation were then presented. The first example used a performance function with one independent parameter and one dependent parameter. The last two examples both dealt with the problem of formation flight for drag reduction. They each used performance functions with two independent parameters and one dependent parameter. In each example the extremum of the performance function was found.

The problem of formation flight for drag reduction was discussed in some detail before the last two examples were presented. The influences of a wing-tip vortex from a leading aircraft on a trailing aircraft were mathematically derived and discussed. A survey of the theoretical and experimental work in this research was presented.

There is still much work to be completed in the further development of the extremum seeking control method and in the problem of formation flight. In the area of extremum seeking control method development, a more general proof of convergence should be developed to include Hessian estimates. The method could also be applied to a plethora of extremum seeking control problems found in various publications such that a direct comparison against other methods could be made.

In the area of formation flight for drag reduction, much more work needs to be conducted. There has been little study on the optimal performance function to be used nor on the problem of large formations of aircraft attempting to achieve drag reduction. This problem will, most likely, need to employ a game theoretic framework to optimize the amount of drag achieved by the entire system. There is also a lack of flight tests conducted with large aircraft and with more than two aircraft. Interest in the area appears to be growing. Over the last few years, the number

of published work has grown.

APPENDIX A

Derivation of the Linear Quadratic Regulator

The Linear Quadratic Regulator (LQR) is an optimal regulator under a quadratic performance index. Given a linear model of a system the methodology can be used to quickly develop a prototype control system. The methodology is particularly useful with the full-information problem, that is when all states of the system are available for feedback through the controller.

When all the system states are not available for feed-back, i.e. in the partial-information problem, a Linear Quadratic Gaussian (LQG) control design may be used. The LQG control design consists of an LQR controller and a Kalman filter used to estimate the unavailable states. The design is typically performed by computing the LQR gains assuming a full-information problem and a Kalman filter separately and then combining the components in the feedback loop.

The LQR can be derived with a variety of methods such as with conventional minimization using Lagrange multipliers [Oga95], with the maximum principle [BH75], or with the use of a projection operator as was used in section 3.3 in the derivation of the Kalman filter. In this section, a dynamic programming approach is used.

A.1 A Dynamic Programming Derivation of the LQR

Consider the discrete linear system

$$x_{k+1} = Ax_k + Bu_k \quad (\text{A.1.1a})$$

$$y_{k+1} = x_k \quad (\text{A.1.1b})$$

with $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$, and the initial state $x(0)$ as given. The linear quadratic regulator minimizes the cost function

$$J = \sum_{k=0}^n x_{k+1}^T Q x_{k+1} + u_k^T R u_k \quad (\text{A.1.2})$$

subject to the constrain given by equation (A.1.1). In equation (A.1.2) $Q \in \mathbb{R}^{n \times n}$ and $R \in \mathbb{R}^{m \times m}$ are weighting matrices which are used to designate the relative importance of each state and input. The LQR problem is to determine the input u that minimizes J . Full state information is assumed available for feedback.

To improve the readability of the subsequent derivation the following notation is introduced.

Let

$$\|x\|_M^2 = x^T M x.$$

Equation (A.1.2) above is then written as

$$J = \sum_{k=0}^n \|x_{k+1}\|_Q^2 + \|u_k\|_R^2$$

To solve this optimization problem we first consider the last stage of the control sequence.

A.1.1 Last Stage

In the last stage of the control sequence the cost function is given by

$$J_n = \|x_{n+1}\|_Q^2 + \|u_n\|_R^2.$$

The constraint given by equation (A.1.1) is used to replace x_{n+1} . This results in

$$J_n = \|Ax_n + Bu_n\|_Q^2 + \|u_n\|_R^2. \quad (\text{A.1.1})$$

The first term on the left hand side is a quadratic and can be expanded as follows

$$\begin{aligned} \|Ax_n + Bu_n\|_Q^2 &= [Ax_n + Bu_n]^T Q [Ax_n + Bu_n] \\ &= [x_n^T A^T + u_n^T B^T] Q [Ax_n + Bu_n] \\ &= x_n^T A^T Q Ax_n + x_n^T A^T Q Bu_n + u_n^T B^T Q Ax_n \\ &\quad + u_n^T B^T Q Bu_n \\ &= x_n^T A^T Q Ax_n + 2x_n^T B^T Q Au_n + u_n^T B^T Q Bu_n \\ &= \|x_n\|_{A^T Q A}^2 + 2x_n^T B^T Q Au_n + \|u_n\|_{B^T Q B}^2. \end{aligned}$$

The cost function in equation (A.1.1) is then written as

$$J_n = \|x_n\|_{A^T Q A}^2 + 2x_n^T B^T Q Au_n + \|u_n\|_{B^T Q B}^2 + \|u_n\|_R^2 \quad (\text{A.1.2a})$$

$$= \|u_n\|_{B^T Q B + R}^2 + 2x_n^T B^T Q Au_n + \|x_n\|_{A^T Q A}^2 \quad (\text{A.1.2b})$$

This is now a quadratic matrix equation in u_n . Its general solution is given in the following section, after which we will return to the optimization problem at hand.

A.1.2 Factorization of a Quadratic Matrix Equation

A quadratic matrix equation is given by

$$f(u) = \|u\|_W^2 + 2V^T u + Z \quad (\text{A.1.1})$$

where $W = W^T$. By adding and subtracting $V^T W^{-1} V$ to equation (A.1.1), it can be factored as

$$\begin{aligned} f(u) &= [u + W^{-1} V]^T W [u + W^{-1} V] + Z - V^T W^{-1} V \\ &= \|u + W^{-1} V\|_W^2 + Z - \|V\|_{W^{-1}}^2. \end{aligned}$$

It is now apparent that the minimizing u is given by

$$u^* = -W^{-1} V \quad (\text{A.1.3})$$

The function therefore has a minimum value

$$f(u^*) = Z - \|V\|_{W^{-1}}^2$$

A.1.3 Back to the Last Stage

Using the above result, Equation (A.1.3) can be solved. Comparing Equation (A.1.3) to Equation A.1.1 we recognize the corresponding terms and rewrite Equation (A.1.3) as

$$J_n = \underbrace{\|x_n\|_{A^T Q A}^2}_{Z} + 2 \underbrace{x_n^T B^T Q A}_{V} u_n + \underbrace{\|u_n\|_{B^T Q B + R}^2}_{W} \quad (\text{A.1.1})$$

Equation (A.1.3) is then minimized by

$$u_n^* = - \underbrace{[B^T Q B + R]^{-1}}_{W^{-1}} \underbrace{B^T Q A x_n}_{V}. \quad (\text{A.1.2})$$

The underbracing in equation (A.1.2) is being used to identify the corresponding parts of equation (A.1.3). The optimal cost for the last stage is then

$$\begin{aligned} J_n^* &= \underbrace{\|x_n\|_{A^T Q A}^2}_{Z} - \underbrace{\|x_n\|_{A^T Q B [B^T Q B + R]^{-1} B^T Q A}^2}_{\|V\|_{W^{-1}}^2} \\ &= \|x_n\|_{A^T Q A - A^T Q B [B^T Q B + R]^{-1} B^T Q A}^2. \end{aligned}$$

A.1.4 Second to Last Stage

Now consider the second to last stage. The cost function is given by

$$\begin{aligned} J_{n-1} &= J_n + \|x_n\|_Q^2 + \|u_{n-1}\|_R^2 \\ &= \|x_n\|_{A^T Q A - A^T Q B [B^T Q B + R]^{-1} B^T Q A}^2 + \|x_n\|_Q^2 + \|u_{n-1}\|_R^2. \end{aligned} \quad (\text{A.1.1a})$$

To further improve the readability the following addition notation is defined. Let

$$P_{n-1} = A^T Q A - A^T Q B [B^T Q B + R]^{-1} B^T Q A + Q. \quad (\text{A.1.2})$$

The cost function in equation (A.1.1a) is then written as

$$J_{n-1} = \|x_n\|_{P_{n-1}}^2 + \|u_{n-1}\|_R^2$$

In this form it is easy to see that this is precisely the problem that was solved in the last stage!

