

2006

Numerical optimization and perturbation-based extremum seeking control and applications

Chunlei Zhang
University of Dayton

Follow this and additional works at: https://ecommons.udayton.edu/graduate_theses

Recommended Citation

Zhang, Chunlei, "Numerical optimization and perturbation-based extremum seeking control and applications" (2006). *Graduate Theses and Dissertations*. 6566.
https://ecommons.udayton.edu/graduate_theses/6566

This Dissertation is brought to you for free and open access by the Theses and Dissertations at eCommons. It has been accepted for inclusion in Graduate Theses and Dissertations by an authorized administrator of eCommons. For more information, please contact mschlange1@udayton.edu, ecommons@udayton.edu.

Numerical Optimization and Perturbation-based Extremum
Seeking Control and Applications

DISSERTATION

Submitted to

The School of Engineering of the

UNIVERSITY OF DAYTON

In Partial Fulfillment of the Requirements for

The Degree

Doctor of Philosophy in Electrical Engineering

by

Chunlei Zhang

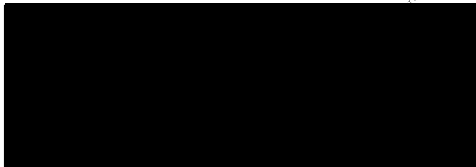
UNIVERSITY OF DAYTON

Dayton, Ohio

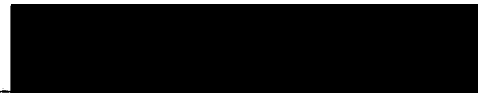
May, 2006

Numerical Optimization and Perturbation based Extremum Seeking
Control and Applications

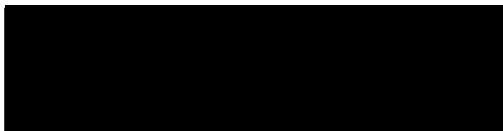
APPROVED BY:



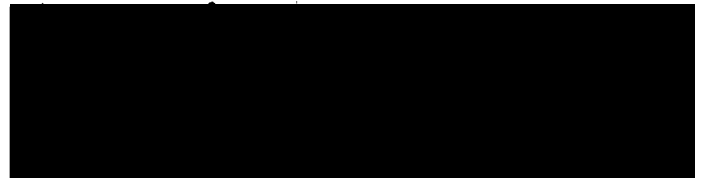
Raúl Ordóñez, Ph.D.
Advisory Committee Chairman
Associate Professor
Department of Electrical and
Computer Engineering



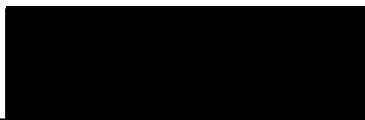
John S. Loomis, Ph.D.
Committee Member
Associate Professor
Department of Electrical and
Computer Engineering



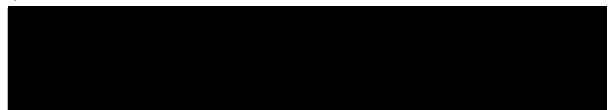
Russell C. Hardie, Ph.D.
Committee Member
Professor
Department of Electrical and
Computer Engineering



Ruihua Liu, Ph.D.
Committee Member
Assistant Professor
Department of Mathematics



Donald L. Moon, Ph.D.
Associate Dean
Graduate Engineering Program &
Research, School of Engineering



Joseph E. Saliba, Ph.D., P.E.
Dean, School of Engineering

To my parents and Ran.

ABSTRACT

Tracking a varying maximum or minimum of a performance (output, cost) function is called *extremum seeking control*. The motivation for the research on extremum seeking control arises from its practical interest, since even small improvements in performance can lead to large savings in raw material and energy consumption. For example, maximize the yield of bioreactors, minimize the power demand in formation control, maximize the traction between the wheel and the road to stop the vehicle faster, or even get a better tuning of the PID coefficients. The main purpose of this dissertation is to study the perturbation-based extremum seeking control, numerical optimization-based extremum seeking control, and their applications. We first introduce a new idea to extend the applicability of perturbation-based extremum seeking control to moderately unstable systems. Then, we propose a novel numerical optimization-based extremum seeking control based on optimization algorithm and state regulation, starting from simple linear time invariant system and extending to a class of feedback linearizable nonlinear systems. We also analyze the robustness of two main optimization algorithms: line search methods and trust region methods. Further design flexibility is achieved via the robustness analysis of the optimization algorithms and the output tracking framework. Robust and adaptive design of numerical optimization-based extremum seeking control are then pursued, where nonlinear

damping and nonlinear adaptive control are used to deal with input disturbances and unmodeled plant dynamics.

On the application aspects, we perform a comparative study of antilock braking system design via different extremum seeking control schemes. Another interesting and promising application studied here is the autonomous vehicle source seeking problem. We especially study the perturbation-based extremum seeking control to the design of autonomous vehicle source seeking, where the stability analysis is conducted for different models of autonomous vehicle. Finally, we make further progress on the swarm seeking problem, where source seeking, formation control, collision avoidance and obstacle avoidance of a group of autonomous vehicles are achieved via extremum seeking control and potential fields.

ACKNOWLEDGMENTS

I give a sincere gratitude to the people who do science with pure, unselfish and honest passion as they are the people who made me grow and appreciate the world in the way I see. I would like to express my gratitude to all those who gave me the possibility to complete my Ph.d study.

My greatest gratitude must go to my advisor Professor Raúl Ordóñez, for his patient guidance for the past five years. He has not only been a great mentor but a good friend to me. I appreciate his passion in control, attitude in research and pleasant personality. Particular, his powerful encouragement every time I make progress or fell down, insightfully spark new ideas, patiently read through my lengthy papers . It is his support, advice and encouragement that have helped me through the Ph.D program.

This work would not have been possible without the help from Professor Miroslav Krstić at University of California San Diego. My Ph.d research benefits greatly from his insightful guidance and suggestions through the summer visiting and continuous collaboration afterwards, especially in the perturbation-based extremum seeking control. Moreover, he is a great leader in the research of extremum seeking control and several other areas, and a tireless person. His devotion to control research sets an excellent paradigm for me to follow.

My thanks must go to Professor Qin Sheng at Baylor University, who is also my advisor for the Applied Mathematics program at University of Dayton. His scientific integrity and unwavering support have been instrumental to my success in the research. I particularly appreciate the trust and encouragement he offered when I began to explore the rigorous world of mathematics.

My gratitude also goes to many professors for various reasons. I would like to thank Professor John Loomis, Professor Russel Hardie, Professor Qin Sheng and Professor Ruihua Liu for serving on my dissertation committee for all the important events of my Ph.D study: qualification, dissertation proposal and defense. I want to thank you Professor Kevin Moore at Colorado School of Mines, Professor Kevin M. Passino at the Ohio State University and Dr. Corey Schumacher at AFRL on the discussions of multiple autonomous vehicles cooperative search problem. I want to thank you Professor Veysel Gazi at TOBB University of Economics and Technology on his pioneer work on swarming source seeking and his generosity in sharing the simulation code. I also want to thank you Professor Umit Ozguner at the Ohio State University for his discussion on sliding mode-based extremum seeking control. I want to thank you Professor Manfredi Maggiore at University of Toronto for his insightful comments on extremum seeking control. I would also thank Professor Paul Elie, Professor Ruihua Liu, Professor John Dennis at Rice University and Professor Mark A. Abramson at Air Force Institute of Technology for their help in numerical optimization and general mathematics.

My research has also benefited from discussions and interactions with many others. I would like to thank you Schreecharan Kanchanavally for his long term collaboration on various research problems, especially in multiple autonomous vehicles control. I

would like to thank you Jinyi Yao for her conversation and collaboration on swarm source seeking problem, Kayode Ajayi-Majebe for the Khepera experimentations, Jia Wang for the potential fields and decentralized control. I would like to thank you Daniel Arnold, Antranik Siranosian and Nima Ghods for their collaborative research on autonomous vehicle source seeking via extremum seeking control. I would also thank Hai Yu for his interesting conversation on extremum seeking control. I will give special thanks for the anonymous reviewers for my submitted papers, they give me precious opinions on my research and its presentation. I would like to thank you Marilyn Knisley and Loretta Christon coordinating and helping for my Ph.D study.

My Ph.D study is support by Dayton Area Graduate Studies Institute Scholarship and Assistantship, and summer research visiting scholarship and other traveling funding from the University Dayton Graduate School, Department of Electrical and Computer Engineering and School of Engineering. I did appreciate the providers of the scholarship for giving me the precious opportunity of top quality education and research.

I would like to thank all my friends in supporting my life and study. I finally thank my family, especially my parents and Ran who give me unlimited love, support and have more confidence in me than myself. The dissertation is dedicated to them.

TABLE OF CONTENTS

	Page
Abstract	iv
List of Figures	xii
Chapters:	
1. Introduction	1
1.1 Motivation	1
1.2 History and Literature Overview	4
1.3 Contributions	10
1.4 Outline	12
1.5 Numerical Optimization	13
1.5.1 Unconstrained Optimization	16
1.5.2 Line Search Methods	17
1.5.3 Trust Region Methods	22
2. Analog Optimization-based Extremum Seeking Control	26
2.1 Gradient-based Extremum Seeking Control	28
2.2 Sliding Mode-based Extremum Seeking Control	31
2.3 Perturbation-based Extremum Seeking Control	33
2.3.1 Sinusoidal Perturbation-based Analog Optimization	33
2.3.2 Perturbation-based Extremum Seeking Control	37
2.4 Perturbation-based Extremum Seeking Control for a Plant with Slightly Unstable Poles	42
2.5 Perturbation-based Extremum Seeking Control for a Plant with Mod- erately Unstable Poles	47

3.	Numerical optimization-based Extremum Seeking Control	53
3.1	Problem Statement	53
3.2	LTI Systems	56
3.3	State Feedback Linearizable Systems	63
3.3.1	Line Search-based Extremum Seeking Control	65
3.3.2	Trust Region-based Extremum Seeking Control	68
3.4	Input-Output Feedback Linearizable Systems	74
3.5	Robustness Issues	79
3.5.1	Robustness of Line Search Methods	86
3.5.2	Robustness of Trust Region Methods	94
3.5.3	Design of Robust Extremum Seeking Control Scheme	97
4.	Robust and Adaptive Design of Extremum Seeking Control	100
4.1	An Output Tracking Framework for Numerical Optimization-based Extremum Seeking Control	101
4.1.1	Construction of a Reference Signal	101
4.1.2	Design of a State Regulator	105
4.1.3	Determination of the Regulation Time	107
4.1.4	Algorithm and Convergence	109
4.1.5	Simulations	112
4.2	Robust Design for Input Disturbance	113
4.2.1	Bounded Input Disturbance	114
4.2.2	Unbounded Input Disturbance	117
4.2.3	Simulations	118
4.3	Robust Design for Unknown Plant Dynamics	119
4.3.1	Indirect Adaptive Control	120
4.3.2	Direct Adaptive Control	125
4.3.3	Simulations	127
5.	Antilock Braking Systems	130
5.1	Model of a Slipping Wheel	130
5.2	Perturbation-based Extremum Seeking Control Design	133
5.3	Sliding Mode-based Extremum Seeking Control Design	134
5.4	Numerical Optimization-based Extremum Seeking Control Design	136
5.5	Comparison	139

6.	Source Seeking	143
6.1	Introduction	143
6.2	A Velocity-Actuated Point Mass (Single Integrators)	146
6.3	A Force-Actuated Point Mass (Double Integrators)	151
6.4	Nonholonomic Unicycle	157
7.	Swarm Source Seeking	172
7.1	Problem Statement	174
7.2	Gradient-based Extremum Seeking Control Design	177
7.3	Perturbation-based Extremum Seeking Control Design	184
7.4	Numerical Optimization-based Extremum Seeking Control Design	189
7.5	Comparison	192
8.	Conclusions and Future Work	200
8.1	Conclusions	200
8.2	Future Work	201
	Bibliography	206
	Vita	219

LIST OF FIGURES

Figure	Page
1.1 Steepest descent method on Rosenbrock's function.	23
2.1 Analog optimization-based extremum seeking control.	28
2.2 Sliding mode-based analog optimization.	32
2.3 Sinusoidal perturbation-based analog optimization.	34
2.4 Perturbation-based extremum seeking control.	37
2.5 Perturbation-based extremum seeking control of a plant with slightly unstable poles.	44
2.6 Perturbation-based extremum seeking control of a plant with slightly unstable poles: (a) performance output; (b) state; (c) control input of x -axis; (d) control input of y -axis.	47
2.7 Perturbation-based extremum seeking control of a plant with moder- ately unstable poles.	48
2.8 Perturbation-based extremum seeking control of a plant with moder- ately unstable poles: (a) performance output; (b) state; (c) control input of x -axis; (d) control input of y -axis.	52
3.1 Numerical optimization-based extremum seeking control.	54
3.2 Line search-based extremum seeking control for controllable and stable LTI system: (a_1, a_2) performance function; (b_1, b_2) state; (c_1, c_2) control input; (d_1, d_2) phase portrait.	64

3.3	Line search-based extremum seeking control for state feedback linearizable nonlinear system with quadratic performance function: (a) performance function; (b) control input; (c) state x ; (d) state z ; (e) phase portrait in x -coordinates; (f) phase portrait in z -coordinates.	69
3.4	Line search-based extremum seeking control for state feedback linearizable nonlinear system with Rosenbrock's performance function: (a) performance function; (b) control input; (c) state x ; (d) state z ; (e) phase portrait in x -coordinates; (f) phase portrait in z -coordinates. .	70
3.5	Trust region-based extremum seeking control for the feedback linearizable nonlinear system: (a_1, a_2) performance function; (b_1, b_2) state; (c_1, c_2) control input; (d_1, d_2) phase portrait.	75
3.6	Line search-based extremum seeking control for input-output feedback linearizable nonlinear system: (a_1, a_2) performance function; (b_1, b_2) state; (c_1, c_2) control input; (d_1, d_2) phase portrait.	80
3.7	Line search-based extremum seeking control for controllable and stable LTI system with input disturbance: (a_1, a_2) performance function; (b_1, b_2) state; (c_1, c_2) control input; (d_1, d_2) phase portrait.	84
3.8	Extremum seeking control for controllable and stable LTI system with input disturbance, given different regulation time: (a_1, a_2) performance function; (b_1, b_2) state; (c_1, c_2) control input; (d_1, d_2) phase portrait. .	85
3.9	Extremum seeking control for controllable but unstable LTI system without and with input disturbance: (a_1, a_2) performance function; (b_1, b_2) state; (c_1, c_2) control input; (d_1, d_2) phase portrait.	87
3.10	Extremum seeking control for controllable but unstable LTI system with input disturbance, given different regulation time: (a_1, a_2) performance function; (b_1, b_2) state; (c_1, c_2) control input; (d_1, d_2) phase portrait.	88
3.11	Extremum seeking control for controllable but unstable LTI system with and without input disturbance, with stabilized pole placement controller: (a_1, a_2) performance function; (b_1, b_2) state; (c_1, c_2) control input; (d_1, d_2) phase portrait.	89
3.12	Robust region (3.26) and (3.27) of the trust region method.	99

4.1	Extremum seeking control based on line search and output tracking, $k_1 = 2, k = 2, T_r = 3, \alpha_k = 0.0012$: (a) performance output; (b) state; (c) control input; (d) phase portrait.	114
4.2	Extremum seeking control based on line search and output tracking, $k_1 = 2, k = 5, T_r = 1, \alpha_k = 0.0012$: (a) performance output; (b) state; (c) control input; (d) phase portrait.	115
4.3	Extremum seeking control based on line search and output tracking, bounded input disturbance, $k_1 = 2, k = 2, \alpha_k = 0.0012$: (a) performance output; (b) state; (c) control input; (d) phase portrait.	120
4.4	Extremum seeking control based on line search and output tracking, unbounded input disturbance, $k_1 = 2, k = 2, \alpha_k = 0.0012$: (a) performance output; (b) state; (c) control input; (d) phase portrait.	121
4.5	Stabilizing controller using signum function for bounded input disturbance, $k_1 = 2, k = 2, 1 \leq \delta_k \leq 3, \alpha_k = 0.001, \beta = 3$: (a) performance output; (b) state; (c) control input; (d) phase portrait.	122
4.6	Smoothed stabilizing controller for bounded input disturbance, $k_1 = 2, k = 2, \alpha_k = 0.0012, \beta = 3, c = 0.1$: (a) performance output; (b) state; (c) control input; (d) phase portrait.	123
4.7	Stabilizing controller for unbounded input disturbance, $k_1 = 1, k = 3, \alpha_k = 0.0012, \eta = 2$: (a) performance output; (b) states; (c) control input; (d) phase portrait.	124
4.8	Indirect adaptive control for unknown plant dynamics with unknown unbounded input disturbance, $k_1 = 2, k = 2, p_1 = p_2 = 3, \alpha_k = 0.0007, g_0 = 0.9, W_1 = 10, W_2 = 10, \gamma_1 = 2, \gamma_2 = 2$: (a) performance output; (b) state; (c) control input; (d) phase portrait.	128
4.9	Direct adaptive control for unknown plant dynamics with unknown unbounded input disturbance, $k_1 = 2, k = 2, p_u = 3, \alpha_k = 0.001, g_0 = 0.9, B = 0, W_u = 20, \gamma_u = 5$: (a) performance output; (b) state; (c) control input; (d) phase portrait.	129
5.1	The wheel forces.	131