The solution is

$$u_{n-1}^* = -[B^T P_{n-1} B + R]^{-1} B^T P_{n-1} A x_{n-1}.$$

The corresponding cost is

$$J_{n-1}^* = \|x_{n-1}\|_{A^T P_{n-1} A - A^T P_{n-1} B [B^T P_{n-1} B + R]^{-1} B^T P_{n-1} A}^2.$$

In the same vein as equation (A.1.2) we define

$$P_{n-2} = A^T P_{n-1} A - A^T P_{n-1} B [B^T P_{n-1} B + R]^{-1} B^T P_{n-1} A + P_{n-1}$$

so that the cost is given as

$$J_{n-1}^* = \|x_{n-1}\|_{P_{n-2} - P_{n-1}}^2.$$

A.1.5 General Stage

At this point it is clear that the general stage of the LQR controller can be written. The control at each k stage is given by

$$u_k^* = -[B^T P_k B + R]^{-1} B^T P_k A x_k$$

where P_k is given by

$$P_{k-1} = A^T P_k A - A^T P_k B [B^T P_k B + R]^{-1} B^T P_k A + P_k.$$

This is the well known Riccati equation. The corresponding cost at the k^{th} stage is

$$J_k^* = \|x_{k-1}\|_{P_{k-1} - P_k}^2.$$

A.2 Time Invariant Linear Quadratic Regulator

The time invariant LQR problem solution can be found by considering the case of $k \rightarrow \infty$. In this case $P_k = P_{k-1}$. The optimal control is given by

$$\begin{aligned} u_k^* &= -(B^T PB + R)^{-1} B^T PAx_k \\ &= -Kx_k \end{aligned}$$

where

$$K = (B^T PB + R)^{-1} B^T PA$$

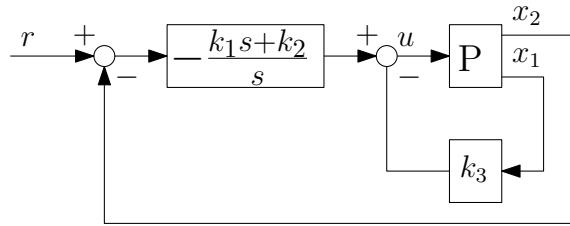
is a constant.

APPENDIX B

Derivation of the Linear Quadratic Tracker

From the internal model principle we know that to track a command with no steady-state error, we must include a model of that command in the closed loop system. This means that in order to track a step command with no steady-state error, there must be an integral in the controller, and in order to track a ramp command, there must be a double integral in the controller.

Let's consider the case of a step command. A reasonable controller to track a step command with no steady-state error may take the form as shown in figure B.1.



B.1: Step-Tracking Control Interconnection

Here r represents a reference command, u a control command into the plant, P which is given in state space form as

$$\dot{x} = Ax + Bu$$

$$y = x.$$

Control system gains are represented by k_1 , k_2 , and k_3 . In this system, the plant can be partitioned in a state space such that the plant output is carried as a state x_2 while all other states are grouped in x_1 .

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} u \quad (\text{B.0.2a})$$

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad (\text{B.0.2b})$$

The control-block on the input signal is written as

$$\dot{x}_3 = r - x_2 \quad (\text{B.0.3a})$$

$$y = k_2 x_3 + k_3(r - x_2) \quad (\text{B.0.3b})$$

The closed loop system is given by

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} A_{11} + B_1 k_1 & A_{12} - B_1 k_3 & B_1 k_2 \\ A_{21} + B_2 k_1 & A_{22} - B_2 k_3 & B_2 k_2 \\ 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} B_1 k_3 \\ B_2 k_3 \\ 1 \end{bmatrix} r \quad (\text{B.0.4a})$$

$$y = I \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \quad (\text{B.0.4b})$$

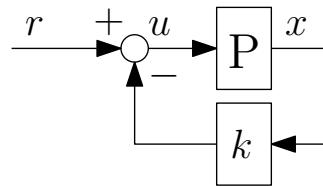
Expanding equation (B.0.4a) to

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} & 0 \\ A_{21} & A_{22} & 0 \\ 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} B_1 k_1 & -B_1 k_3 & B_1 k_2 \\ B_2 k_1 & -B_2 k_3 & B_2 k_2 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} B_1 k_3 \\ B_2 k_3 \\ 1 \end{bmatrix} r \quad (\text{B.0.5})$$

we recognize this as taking the form

$$\dot{x} = (A - Bk)x + Bkr \quad (\text{B.0.6})$$

which represents the standard regulator closed-loop of figure B.2. We therefore conclude that the



B.2: Regulator Closed Loop Interconnection

desired control system can be designed by forming the augmented plant model

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} & 0 \\ A_{21} & A_{22} & 0 \\ 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} u \quad (\text{B.0.7a})$$

$$y = [I] \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \quad (\text{B.0.7b})$$

and using any state-space control design approach to find the control gains. If a Linear-Quadratic-Regulator formulation is used then the design is called an LQR-tracker.

APPENDIX C

Extremum Seeking Control Design Tutorial

This appendix is intended to provide a step by step design guide for the extremum seeking method previously introduced. Each step is accompanied by a discussion of its purpose and a number of items to be considered during design. A discussion of the iterative nature of the tuning is also included.

The order of the design steps is in part dictated by the constraints of the system. For instance if the computer processing capability of the system is limited, the designer may need to adjust the persistent excitation frequency and the number of measurements taken to accommodate this constraint. Or if the closed loop system has a limited response time, the iteration rate may need to be adjusted to accommodate this constraint. In the following it is assumed that the plant dynamics are the driving constraint. The design steps are presented with this consideration.

C.1 Step 1: Simulation Development

As with most control system designs, the first step in the design process is to develop a simulation of the process to be controlled. The fidelity of this simulation is a matter of choice and need of the designer. If the simulation is of high fidelity, the designer can have high confidence that a control system designed with the simulation will be successful on the true physical process. If the simulation is of low fidelity, the designer will have less confidence that the control system design will be successful. In this case a larger level of robustness should be provided in the control system design.

For extremum-seeking control the simulation must model both the system plant and the performance function. In many cases the performance function may be modeled by a static function represented by an empirical equation or by a look-up-table. Uncertainties in the plant and performance function can be modeled with random noise superimposed upon simulation

parameters.