5.2	Friction force coefficient.	132
5.3	ABS design via perturbation-based extremum seeking control.	134
5.4	ABS design via perturbation-based extremum seeking control: (a) friction force coefficient; (b) slip; (c) linear velocity and angular velocity; (d) braking torque.	135
5.5	ABS design via sliding mode-based extremum seeking control.	136
5.6	ABS design via sliding mode-based extremum seeking control: (a) friction force coefficient; (b) slip; (c) linear velocity and angular velocity; (d) braking torque.	137
5.7	ABS design via numerical optimization-based extremum seeking control.	138
5.8	ABS design via line search-based extremum seeking control: (a) friction force coefficient; (b) slip; (c) linear velocity and angular velocity; (d) braking torque.	139
5.9	ABS design via trust region-based extremum seeking control: (a) friction force coefficient; (b) slip; (c) linear velocity and angular velocity; (d) braking torque.	140
5.10	ABS design via extremum seeking control based on line search and output tracking: (a) friction force coefficient; (b) slip; (c) linear velocity and angular velocity; (d) braking torque.	141
5.11	ABS design via extremum seeking control based on trust region and output tracking: (a) friction force coefficient; (b) slip; (c) linear velocity and angular velocity; (d) braking torque.	142
6.1	Extremum seeking for velocity-actuated point mass.	146
6.2	Extremum seeking for velocity-actuated point mass, stationary case: (a) output; (b) vehicle trajectory starts from (1,1); (c) control input of x -axis; (d) control input of y -axis.	152

6.3	Extremum seeking for velocity-actuated point mass, slowly time-varying case: (a) output; (b) vehicle trajectory starts from (1,1) and source trajectory starts from (0,0); (c) control input of x -axis; (d) control input of y -axis.	153
6.4	Extremum seeking for velocity-actuated point mass, slowly time-varying case: (a) large adaption gain, (b) slower time varying maximizer. . . .	154
6.5	Extremum seeking for force-actuated point mass.	155
6.6	Extremum seeking for force-actuated point mass, stationary case: (a) output; (b) vehicle trajectory starts from (1,1); (c) control input of x -axis; (d) control input of y -axis.	158
6.7	Extremum seeking for force-actuated point mass, slowly time-varying case: (a) output; (b) vehicle trajectory starts from (1,1) and source trajectory starts from (0,0); (c) control input of x -axis; (d) control input of y -axis.	159
6.8	Unicycle model with non-collocated sensor.	160
6.9	Extremum seeking for unicycle model.	161
6.10	Extremum seeking of unicycle with non-collocated sensor, stationary target case, $\omega_0 = \omega/5$: (a) output of nonlinear map; (b) vehicle center and sensor trajectories; (c) forward velocity before modulation	169
6.11	Extremum seeking of unicycle with non-collocated sensor, stationary target case, $\omega_0 = \omega/3$: (a) vehicle center and sensor trajectories; (b) triangular pattern of the vehicle center movement (c) sensor position trajectory	170
6.12	Extremum seeking of unicycle with non-collocated sensor, moving target case, $\omega_0 = \omega/5$: (a) vehicle center and target trajectories; (b) output of the nonlinear map.	171
7.1	Swarm source seeking.	178
7.2	Swarm source seeking via gradient-based extremum seeking control, velocity-actuated point mass: (a) trajectories of the swarm vehicles; (b) final formation; (c) potential functions.	182

7.3	Swarm source seeking via gradient-based extremum seeking control, force-actuated point mass: (a) trajectories of the swarm vehicles; (b) final formation; (c) potential functions.	183
7.4	Swarm source seeking via perturbation-based extremum seeking control, velocity-actuated point mass: (a) trajectories of the swarm vehicles ; (b) final formation; (c) potential functions.	185
7.5	Potential function of a rectangular obstacle.	186
7.6	Swarm source seeking via perturbation-based extremum seeking control, velocity-actuated point mass, obstacle avoidance: (a) trajectories of the swarm vehicles; (b) obstacle avoidance; (c) potential functions.	188
7.7	Swarm source seeking via perturbation-based extremum seeking control, force-actuated point mass: (a) trajectories of the swarm vehicles; (b) final formation; (c) potential functions.	190
7.8	Swarm source seeking via perturbation-based extremum seeking control, force-actuated point mass, obstacle avoidance: (a) trajectories of the swarm vehicles; (b) obstacle avoidance; (c) potential functions.	191
7.9	Swarm source seeking via numerical optimization-based extremum seeking control, velocity-actuated point mass: (a) trajectories of the swarm vehicles; (b) final formation; (c) potential functions.	193
7.10	Swarm source seeking via numerical optimization-based extremum seeking control, velocity-actuated point mass, obstacle avoidance: (a) trajectories of the swarm vehicles; (b) obstacle avoidance; (c) potential functions.	194
7.11	Swarm source seeking via numerical optimization-based extremum seeking control, force-actuated point mass: (a) trajectories of the swarm vehicles; (b) final formation; (c) potential functions.	195
7.12	Swarm source seeking via numerical optimization-based extremum seeking control, force-actuated point mass, obstacle avoidance: (a) trajectories of the swarm vehicles; (b) obstacle avoidance; (c) potential functions.	196

7.13	Swarm source seeking via numerical optimization-based extremum seeking control, force-actuated point mass, obstacle avoidance with input disturbance: (a) trajectories of the swarm vehicles; (b) obstacle avoidance; (c) potential functions.	197
8.1	(a) A two-wheel autonomous vehicle for future experiments, (b) Test bed: a vehicle with a source field represented with a gray scale image; (c) Khepera robot.	205

CHAPTER 1

INTRODUCTION

1.1 Motivation

Traditional control system design deals with the problem of *stabilization* of a known reference trajectory or set point, that is, so called “tracking” and “regulation” problem. The reference is often easily determined. However, in some occasions it can be very difficult to find a suitable reference value. For instance, the fuel consumption of a car depends on the ignition angle. It is necessary to change the ignition angle as the condition of the road and the load of the car change to maintain optimal efficiency. Also in the Antilock Braking Systems (ABS) design [1, 2, 3], there is an optimal slip that maximizes the friction force coefficient; however, such slip is not known in advance and we have to seek it and operate at that optimal slip.

Tracking a varying maximum or minimum (extremum, optimum) of a performance (output, cost) function is called *extremum seeking control* [4]. For example, maximize the yield of bioreactors [5, 6], minimize the power demand in formation control [7], maximize the traction between the wheel and the road to stop the vehicle faster [1, 2, 3], or even get a better tuning of the PID coefficients [8]. It has two layers of meaning: first, we need to seek an extremum of the output function; secondly, we

need to be able to control (stabilize) the system and drive the performance output to that extremum. In the traditional optimal control problem, the performance function generally is an integral function of the state and control, that is, the extremum of the performance function is a trajectory, therefore, the calculus of variation method is involved in the design, and the analytical form of the performance function is needed to obtain the necessary conditions of the optimal control. However, in extremum seeking control, the performance function is generally a function of the state and not known or poorly known in the design, we can only design an extremum seeking controller based on the measurements of the performance output or its derivatives if available. In ABS design, the friction force coefficient is a function of the slip, but such function is not known and is also varying with different road conditions and weather. However, we can get measurements of the friction force coefficient. Another example can be found in minimizing the effect of an unknown vortex field in the formation flight [9]. On the other hand, even if the performance function is known perfectly, we can get optimal points by finding the roots of its gradient, which may be very difficult as well. Mathematically, the root finding and optimization problem are equivalently difficult to solve.

Without assuming the knowledge of the performance function, many of the ideas used in extremum seeking control have been transferred from numerical optimization, where only the measurements of the performance output or its gradient are required. It is therefore also called dynamic optimization due to the fact that the arguments of the performance function to be optimized are constrained by a dynamic system [10]. Also such controller needs to track the extremum if it is varying. Control of air-fuel ratio of combustion is one example, where the extremum will change with the

temperature and fuel quality. Also, the maximum of tire friction force coefficient will change with temperature and road condition. Whenever off-line calculation of the optimal parameters is impractical or when no reliable model is available to predict the variation of the cost function with time, extremum seeking algorithms are used to determine and track the parameters that optimize a system level cost function in real time [11]. The goal of an extremum seeking control is to operate at a set-point that represents the optimal value of a function being optimized in the control loop [12]. It is also known as extremum control and self-optimizing control: a control system which is used to determine and to maintain the extreme value of a function is called "an extremum control system" [13].

As stated in [3], extremum seeking control is applicable in situations where there is a nonlinearity in the control problem, and the nonlinearity has a local minimum or a maximum. The nonlinearity may be in the plant, as a physical nonlinearity, possibly manifesting itself through an equilibrium, or it may be in the control objective, added to the system through a cost functional of an optimization problem. Hence, one can use extremum seeking control both for tuning a set point to achieve an optimal value of the output, or for tuning parameters of a feedback law.

There are many aspects in extremum seeking control that need further explorations: the modeling of the processes and nonlinearity [4]; transient dynamical phenomena will have to be taken into account; performance measurements with noise, low-pass filtering the output will take care the measurement noise but will introduce additional dynamics into the problem [11]. In Section 13.3 in Astrom and Wittenmark [4], the authors put extremum seeking control among the most promising future areas for adaptive control. Extremum seeking control is of great practical interest, since

even small improvements in the performance can lead to large savings in raw material and energy consumption. There are commercial extremum seeking controllers.

1.2 History and Literature Overview

In this section, we first present a historical development of extremum seeking control stated in [3] and then review the existing literature especially for the last decades. Extremum seeking control was popular in 1950s and 1960s [14, 15, 16, 17, 18, 19, 20, 21, 22], which have been pursued largely by Russian researchers and much before the theoretical breakthroughs in adaptive linear control of the 1980's. Several books in the 1950s-60s have been exclusively or partly dedicated to extremum seeking control, including Tsien [23] (1954), Feldbaum [24] (1959), Krasovskii [25] (1964), Wilde [26], and Chinaev [27] (1969). It returns as an exciting research topic and industrial real-time optimization tool in the 1990's. Extremum seeking control is also a method of adaptive control but it does not fit into the classical paradigm of model reference and related schemes, which deal with the problem of stabilization of known reference trajectory or set point. In fact, the emergence of extremum seeking control dates as far back as the 1922 paper of Leblanc [28], whose scheme may very well have been the first "adaptive" controller reported in the literature.

In the 1960s, extremum seeking control branches in two directions. On one side, the emergence of computers steered the effort on real-time optimization toward general purpose optimization algorithms (numerical optimization). On the other side, a distinction between stabilization and optimization objectives for adaptive control crystallized, and model reference adaptive control methods appeared, which are analytically tractable by simpler Lyapunov tools. As a result, extremum seeking control

as a research topic goes dormant for some 30 years. In the traditional application of extremum seeking control, the optimal parameters were assumed to vary rather slowly. Then, system dynamics were typically neglected and the algorithm was analyzed and designed using traditional “static optimization” techniques [29, 11]. Many investigations of extremum seeking control systems assume that the dynamic system is static, which can be justified if the time between the changes in the optimal reference is sufficiently long. Extremum seeking control of static systems is in essence a problem of numerical optimization [4], which can be approached by the continuous (analog) implementations of some optimization methods, the so called “analog optimization” approach (such as sinusoidal perturbation and sliding mode-based analog optimization).

Extremum seeking control witnessed a resurgence of interest after Krstić’s publication of stability studies on perturbation-based extremum seeking control in [30] and [31]. By posing certain assumptions on a nonlinear system [3], one can reduce the extremum seeking control problem to a one dimensional optimization problem. Therefore, the design of extremum seeking control focuses on how to find an updating law for the single parameter, where several interesting analog optimizers [30, 32, 33] come into the context of the extremum seeking control. Among these, the method of sinusoidal perturbation-based extremum seeking control has been the most popular one. The first stability analysis of local stability of perturbation-based extremum seeking control for a general nonlinear system was developed based on averaging analysis and singular perturbation [30], where a high-pass filter and slow perturbation signal are employed to derive the gradient information. The pioneering averaging studies of Meerkov [19, 20, 21] stand out as a precursor to the stability results in [30]. In

[31], dynamic compensation was proposed for providing stability guarantees and fast tracking of changes in the plant operating conditions for single parameter extremum seeking. Discrete time perturbation-based extremum seeking control was studied in [34]. An accelerator designed based on polynomial identification and Butterworth filter is studied in [35] to speed the convergence of the perturbation-based extremum seeking control. Roeta [11] and Walsh [36] provided the first studies of multiparameter extremum seeking control schemes, their results were for plants with constant parameters and a systematic design procedure is absent. Stability analysis for general multiparameter extremum seeking control and systematic design guidelines for stability/performance are supplied in [37]. The results obtained in [38] constitute a generalization of perturbation-based extremum seeking control, which seeks a point of zero slopes, to the problem of seeking a general slope. The book by Ariyur and Krstić [3] presents a systematic description of the perturbation-based extremum seeking control and its applications. New progress in semi-global stability appears in [39]. The perturbation-based extremum seeking control above relies on time scale decomposition and as such has so far been developed only for plants that are open loop stable, with poles that are sufficiently well damped. A new idea based on phase lead compensator extends the applicability to moderately unstable systems [40].

Sliding mode control is also a tool used for optimization [41]. Sliding mode-based extremum seeking control is introduced by Korovin and Utkin in [42], and analyzed and applied by Ozguner and his -workers [43, 44, 45, 46, 47, 48, 49, 32, 50], and others [51, 52] on a variety of automotives problems, especially in ABS design [53, 1, 54, 55]. Lyapunov method is used for the analysis of convergence and stability of the closed loop system.

Gradient or its estimation-based extremum seeking control is the most straightforward approach, and has been studied by Banavar and his co-workers in [12, 56, 57, 58, 59], which is strongly reminiscent of steepest descent type algorithms in numerical optimization. Such similar idea also appears in parameter updating law design in adaptive control [60]. The self-optimizing control studied by Horowitz and his co-workers [61, 62, 63] belongs to estimated gradient-based extremum seeking control. Recent developments of gradient-based extremum seeking control with adaptive design are pursued by Guay and his co-workers. An extremum seeking control problem is proposed and solved in [64] for a class of nonlinear systems with unknown parameters, where an explicit structure information for the performance function to be maximized is required. Assuming that one can provide a suitable functional expression for the plant profit, an adaptive receding horizon controller design technique is developed that is able to steer the process state of the closed-loop system to an unknown optimum while ensuring transient performance and process regulation about the unknown optimum [65]. They continue the work for output feedback extremum seeking control for linear uncertain plant in [66], state constrained nonlinear systems in [67] and a flatness based approach for nonlinear systems in [68].

The above mentioned perturbation-based, sliding mode-based and gradient-based extremum seeking control can be considered in one big category, the so called analog optimization-based extremum seeking control. The reason is the optimization process involved in the extremum seeking loop is continuous as supposed to general iterative numerical optimization algorithms. The requirement for continuous measurements of the gradient is very strong, thus the former two methods are very appealing because

they are non-gradient based. Other non-gradient analog optimization such as neural-network based can be found in [69, 33]. Furthermore, time scale separation is generally needed in the analog optimization-based extremum seeking control.

As opposed to analog optimization-based extremum seeking control, numerical optimization-based extremum seeking control methods were used successfully. Now, the requirement for gradient measurements is not continuous, and it may have time to collect enough output measurements to estimate the gradient. Moreover, there are non-gradient numerical optimization algorithms that can be implemented as well. Extremum seeking control via triangular search as in Zhang [70] was employed to attenuate combustor thermoacoustic oscillations and minimize diffuser losses at United Technologies Research Center (UTRC). Nonlinear programming was successfully used in extremum seeking control by defining a readout map as a steady state output function [71]. Simultaneous perturbation stochastic approximation (SPSA) recursive algorithm is used in the extremum seeking control design [10]. More systematic studies by Zhang and Ordóñez on numerical optimization-based extremum seeking control first appeared in [72], and then in [73, 74, 75, 76], where numerical optimization algorithm and state regulation are combined to design the extremum seeking control scheme, and will be presented in this dissertation.

Other research in extremum seeking control is listed for completeness. An extremum controller having no special trial steps or oscillations can be designed if some a priori knowledge about the plant and its disturbance exist [77]. That is, the extremum characteristic of the plant is unimodal and can be approximated by second or third-order polynomials, the distribution of the disturbances is close to uniform and the value of the optimum changes slowly. A self tuning concept is applied to

the extremum control problem in [78]. It can be viewed as the case of online optimization of a static performance criterion. When placed in a self-tuning context, this performance index will be parameterized as a quadratic function so that a recursive estimator can be applied to determine the coefficients of its parametric form. The mixing of dynamics with the static optimization task presents problems for the self-tuning identifier [79], which is a companion paper concerning the application of self-tuning extremum control to automotive engine management. Adaptive extremum control [80, 78, 81, 82] is intended to optimize the output of static nonlinear processes under noisy conditions. The concept of explicitly parameterising the performance index and estimating the coefficients has occurred in [83]. Extremum seeking control dealing with the steady-state optimization of noisy nonlinear systems is studied in [82]. The control action is calculated by making use of a model obtained by some kind of system identification. A new approach called Dynamic Extremum Control was studied in [84, 85] and earlier references of the authors. The authors explicitly solve the optimal condition given the form of the cost function, and a self-tuning algorithm is then developed, where a dither signal is used to prevent identifiability problems.

Extremum seeking control of Wiener type systems is considered in [86], the linear subsystem is described by a discrete-time system with delay and Gaussian distributed white noise. The nonlinearity is described as a quadratic function with a unique minimum. Therefore, the author solves the minimum of the output and the purpose of the control is to keep the output as close as possible to the minimum. Extremum seeking control based on probing strategy can be found in [87], the stability and performance issues are performed further in [88]. Discrete-time extremum seeking

algorithms for SISO nonlinear systems are proposed in [89], where the reference-to-output equilibrium map is approximated via a quadratic polynomial. The extremum can be explicitly solved by the polynomial parameters, then three extremum seeking algorithms are proposed based on the way how they estimate the polynomial parameters, which are least square estimation, parabola approximation and ellipse approximation. A similar study can be found in [90].

Among the many applications of extremum seeking control overviewed in [29] and [4] are combustion processes, grinding processes, solar cell and radio telescope antenna adjustment to maximize the received signal, and blade adjustment in water turbines and wind mills to maximize the generated power. Recent applications include ABS design [53, 1, 2, 55, 3], exercise machine [62, 63, 91], optimizing bioreactor [5, 6, 92, 93, 94, 95], formation control [96, 7, 9, 97], limit cycle minimization [98], axial-flow compressor [99], combustion instability [100, 101, 102], fermentation processes [103], engine optimization [104, 105, 106], wind turbine [107], flow separation control in diffusers [108], tubular reactor [95, 109], electromechanical valve actuator [110], thermoacoustic cooler [111], blending processes [112], plasma control [113], PID tuning [8], maximum power point tracking [114] and source seeking without position measurements [40, 115].

1.3 Contributions

In short, the main goal of this dissertation is to study and develop extremum seeking control algorithms, and explore the applicability of extremum seeking control in solving interesting engineering problems. We first introduce a new idea based on phase leader compensator to extend the applicability to moderately unstable systems

[40], which can be found in Section 2.4 and 2.5. The extension to marginally unstable systems draws motivation from the application of autonomous vehicle source seeking problems.

In Chapter 3 and 4, we propose a systematic approach on numerical optimization-based extremum seeking control, which combines the numerical optimization algorithm and state regulator design to achieve the extremum seeking purpose. We start from simple linear time invariant systems [72], then extend the result to a class of feedback linearizable nonlinear systems [73, 74]. The robustness of two main unconstrained optimization algorithms, line search methods and trust region methods, are analyzed. A new design of numerical optimization-based extremum seeking control is made possible by the robustness of the optimization algorithms and the output tracking framework [75, 76]. Then the robust and adaptive design of extremum seeking control is studied, where nonlinear damping and nonlinear adaptive control are used to deal with input disturbance and unmodeled plant dynamics.

On the application aspect, we apply different extremum seeking control schemes in the ABS design in Chapter 5. The simulation results obtained are used for a comparative study of different designs and successfully illustrate the feasibility of the new numerical optimization-based approach [73]. A more interesting and promising application studied in Chapter 6 is the autonomous vehicle source seeking problem. The design via gradient and numerical optimization-based extremum seeking control is straightforward. We especially extend the perturbation-based extremum seeking control for the design of autonomous vehicle control law in performing source seeking [40, 115]. Several models of the autonomous vehicle are considered, including velocity-actuated point mass, force-actuated point mass and nonholonomic unicycle. The

delicate stability analysis is presented based on the averaging method. A notable feature of the perturbation-based extremum seeking design is the source seeking can be realized without the position measurements of the autonomous vehicle.

Then, we make further progress on the swarm seeking problem in Chapter 7, where the goal is to design a control law for each individual vehicle to achieve source seeking, formation control, collision avoidance and obstacle avoidance of the entire group of autonomous vehicles. The problem is approached via potential fields and three different extremum seeking control schemes: gradient-based, perturbation-based and numerical optimization-based extremum seeking control. A comparative study is performed based on the simulation results, where the results shown is beyond what the theory tells us extremum seeking control can do and shows the great potential of extremum seeking control in the field of autonomous vehicle control.

1.4 Outline

The dissertation mainly contains two parts: the theory part as in Chapters 2, 3 and 4, and application part as in Chapters 5, 6 and 7. The remainder of the dissertation is organized as follows. A brief review of numerical optimization, the unconstrained optimization algorithms (line search methods and trust regions methods), is presented in Section 1.5. In Chapter 2, we review the existing techniques grouped as analog optimization-based extremum seeking control, which are gradient-based, sliding mode-based and perturbation-based extremum seeking control. Then, we introduce a new idea to extend the applicability of the perturbation-based extremum seeking control to moderately unstable systems. A novel numerical optimization-based extremum seeking control is proposed in Chapter 3. The extremum seeking

control scheme is first applied to LTI systems, then extended to a class of feedback linearizable nonlinear systems. The convergence of the extremum seeking scheme is shown via Lyapunov method. The robustness of line search methods and trust region methods is then studied. Then, in Chapter 4, a new design of numerical optimization algorithm is made possible by the robustness of the optimization algorithms and the output tracking framework. Furthermore, the robust and adaptive design of extremum seeking control is performed, where nonlinear damping and nonlinear adaptive control are used to deal with input disturbance and unmodeled plant dynamics, respectively. A comparative study of ABS design via different extremum seeking control schemes is presented in Chapter 5. In Chapter 6, we especially extend the perturbation-based extremum seeking control for the design of autonomous vehicle control law in performing source seeking, stability proof is shown for different vehicle models. Moreover, swarm seeking problem is considered in Chapter 7. Different extremum seeking control designs are used to achieve the objective of source seeking, formation control, collision avoidance and obstacle avoidance of a group of autonomous vehicles. Finally, Chapter 8 concludes the dissertation and presents some future directions.

1.5 Numerical Optimization

In this section, we present a necessary background on numerical optimization, which lies as an fundamental tool for extremum seeking control. Given a sufficiently smooth function $J : \mathbb{R}^n \rightarrow \mathbb{R}$, denote $g(x) = \nabla J(x) = \left[\frac{\partial J(x)}{\partial x_1}, \dots, \frac{\partial J(x)}{\partial x_n} \right]^T$ as the *gradient*, and $H(x) = \nabla^2 J(x)$ as the *Hessian matrix*. The (i, j) component of H is $\partial^2 J(x) / \partial x_i \partial x_j$, $1 \leq i, j \leq n$. H is symmetric if $J \in C^2$. In the case of $x = x(t)$ for

$t \in \mathbb{R}^m$, by using the chain rule we have

$$\nabla J(x(t)) = \sum_{i=1}^n \frac{\partial J}{\partial x_i} \nabla x_i(t) = \sum_{i=1}^n \frac{\partial J}{\partial x_i} \left[\frac{\partial x_i(t)}{\partial t_1}, \dots, \frac{\partial x_i(t)}{\partial t_m} \right]^T.$$

THEOREM 1.5.1 (TAYLOR'S THEOREM [116]) *Let $J : \mathbb{R}^n \rightarrow \mathbb{R}$ be continuously differentiable and $p \in \mathbb{R}^n$. Then*

$$J(x+p) = J(x) + \nabla J^T(x + \alpha p)p,$$

for some $\alpha \in (0, 1)$. Moreover, if J is twice continuously differentiable then

$$\nabla J(x+p) = \nabla J(x) + \int_0^1 \nabla^2 J(x + \alpha p) p d\alpha,$$

and

$$J(x+p) = J(x) + \nabla J(x)^T p + \frac{1}{2} p^T \nabla^2 J(x + \alpha p) p,$$

for some $\alpha \in (0, 1)$.

Note that a set $S \subseteq \mathbb{R}^n$ is *convex* if for any $x, y \in S$ we have $\beta x + (1-\beta)y \in S$ for all $\beta \in [0, 1]$. Further, S is *compact* if every sequence of elements of S has a subsequence that converges to an element of S . A function J is a *convex function* if its domain is convex and if for any x, y in this domain we have $J(\beta x + (1-\beta)y) \leq \beta J(x) + (1-\beta)J(y)$ for all $\beta \in [0, 1]$.

DEFINITION 1.5.2 *Let J be defined on $S \subseteq \mathbb{R}^n$. Point $x^* \in S$ is a global minimizer of J if $J(x^*) \leq J(x)$ for all $x \in S$; it is a strict global minimizer of J if $J(x^*) < J(x)$ for all $x \in S, x \neq x^*$. Correspondingly, we say that $J(x^*)$ is a (strict) global minimum of J .*

DEFINITION 1.5.3 *Let J be defined on $S \subseteq \mathbb{R}^n$. Point $x^* \in S$ is a local minimizer of J if there exists an open neighborhood B of x^* such that $J(x^*) \leq J(x)$ for all*

$x \in B \cap S$; it is a strict local minimizer if $J(x^*) < J(x)$ for all $x \in B \cap S$, $x \neq x^*$.

Correspondingly, we say that $J(x^*)$ is a (strict) local minimum of J .

DEFINITION 1.5.4 We say that x^* is a stationary point of J defined on $S \subseteq \mathbb{R}^n$ if $\nabla J(x^*) = 0$.

DEFINITION 1.5.5 Let J be defined on \mathbb{R}^n and $\gamma > 0$. We define the level set with respect to γ as

$$\mathcal{L}_\gamma = \{x \in \mathbb{R}^n : J(x) \leq \gamma\}.$$

Computational procedures of minimizer and minima are called *optimization*. The optimization is often achieved via different *iterative algorithms*. The algorithms begin with a *initial guess* x_0 and generate a *sequence* $\{x_k\}$ leading to a possible solution, that is, a stationary point, local minimizer or global minimizer.

DEFINITION 1.5.6 Let $S \subseteq \mathbb{R}^n$ and $\{x_k\} \subseteq S$ be a sequence generated by an optimization algorithm. If $\lim_{k \rightarrow \infty} x_k = x^* \in S$ for any $x_0 \in S$, then we say that the algorithm is globally convergent. If such a convergence only exists for some $x_0 \in S$, then we say the algorithm is locally convergent.

DEFINITION 1.5.7 Let $\{x_k\}$ be a locally convergent sequence in $S \subseteq \mathbb{R}^n$. We say that $\{x_k\}$ is Q -order convergent if

$$\lim_{k \rightarrow \infty} \frac{\|x_{k+1} - x^*\|}{\|x_k - x^*\|^p} = M$$

exists for certain $p, M > 0$. In particular, we say that $\{x_k\}$ is Q -linear convergent if $p = 1$; and Q -superlinear or Q -quadratic convergent if $1 < p < 2$ or $p = 2$, respectively.

The following standard stopping criteria are frequently employed in optimization computations. While $x_k, J(x_k) \neq 0$ for sufficiently large k , computation terminates as $\|x_{k+1} - x_k\| / \|x_k\| \leq \epsilon, |J(x_k) - J(x_{k+1})| / |J(x_k)| \leq \epsilon$, otherwise it terminates as $\|x_{k+1} - x_k\| \leq \epsilon, |J(x_k) - J(x_{k+1})| \leq \epsilon$, where $\epsilon > 0$ is a controlling parameter. More sophisticated stopping criteria may also be considered.

Since most of the optimization algorithms are iterative, there has been a fundamental tradeoff between their efficiency and robustness [117]. In general, algorithms designed to be very efficient on one type of problems tend to be brittle in the sense that they may not be ideally used for other type problems. Such a lack of universally best algorithm is a manifestation of the so-called *No Free Lunch (NFL) theorems* [118]. The NFL theorems serve as a fundamental barrier to exaggerated claims of the power and efficiency of any specific algorithm in numerical optimizations. A way to cope with negative implications of the barrier is to restrict an algorithm to a particular class of problems and to design the algorithm structures only for the anticipated class. This has become a general principle in optimization.

1.5.1 Unconstrained Optimization

Let $J : \mathbb{R}^n \rightarrow \mathbb{R}$ be a sufficiently smooth objective function. Consider

$$y^* = \min_{x \in \mathbb{R}^n} J(x). \quad (1.1)$$

The above function J is referred as an *objective function*. We do not consider a maximization optimization problem due to the fact that $\max J(x) = -\min(-J(x))$, $x \in S \subseteq \mathbb{R}^n$. The existence of a global minimizer for (1.1) has been shown in cases where the level sets of J are compact for certain γ [119].

THEOREM 1.5.8 (FIRST-ORDER NECESSARY CONDITIONS, [116]) *If x^* is a local minimizer and J is continuously differentiable in an open neighborhood of x^* , then $\nabla J(x^*) = 0$.*

THEOREM 1.5.9 (SECOND-ORDER NECESSARY CONDITIONS, [116]) *If x^* is a local minimizer and $\nabla^2 J$ is continuous in an open neighborhood of x^* , then $\nabla J(x^*) = 0$ and $\nabla^2 J(x^*)$ is positive semidefinite.*

THEOREM 1.5.10 (SECOND-ORDER SUFFICIENT CONDITIONS, [116]) *If $\nabla J(x^*) = 0$, $\nabla^2 J$ is continuous in an open neighborhood of x^* and $\nabla^2 J(x^*)$ is positive definite, then x^* is a strict local minimizer of J .*

THEOREM 1.5.11 ([116]) *If J is convex, then any local minimizer x^* is a global minimizer of J . If in addition J is differentiable, then any stationary point x^* is a global minimizer of J .*

The above conditions provide a basis for the developments and analysis of various algorithms. In particular, algorithms fulfilled should verify solutions if they satisfy certain optimality conditions, and detect if a minimizer has been approximated satisfactorily. To determine a global minimizer of a given problem is in general difficult, therefore many algorithms used can only guarantee the convergence to a stationary point.

1.5.2 Line Search Methods

Each iteration in a line search method starts from x_k , computes a *search direction* p_k , and then decides *how far* to move along that direction. This iterative process can

be illustrated by

$$x_{k+1} = x_k + \alpha_k p_k,$$

where the positive scalar α_k is called the *step length*. Most line search methods require p_k to be a *descent direction*, that is $p_k^T \nabla J(x_k) < 0$, to guarantee that the objective function value is reduced along that direction if the step length is sufficiently small.

Search Directions

As mentioned above, a descent direction may guarantee a decrease of J . This is obvious by using Taylor's theorem, which offers

$$J(x_k + \alpha_k p_k) = J(x_k) + \alpha_k p_k^T \nabla J(x_k) + O(\alpha_k^2).$$

It follows therefore that $J(x_k + \alpha_k p_k) < J(x_k)$ for all positive but sufficiently small α_k if $p_k^T \nabla J(x_k) < 0$. The *steepest-descent direction*,

$$p_k = -\nabla J(x_k),$$

is the most obvious choice for search direction. It is among all the directions we could select from x_k , that ensures J decreases most rapidly. According to Taylor's theorem, the rate of change in J along p_k at x_k is $p_k^T \nabla J(x_k)$. Thus, whenever $\|p_k\| = 1$, the most rapid decrease is the solution of the problem

$$\min_{p_k} (p_k^T \nabla J(x_k)).$$

Therefore, the solution is $p_k = -\nabla J(x_k) / \|\nabla J(x_k)\|$, and we have $p_k^T \nabla J(x_k) = -\|\nabla J(x_k)\|$.

Other frequently used search directions include the *Newton direction*,

$$p_k = -(\nabla^2 J(x_k))^{-1} \nabla J(x_k).$$

The basic idea here is to minimize the quadratic approximation of J instead of the objective function at x_k . A Newton direction can be adopted in a line search method when the Hessian matrix is positive definite. Therefore $p_k^T \nabla J(x_k) = -p_k^T \nabla^2 J(x_k) p_k \leq -\sigma_k \|p_k\|$ for some $\sigma_k > 0$. Unless the gradient $\nabla J(x_k)$ is zero, $p_k^T \nabla J(x_k) < 0$. Therefore, a Newton direction is a descent direction and a natural step length of unit is often utilized. Needless to mention that calculations of the Hessian matrix may involve large amounts of computation. A *Quasi-Newton method* is designed to avoid the disadvantage via features of $J(x_k)$ and $\nabla J(x_k)$. Their curvature information is used to construct B_k for approximating the Hessian matrix. The standard Quasi-Newton search routine is

$$p_k = -B_k^{-1} \nabla J(x_k).$$

A popular formula for obtaining B_k is the BFGS formula, named after Broyden, Fletcher, Goldfarb, and Shanno:

$$B_k = B_{k-1} - \frac{B_{k-1}^T s_{k-1} s_{k-1}^T B_{k-1}}{s_{k-1}^T B_{k-1} s_{k-1}} + \frac{y_{k-1} y_{k-1}^T}{y_{k-1}^T s_{k-1}}, \quad B_0 = I,$$

where $s_{k-1} = x_k - x_{k-1}$ and $y_{k-1} = \nabla J(x_k) - \nabla J(x_{k-1})$. Factorizations of B_k can be achieved through updating the inverse of B_{k-1} [116].

On the other hand, a conjugate gradient direction is derived through the following,

$$p_k = -\nabla J(x_k) + \beta_k p_{k-1}, \quad p_0 = -\nabla J(x_0),$$

where $\beta_k = \frac{\nabla J^T(x_k) \nabla J(x_k)}{\nabla J^T(x_{k-1}) \nabla J(x_{k-1})}$ (Fletcher-Reeves formula) or $\beta_k = -\frac{\nabla J^T(x_k) \nabla J(x_k)}{p_{k-1}^T \nabla J(x_{k-1})}$ (Dixon formula). The above value of β_k ensures p_k and p_{k+1} are conjugate, a concept originally developed for solutions of linear systems.

Step Length

Typical step length selection algorithms consist of two phases: *bracketing phase* and *selection phase*. The former finds an interval $[a, b]$ containing acceptable step lengths, while the latter zooms in the interval to locate the final step length.