C.2 Step 2: Inner Loop Control Design

Once the simulation is built, a stabilizing inner loop control system must be designed around the system plant. Any control system design technique may be used. The extremum seeking control is designed independently of the stabilization method; it assumes that the inner loop will adequately track a step command. The response time and robustness of the inner loop is left to the discretion of the designer; however, the designer must recognize that the extremum seeking control system performance is limited by that of the inner loop system. For simple control systems, static proportional-integral control systems may be sufficient. For more elaborate systems, gain scheduled or nonlinear designs may be necessary.

The inner-loop design should be thoroughly evaluated via simulation to ensure adequate response characteristics. The designer must ensure actuators are not saturated and system states stay within their limits.

C.3 Step 3: Persistent Excitation Design

The third step in the design process is to choose a persistent excitation signal. This choice and those of the remaining steps are typically settled upon in an iterative fashion. With this in mind, the designer should not spend an inordinate amount of time on the initial choice.

The persistent excitation can take almost any form the designer chooses. It need not be a sinusoidal function although this is often a convenient choice. The only requirement is that it spans all dimensions of the performance function to ensure observability. For example, in a system whose performance function has two independent parameters, the persistent excitation signal must span the independent parameter space. If it spans only one dimension, then there will be no observance into the second dimension making gradient and Hessian estimation of that dimension impossible.

The form of the persistent excitation should be based upon the problem at hand. Its magnitude and frequency should be based upon the characteristics of the performance function and of the stabilized plant. For example, if the performance function exhibits much noise or many local

minimums, the magnitude of the persistent excitation should be chosen large enough to minimize their influence.

The frequency of the persistent excitation signal must be selected at a rate that is lower than or equivalent to the response rate achievable by the inner loop control system. If the inner loop control system can not track the command, then measurements can not be taken at the intended locations along the persistent excitation path.

Unlike other extremum-seeking methods, the method at hand *does not* require frequency separation of the persistent excitation in each axis. This is because the method does not use the frequency of the persistent excitation to differentiate between the axes. As discussed in Chapter 4, the Kalman filter fits a quadratic function to the performance function measurements thus estimating the gradient and Hessian.

This provides large latitude for the designer to construct the persistent excitation signal. For instance, in Chapter 8, the persistent excitation is designed such that it perturbs the system only in directions perpendicular to the direction of the current position command.

C.4 Step 4: Measurement Rate Selection

The fourth step in the design process is to choose the measurement rate of the system. It should be chosen with consideration of the response time of the closed loop plant and noise and slope characteristics of the performance function. In particular the signal-to-noise ratio of the change in performance function magnitude with respect to the change in performance function coordinates should be considered. For example, if for n measurements at a chosen measurement rate, the change in performance function magnitude is obscured by noise, then the measurement rate should be reduced to allow a larger change, between measurements, of the independent parameters such that a larger change in performance function magnitude results. If the signal to noise ratio is too small the Kalman filter will not be able to easily estimate the true gradient nor Hessian.

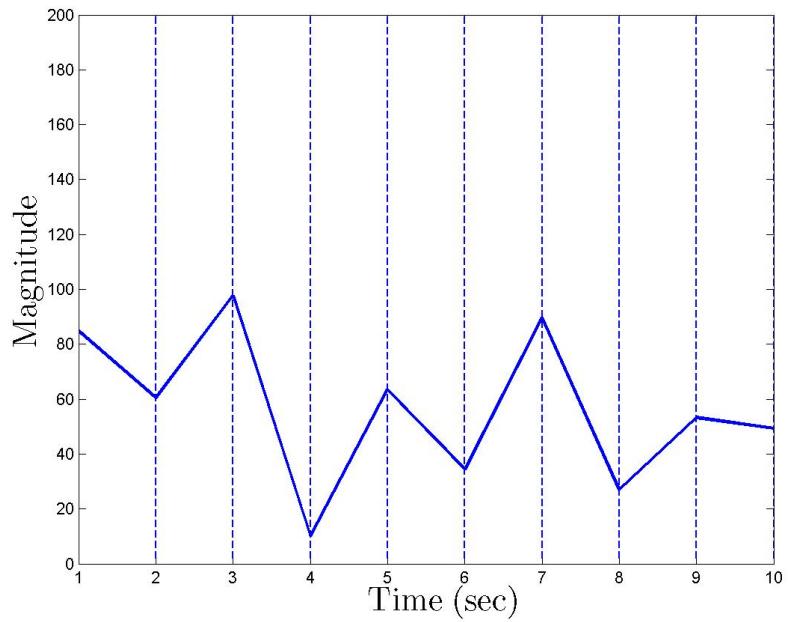
Figure C.1 illustrates the point. Here measurements of a noisy performance function are being taken. If figure C.1a the measurement rate is chosen to be 10 Hz. At this rate, after 10 iterations, it appears that the gradient is negative. In Figure C.1b, the measurement rate has been slowed to 1.0 Hz. At this rate, the gradient is revealed to be positive. The larger change in the independent

coordinates of the performance function has allowed a larger change in the performance function magnitude revealing the true gradient. The Kalman filter will now be able to better estimate the gradient and Hessian. This example is intentionally oversimplified for impact; it intentionally ignores the recursive character of the Kalman filter and only considers a constant gradient.

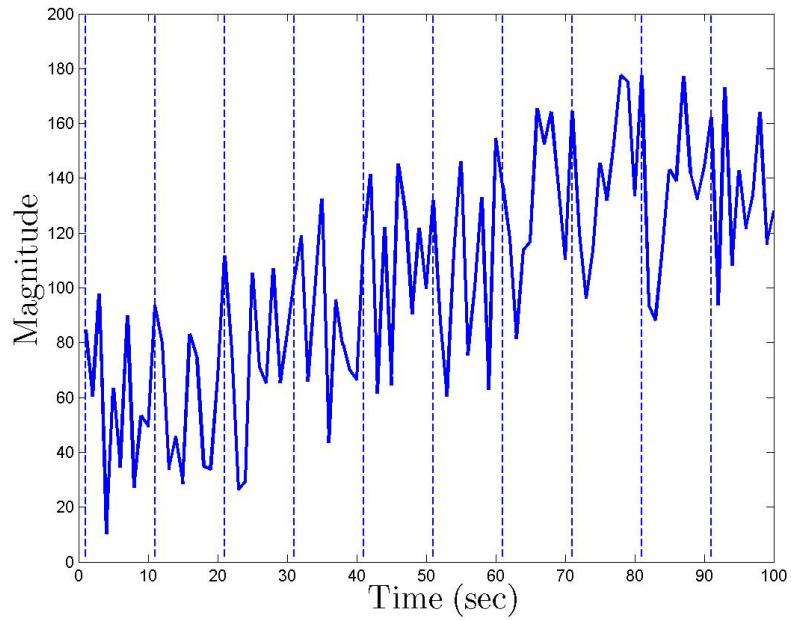
The choice of measurement rate must also take into account the frequency of the persistent excitation. For example, if the iteration rate is chosen to be equal to the frequency of a sinusoidal persistent excitation, then the measurements will always be taken at the same location on the wave form. Figure C.2 illustrates some poor choices of iteration and frequency combination. Figure C.3 illustrates some good choices of iteration and frequency combination. Figure C.2a illustrates a case where the persistent excitation is equivalent to the iteration rate. The net effect is that measurements do not span the independent parameter space of the performance function. Figure C.2b has slightly improved the situation by choosing the iteration rate to be slightly away from the persistent excitation frequency. Figure C.3a illustrates a better choice of measurement rate. The measurements span the independent parameter space of the performance function which allows estimation of the gradient and Hessian of the system. Figure C.3b illustrates another good choice of measurement rate. Measurements at intermediate spacing of the sinusoidal form have been added to those of Figure C.3a.

C.5 Step 5: Selection of Number of Measurement

As discussed in Chapter 4, the number of measurements is a tunable parameter. When selecting this number, the topology of the performance function and the required speed of convergence must be considered. A larger number of measurements causes the Kalman filer to estimate an average gradient and Hessian across the range of measurements. If the designer chooses to use only one measurement, then only the change in magnitude and change in location of the current and previous measurements will be used in the Kalman filter. This, ignoring the recursive nature of the Kalman filter, will then provide the gradient and Hessian largely based on only the last two measurements. If these measurements are noisy, the gradient and Hessian could be largely in error. If, on the other hand, the designer chooses to use 10 measurements, the Kalman filter will use the difference between the current measurement and the previous 10 to estimate the local gradient and Hessian. This has the affect of averaging the gradient and Hessian of the last 10

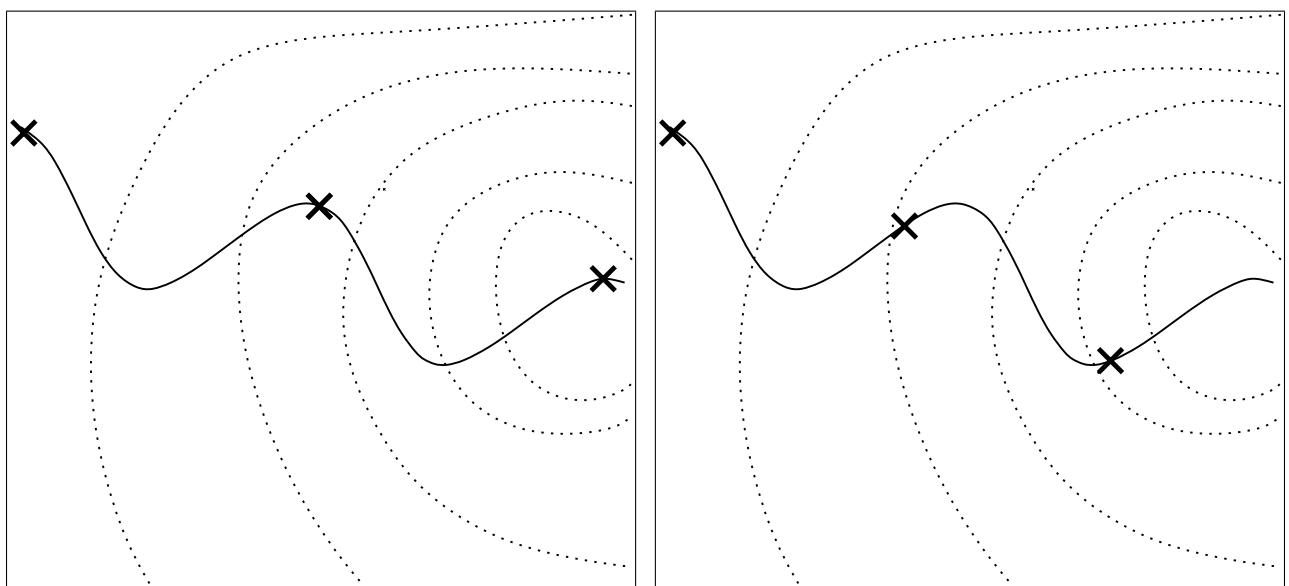


(a) 10 iterations at 0.1 Hz



(b) 10 iterations at 1.0 Hz

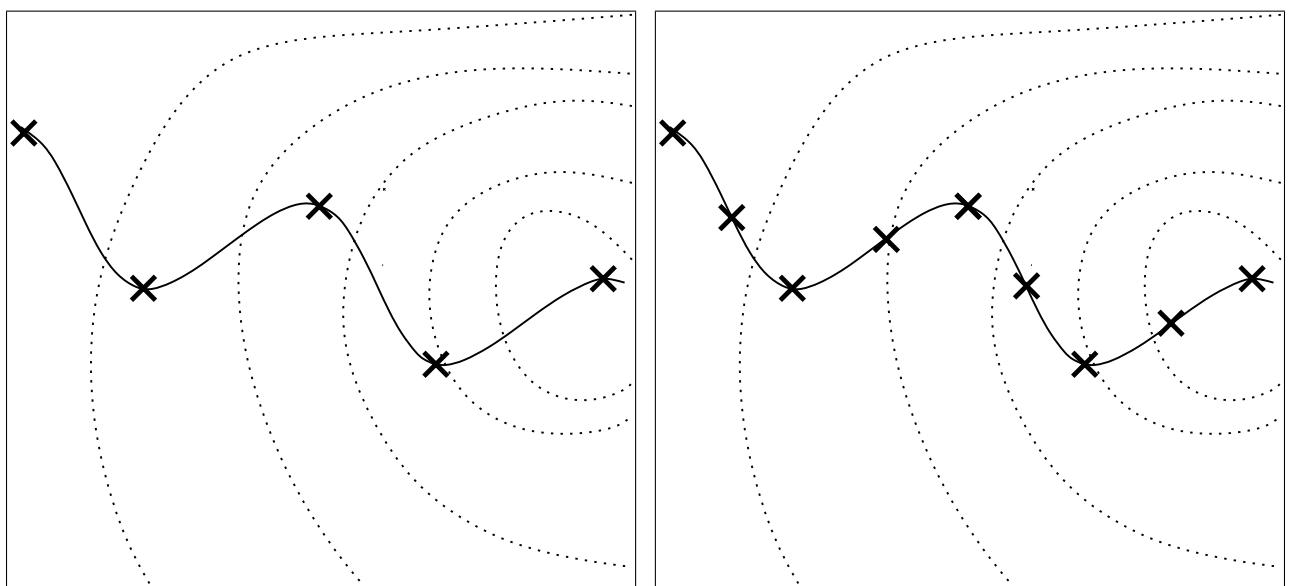
C.1: Illustration of various iteration rates for a noisy signal. The vertical lines represent the instants of measurement.



(a) Persistent excitation frequency equal to iteration rate (b) Persistent excitation frequency too similar to iteration rate

C.2: Illustration of poor persistent excitation and iteration rate combinations on a performance function with two independent dimensions.

The dotted lines denote contours of the performance function. The solid line denotes the persistent excitation path. The x's denote the locations at which measurements are taken.