The second phase can be implemented by approximating solutions of the following scalar minimization problem:

$$\min_{\alpha > 0} \phi(\alpha) = \min_{\alpha > 0} J(x_k + \alpha p_k), \quad \alpha > 0, \quad (1.2)$$

which brings the name “line search.” An exact solution for the above is expensive and frequently not necessary. More practical strategies suggest an *inexact line search* to determine a step size that makes an adequate reduction in J at minimal costs. To achieve so, conditions often used include

$$J(x_k + \alpha_k p_k) \leq J(x_k) + c_1 \alpha_k p_k^T \nabla J(x_k) \quad (1.3)$$

which prevents steps that are too long via a sufficient decrease criterion, and

$$p_k^T \nabla J(x_k + \alpha_k p_k) \geq c_2 p_k^T \nabla J(x_k), \quad (1.4)$$

which prevents steps that are too short via a curvature criterion, for $0 < c_1 < c_2 < 1$. Condition (1.3) is sometimes called the *Armijo condition*, while (1.4) is called the *Wolfe condition*. Moreover, in order to avoid poor choices of descent directions, an *angle condition* [120] can be introduced to enforce a uniformly lower bound on the angle θ_k between p_k and $-\nabla J(x_k)$:

$$\cos \theta_k = \frac{-p_k^T \nabla J(x_k)}{\|p_k\| \|\nabla J(x_k)\|} \geq c_3 > 0, \quad (1.5)$$

where c_3 is independent of k . The above holds naturally in the method of steepest descent.

Convergence and Rate of Convergence

A standard first-order global convergence result for line search methods is

THEOREM 1.5.12 [120] *Let $J : \mathbb{R}^n \rightarrow \mathbb{R}$ be continuously differentiable and bounded below. Further, let ∇J be Lipschitz continuous with constant $L > 0$, that is,*

$$\|\nabla J(y) - \nabla J(x)\| \leq L\|y - x\| \quad \text{for all } x, y \in \mathbb{R}^n.$$

If the sequence $\{x_k\}$ satisfies conditions (1.3), (1.4) and (1.5), then

$$\lim_{k \rightarrow \infty} \|\nabla J(x_k)\| = 0.$$

We can relax the assumptions for J to be bounded below on \mathbb{R}^n and continuous differentiable in an open set \mathcal{N} containing the level set $\{x | J(x) \leq J(x_0)\}$, where x_0 is the starting point of the iteration. And the gradient ∇J is Lipschitz continuous on \mathcal{N} [116]. Moreover, the following theorem shows the linear convergence rate of steepest descent algorithm.

THEOREM 1.5.13 [121] *Let $J : \mathbb{R}^n \rightarrow \mathbb{R}$ be twice continuously differentiable, and the Hessian matrix is positive definite. If $\{x_k\}$ generated by a steepest descent method with exact line search converges to x^* , then*

$$J(x_{k+1}) - J(x^*) \leq \left(\frac{\lambda_n - \lambda_1}{\lambda_n + \lambda_1} \right) [J(x_k) - J(x^*)],$$

where $0 < \lambda_1 \leq \dots \leq \lambda_n$ are the eigenvalues of the Hessian matrix of J .

It has been shown that numerical methods using Newton directions have a fast rate of local convergence, typically quadratic. Their main drawback, however, is the need of the Hessian matrix. There have been numerous recent discussions in the simplification of underlying computation procedures.

In the particularly practical case of the quasi-Newton method, if its search direction approximates the Newton direction accurately enough, then the unit step length can satisfy the Wolfe conditions as the iterates converge to a minimizer. Further, if for search direction $\lim_{k \rightarrow \infty} \|\nabla J(x_k) + \nabla^2 J(x_k)p_k\| / \|p_k\| = 0$, then the quasi-Newton method offers a superlinearly convergent iteration. It is also known that for any quadratic objective function, a conjugate gradient method terminates with an optimal solution within n steps.

Example: Minimization of the Rosenbrock's Function

The Rosenbrock's function, also known as the banana function,

$$J(x) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2, \quad x \in \mathbb{R}^2,$$

is notorious in unconstrained optimization due to its curvature bends around the origin. The only global minimizer occurs at $x^* = [1, 1]^T$, where $J(x^*) = 0$. A sequence $\{x_k\}$ obtained via a steepest descent method with inexact line search starting from $x_0 = [-1.9, 0]^T$ is shown in 1.1.

1.5.3 Trust Region Methods

Trust region methods are iterative and build, around the current iterate x_k , a model function m_k of the objective function J that is cheaper to evaluate and easier to minimize than the objective function. Because the model function may not be a good approximation of J when x is far away from x_k , we have to restrict the search for a minimizer of m_k to a local region involving x_k . Such a region is called a *trust region*, typically a ball centered at the current iterate, x_k , of the form $R_k = \{x \in \mathbb{R}^n \mid \|x - x_k\| \leq \Delta_k\}$. The radius of the ball, Δ_k , is called the trust region radius. A

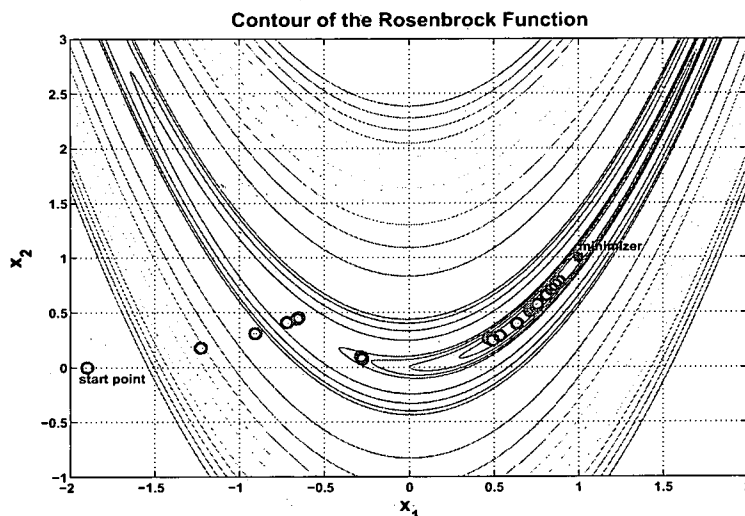


Figure 1.1: Steepest descent method on Rosenbrock's function.

new trial point is then computed, which minimizes or sufficiently reduces the model within the trust region and the true objective function is evaluated at this point. If the achieved objective function reduction is sufficiently compared with the reduction predicted by the model, the trial point is accepted as the new iterate and the trust region is centered at the new point and possible enlarged. On the other hand, if the achieved reduction is poor compared with the predicted one, the current iterate is typically unchanged and trust region is reduced. This process is then repeated until convergence occurs.

The Basic Trust Region Algorithm

Step 0 [Initialization] x_0 and Δ_0 are given. Choose $0 < \eta_1 \leq \eta_2 < 1$ and $0 < \gamma_1 \leq \gamma_2 <$

1. Compute $J(x_0)$ and set $k = 0$.

Step 1 [Model definition] Define a model $m_k(x_k + p)$ in R_k .

Step 2 [Step calculation] Compute a step p_k that “sufficiently reduces the model” m_k with $x_k + p_k \in R_k$.

Step 3 [Acceptance of the trial point] Compute $J(x_k + p_k)$ and evaluate

$$\rho_k = \frac{J(x_k) - J(x_k + p_k)}{m_k(x_k) - m_k(x_k + p_k)}. \quad (1.6)$$

If $\rho_k \geq \eta_1$, then define $x_{k+1} = x_k + p_k$; otherwise define $x_{k+1} = x_k$.

Step 4 [Trust region radius update] Update Δ_{k+1} as

$$\Delta_{k+1} = \begin{cases} [\Delta_k, \infty), & \rho_k \geq \eta_2, \\ [\gamma_2 \Delta_k, \Delta_k], & \rho_k \in [\eta_1, \eta_2), \\ [\gamma_1 \Delta_k, \gamma_2 \Delta_k], & \rho_k < \eta_1. \end{cases} \quad (1.7)$$

Set $k = k + 1$ and go to step 1.

The trust-region methods retain the quadratic convergence rate while being globally convergent. The following is a standard global convergence result for trust region methods [116].

THEOREM 1.5.14 *Let $J : \mathbb{R}^n \rightarrow \mathbb{R}$ be Lipschitz continuously differentiable and bounded below on the level set $\{x | J(x) \leq J(x_0)\}$, where x_0 is the starting point of the iteration. Suppose that $\|B_k\| \leq \beta$ for some constant β , where B_k is the approximation or the Hessian matrix of J itself, and that all trial steps p_k satisfy the inequality*

$$m_k(x_k) - m_k(x_k + p_k) \geq c_t \|\nabla J(x_k)\| \min \left(\Delta_k, \frac{\|\nabla J(x_k)\|}{\|B_k\|} \right) \quad (1.8)$$

for some constant c_t , and $\|p_k\| \leq \gamma \Delta_k$ for some constant $\gamma \geq 1$. Then

$$\lim_{k \rightarrow \infty} \|\nabla J(x_k)\| = 0.$$

Generally, quadratic approximations of J are often used for constructing m_k ,

$$m_k(p) = J(x_k) + g_k^T p + \frac{1}{2} p^T B_k p. \quad (1.9)$$

The vector g_k is either the gradient $\nabla J(x_k)$ or an approximation of it, and the matrix B_k is either the Hessian matrix $\nabla^2 J(x_k)$ or an approximation of it. Thus, such construction still requires the gradient information. However, the trust region framework provides large flexibility in designing derivative free optimization methods. This compares very favorably with most line search methods which do require gradient measurements of the objective function. Derivative free trust region algorithms proposed in [122, 123, 124] use multivariate interpolation to construct the model function m_k , where only an interpolation set Y containing the interpolating nodes and their objective function values are needed. The authors also impose certain geometric condition on Y to ensure the existence and quality of the interpolation. Moreover, both Lagrange fundamental polynomials and Newton fundamental polynomials can be used to construct the m_k , while the latter is more efficient in updating the model function given new nodes. Later, we will use the derivative free trust region algorithm [123] in the extremum seeking control scheme; this is how we claim the extremum seeking control can be gradient free.

The optimality and algorithms for constrained optimization are based on the results of unconstrained optimization, additional techniques such as penalty methods and barrier methods are used to address the equality and inequality constraints [125, 116, 126].

CHAPTER 2

ANALOG OPTIMIZATION-BASED EXTREMUM SEEKING CONTROL

Consider the nonlinear system

$$\dot{x} = f(x, u), \quad (2.1)$$

$$y = J(x), \quad (2.2)$$

where $x \in \mathbb{R}^n$ is the state, $u \in \mathbb{R}$ is the input, $y \in \mathbb{R}$ is the performance output, and $f : D \times \mathbb{R} \rightarrow \mathbb{R}^n$ and $J : D \rightarrow \mathbb{R}$ are generally smooth functions on $D \subseteq \mathbb{R}^n$. For the simplification of analysis, we assume $D = \mathbb{R}^n$ throughout the dissertation unless otherwise mentioned. Without loss of generality, we consider the design of an extremum seeking controller to find the maximum of the performance function (2.2), i.e., a maximum seeking controller design (we can achieve minimum seeking by replacing y with $-y$).

In this chapter, we focus on the general problem where the nonlinearity with an extremum arises as a reference-to-output equilibrium map of a general nonlinear system [3]. Such system is assumed to be stable or stabilizable at each of these equilibria by a local feedback controller. Suppose that we know a smooth control law

$$u = \alpha(x, \theta) \quad (2.3)$$

parameterized by a scalar parameter θ . The closed-loop system

$$\dot{x} = f(x, \alpha(x, \theta))$$

then has equilibria parameterized by θ . We make the following assumptions about the closed-loop system.

ASSUMPTION 2.0.15 *There exists a smooth function $l : \mathbb{R} \rightarrow \mathbb{R}^n$ such that*

$$f(x, \alpha(x, \theta)) = 0 \quad \text{if and only if} \quad x = l(\theta).$$

ASSUMPTION 2.0.16 *For each $\theta \in \mathbb{R}$, the equilibrium $x = l(\theta)$ of the system (2.1) is locally exponentially stable.*

Hence we assume that we have a control law (2.3) which exponentially stabilizes any of the equilibria that θ may produce. It simply means that we have a control law designed for local stabilization and this control law need not to be based on modeling knowledge of either $f(x, u)$ or $l(\theta)$. The next assumption is central to the problem of extremum seeking.

ASSUMPTION 2.0.17 *There exists $\theta^* \in \mathbb{R}$ such that*

$$(J \circ l)'(\theta^*) = 0, (J \circ l)''(\theta^*) < 0.$$

Thus, we assume that the output equilibrium map $y = J(l(\theta))$ has a maximum¹ at $\theta = \theta^*$. Our objective is to develop a feedback mechanism which maximizes the steady state value of y but without requiring knowledge of either θ^* or the functions J and l .

¹We certainly will assume $(J \circ l)''(\theta^*) > 0$ for a minimization problem

The above three assumptions were first proposed in [30], by doing so, one reduces the optimization problem of an n dimensional performance function (2.2) to a one dimensional problem of optimizing $Jol(\theta)$ in the steady state. Therefore, the design of extremum seeking control focuses on how to find an optimizing law for the parameter θ , where several interesting analog optimizers come into the context of the extremum seeking. A basic block diagram can be found in Figure 2.1.

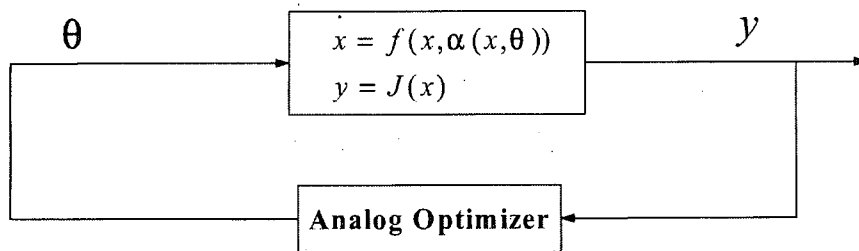


Figure 2.1: Analog optimization-based extremum seeking control.

2.1 Gradient-based Extremum Seeking Control

Consider the maximization of a performance function $y = J(\theta)$, if we know the derivative $\frac{dJ}{d\theta}$, we can choose the optimizing law for θ as

$$\dot{\theta} = k \frac{dJ}{d\theta}, k > 0. \quad (2.4)$$

By letting θ^* be an isolated local maximizer of $J(\theta)$, we can choose a Lyapunov candidate $V = J(\theta^*) - J(\theta)$, then

$$\dot{V} = -\frac{dJ}{d\theta} \dot{\theta} = -k \left(\frac{dJ}{d\theta} \right)^2 \leq 0,$$

thus, we will have that θ converges to the invariant set of $\dot{V} = 0$, that is, $\frac{dJ}{d\theta} = 0$, which can only occur at $\theta = \theta^*$ for a local region. Thus, the optimizing law (2.4) can successfully maximize $J(\theta)$, we can easily change it into a minimizing law by changing the sign of k . Assuming the knowledge of the derivative amounts to knowing $J(\theta)$ as well, and we can obtain θ^* by solving equation $\frac{dJ}{d\theta} = 0$, which also means root finding and optimization problem are somehow equivalent mathematically. In the one dimensional case, estimating the derivative is not a difficult thing and we do not really need the precise gradient information, one can choose

$$\dot{\theta} = k \operatorname{sgn} \left(\frac{dJ}{d\theta} \right), k > 0,$$

where sgn is the signum function. Then taking the same Lyapunov candidate, we have

$$\dot{V} = -k \frac{dJ}{d\theta} \operatorname{sgn} \left(\frac{dJ}{d\theta} \right) = -k \left| \frac{dJ}{d\theta} \right| \leq 0$$

Assuming knowledge of the sign of the derivative is still a strong assumption, however, we can then try to estimate the sign, which is exactly what the sliding mode approach tries to achieve. One can then form a gradient-based extremum seeking control as the framework in Figure 2.1 by using the above updating law (2.4) to be the analog optimizer.

Below we will see that a general design of extremum seeking control based on gradient feedback is difficult without reducing it to the framework in Figure 2.1. Now, consider a general n dimensional gradient system

$$\dot{x} = k \nabla J(x), \quad k > 0. \quad (2.5)$$

It is well known [127, 128] that the maximal points of J are stable equilibria of the gradient system (2.5), and that, if the level sets of J are bounded, then the trajectory

of x will converge asymptotically to the set of stationary points of J . So if we have the gradient $\nabla J(x)$ and if we can design a control law u to force the nonlinear system (2.1) with performance function (2.2) behave as the gradient system (2.5), then we can achieve the extremum seeking control. Moreover, we can also try to form the following system

$$\dot{x} = kp(x), \quad k > 0,$$

where $p(x)$ is required to be a ascent direction (that is $p(x)^\top \nabla J(x) > 0$) to guarantee the performance function is increased along the direction of \dot{x} , which can be seen from $\dot{J} = \nabla J^\top \dot{x} = k \nabla J^\top p(x) > 0$. Recent development in [129] shows that we can achieve finite convergence to the stationary point by choosing a control law u to force the system to be

$$\dot{x} = \frac{\nabla J(x)}{\|\nabla J(x)\|} \quad \text{or} \quad \dot{x} = \text{sgn}(\nabla J(x)).$$

Now, we want to see how to design the control u to force the dynamic system behave like the gradient system (2.5). Consider a linear time invariant (LTI) system:

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

where $x \in \mathbb{R}^n$, suppose x^* is a local maximum of the performance function $J(x)$, let $V = J(x^*) - J(x)$ be a Lyapunov candidate, then

$$\dot{V} = -\nabla J(x)^\top \dot{x} = -\nabla J(x)^\top (Ax + Bu).$$

We need to find a control law such that $\dot{V} \leq 0$, which generally is very difficult even if we know the gradient. Given the LTI system is square, that is $u \in \mathbb{R}^n$ and B is nonsingular, then we can choose

$$u = kB^{-1}(\nabla J(x) - Ax), k > 0$$

then $\dot{V} = -k\|\nabla J(x)\|^2 \leq 0$ and we can conclude the state will converge to the stationary points of J . Of course, we can have a more flexible design by choosing $u = kB^{-1}(p(x) - Ax)$, $k > 0$ as long as $p(x)^T \nabla J(x) > 0$. Recent research on gradient or its estimation based extremum seeking can be found in [12, 56, 57, 58, 59, 64, 65]. However, the requirement of knowing gradient $\nabla J(x)$ is a very strong assumption, and moreover for a general single input single output n dimensional nonlinear system (2.1) and (2.2), such control law to put the nonlinear system into a gradient system (2.5) may not exist or may be very difficult to find. Thus, by posing Assumptions 2.0.15-2.0.17, we can reduce the extremum seeking control design to find a controller parameter update law for θ , which optimizes the unimodal reference-to-output equilibrium map $J \circ l(\theta)$. Thus, such analog optimization laws are intensively explored.