(a) Persistent excitation frequency half the measurement rate (b) Persistent excitation frequency one-fourth the measurement rate

C.3: Persistent excitation frequency and iteration rate allow good observability on a performance function with two independent dimensions. The dotted lines denote contours of the performance function. The solid line denotes the persistent excitation path. The x denote the locations at which measurements are taken.

iterations. This, depending on the particular characteristics of the performance function, may slow the convergence of the system to the extremum. It will also lessen the effect of noise. Thus there is a trade-off. A larger number of measurements will make the system more tolerant to noise but will slow system convergence.

C.6 Step 6: Kalman Filter Construction and Tuning

The sixth step in the design process is the construction and tuning of the Kalman filter. The designer must first decide if the gradient of the performance function or both the gradient and Hessian will be used to guide the system toward the coordinates of the performance function extremum. Including the Hessian will allow the system to move more quickly to the optimum location by utilizing a Newton-Raphson approach instead of a steepest descent approach. Discussion of both techniques and further guidance can be found in [Lue73]. Use of the Hessian requires more parameters to be estimated. While the number of parameters per dimension varies as m for a steepest decent approach, it varies by $\frac{m^2+3m}{2}$ for the Newton-Raphson approach; this is shown in chapter 4. This obviously makes the estimation more difficult. This trade-off must be considered carefully by the designer. Hybrid approaches may also be used such as using a logical switch between steepest-descent and Newton-Raphson approaches based on a parameter such as the error-covariance of the Kalman filter.

Once the steepest descent or Newton-Raphson approach is selected, the designer must build the Kalman filter according to Section 4.4. The Kalman filter must then be tuned for the amount of noise present in the measurements and the expected rate of change of the gradient and Hessian of the performance function. A discussion of how to tune a Kalman filter can be found in [SC08].

C.7 Step 7: Command Conditioning Design

The seventh step is to design a command conditioning function. This function is used to provide a gain on the command to control the convergence rate and to condition the command before it is sent to the inner-loop-system. The designer may simply choose this function to be a constant gain to slow or accelerate the plant response. The designer may also choose a function which will smooth the command such as $0.1/(z - 1)$ in order to avoid over-shoot of the inner-loop-system

when responding to a step command. Such a function will provide a smoother plant response and avoid large actuator commands.

C.8 Step 8: Design System Initialization

Consideration must be given to system initialization. The Kalman filter must be initialized with an initial guess of the gradient and Hessian and an initial position command must be provided to the inner-loop control system.

The initial guess of gradient and Hessian may be determined from a priori knowledge of the performance function, or by using an initialization stage. Such an initialization stage occurs before the extremum seeking algorithm is started. It typically entails issuing a series of commands to the inner loop control system such that measurements of the performance function can be taken which span the independent parameter space of the performance function. These measurements are then used to generate the initial guesses of the gradient and Hessian. One such scheme is discussed in Chapter 8.

The initial position command may also be generated from a priori knowledge of the performance function, by pure luck, or by utilizing the initial guesses of the gradient and Hessian. Even if the initial guesses are vastly different than the true values, if the performance function is unimodal the extremum seeking control algorithm will eventually command the system in the correct direction.

C.9 Step 9: Simulate response

The final step of system design is to evaluate the system performance. Once the system is built, it should then be analyzed via simulation. The designer must examine the impact of the previous design choices on items such as rate of convergence, actuator usage, and inner-loop-system state response. Any design limits or requirements must be examined and verified to be met. If the design does not meet all requirements, then the design choices should be re-considered in an iterative fashion until the requirements are met.

C.10 Design Process Summary

The design process is summarized in the following steps

1. Develop a computer simulation of the plant and performance function.
2. Design the inner loop control system.
3. Design the persistent excitation signal.
4. Choose a Measurement rate.
5. Choose the number of measurements to be used in each iteration .
6. Build and tune the Kalman filter.
7. Choose the command conditioning function.
8. Design the method of system initialization.
9. Simulate the full system and analyze the response.
10. Repeat the steps to refine the design if necessary.

C.11 Special design considerations

While there are countless design variations and possibilities, a few are encountered frequently enough to warrant discussion. These are the cases of inherent persistent excitation, non-static performance functions, and slope-seeking problems. Each will be discussed in turn.

C.11.1 Inherent persistent excitation

In some problems persistent excitation is inherent in the design problem. This may take the form of noise as discussed in [Yin08] and [LK10]. It may also be inherent to the particular problem. For instance in the case of a unimodal static performance function with one independent and one dependent axis, an extremum-seeking control system can be designed such that no external persistent excitation signal is needed in order to find the extremum. The inherent persistent excitation is created as the inner-loop system moves across the performance function. As this

occurs, more measurements are taken, new gradient and Hessian estimates are made, and updated position commands continue to move the inner-loop system. Once the extremum is reached, the system stops moving and no further improvement can be made. There is no guarantee of convergence in this case; the system could stop at a position that is not the extremum and there remain. An example of this type of persistent excitation can be found in Chapter 6

This method of persistent excitation will not be effective in the case of a nonstatic performance functions because the system will reach a minimum and stop. It will not have the ability to continually gather measurements to estimate the current gradient nor Hessian. This limitation, however can be overcome by providing periodic perturbations to a stopped system to determine if the performance function minimum remains in the same location.

C.11.2 Non-static performance function

When the performance function is not static, the extremum-seeking control method at hand is capable of tracking the minimum with little modification. The performance function must vary at a rate that is slow relative to the measurement rate and inner-loop response rate. The largest adjustment to the extremum seeking algorithm that must be made is to the Kalman filter. It must be tuned to indicate the larger uncertainty in the process equation. In addition to this, measurement rate may need be adjusted. These adjustments should be made in an iterative fashion.

C.11.3 Slope seeking problems

The extremum seeking method can also be modified to address the slope-seeking problem. In this problem, the system seeks a given slope instead of an extremum. In this case, Equation 4.3.6 is modified to find the coordinate at which the gradient is G .