2.2 Sliding Mode-based Extremum Seeking Control

This section mainly follows [46]. The use of sliding mode for analog optimization of an analytically unknown one dimensional function $J(\theta)$ has been reported in [42, 130]. The basic idea is to make J follow an increasing/decreasing time function via sliding mode motions. The main difficulty with such set up is that the unknown gradient term multiplies the control at the differential equation of J so that the system itself possesses a variable structure behavior. This idea has been extended in [43] with the introduction of the notion of periodic switching function and then studied in [44, 45, 46, 89, 47, 48, 49, 32, 50, 51, 52] on a variety of automotives problems, especially in ABS design [53, 1, 54, 55].

Consider the maximization of a performance function $y = J(\theta)$, the performance output y is forced to track an increasing time function irrespective of the unknown

gradient via sliding mode. A basic sliding mode-based analog optimization method can be found in Figure 2.2. Pick any increasing function $g(t)$ and try to keep $J(\theta) - g(t)$

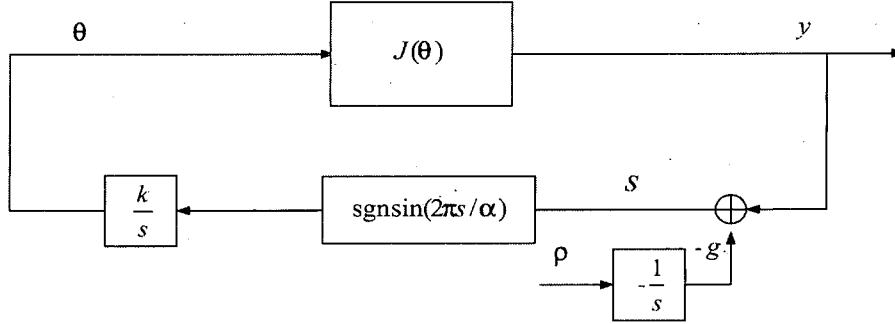


Figure 2.2: Sliding mode-based analog optimization.

at a constant by a proper choice of $\dot{\theta}$. If so, $J(\theta)$ increases at the same rate with $g(t)$ independent of whether $\theta < \theta^*$ or $\theta > \theta^*$. To this end, let

$$s = J(\theta) - g(t), \quad (2.6)$$

so that

$$\dot{s} = \frac{dJ}{d\theta} \dot{\theta} - \dot{g}(t)$$

With the optimizing law of

$$\dot{\theta} = k \operatorname{sgn} \sin(2\pi s/\alpha), k > 0 \quad (2.7)$$

as in [43] with α being a small positive constant, a sliding motion occurs for $k|dJ/d\theta| > |\dot{g}(t)|$ and θ is steered towards θ^* while y tracks $g(t)$. The region defined by $|dJ/d\theta| < |\dot{g}(t)|/k$ quantifies the region in which θ will be confined with this optimizing law. The idea can be extended to more general dynamics by adding the derivatives of the

performance function as well as those of $g(t)$ to the sliding manifold expression so as to compensate the relative degree deficit. In [45], this optimization idea has further been developed for online operating point and set point optimization purpose by ending up a two-time scale sliding mode optimization design. The resulting method allows the optimization of the closed loop operation of a system by exploiting the extra degree of freedom in the available control authority possibly in a different time scale, which is exactly what we mean to be sliding mode-based extremum seeking control. We will further see the details of time scale separation in the next section.

2.3 Perturbation-based Extremum Seeking Control

The method of sinusoidal perturbation introduced in this section has been the most popular of extremum seeking control schemes. Due to its continuous way to perform gradient type optimization, it permits fast adaptation and easy implementation. This section is mainly based on Chapter 5 of [3] and later we extend the perturbation-based extremum seeking control to unstable systems by incorporating phase lead compensators in the extremum seeking loop.

2.3.1 Sinusoidal Perturbation-based Analog Optimization

Consider the maximization of a performance function $y = J(\theta)$. A basic sinusoidal perturbation-based continuous maximization method can be found in Figure 2.3.

We want to lay a conceptual foundation for perturbation-based extremum seeking control and develop familiarity with the methods used for analysis and thereby ease understanding of the more intricate analysis in the subsequent sections. Now, consider

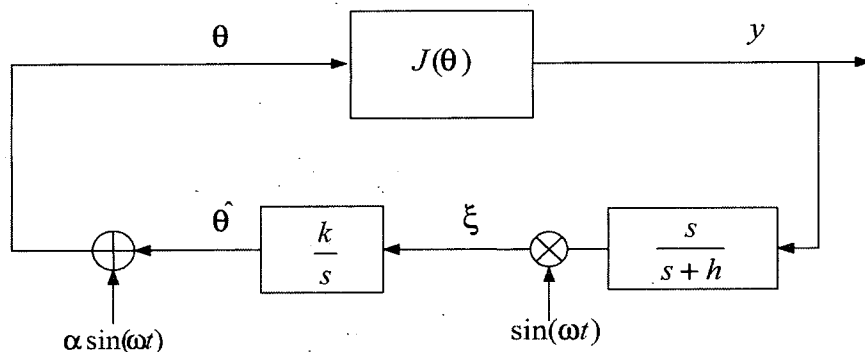


Figure 2.3: Sinusoidal perturbation-based analog optimization.

the performance function

$$J(\theta) = J^* + \frac{J''}{2}(\theta - \theta^*)^2, \quad (2.8)$$

where $J'' < 0$. The global maximizer is θ^* and the global maximum is $J^* = J(\theta^*)$. Any twice continuous differentiable function can be approximated locally by equation (2.8). The assumption of $J'' < 0$ is made without loss of generality, that is, we focus on the maximization problem. We can easily change such maximization method to minimization by replacing k ($k > 0$) with $-k$. The purpose of the algorithm is to make $\theta - \theta^*$ as small as possible, so that the output $J(\theta)$ is driven to its maximum J^* .

The perturbation signal $\alpha \sin(\omega t)$ fed into the function helps to get a measure of the gradient information of $J(\theta)$. We give next an elementary intuitive explanation of how the scheme works with a rigorous analysis to follow subsequently. We start by noting that $\hat{\theta}$ in Figure 2.3 denotes the estimate of the unknown optimal input θ^* .

Let

$$\tilde{\theta} = \theta^* - \hat{\theta}$$

denote the estimation error. Thus,

$$\theta - \theta^* = \alpha \sin \omega t - \tilde{\theta},$$

which, when substituted into Equation (2.8), gives

$$\begin{aligned} y &= J^* + \frac{J''}{2}(\tilde{\theta} - \alpha \sin(\omega t))^2 \\ &= J^* + \frac{J''}{2}\tilde{\theta}^2 - \alpha J''\tilde{\theta} \sin(\omega t) + \frac{\alpha^2 J''}{2} \sin^2(\omega t) \\ &= J^* + \frac{\alpha^2 J''}{4} + \frac{J''}{2}\tilde{\theta}^2 - \alpha J''\tilde{\theta} \sin(\omega t) - \frac{\alpha^2 J''}{4} \cos(2\omega t) \\ &\approx J^* + \frac{\alpha^2 J''}{4} - \alpha J''\tilde{\theta} \sin(\omega t) - \frac{\alpha^2 J''}{4} \cos(2\omega t). \end{aligned} \quad (2.9)$$

We are interested only in local analysis, therefore we can neglect the quadratic term $\frac{J''}{2}\tilde{\theta}^2$. The washout (high pass) filter $\frac{s}{s+h}$ applied to the output, serves to remove $J^* + \frac{\alpha^2 J''}{4}$, namely, $-\alpha J''\tilde{\theta} \sin(\omega t) - \frac{\alpha^2 J''}{4} \cos(2\omega t)$, then the signal after demodulation is

$$\xi = \sin(\omega t) \frac{s}{s+h} [y] \approx -\alpha J''\tilde{\theta} \sin^2(\omega t) - \frac{\alpha^2 J''}{4} \cos(2\omega t) \sin(\omega t),$$

where the notation $\frac{s}{s+h} [y]$ is used to represent the output after the block whose transfer function is $\frac{s}{s+h}$ and input is y . Such notation will be used throughout the dissertation. By using trigonometric identity $2\sin^2(\omega t) = 1 - \cos(2\omega t)$ and $2\cos(2\omega t) \sin(\omega t) = \sin(3\omega t) - \sin(\omega t)$, we arrive at

$$\xi \approx -\frac{\alpha J''}{2}\tilde{\theta} + \frac{\alpha J''}{2}\tilde{\theta} \cos(2\omega t) + \frac{\alpha^2 J''}{8}(\sin(\omega t) - \sin(3\omega t))$$

Note that, because θ^* is a constant and $\tilde{\theta} = \theta^* - \hat{\theta}$, then

$$\dot{\tilde{\theta}} = -\dot{\hat{\theta}},$$

we get

$$\tilde{\theta} = -\frac{k}{s}[\xi] \approx -\frac{k}{s} \left[-\frac{\alpha J''}{2}\tilde{\theta} + \frac{\alpha J''}{2}\tilde{\theta} \cos(2\omega t) + \frac{\alpha^2 J''}{8}(\sin(\omega t) - \sin(3\omega t)) \right]$$

The last two terms are high frequency signals. When passed through an integrator, they get greatly attenuated. Thus, we neglect them, getting

$$\tilde{\theta} \approx -\frac{k}{s} \left[-\frac{\alpha J''}{2}\tilde{\theta} \right]$$

or

$$\dot{\tilde{\theta}} \approx \frac{k\alpha J''}{2}\tilde{\theta} \quad (2.10)$$

Since $k\alpha J'' < 0$, this is a stable system. Thus, we conclude that $\tilde{\theta} \rightarrow 0$, or, in terms of the original problem, $\hat{\theta}(t)$ converges to small distance of θ^* . It is important to note that the approximations hold only when ω is large (in a qualitative sense) relative to k, α, h , and J'' . The following result sums up the properties of the basic perturbation-based extremum seeking loop in Figure 2.3:

THEOREM 2.3.1 *For sufficiently large ω there exists a unique exponentially stable periodic solution of period $2\pi/\omega$ for system in the Figure 2.3 and it satisfies*

$$\left| \tilde{\theta}^{2\pi/\omega}(t) \right| + \left| e^{2\pi/\omega}(t) - \frac{\alpha^2 J''}{4} \right| \leq O\left(\frac{1}{\omega}\right), \forall t \geq 0.$$

This theorem states a local convergence property of the perturbation-based continuous optimization of a single parameter function. The output $y - J^* \rightarrow J'' O\left(\frac{1}{\omega^2} + \alpha^2\right)$, the convergence result is second order and convergence speed is proportional to $1/\omega, \alpha, k, J''$. The rigorous proof is subsumed in a more general result we will cover in the following section.

2.3.2 Perturbation-based Extremum Seeking Control

The perturbation-based extremum seeking control scheme is shown in Figure 2.4. Here, we review the results of the stability analysis appeared in the important paper [30] and later in Chapter 5 of [3], which lays a foundation of the techniques we used to extend the perturbation-based extremum seeking control to moderately unstable system and the autonomous vehicle source seeking problem. Tools of averaging and singular perturbation are employed to show that solutions of the closed loop system converge to a small neighborhood of the extremum of the equilibrium map. The size of the neighborhood is inversely proportional to the adaptation gain and the amplitude and frequency of a periodic signal used to achieve extremum seeking. The low pass

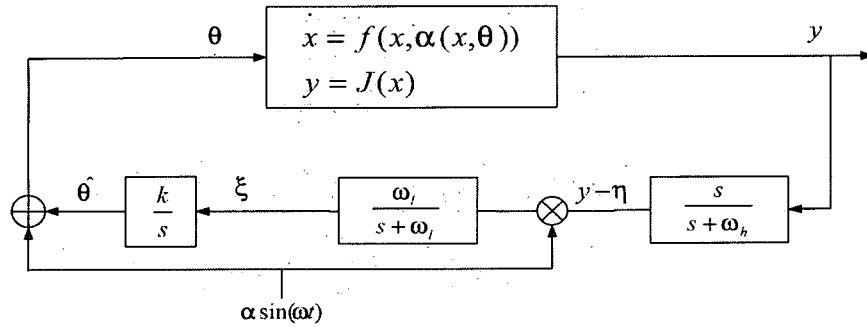


Figure 2.4: Perturbation-based extremum seeking control.

filter $\frac{\omega_l}{s + \omega_l}$, which did not appear in the previous section, is not necessary, but it is helpful in filtering out a $\cos(2\omega t)$ signal after the multiplier (demodulator). The

design parameters are selected as

$$\omega_h = \omega\omega_H = \omega\delta\omega'_H = O(\omega\delta) \quad (2.11)$$

$$\omega_l = \omega\omega_L = \omega\delta\omega'_L = O(\omega\delta) \quad (2.12)$$

$$k = \omega K = \omega\delta K' = O(\omega\delta), \quad (2.13)$$

where ω and δ are small positive constants and ω'_H, ω'_L , and K' are $O(1)$ positive constants. As it will become apparent later, α also needs to be small. From (2.11) and (2.12) we see that the cut-off frequencies of the filters need to be lower than the frequency of the perturbation signal. In addition, the adaptation gain k needs to be small. Thus, the overall feedback system has three time scales:

1. fastest – the plant with the stabilizing controller,
2. medium – the periodic perturbation,
3. slow – the filters in the extremum seeking scheme.

The analysis that follows treats first the static case (“freeze” x at its equilibrium) using the method of averaging. Then we use the singular perturbation method for the full system in Figure 2.4. Let us introduce the new coordinates $\tilde{\theta} = \hat{\theta} - \theta^*$ and $\tilde{\eta} = \eta - J \circ l(\theta^*)$. Then, in the time scale $\tau = \omega t$, the system in Figure 2.4 is written as

$$\omega \frac{dx}{d\tau} = f(x, \alpha(x, \theta^* + \tilde{\theta} + \alpha \sin \tau)) \quad (2.14)$$

$$\frac{d}{d\tau} \begin{bmatrix} \tilde{\theta} \\ \xi \\ \tilde{\eta} \end{bmatrix} = \delta \begin{bmatrix} K' \xi \\ -\omega'_L \xi + \omega'_L (J(x) - J \circ l(\theta^*) - \tilde{\eta}) \alpha \sin \tau \\ -\omega'_H \tilde{\eta} + \omega'_H (J(x) - J \circ l(\theta^*)) \end{bmatrix}. \quad (2.15)$$

The first step in the analysis is to study the system in the steady state. If perturbation $\alpha \sin(\omega t)$ is slow, the plant can be viewed as a static map. That is, we “freeze”

x at its equilibrium value

$$x = l(\theta) = l(\theta^* + \tilde{\theta} + \alpha \sin(\omega t))$$

and substitute it into (2.15), getting the "reduced system"

$$\frac{d}{d\tau} \begin{bmatrix} \tilde{\theta}_r \\ \xi_r \\ \tilde{\eta}_r \end{bmatrix} = \delta \begin{bmatrix} K' \xi_r \\ -\omega'_L \xi_r + \omega'_L (v(\tilde{\theta}_r + \alpha \sin \tau) - \tilde{\eta}_r) \alpha \sin \tau \\ -\omega'_H \tilde{\eta}_r + \omega'_H v(\tilde{\theta}_r + \alpha \sin \tau) \end{bmatrix}, \quad (2.16)$$

where

$$v(\tilde{\theta}_r + \alpha \sin \tau) = J \circ l(\theta^* + \tilde{\theta}_r + \alpha \sin \tau) - J \circ l(\theta^*).$$

The system (2.16) is in the form to which the averaging method [131] is applicable.