$$\frac{\partial \Delta f_k}{\partial \Delta x_k} = G$$

This leads to

$$\Delta x_k = (G - b_{x_k}^T) M_{x_k}^{-1}.$$

The coordinates of the desired slope are then found to be

$$x_{k+1} = x_k + (G - b_{x_k}^T) M_{x_k}^{-1}.$$

This equation is then used to replace Equation 4.4.7 of Chapter 4

REFERENCES

- [AGA91] AGARD, editor. *Conference Proceedings 494 Symposium on Vortex Flow Aerodynamics*, CP-494. AGARD, AGARD, July 1991.
- [AGA96] AGARD, editor. *Conference Proceedings 584 The Characterisation & Modification of Wakes from Lifting Vehicles in Fluids*, Trondheim, Norway, November 1996. AGARD, AGARD. Published in November. Papers presented in May 20-23.
- [AK02] KB Ariyur and M. Krstic. “Analysis and design of multivariable extremum seeking.” In *American Control Conference, 2002. Proceedings of the 2002*, volume 4, 2002.
- [AK03] KB Ariyur and M. Krstic. *Real-Time Optimization by Extremum-Seeking Control*. Wiley-Interscience, 2003. Hoboken Editor.
- [AK04] Kartik B. Ariyur and Miroslav Krstic. “Slope seeking: a generalization of extremum seeking.” *International Journal of Adaptive Control and Signal Processing*, **18**:1–22, 2004.
- [BAK03] Paolo Binetti, Kartik B. Ariyur, Miroslav Krstic, and Franco Bernelli. “Formation Flight Optimization Using Extremum Seeking Feedback.” *Journal Of Guidance, Control, And Dynamics*, **26**:132–142, 2003.
- [BAK04] A. Banaszuk, K.B. Ariyur, M. Krstić, and C.A. Jacobson. “An adaptive algorithm for control of combustion instability.” *Automatica*, **40**(11):1965–1972, 2004.
- [BFK09] Geoffrey C. Bower, Tristan C. Flanzer, and Ilan M. Kroo. “Formation Geometries and Route Optimization for Commercial Formation Flight.” In *27th AIAA Applied Aerodynamics Conference*, San Antonio, Texas, June 2009.
- [BH75] Arthur Bryson and Yu-Chi Ho. *Applied Optimal Control: Optimization, Estimation, and Control*. Hemisphere Publishing Corporation, 1975.
- [BH82] D. C. Burnham and J. N. Hallock. “Volume IV. Wake Vortex Decay.” In *Chicago Monostatic Acoustic Vortex Sensing System*. Transportation Systems Center Cambridge MA, 1982.
- [BH90] Marku Beukenberg and Dietrich Hummel. “Aerodynamics, Performance and Control of Airplanes in Formation Flight.” In *Proceedings of the 17th Congress of the International Council of the Aeronautical Sciences*. ICAS-90-5.9.3, September 9-14 1990.
- [BH97] R.G. Brown and P.Y.C. Hwang. *Introduction to Random Signals and Applied Kalman Filtering: With MATLAB Exercises and Solutions*. Wiley, 1997.
- [BI72] S. Bernstein and J.D Iversen. “Dynamic Simulation of an Aircraft Under the Effect of Vortex Wake Turbulence.” In *Proc. AIAA Atmospheric Flight Mechanics Conference*, 1972.
- [BK07] Ralf Becker and Rudibert King. “Adaptive Closed-Loop Separation Control on a High-Lift Configuration Using Extremum Seeking.” *AIAA Journal*, **45**(6):1382–1392, 2007.

- [Bla62] P. F. Blackman. “Extremum-Seeking Regulators.” In J. H. Westcott, editor, *An Exposition of Adaptive Control*, p. 3650. The Macmillan Company, New York, NY, 1962.
- [Bla00] W. B. Blake. “An Aerodynamic Model For Simulation Of Close Formation Flight.” In *Proceedings of AIAA Modeling and Simulation Technologies Conference and Exhibit*, number 4304 in AIAA-2000, Denver, CO, August 2000. American Institute of Aeronautics & Astronautics.
- [Bra06] Gtz Bramesfeld. *A Higher Order Vortex-Lattice Method with a Force-Free Wake*. PhD thesis, The Pennsylvania State University, August 2006.
- [CKA02] Joon-Young Choi, Miroslav Krstic, Kartik B. Ariyur, and Jin S. Lee. “Extremum Seeking Control for Discrete-Time Systems.” *IEEE Transactions on Automatic Control*, **47(2)**:318–323, February 2002.
- [CS94] C. J. Cutts and J. R. Speakman. “Energy Savings In Formation Flight Of Pink-Footed Geese.” *J. exp. Biol.*, **189**:251261, 1994.
- [CSF06] D.F. Chichka, J.L. Speyer, C. Fanti, and C.G. Park. “Peak-Seeking Control for Drag Reduction in Formation Flight.” *J. Guid. Control Dyn.*, **29(5)**:1221–1230, September–October 2006.
- [DL51] C. S. Draper and Y. J. Li. “Principles of optimalizing control systems and an application to an internal combustion engine.” *ASME Publications*, pp. 160–168, September 1951.
- [EL05] Bo Egardt and Stefan Larsson. “On a Parameter Adaptive Extremum Controller.” In *Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference*, pp. 4809–4814, Seville, Spain, December 2005.
- [Eug01] Jr. Eugene H. Wagner. “*An Analytical Study Of T-38 Drag Reduction In Tight Formation Flight*.” Master’s thesis, Air Force Institute Of Technology, March 2001.
- [FDA66] A. L. Frey, W. B. Deem, and R. J. Altpeter. “Stability and optimal gain in extremum seeking adaptive control of a gas furnace.” In *Proceedings of the Third IFAC World Congress*, London, 1966.
- [Fil96] A Filippone. “Heuristic optimization applied to an intrinsically difficult problem: birds formation flight.” In *AIAA*, number 0515 in AIAA-96, Reno, NV, 1996.
- [GA01] M.S. Grewal and A.P. Andrews. *Kalman Filtering: Theory and Practice Using MATLAB*. Wiley, 2001.
- [GDP03] M. Guay, D. Dochain, and M. Perrier. “Adaptive Extremum Seeking Control of Nonisothermal Continuous Stirred Tank Reactors.” In *Proceedings of 42nd IEEE Conference on Decision and Control*, volume 5, pp. 4753 – 4758, December 2003.
- [GSC06] Yu Gu, Brad Seanor, Giampiero Campa, Marcello R Napolitano, Larry Rowe, Srikanth Gururajan, and Sheng Wan. “Design and Flight Testing Evaluation of Formation Control Laws.” *IEEE Transactions On Control Systems Technology*, **14(6)**:1105 –1112, November 2006.

- [Hal91] J. N Hallock. *Aircraft Wake Vortices: An Annotated Bibliography (1923 - 1990)*. U.S. Dept. of Transportation, 1991. at DTIC.
- [Han03] Curtis E. Hanson. “A Fuzzy Technique for Performing Lateral- Axis Formation Flight Navigation Using Wingtip Vortices.” Technical Report NASA/TM-2003-212033, NASA Dryden Flight Research Center, Edwards, California, 2003.
- [HC02] Jennifer L. Hansen and Brent R. Cobleigh. “Induced Moment Effects Of Formation Flight Using Two F/A-18 Aircraft.” In *the AIAA Guidance, Navigation and Control Conference*, pp. 2002–4489, 2002.
- [HK82] B.C. Jr. Hastings and Jr Keyser, G.L. “Simulator Studies of Vortex Encounters by a Twin- Engine Commercial Jet Transport Airplane.” Technical Paper 1966, NASA, 1982.
- [Hum83] D. Hummel. “Aerodynamic aspects of formation flight in birds.” *J. Theor. Biol.*, **104**:321347, 1983.
- [Hum96] Ing. D. Hummel. “The Use of Aircraft Wakes to Achieve Power Reductions in Formation Flight.” In *Proceedings of the 1996 AGARD FDP Symposium on The Characterisation & Modification of Wakes from Lifting Vehicles in Fluid*, pp. 36–1–36–13. AGARD CP-584, May 20-23 1996.
- [IM02] S. Iglesias and W.H. Mason. “Optimum Spanloads in Formation Flight.” In *40th AIAA Aerospace Sciences Meeting & Exhibit*, Reno, NV, January 2002. AIAA.
- [Jaz70] Andrew Jazwinski. *Stochastic Processes and Filtering Theory*. Academic Press, Inc, New York, 1970.
- [JCR95] L.R. Jenkinson, R.E. Caves, and D.P. Rhodes. “Automatic Formation Flight “A preliminary investigation into the application to civil operations”.” In *Proceedings of 1st AIAA Aircraft Engineering, Technology, and Operations Congress*, Los Angeles, CA, September 1995.
- [JS68] O. L. R. Jacobs and G. C. Shering. “Design of a single-input sinusoidal-perturbation extremum-control system.” *Proceedings IEE*, **115**:212–217, 1968.
- [Kal60] R. E. Kalman. “A New Approach to Linear Filtering and Prediction Problems.” *Transactions of the ASME Journal of Basic Engineering*, **82 (Series D)**:35–45, March 1960.
- [Kaz43] V. V. Kazakevich. “Technique of automatic control of different processes to maximum or to minimum.”, 1943.
- [Kaz44] V. V. Kazakevich. *On extremum seeking*. PhD thesis, Moscow High Technical University, 1944.
- [KD04] Maziar E. Khatir and Edward J. Davison. “Bounded Stability and Eventual String Stability of a Large Platoon of Vehicles Using Non-Identical Controllers.” In *Proceedings of IEEE Conference on Decision and Control*, Atlantis, Paradise Island, Bahamas, December 2004.

- [KW97] Miroslav Krstic and Hsin-Hsiung Wang. “Extremum Seeking Feedback: Stability Proof and Application to an Aeroengine Compressor Model.” Published in *Automatica* as Stability of extremum seeking feedback for general nonlinear dynamic systems, 1997.
- [KW00] M. Krstic and H. H. Wang. “Stability of extremum seeking feedback for general dynamic systems.” *Automatica*, **36**:595–601, 2000.
- [Lar01] S. Larsson. “Literature Study on Extremum Control.” Technical report, Chalmers University of Technology, 2001.
- [Lav02] Eugene Lavretsky. “F\A-18 Autonomous Formation Flight Control System Design.” In *Proceedings of AIAA Guidance, Navigation, and Control Conference and Exhibit*, Monterey, California, August 2002.
- [Leb22] M. Leblanc. “Sur l’electrification des chemins de fer au moyen de courants alternatifs de frequence elevee.” In *Revue Generale de l’Electricite*, Frances, 1922.
- [LHC03a] E Lavretsky, N Hovakimyan, and A Calise. “Adaptive extremum seeking control design.” In *American Control Conference, 2003. Proceedings of the 2003*, pp. 567 – 572, Denver, CO, June 2003. ACC.
- [LHC03b] E Lavretsky, N Hovakimyan, A Calise, and V Stepanyan. “Adaptive vortex seeking formation flight neurocontrol.” In *in AIAA Guidance, Navigation, and Control Conference and Exhibit*, pp. 11–14, Austin, Texas, 2003.
- [LK10] Shu-Jun Liu and Miroslav Krstic. “Stochastic Averaging in Continuous Time and Its Applications to Extremum Seeking.” *IEEE Transactions On Automatic Control*, **55**:2235–2250, 2010.
- [Lue73] D. G. Luenberger. *Introduction to Linear and Nonlinear Programming*. Wiley, New York, 1973.
- [MB10] Richard H. Middleton and Julio H. Braslavsky. “String Instability in Classes of Linear Time Invariant Formation Control With Limited Communication Range.” *IEEE Transactions on Automatic Control*, **55**:1519–1530, 2010.
- [McG61] W.A McGowan. “Calculated Normal Load Factors on Light Airplanes Traversing the Trailing Vortices of Heavy Transport Airplanes.” Technical Note D-829, NASA, 1961.
- [MMB08a] William H. Moase, Chris Manzie, and Michael J. Brear. “Newton-Like Extremum-Seeking Part I: Theory.” In *Proceedings of Joint 48th IEEE Conference on Decision and Control*, 2008.
- [MMB08b] William H. Moase, Chris Manzie, and Michael J. Brear. “Newton-Like Extremum-Seeking Part II: Simulations and Experiments.” In *Proceedings of Joint 48th IEEE Conference on Decision and Control*, 2008.
- [Moa10] William H. Moase. “Newton-Like Extremum-Seeking for the Control of Thermoacoustic Instability.” *IEEE Transactions On Automatic Control*, **55**(8):2094–2105, September 2010.

- [Nel74] R.C Nelson. “The Response of Aircraft Encountering Wake Turbulence.” Technical Report 74-29, Air Force Flight Dynamics Laboratory, 1974.
- [NFG04] Jonathan G. Nehrbass, Joshua B. Frommer, Loren A. Garrison, David N. Loffing, and William A. Crossley. “Point to Point Commercial Aircraft Service Design Study Including Formation Flight and Morphing.” In *Proceedings of AIAA 4th Aviation Technology, Integration and Operations (ATIO) Forum*, number 6404 in AIAA-2004, Chicago, Illinois, September 2004. AIAA.
- [NFK10] S. Andrew Ning, Tristan C. Flanzer, and Ilan M. Kroo. “Aerodynamic Performance of Extended Formation Flight.” In *448th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, January 2010.
- [NS02] F. Najson and J.L. Speyer. “Extremum Seeking Loop for a Class of Performance Functions.” In *15th IFAC World Congress on Automatic Control*, Barcelona, Spain, July 2002.
- [Oga95] K. Ogata. *Digital-Time Control Systems*. Prentice-Hall, 2 edition, 1995.
- [OPJ04] Ryan K. Osteros, Meir Pachter, and David R. Jacques. “Full Capability Formation Flight Control.” In *Proceedings of AIAA Guidance, Navigation, and Control Conference and Exhibit*, Providence, Rhode Island, August 2004.
- [Ost57] I. I. Ostrovskii. “Extremum regulation.” *Automatic and Remote Control*, **18**:900–907, 1957.
- [PAO02] Yaidong Pan, Tankut Acarman, and Umit Ozguner. “Nash Solution by Extremum Seeking Control Approach.” In *Proceedings of the IEEE Conference on Decision and Control*, Las Vegas, NV, December 2002.
- [PDP01] Meir Pachter, John J. Dazzo, and Andrew W. Proud. “Tight Formation Flight Control.” *Journal Of Guidance, Control, And Dynamics*, **24**(2):246–254, MarchApril 2001.
- [PJM06] D. Popović, M. Jankovic, S. Magner, and A.R. Teel. “Extremum seeking methods for optimization of variable cam timing engine operation.” *IEEE Transactions on Control Systems Technology*, **14**(3):398–407, 2006.
- [PO02] Yaodong Pan and Umit Ozguner. “Extremum Seeking Control with Sliding Mode.” In *Proceedings of the IFAC 15th Triennial World Congress*, Barcelona, Spain, 2002.
- [Pop04] Dobrivoje Popović. *Topics in Extremum Seeking*. PhD thesis, University of California, Santa Barbara, 2004.
- [Pro99] Andrew W Proud. “Close formation flight control.” In *AIAA Guidance, Navigation, and Control Conference and Exhibit*, number 4207 in AIAA-98, Portland, OR, August 1999. AIAA. Does Not use extremum seeking.
- [RCV02] Ron. J. Ray, Brent R. Cobleigh, M. Jake Vachon, and Clinton St. John. “Flight Test Techniques Used To Evaluate Performance Benefits During Formation Flight.” In *the AIAA Guidance, Navigation and Control Conference*, pp. 2002–4492, 2002.