The average model of (2.16) is

$$\frac{d}{d\tau} \begin{bmatrix} \tilde{\theta}_r^a \\ \xi_r^a \\ \tilde{\eta}_r^a \end{bmatrix} = \delta \begin{bmatrix} K' \xi_r^a \\ -\omega'_L \xi_r^a + \frac{\omega'_L}{2\pi} \alpha \int_0^{2\pi} v(\tilde{\theta}_r^a + \alpha \sin \tau) \sin \tau d\tau \\ -\omega'_H \tilde{\eta}_r^a + \frac{\omega'_H}{2\pi} \alpha \int_0^{2\pi} v(\tilde{\theta}_r^a + \alpha \sin \tau) d\tau \end{bmatrix}. \quad (2.17)$$

After certain calculations, the equilibrium of the average model (2.17) is

$$\begin{bmatrix} \tilde{\theta}_r^{a,e} \\ \xi_r^{a,e} \\ \tilde{\eta}_r^{a,e} \end{bmatrix} = \begin{bmatrix} -\frac{v'''(0)}{8v''(0)} \alpha^2 + O(\alpha^3) \\ 0 \\ -\frac{v''(0)}{4} \alpha^2 + O(\alpha^3) \end{bmatrix}.$$

The Jacobian of (2.17) at the equilibrium $(\theta_r^{a,e}, \xi_r^{a,e}, \tilde{\eta}_r^{a,e})$ is

$$J_r^a = \delta \begin{bmatrix} 0 & K' & 0 \\ \frac{\omega'_L}{2\pi} \alpha \int_0^{2\pi} v'(\tilde{\theta}_r^{a,e} + \alpha \sin \tau) \sin \tau d\tau & -\omega'_L & 0 \\ \frac{\omega'_H}{2\pi} \alpha \int_0^{2\pi} v'(\tilde{\theta}_r^{a,e} + \alpha \sin \tau) d\tau & 0 & -\omega'_H \end{bmatrix}.$$

More calculations give

$$\det(\lambda I - J_r^a) = \left(\lambda^2 + \delta \omega'_L \lambda - \frac{\delta^2 \omega'_L K'}{2} v''(0) \alpha^2 + O(\delta^2 \alpha^3) \right) (\lambda + \delta \omega'_H),$$

which proves that J_r^α is Hurwitz for sufficiently small α . This in turn, implies the equilibrium of the average system (2.17) is exponentially stable for sufficiently small α . Then, according to the Averaging Theorem in [131] we have the following result.

THEOREM 2.3.2 *Consider the system (2.16) under Assumption 2.0.17. There exist $\bar{\delta}$ and $\bar{\alpha}$ such that for all $\delta \in (0, \bar{\delta})$ and $\alpha \in (0, \bar{\alpha})$ the system (2.0.17) has a unique exponentially stable periodic solution $(\theta_r^{2\pi}(\tau), \xi_r^{2\pi}(\tau), \tilde{\eta}_r^{2\pi}(\tau))$ of period 2π and this solution satisfies*

$$\left\| \begin{bmatrix} \theta_r^{2\pi}(\tau) + \frac{v'''(0)}{8v''(0)}\alpha^2 \\ \xi_r^{2\pi}(\tau) \\ \tilde{\eta}_r^{2\pi}(\tau) + \frac{v''(0)}{4}\alpha^2 \end{bmatrix} \right\| \leq O(\delta) + O(\alpha^3), \forall \tau \geq 0.$$

Now we address the full system in Figure 2.4 whose state space model is given by (2.14) and (2.15) in the time scale $\tau = \omega t$. We simplify (2.15) as

$$\frac{dz}{d\tau} = \delta G(\tau, x, z), \quad (2.18)$$

where $z = (\tilde{\theta}, \xi, \tilde{\eta})$. By Theorem 2.3.2, there exists an exponentially stable periodic solution $z_r^{2\pi}(\tau)$. We shift the state z using the transformation $\tilde{z} = z - z_r^{2\pi}$ and bring the full system (2.14) and (2.15) into the standard singular perturbation form:

$$\frac{d\tilde{z}}{d\tau} = \delta \tilde{G}(\tau, x, \tilde{z}) \quad (2.19)$$

$$\omega \frac{dx}{d\tau} = \tilde{F}(\tau, x, \tilde{z}), \quad (2.20)$$

where

$$\tilde{G}(\tau, x, \tilde{z}) = G(\tau, x, \tilde{z} + z_r^{2\pi}) - G(\tau, L(\tau, z_r^{2\pi}), z_r^{2\pi})$$

$$\tilde{F}(\tau, x, \tilde{z}) = f\left(x, \alpha(x, \theta^* + \tilde{\theta} + \alpha \sin \tau)\right)$$

$$L(\tau, z) = l(\theta^* + \tilde{\theta} + \alpha \sin \tau).$$

We note that $x = L(\tau, \tilde{z} + z_r^{2\pi})$ is the quasi-steady state, and that the reduced model

$$\frac{d\tilde{z}_r}{d\tau} = \delta\tilde{G}(\tau, L(\tau, \tilde{z}_r + z_r^{2\pi}), \tilde{z}_r + z_r^{2\pi}) \quad (2.21)$$

has an equilibrium at the origin $\tilde{z}_r = 0$. And this equilibrium has been shown above to be exponentially stable for sufficiently small α . Moreover, we study the boundary layer model (in the time scale $t = \tau/\omega$):

$$\frac{dx_b}{dt} = \tilde{F}(\tau, x_b + L(\tau, \tilde{z} + z_r^{2\pi}), \tilde{z}) = f(x_b + l(\theta), \alpha(x_b + l(\theta), \theta)). \quad (2.22)$$

Since $f(l(\theta), \alpha(l(\theta), \theta)) = 0$, then x_b is an equilibrium of (2.22). By Assumption 2.0.16, this equilibrium is exponentially stable.

By combining exponential stability of the reduced model (2.21) with the exponential stability of the boundary layer model (2.22), using Tikhonov's Theorem on the Infinite Interval [131], we can conclude that

1. The solution $z(\tau)$ of (2.18) is $O(\omega)$ -close to the solution of $z_r(\tau)$ of (2.21), and therefore, it converges exponentially to an $O(\omega)$ -neighborhood of the periodic solution of $z_r^{2\pi}(\tau)$, which is $O(\delta)$ -close to the equilibrium $z_r^{\alpha, e}$. It follows then that $\theta(\tau) = \theta^* + \tilde{\theta}(\tau) + \alpha \sin \tau$ converges to an $O(\omega + \delta + \alpha)$ -neighborhood of θ^* .
2. The solution $x(\tau)$ of (2.20) satisfies

$$x(\tau) - l(\theta^* + \tilde{\theta}_r(\tau) + \alpha \sin \tau) - x_b(t) = O(\omega),$$

where $\tilde{\theta}_r(\tau)$ is the solution of the reduced system (2.16) and $x_b(t)$ is the solution of the boundary layer model (2.22). So we further obtain

$$x(\tau) - l(\theta^*) = O(\omega) + l(\theta^* + \tilde{\theta}_r(\tau) + \alpha \sin \tau) - l(\theta^*) - x_b(t).$$

Since $\tilde{\theta}_r(\tau)$ converges exponentially to the periodic solution $\tilde{\theta}_r^{2\pi}(\tau)$, which is $O(\delta)$ -close to the average equilibrium $\tilde{\theta}_r^{a,e}$, and since the solution $x_b(t)$ is exponentially decaying, then $x(\tau) - l(\theta)$ exponentially converges to an $O(\omega + \delta + \alpha)$ -neighborhood of zero. Consequently, $y = J(x)$ converges exponentially to an $O(\omega + \delta + \alpha)$ -neighborhood of its maximal equilibrium value $J \circ l(\theta^*)$.

We summarize the above conclusion in the following theorem.

THEOREM 2.3.3 *Consider the feedback system (2.14) and (2.15) under Assumptions 2.0.15–2.0.17. There exists a ball of initial conditions around the point $(x, \hat{\theta}, \xi, \eta) = (l(\theta^*), \theta^*, 0, J \circ l(\theta^*))$ and contains $\bar{\omega}, \bar{\delta}$, and $\bar{\alpha}$ such that for all $\omega \in (0, \bar{\omega}), \delta \in (0, \bar{\delta})$, and $\alpha \in (0, \bar{\alpha})$, the solution $(x(t), \hat{\theta}(t), \xi(t), \eta(t))$ converges exponentially to an $O(\omega + \delta + \alpha)$ -neighborhood of that point. Furthermore, $y(t)$ converges to an $O(\omega + \delta + \alpha)$ -neighborhood of $J \circ l(\theta^*)$.*

2.4 Perturbation-based Extremum Seeking Control for a Plant with Slightly Unstable Poles

The perturbation-based extremum seeking control above and in [3] relies on time scale decomposition and as such has so far been developed only for plants that are open loop stable, with poles that are sufficiently well damped. In current and following sections, we introduce a new idea regarding how to extend the applicability of extremum seeking to moderately unstable systems. The extension to marginally unstable systems draws motivation from the application of autonomous vehicle source seeking and will be presented in Chapter 6.

We present an example of a MIMO plant with slightly unstable poles that can be stabilized, in the absence of its output measurements, with extremum seeking.

Consider a two-input-two-output system

$$\begin{aligned}\dot{x} &= v_x + \epsilon_x x \\ \dot{y} &= v_y + \epsilon_y y,\end{aligned}\tag{2.23}$$

with performance function

$$J = f(x, y) = f^* - q_x(x - x^*)^2 - q_y(y - y^*)^2,\tag{2.24}$$

where $\epsilon_x, \epsilon_y > 0$ are constant and v_x, v_y are the inputs. The (x^*, y^*) is a maximizer, $f^* = f(x^*, y^*)$ is the maximum and q_x, q_y are some unknown positive constants (since the Hessian is negative). General non-quadratic maps with non-diagonal Hessians are equally amenable to analysis, using the same technique as in [3, 30]. A block diagram of extremum seeking is shown in Figure 2.5. If ϵ_x, ϵ_y are very small, the robustness of the perturbation-based extremum seeking loop itself will be able to compensate their effect without resorting to the phase lead compensators.

The analysis that follows employs the method of averaging. Let

$$e = \frac{h}{s + h}[J] - f^*,\tag{2.25}$$

then the signal after the washout filter can be expressed as

$$\frac{s}{s + h}[J] = J - \frac{h}{s + h}[J] = J - f^* - e.$$

Now, let us introduce the new coordinates

$$\tilde{x} = x - x^* - \alpha \sin(\omega t)\tag{2.26}$$

$$\tilde{y} = y - y^* + \alpha \cos(\omega t).\tag{2.27}$$

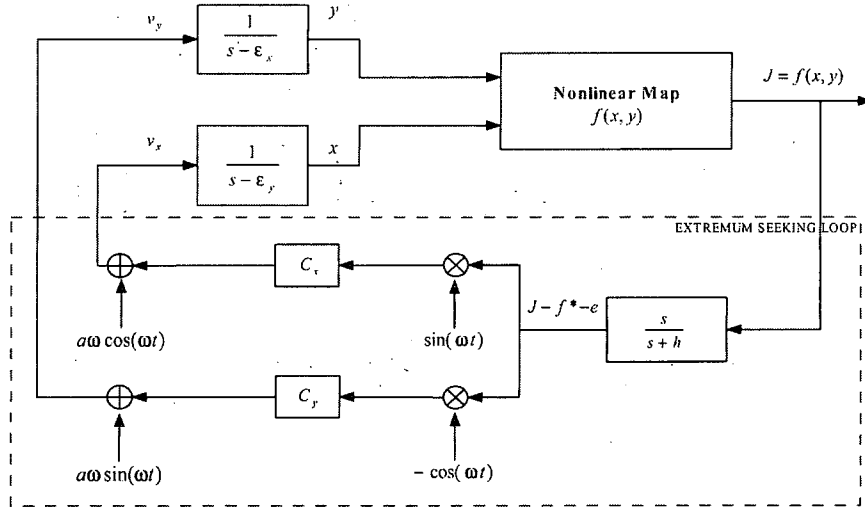


Figure 2.5: Perturbation-based extremum seeking control of a plant with slightly unstable poles.

Then, in the time scale $\tau = \omega t$, we summarize the system in Figure 2.5 as

$$\begin{aligned}
 \frac{d\tilde{x}}{d\tau} &= \frac{1}{\omega} [c_x \Delta \sin \tau + \epsilon_x (\tilde{x} + x^* + a \sin \tau)] \\
 \frac{d\tilde{y}}{d\tau} &= \frac{1}{\omega} [-c_y \Delta \cos \tau + \epsilon_y (\tilde{y} + y^* - a \cos \tau)] \\
 \frac{de}{d\tau} &= \frac{h}{\omega} \Delta,
 \end{aligned} \tag{2.28}$$

where Δ is defined as

$$\Delta = (J - f^* - e) = -[q_x(\tilde{x} + \alpha \sin \tau)^2 + q_y(\tilde{y} - \alpha \cos \tau)^2 + e]. \tag{2.29}$$

The system (2.28) is in the form to which the averaging method is applicable. The averaging model of (2.28) is

$$\begin{aligned}\frac{d\tilde{x}_{avg}}{d\tau} &= \frac{1}{\omega} [(\epsilon_x - \alpha c_x q_x) \tilde{x}_{avg} + \epsilon_x x^*] \\ \frac{d\tilde{y}_{avg}}{d\tau} &= \frac{1}{\omega} [(\epsilon_y - \alpha c_y q_y) \tilde{y}_{avg} + \epsilon_y y^*] \\ \frac{de_{avg}}{d\tau} &= \frac{1}{\omega} (-h) \left[q_x \tilde{x}_{avg}^2 + q_y \tilde{y}_{avg}^2 + e_{avg} + \frac{\alpha^2}{2} (q_x + q_y) \right].\end{aligned}\quad (2.30)$$

Then the equilibrium of the average model (2.30) is

$$\begin{aligned}\tilde{x}_{avg}^e &= \frac{\epsilon_x x^*}{\alpha c_x q_x - \epsilon_x} \\ \tilde{y}_{avg}^e &= \frac{\epsilon_y y^*}{\alpha c_y q_y - \epsilon_y} \\ e_{avg}^e &= -\frac{\alpha^2}{2} (q_x + q_y) - q_x \left(\frac{\epsilon_x x^*}{\alpha c_x q_x - \epsilon_x} \right)^2 - q_y \left(\frac{\epsilon_y y^*}{\alpha c_y q_y - \epsilon_y} \right)^2.\end{aligned}\quad (2.31)$$

The Jacobian of (2.30) at $(\tilde{x}_{avg}^e, \tilde{y}_{avg}^e, e_{avg}^e)$ is

$$J_{avg} = \frac{1}{\omega} \begin{bmatrix} \epsilon_x - \alpha c_x q_x & 0 & 0 \\ 0 & \epsilon_y - \alpha c_y q_y & 0 \\ -2h q_x \tilde{x}_{avg}^e & -2h q_y \tilde{y}_{avg}^e & -h \end{bmatrix}.\quad (2.32)$$

Therefore, J_{avg} will be Hurwitz if and only if

$$\epsilon_x - \alpha q_x c_x > 0, \quad \epsilon_y - \alpha q_y c_y > 0.$$

Given the knowledge that q_x, q_y are some unknown positive constants, and the bound of ϵ_x, ϵ_y , we are able to choose α, c_x, c_y to make the Jacobian (2.32) be Hurwitz, which implies that the equilibrium (2.31) of the average system (2.30) is exponentially stable. Then according to the Averaging Theorem [131], we have the following result.

THEOREM 2.4.1 *Consider the system in the Figure 2.5, where the nonlinear map has the form of (2.24). There exist $\bar{\epsilon}, \bar{\omega}$ such that for all $\epsilon_x, \epsilon_y \in (0, \bar{\epsilon})$ and for all $\frac{1}{\omega} \in$*

$(0, \frac{1}{\omega})$ the system has a unique exponentially stable periodic solution $(\tilde{x}^{2\pi/\omega}, \tilde{y}^{2\pi/\omega}, e^{2\pi/\omega})$ of period $\frac{2\pi}{\omega}$ and this solution satisfies

$$\left\| \begin{bmatrix} \tilde{x}^{2\pi/\omega} - \tilde{x}_{avg}^e \\ \tilde{y}^{2\pi/\omega} - \tilde{y}_{avg}^e \\ e^{2\pi/\omega} - e_{avg}^e \end{bmatrix} \right\| \leq O(1/\omega), \quad \forall \tau \geq 0, \quad (2.33)$$

$$\text{where } \tilde{x}_{avg}^e = \frac{\epsilon_x x^*}{\alpha c_x q_x - \epsilon_x}, \tilde{y}_{avg}^e = \frac{\epsilon_y y^*}{\alpha c_y q_y - \epsilon_y} \text{ and}$$

$$e_{avg}^e = - \left[\frac{\alpha^2}{2} (q_x + q_y) + q_x \left(\frac{\epsilon_x x^*}{\alpha c_x q_x - \epsilon_x} \right)^2 + q_y \left(\frac{\epsilon_y y^*}{\alpha c_y q_y - \epsilon_y} \right)^2 \right].$$

Since

$$x - x^* = \tilde{x} + \alpha \sin(\omega t) = (\tilde{x} - \tilde{x}^{2\pi/\omega}) + \left(\tilde{x}^{2\pi/\omega} - \frac{\epsilon_x x^*}{\alpha c_x q_x - \epsilon_x} \right) + \frac{\epsilon_x x^*}{\alpha c_x q_x - \epsilon_x} + \alpha \sin \tau.$$

The above result (2.33) implies that the first term converges to zero, then second term is $O(1/\omega)$, the third term is $O(\epsilon)$ and the fourth term $O(\alpha)$, then

$$\limsup_{\tau \rightarrow \infty} |x - x^*| = O(\alpha + 1/\omega + \epsilon).$$

Similarly, we can obtain $\limsup_{\tau \rightarrow \infty} |y - y^*| = O(\alpha + 1/\omega + \epsilon)$. Thus, eventually we get

$$\limsup_{\tau \rightarrow \infty} |f - f^*| = O(\alpha^2 + (1/\omega)^2 + \epsilon^2),$$

which characterizes the asymptotic performance of the extremum seeking loop in Figure 2.5, meaning that $(x(t), y(t))$ eventually converge to a neighborhood of the maximum. The size of the neighborhood is proportional to the amplitude of the periodic perturbation, the inverse of the perturbation frequency and the value of the unstable poles.

In the simulation results shown in Figure 2.6, we have $\epsilon_x = \epsilon_y = 0.05$, the perturbation frequency $\omega = 20$, perturbation amplitude $a = 0.05$, adaptation gains $c_x = c_y = 10$ and washout filter $h = 1$. The parameters of the nonlinear map (2.24) are $f^* = 1, q_x = 1$ and $q_y = 0.5$. The start position of the state is $(x(0), y(0)) = (0, 0)$.

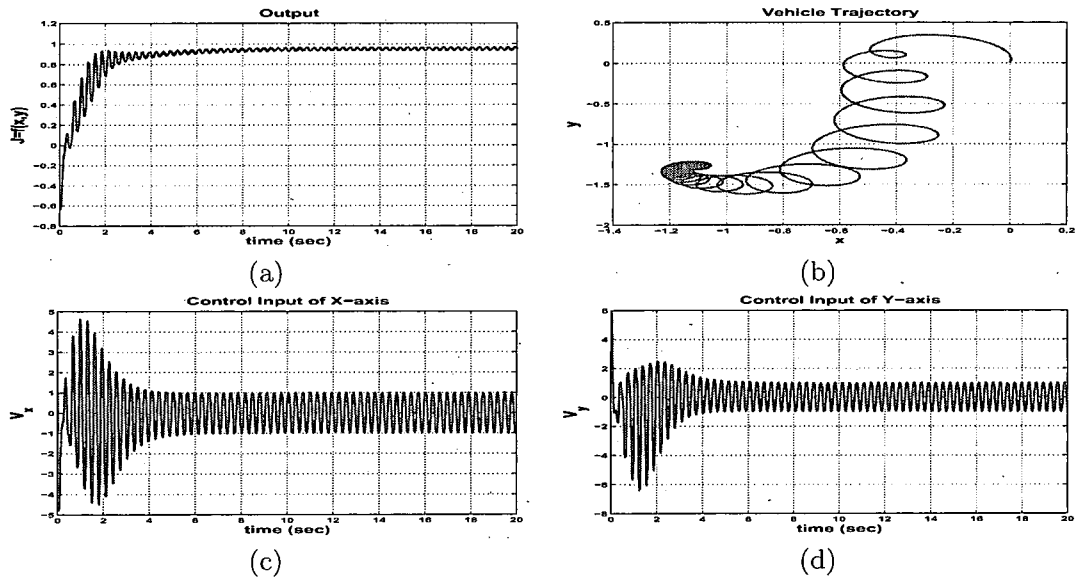


Figure 2.6: Perturbation-based extremum seeking control of a plant with slightly unstable poles: (a) performance output; (b) state; (c) control input of x -axis; (d) control input of y -axis.

2.5 Perturbation-based Extremum Seeking Control for a Plant with Moderately Unstable Poles

If, however, ϵ_x and ϵ_y in (2.23) are not very small but of “medium” size, then the robustness of the extremum seeking loop itself cannot stabilize the system, therefore we include a phase lead compensator to make up for the phase lag introduced by the unstable first order dynamics. Thus, the extremum seeking scheme in Figure 2.7 employs phase lead compensators for achieving robustness against the destabilizing effect of $\epsilon_x, \epsilon_y > 0$.

The transfer function of PD compensator is designed as

$$G(s) = k_c \frac{s - z_0}{s - p_0} \quad (2.34)$$

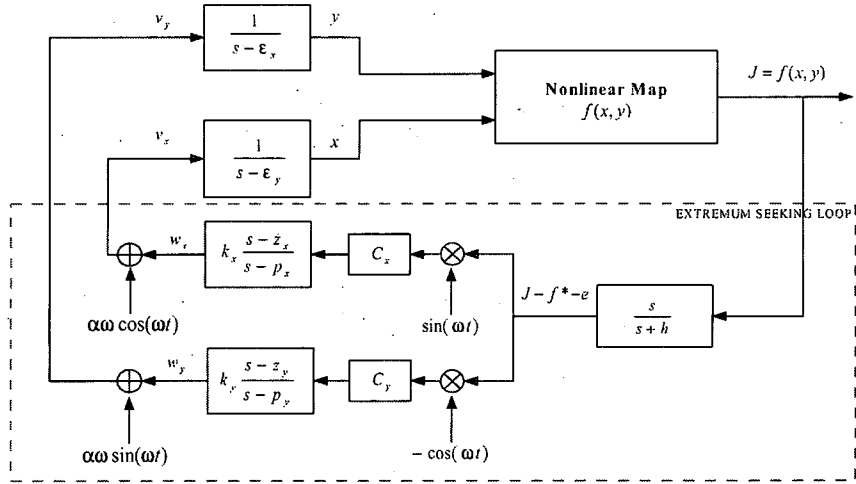


Figure 2.7: Perturbation-based extremum seeking control of a plant with moderately unstable poles.

Then, in the time scale $\tau = \omega t$, we summarize the system in Figure 2.7 as

$$\begin{aligned}
 \frac{d\tilde{x}}{d\tau} &= \frac{1}{\omega} [w_x + \epsilon_x(\tilde{x} + x^* + \alpha \sin \tau)] \\
 \frac{d\tilde{y}}{d\tau} &= \frac{1}{\omega} [w_y + \epsilon_y(\tilde{y} + y^* - \alpha \cos \tau)] \\
 \frac{de}{d\tau} &= \frac{h}{\omega} \Delta \\
 \frac{dw_x}{d\tau} &= \frac{1}{\omega} \left[p_x w_x - c_x k_x z_x \Delta \sin \tau + c_x k_x \omega \Delta \cos(\omega t) + c_x k_x \frac{d\Delta}{dt} \sin(\omega t) \right] \\
 \frac{dw_y}{d\tau} &= \frac{1}{\omega} \left[p_y w_y + c_y k_y z_y \Delta \cos \tau + c_y k_y \omega \Delta \sin(\omega t) - c_y k_y \frac{d\Delta}{dt} \cos(\omega t) \right] \quad (2.35)
 \end{aligned}$$

where Δ is defined in (2.29), and

$$\frac{d\Delta}{dt} = -2q_x(\tilde{x} + \alpha \sin(\omega t)) \left(\frac{d\tilde{x}}{dt} + \alpha \omega \cos(\omega t) \right) - 2q_y(\tilde{y} - \alpha \cos(\omega t)) \left(\frac{d\tilde{y}}{dt} + \alpha \omega \sin(\omega t) \right) - \frac{de}{dt}$$

The average model of (2.35) is

$$\begin{aligned}
\frac{d\tilde{x}_{avg}}{d\tau} &= \frac{1}{\omega} [\epsilon_x(\tilde{x}_{avg} + x^*) + w_{xavg}] \\
\frac{d\tilde{y}_{avg}}{d\tau} &= \frac{1}{\omega} [\epsilon_y(\tilde{y}_{avg} + y^*) + w_{yavg}] \\
\frac{de_{avg}}{d\tau} &= \frac{1}{\omega} (-h) \left[q_x \tilde{x}_{avg}^2 + q_y \tilde{y}_{avg}^2 + e_{avg} + \frac{\alpha^2}{2} (q_x + q_y) \right] \\
\frac{dw_{xavg}}{d\tau} &= \frac{1}{\omega} \left[(p_x - \alpha c_x k_x q_x) w_{xavg} + \alpha c_x k_x q_x (z_x - 2\epsilon_x + h) \tilde{x}_{avg} - \alpha c_x k_x q_x \epsilon_x x^* \right] \\
\frac{dw_{yavg}}{d\tau} &= \frac{1}{\omega} \left[(p_y - \alpha c_y k_y q_y) w_{yavg} + \alpha c_y k_y q_y (z_y - 2\epsilon_y + h) \tilde{y}_{avg} - \alpha c_y k_y q_y \epsilon_y y^* \right].
\end{aligned} \tag{2.36}$$

Then the equilibrium of the average model (2.36) is

$$\begin{aligned}
\tilde{x}_{avg}^e &= \frac{p_x \epsilon_x x^*}{\alpha c_x k_x q_x (z_x - \epsilon_x + h) - p_x \epsilon_x} \\
\tilde{y}_{avg}^e &= \frac{p_y \epsilon_y y^*}{\alpha c_y k_y q_y (z_y - \epsilon_y + h) - p_y \epsilon_y} \\
e_{avg}^e &= -\frac{\alpha^2}{2} (q_x + q_y) - q_x \left(\frac{p_x \epsilon_x x^*}{\alpha c_x k_x q_x (z_x - \epsilon_x + h) - p_x \epsilon_x} \right)^2 \\
&\quad - q_y \left(\frac{p_y \epsilon_y y^*}{\alpha c_y k_y q_y (z_y - \epsilon_y + h) - p_y \epsilon_y} \right)^2 \\
w_{xavg}^e &= \frac{-\alpha c_x k_x q_x \epsilon_x (z_x - \epsilon_x + h) x^*}{\alpha c_x k_x q_x (z_x - \epsilon_x + h) - p_x \epsilon_x} \\
w_{yavg}^e &= \frac{-\alpha c_y k_y q_y \epsilon_y (z_y - \epsilon_y + h) y^*}{\alpha c_y k_y q_y (z_y - \epsilon_y + h) - p_y \epsilon_y}.
\end{aligned} \tag{2.37}$$

The Jacobian of (2.36) at $(\tilde{x}_{avg}^e, w_{xavg}^e, \tilde{y}_{avg}^e, w_{yavg}^e, e_{avg}^e)$ is

$$J_{avg} = \frac{1}{\omega} \begin{bmatrix} \epsilon_x & 1 & 0 & 0 & 0 \\ a_1 & a_2 & 0 & 0 & 0 \\ 0 & 0 & \epsilon_y & 1 & 0 \\ 0 & 0 & b_1 & b_2 & 0 \\ -2hq_x x_{avg}^e & 0 & -2hq_y y_{avg}^e & 0 & -h \end{bmatrix}, \tag{2.38}$$

where $a_1 = \alpha c_x k_x q_x (z_x - 2\epsilon_x + h)$, $a_2 = (p_x - \alpha c_x k_x q_x)$, $b_1 = \alpha c_y k_y q_y (z_y - 2\epsilon_y + h)$ and $b_2 = (p_y - \alpha c_y k_y q_y)$. Therefore, J_{avg} will be Hurwitz if and only if the following

inequalities hold

$$\begin{aligned}
\alpha c_x k_x q_x - \epsilon_x - p_x &> 0 \\
(\alpha c_x k_x q_x + p_x) \epsilon_x - \alpha c_x k_x q_x (z_x + h) &> 0 \\
\alpha c_y k_y q_y - \epsilon_y - p_y &> 0 \\
(\alpha c_y k_y q_y + p_y) \epsilon_y - \alpha c_y k_y q_y (z_y + h) &> 0. \\
h &> 0
\end{aligned} \tag{2.39}$$

If $q_x, q_y \geq \underline{q}$ and $\epsilon_x, \epsilon_y \leq \bar{\epsilon}$, one possible design to satisfy the inequalities (2.39) is

1. Choose $\alpha > 0$ to be small, $h > 0$.
2. Choose $c_x > 0, k_x > 0$ such that $c_x k_x > \frac{\bar{\epsilon}}{2\alpha \underline{q}}$.
3. Choose $p_x = -\alpha c_x k_x \underline{q}$ and $z_x < -h$.
4. Choose $c_y > 0, k_y > 0$ such that $c_y k_y > \frac{\bar{\epsilon}}{2\alpha \underline{q}}$.
5. Choose $p_y = -\alpha c_y k_y \underline{q}$ and $z_y < -h$.

Then according to the averaging theorem [131], we have the following result.

THEOREM 2.5.1 *Consider the system in the Figure 2.7, where the nonlinear map has the form of (2.24). If the conditions (2.39) are satisfied by design, then there exists $\bar{\omega}$ such that for all $\frac{1}{\omega} \in (0, \frac{1}{\bar{\omega}})$ the system has a unique exponentially stable periodic solution $(\tilde{x}^{2\pi/\omega}, \tilde{y}^{2\pi/\omega}, e^{2\pi/\omega})$ of period $\frac{2\pi}{\omega}$ and this solution satisfies*

$$\left\| \begin{bmatrix} \tilde{x}^{2\pi/\omega} - \tilde{x}_{avg}^e \\ \tilde{y}^{2\pi/\omega} - \tilde{y}_{avg}^e \\ e^{2\pi/\omega} - e_{avg}^e \\ w_x^{2\pi/\omega} - w_{x,avg}^e \\ w_y^{2\pi/\omega} - w_{y,avg}^e \end{bmatrix} \right\| \leq O(1/\omega), \quad \forall \tau \geq 0, \tag{2.40}$$

where $(\tilde{x}_{avg}^e, \tilde{y}_{avg}^e, e_{avg}^e, w_{x_{avg}}^e, w_{y_{avg}}^e)$ is the equilibrium (2.37) of the average model (2.36).

Since

$$\begin{aligned} x - x^* &= \tilde{x} + \alpha \sin(\omega t) \\ &= (\tilde{x} - \tilde{x}^{2\pi/\omega}) + \left(\tilde{x}^{2\pi/\omega} - \frac{p_x \epsilon_x x^*}{\alpha c_x k_x q_x (z_x - \epsilon_x + h) - p_x \epsilon_x} \right) \\ &\quad + \frac{p_x \epsilon_x x^*}{\alpha c_x k_x q_x (z_x - \epsilon_x + h) - p_x \epsilon_x} + \alpha \sin \tau, \end{aligned}$$

the above theorem implies that the first term converges to zero, the second term is $O(1/\omega)$, the third term is $O(\epsilon)$ and the fourth term $O(\alpha)$, guaranteeing

$$\limsup_{\tau \rightarrow \infty} |x - x^*| = O(\alpha + 1/\omega + \epsilon).$$

Similarly, we can obtain $\limsup_{\tau \rightarrow \infty} |y - y^*| = O(\alpha + 1/\omega + \epsilon)$. Thus, eventually we get

$$\limsup_{\tau \rightarrow \infty} |f - f^*| = O(\alpha^2 + (1/\omega)^2 + \epsilon^2),$$

which characterizes the asymptotic performance of the extremum seeking loop in Figure 2.7, implying that the system eventually converges to the neighborhood of the maximum. The size of the neighborhood is proportional to the amplitude of the periodic perturbation, the inverse of the perturbation frequency and the value of the unstable poles.

The simulation results are shown in Figure 2.8, where the two unstable poles are $\epsilon_x = \epsilon_y = 0.5$. The parameters of the nonlinear map (2.24) are $f^* = 1, q_x = 1$ and $q_y = 0.5$. The parameters of the PD compensator (2.34) are $k_c = 2, z_0 = -5, p_0 = -1$, the perturbation frequency $\omega = 30$, perturbation amplitude $a = 0.05$, adaptation gain $c_x = c_y = 15$ and washout filter $h = 1$. The system's start position is set to $(0, 0)$.

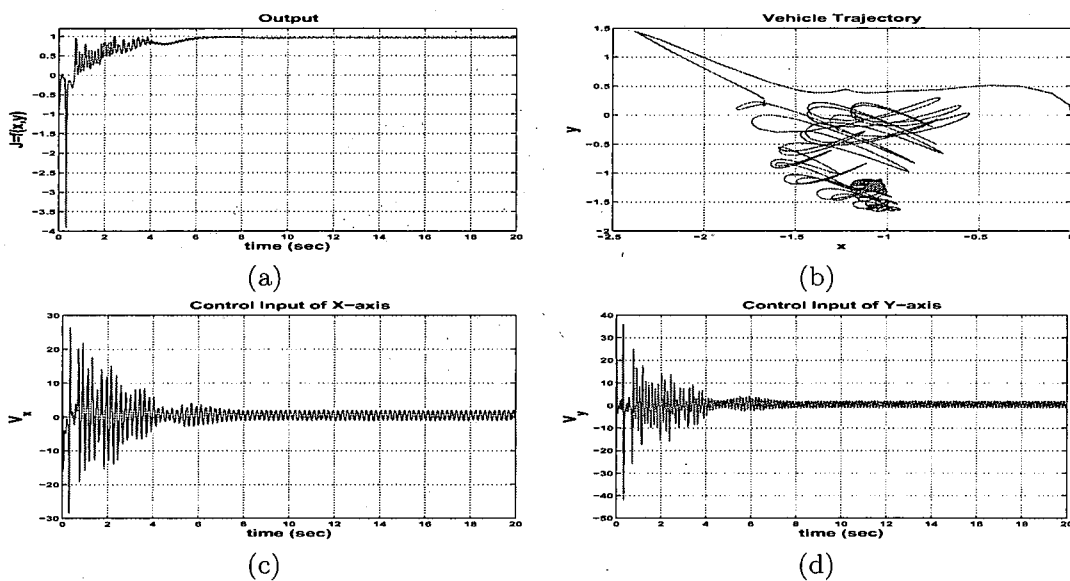


Figure 2.8: Perturbation-based extremum seeking control of a plant with moderately unstable poles: (a) performance output; (b) state; (c) control input of x -axis; (d) control input of y -axis.