- [Rot00] M. A. Rotea. “Analysis of multivariable extremum seeking algorithms.” In *Proc. of the American Control Conference*, pp. 433–437, Chicago, IL, June 2000. extends Kristic methods to multivariable extremum seeking schemes.
- [RV96] H.M. Reimer and D.D Vicroy. “A Preliminary Study of a Wake Vortex Encounter Hazard Boundary for a B737-100 Airplane.” Technical Memorandum 110223, NASA, 1996.
- [SBC00] JL Speyer, RN Banavar, DF Chichka, and I. Rhee. “Extremum seeking loops with assumed functions.” In *Decision and Control, 2000. Proceedings of the 39th IEEE Conference on*, volume 1, 2000.
- [SC08] Jason Lee Speyer and Walter H. Chung. *Stochastic processes, estimation, and control*. Advances in Design and Control. SIAM, 1 edition, 2008. ISBN 978-0-898716-55-9.
- [Sch42] H. Schlichting. “Leistungsersparnis in verbandsflug.” *Bericht Aerodyn. Inst. T. H. Braunschweig*, **42/6**:English translation: TMB Translation 1951 239, US Navy, 1942.
- [Sch44] H. Schlichting. “Verbandsflug mit hohenstaffelung.” *Bericht Aerodyn. Inst. T. H. Braunschweig*, **44/7**:English translation: TMB Translation 1950 240, US Navy, 1944.
- [SD90] S. Sheikholeslam and C. Desoer. “Longitudinal control of a platoon of vehicles.” In *Proceedings of the American Control Conference*, 1990.
- [SH07] Elaine Shaw and J. Karl Hedrick. “String Stability Analysis for Heterogeneous Vehicle Strings.” In *Proceedings of the American Control Conference*, New York City, July 2007.
- [SMT70] T.K. Speer, E.C. Mills, and J. L. Tate. “Formation Flight Technology.” In *AIAA 7th Annual Meeting and Technical DisPlay*. AIAA, October 1970.
- [SPC00] Sahjendra N. Singh, Meir Pachter, Phil Chandler, Siva Banda, Steve Rasmussen l, and Corey Schumacher. “Input-Output Invertibility and Sliding Mode Control for Close Formation Flying.” In *Proceedings of AIAA Guidance, Navigation, and Control Conference and Exhibit*, number 4374 in AIAA-2000, Denver, CO, August 2000.
- [SPH04] P. Seiler, A. Pant, and J. K. Hedrick. “String Instabilities in Formation Flight: Limitations Due to Integral Constraints.” *Journal of Dynamic Systems, Measurement, and Control*, **126**:873–879, December 2004.
- [SS09] Milos S. Stankovic and Dusan M. Stipanovic. “Stochastic Extremum Seeking With Applications to Mobile Sensor Networks.” In *American Control Conference*, 2009.
- [SSL77] R.I. Sammonds, G.W. Jr Stinnett, and W. Larson. “Criteria Relating Wake Vortex Encounter Hazard to Aircraft Response.” *Journal of Aircraft*, **14**:981–987, 1977.
- [Sug03] Takeshi Sugimoto. “A theoretical analysis of formation flight as a nonlinear self-organizing phenomenon.” *IMA Journal of Applied Mathematics*, **68**(5):441–470, oct 2003.
- [SXT07] Eugenio Schuster, Chao Xu, Nicholas Torres, Eiji Morinaga, Christopher Allen, and Miroslav Krstic. “Beam Matching Adaptive Control via Extremum Seeking.” *Nuclear Instruments and Methods in Physics Research A*, **581**:799815, 2007.

- [TMM10] Y. Tan, W.H. Moase, C. Manzie, D. Nesic, and I.M.Y. Mareels. “Extremum Seeking From 1922 To 2010.” In *Proceedings of the 29th Chinese Control Conference*, Beijing, China, July 2010.
- [TS10] Rodney Teo, Dusan M. Stipanovic, , and Claire Tomlin. “Decentralized Spacing Control of a String of Multiple Vehicles Over Lossy Datalinks.” *IEEE Transactions On Control Systems Technology*, **18**(2):469–473, March 2010.
- [VRW02] M. Jake Vachon, Ronald J. Ray, Kevin R. Walsh, and Kimberly Ennix. “F\A-18 Aircraft Performance Benefits Measured During the Autonomous Formation Flight Project.” In *the AIAA Guidance, Navigation and Control Conference*, 2002.
- [Wal00] G. C. Walsh. “On the application of multi-parameter extremum seeking control.” In *Proc. of the American Control Conference*, pp. 411–415, Chicago, IL, June 2000.
- [Wie14] C. Wieselsberger. “Beitrag zur Erklärung des Winkelfluges eiger Zugvögel.” *Z. Flugtechnik u. Motorluftschiffahrt*, **5**:225–229, 1914.
- [WJB02] Geno Wagner, Dave Jacques, William Blake, and Meir Pachter. “Flight Test Results Of Close Formation Flight For Fuel Savings.” In *Procedings of AIAA Atmospheric Flight Mechanics Conference and Exhibit*, number 4490 in AIAA 2002, Monterey, California, August 2002 2002. AIAA.
- [WKB99] Hsin-Hsiung Wang, Miroslav Krstic, and Georges Bastin. “Optimizing Bioreactors by Extremum Seeking.” *International Journal of Control Adaptive Control and Signal Processing*, **13**:651–669, 1999.
- [WMC01] H. Weimerskirch, J. Martin, Y. Clerquin, P. Alexandre, and S. Jiraskova. “Energy saving in flight formation.” *nat*, **413**:697–698, October 2001.
- [WPB07] David J. Willis, Jaime Peraire, and Kenneth S. Breuer. “A Computational Investigation of Bio-Inspired Formation Flight and Ground Effect.” In *25th AIAA Applied Aerodynamics Conference*, Miami, FL, June 2007.
- [WYK00] Hsin-Hsiung Wang, Simon Yeung, and Miroslav Krstic. “Experimental Application of Extremum Seeking on an Axial-Flow Compressor.” *IEEE Transactions On Control Systems Technology*, **8**(2):300–309, 2000.
- [XG11] Lingyun Xiao and Feng Gao. “Practical String Stability of Platoon of Adaptive Cruise Control Vehicles.” *IEEE Transactions on Intelligent Transportation Systems*, 2011.
- [Yin08] Iven Mareels Ying Tan, Dragan Nesic. “On the choice of dither in extremum seeking systems: A case study.” *Automatica*, **44**:1446–1450, March 2008.
- [Yu06] Hai Yu. *The Adaptive Seeking Control Strategy And Applications In Automotive Control Technology*. PhD thesis, The Ohio State University, 2006.
- [ZO05] Chunlei Zhang and Raul Ordonez. “Numerical Optimization-based Extremum Seeking Control of LTI Systems.” In *Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference*, pp. 4428–4433, 2005.