CHAPTER 3

NUMERICAL OPTIMIZATION-BASED EXTREMUM SEEKING CONTROL

3.1 Problem Statement

In this chapter, we consider the minimum seeking control design of nonlinear system (2.1) with performance function (2.2). Here, we treat extremum seeking control from the perspective of optimization. Now extremum seeking control can be considered as a kind of constrained optimization problem, whose constraint is the differential equation (2.1) as compared to the traditional algebraic constraints, and the manipulation of x has to be done indirectly through the control input u . The extremum seeking control problem then can be stated as:

$$\min_{x \in \mathbb{R}^n} J(x) \quad \text{subject to} \quad \dot{x} = f(x, u).$$

Now the state x is feasible if it is a solution of the dynamic system. In the case when (2.1) is controllable, there always exists an input u that transfers x to any where in \mathbb{R}^n in a finite time. Although controllable dynamic system constraints do allow x to be anywhere in the state space where the numerical optimizer wants, the way in which x reaches the particular place is determined by the dynamic system and the controller to be implemented.

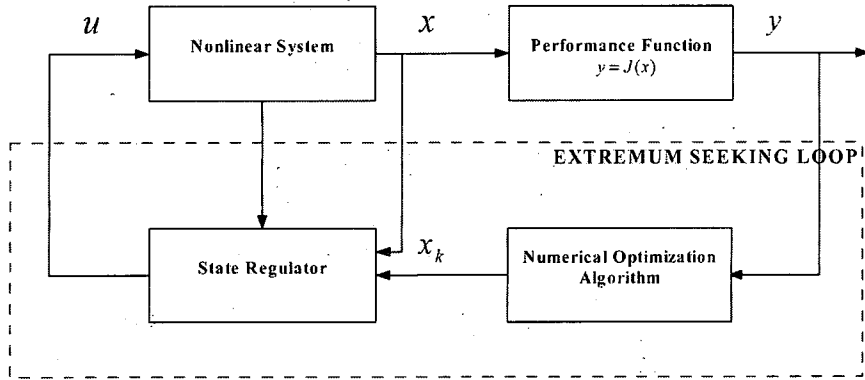


Figure 3.1: Numerical optimization-based extremum seeking control.

A block diagram of numerical optimization-based extremum seeking control is proposed in Figure 3.1, where the nonlinear system is modeled as (2.1) and the performance function is (2.2). The extremum seeking controller (state regulator) ensures that the state x travels along the set point sequence $\{x_k\}$ generated by the numerical optimization algorithm, which eventually converges to a minimizer of the performance function. A basic framework for such extremum seeking control is

Numerical Optimization-based Extremum Seeking Control

Step 0 Given $t_0 = 0$, let $x_0 = x(t_0)$ and $k = 0$.

Step 1 Use an optimization algorithm produce x_{k+1} based on current state $x(t_k)$, the measurements of $J(x(t_k))$, $\nabla J(x(t_k))$ or $\nabla^2 J(x(t_k))$. Denote

$$x_{k+1} = \text{OPTIMIZER}(x(t_k)).$$

Step 2 Design a state regulator u that regulates the state $x(t_k)$ to x_{k+1} in a finite time δ_k , let $t_{k+1} = t_k + \delta_k$.

Step 3 Set $k \leftarrow k + 1$. Go to step 1.

In order to ensure the feasibility of the extremum seeking control design, we need the following assumptions:

ASSUMPTION 3.1.1 (EXISTENCE OF A MINIMUM) *The performance function $J(x)$ is continuous on the compact level sets $L(x_0) = \{x | J(x) \leq J(x_0)\}$ for all x_0 in D .*

ASSUMPTION 3.1.2 *The performance function $J(x)$ is convex on the domain D .*

ASSUMPTION 3.1.3 *The global minimizer $x^* \in \mathbb{R}^n$ of $J(x)$ is an isolated minimizer.*

ASSUMPTION 3.1.4 *The global minimizer $x^* \in \mathbb{R}^n$ of $J(x)$ is a stabilizable equilibrium point of the closed loop system.*

Assumption 3.1.1 is important since it assumes the existence of a minimum, while the remaining assumptions are for the simplicity of analysis. Assumption 3.1.2 means the stationary point condition becomes a sufficient and necessary condition to identify a global minimizer. Optimization algorithms with global convergence² are able to converge to a global minimum x^* ; without such assumption, we just reduce the convergence to a global minimizer to the convergence to a stationary point. Assumptions 3.1.3 and 3.1.4 are required for the applicability of Lyapunov analysis. Assumption 3.1.4 also ensures that there is a controller that can operate the system at the desired set point x^* ; otherwise, we may still be able to design a controller that enables the

²In the optimization literature, first-order convergence of an optimization algorithm means that one (or some, or all) of the limit points of the iterate sequence is a stationary point of $J(x)$. Global convergence is used to mean first-order convergence from an arbitrary starting point. In contrast, local convergence is used to mean convergence when the initial point is close enough to a stationary point.

system to track a reference trajectory containing the minimizer x^* . In the latter case, the steady state performance output will be oscillating. Note that Assumption 3.1.4 is more practical than the assumptions in Chapter 5 of [3] (presented in Assumptions 2.0.15 and 2.0.16 in Chapter 2), where the closed loop system is assumed to have an equilibrium set parameterized by the control argument and extremum seeking is achieved by tuning the single parameter. Therefore, the goal of extremum seeking control is to design a controller based on output measurements and state measurements to regulate the state to an unknown minimizer of an unknown performance function.

3.2 LTI Systems

First, we consider a single input single output (SISO) linear time invariant (LTI) system

$$\dot{x} = Ax + Bu \tag{3.1}$$

with the performance function defined in (2.2), where $x \in \mathbb{R}^n$ is the state, $u \in \mathbb{R}$ is the input. The matrices A, B are given as a model of a real system. However, the explicit form of the performance function $J(x)$ and its minimum are not known, and we can only measure the function value y or its derivatives. We need the following assumption to ensure the feasibility of extremum seeking control for the LTI system (3.1).

ASSUMPTION 3.2.1 *The LTI system (3.1) is controllable.*

First, recall the definition of controllability:

DEFINITION 3.2.2 *The linear system (3.1) or the pair (A, B) is said to be controllable if for any initial state $x(0) = x_0$ and any final state x_1 , there exists an input that transfers x_0 to x_1 in a finite time. Otherwise (3.1) or (A, B) is said to be uncontrollable.*

The following theorem can be found as Theorem 6.1 in [132].

THEOREM 3.2.3 *For LTI system (3.1), if (A, B) is controllable, then for any initial state $x(t_0) = x_0$ and any final state x_1 there exists an input u that transfers x_0 to x_1 in a finite time. For single input LTI system, the input*

$$u(t) = -B^T e^{A^T(t_1-t)} W_c^{-1}(t_1) [e^{A(t_1-t_0)} x_0 - x_1]$$

will transfer x_0 to x_1 at time t_1 , where W_c is the controllability Gramian and can be expressed as

$$W_c(t) = \int_{t_0}^t e^{A(t-\tau)} B B^T e^{A^T(t-\tau)} d\tau = \int_0^{t-t_0} e^{A\tau} B B^T e^{A^T\tau} d\tau.$$

Now, given a controllable LTI system, we can combine a line search method and a state regulator to form an extremum seeking control scheme:

Line Search-based ESC for LTI systems

Step 0 Given $t_0 = 0$, let $x_0 = x(t_0)$ and $k = 0$.

Step 1 Use a line search algorithm with first order convergence to produce x_{k+1} based on current state $x(t_k)$, the measurements of $J(x(t_k))$ and $\nabla J(x(t_k))$. Denote

$$x_{k+1} = x(t_k) + \alpha_k p_k.$$

Step 2 Choose a regulation time δ_k , let $t_{k+1} = t_k + \delta_k$, and design the control input during $t_k \leq t \leq t_{k+1}$ to be

$$u(t) = -B^\top e^{A^\top(t_{k+1}-t)} W_c^{-1}(\delta_k) [e^{A\delta_k} x(t_k) - x_{k+1}], \quad (3.2)$$

where

$$W_c(\delta_k) = \int_0^{\delta_k} e^{A\tau} B B^\top e^{A^\top \tau} d\tau. \quad (3.3)$$

Step 3 Set $k \leftarrow k + 1$. Go to step 1.

In the above scheme, if a line search method with the steepest descent direction is used in the Step 1, we will have $x_{k+1} = x(t_k) - \alpha_k \nabla J(x(t_k))$. Note that the current extremum seeking control only requires gradient information every δ_k time, not a continuously feedback of the gradient measurements. Thus, given enough regulation time δ_k and continuous measurements of the performance output, it is possible to get a good estimation of the gradient using finite difference or other automatic differentiation techniques. We can also use other available optimization algorithms to replace the line search method. Then, the extremum seeking scheme can be derivative free if the optimization algorithm does not require gradient information, such as direct search [120] or derivative free trust region methods [122, 123, 124], which will be shown later. Now, we present the convergence result for the extremum seeking control scheme.

THEOREM 3.2.4 *If the LTI system (3.1) satisfies Assumption 3.2.1 and the performance function (2.2) satisfies Assumptions 3.1.1 – 3.1.4, and ∇J is Lipschitz continuous with constant L ; moreover, if the above line search-based extremum seeking control scheme is applied, then the state x will globally asymptotically converge to the global minimizer x^* of $J(x)$.*

Proof: Given the the LTI system (3.1) is controllable, the controller (3.2) is feasible since $W_c(\delta_k)$ is non-singular. First, at $t = t_0$, the state $x(t_0) = x_0$, we have $x_1 = x_0 - \alpha_0 p_0$. Then, at time $t_1 = t_0 + \delta_0$, we will have

$$x(t_1) = e^{A\delta_0}x_0 + \int_{t_0}^{t_1} e^{A(t_1-\tau)}Bu(\tau)d\tau = x_1,$$

where the control input during (t_0, t_1) is

$$u(t) = -B^\top e^{A^\top(t_1-t)} \left[\int_0^{\delta_0} e^{A\tau}BB^\top e^{A^\top\tau}d\tau \right]^{-1} [e^{A\delta_0}x_0 - x_1].$$

By induction, at $t = t_k$, we suppose the state $x(t_k) = x_k$, then we obtain $x_{k+1} = x_k + \alpha_k p_k$. At time $t_{k+1} = t_k + \delta_k$, we will have

$$x(t_{k+1}) = e^{A\delta_k}x_k + \int_{t_k}^{t_{k+1}} e^{A(t_{k+1}-\tau)}Bu(\tau)d\tau = x_{k+1},$$

where $u(t)$ is defined in (3.2). Thus, the controller (3.2) interpolates between the $\{x_k\}$ precisely within finite time δ_k .

According to Assumptions 3.1.3 and 3.1.4, we suppose that the unknown global minimizer x^* is an equilibrium point. Therefore, let $e = x - x^*$, and we choose the Lyapunov candidate

$$V(e, k) = J(x(t_k)) - J(x^*),$$

which is positive whenever $e \neq 0$ and zero for $e = 0$. Since we know $\{x_k\}$ is a descent sequence converging to x^* as $k \rightarrow \infty$ due to the first order convergence of the line search algorithm, moreover, we have $J(x_{k+1}) < J(x_k)$ from Theorem 3.5.2. Also we know that the controller (3.2) will ensure that the state x crosses x_k and x_{k+1} at time $t = t_k$ and t_{k+1} respectively. Therefore,

$$\begin{aligned} \Delta V &= V(e, k+1) - V(e, k) = J(x(t_{k+1})) - J(x(t_k)) \\ &= J(x_{k+1}) - J(x_k) < 0 \end{aligned}$$

Thus, we will have the $e = 0$ as a globally uniformly asymptotically stable equilibrium point. That is, the performance function achieves its global minimum as $t \rightarrow \infty$ and the state x will globally asymptotically converge to the global minimizer x^* . ■

REMARK 3.2.5 *The convergence result for the extremum seeking control scheme is global since the line search algorithm used is of first order global convergence.*

REMARK 3.2.6 *The design of controller (3.2) is just one way to fulfill the regulation task in a finite time. Other choices of regulator are also possible.*

REMARK 3.2.7 *If the performance function (2.2) is not convex, then the convergence to global minimum will reduce to the convergence to a stationary point.*

COROLLARY 3.2.8 *In addition to the assumptions in Theorem 3.2.4, if J is strongly convex on D , and exact line search with steepest descent direction is used in the extremum seeking control scheme. Then the state x will converge to the ϵ neighborhood of x^* at most at time $t = \sum_{k=1}^N \delta_k$, where $N = \frac{\log((f(x_0) - f(x^*))/\epsilon)}{\log(1/h)}$ for some $0 < h < 1$.*

Proof: If J is strongly convex on \mathbb{R}^n , which means there exist constants $Q, q > 0$ such that [133]

$$qI \leq \nabla^2 f(x) \leq QI$$

for all $x \in \mathbb{R}^n$. And if an exact line search with steepest descent direction is used in the extremum seeking control scheme, letting $h = 1 - q/Q < 1$, we must have $f(x_k) - f(x^*) \leq \epsilon$ after at most

$$N = \frac{\log((f(x_0) - f(x^*))/\epsilon)}{\log(1/h)}$$

iterations of the exact line search with steepest descent direction [133]. Thus, the state x will converge to the ϵ neighborhood of x^* at most at $t = \sum_{k=1}^N \delta_k$. ■

For the purpose of illustration, a second order system and two different performance functions are used in the simulation. Let $x = [x_1, x_2]^\top$, $x_k = [x_1^k, x_2^k]^\top$, and the first performance function is a quadratic function

$$J(x) = 5x_1^2 + x_2^2 + 4x_1x_2 - 14x_1 - 6x_2 + 20. \quad (3.4)$$

It has a global minimizer at $x^* = [1, 1]^\top$ and $J(1, 1) = 10$. Here we postulate the explicit form of the performance function is only for simulation purpose. The explicit form of the performance function and its minimizer are both unknown to the designer. In the extremum seeking control, we only need to access the function value y and its gradient³ for arbitrary $x \in \mathbb{R}^n$. Then, for exact line search with steepest descent direction [134], we can compute the search direction $p_k = -\nabla J(x_1^k, x_2^k) = [-10x_1^k - 4x_2^k + 14, -2x_2^k - 4x_1^k + 6]^\top = [p_1^k, p_2^k]^\top$. Then in this example, we can derive an explicit expression of the optimal step length $\alpha_k = \operatorname{argmin}_\alpha f(x_k + \alpha p_k) = \frac{(p_1^k)^2 + (p_2^k)^2}{2(5(p_1^k)^2 + (p_2^k)^2 + 4p_1^k p_2^k)}$. Note that generally speaking, exact line search is not possible.

The second cost function is the famous Rosenbrock's function (banana function),

$$J(x) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2. \quad (3.5)$$

The objective function $J(x)$ has minimizer $x^* = [1, 1]^\top$, and $J(1, 1) = 0$. Again, we can compute the steepest descent direction $p_k = -\nabla J((x_k)_1, (x_k)_2) = [-2(1 - (x_k)_1) -$

³The gradient information is required since we use steepest descent direction in the extremum seeking scheme. We can use derivative free optimization methods to avoid the requirement of gradient information, and the same analysis of the extremum seeking scheme based on steepest descent can be extended to the derivative free methods.

$400((x_k)_2 - (x_k)_1^2)x_1, 200((x_k)_2 - (x_k)_1)]^\top$. Then an inexact line search satisfying Armijo-Wolfe conditions with initial guess of step length at 1 is used to determine a suitable step length α_k . Such determination of α_k may not be feasible in practical applications due to the requirement of one dimensional search to obtain α_k . However, we can always choose a constant step size, diminishing step size [135] in advance.

Here, consider a second order stable LTI system in its controllable canonical form,

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u, \quad (3.6)$$

The eigenvalues of matrix A are $-1, -2$, and the controllability matrix $[B, AB] = \begin{bmatrix} 0 & 1 \\ 1 & -3 \end{bmatrix}$ is of full rank. That is, the LTI system is stable and controllable. The transition matrix is

$$e^{At} = \begin{bmatrix} -e^{-2t} + 2e^{-t} & -e^{-2t} + e^{-t} \\ 2e^{-2t} - 2e^{-t} & 2e^{-2t} - e^{-t} \end{bmatrix},$$

and the Grammian matrix is

$$\begin{aligned} W_c(t_{k+1}) &= \int_0^{t_{k+1}-t_k} e^{At} B B^\top e^{A^\top t} dt \\ &= \int_0^{\delta_k} \begin{bmatrix} (-e^{-2t} + e^{-t})^2 & (-e^{-2t} + e^{-t})(2e^{-2t} - e^{-t}) \\ (-e^{-2t} + e^{-t})(2e^{-2t} - e^{-t}) & (2e^{-2t} - e^{-t})^2 \end{bmatrix} dt. \end{aligned}$$

Now we apply the extremum seeking control scheme above, where line search method with steepest descent direction is used. The simulation results with the first 15 steps⁴ are shown in Figure 3.2, where $x_0 = [-10, 10]^\top$ for the quadratic function (3.4) and $x_0 = [-1.9, 0]^\top$ for the Rosenbrock's function (3.5), and the regulation time $\delta_k = 2$. The performance output (Figure 3.2 (a_1, a_2)) approaches its minimum at $J(1, 1)$ and the state (Figure 3.2 (b_1, b_2)) accordingly converges to the minimizer $[1, 1]^\top$. The

⁴Since it will take thousands of steps for the line search method with steepest descent direction to converge to the minimizer of the Rosenbrock's function, we only simulate the first 15 steps for illustration purpose.

line search method produces a sequence $\{x_k\}$ as a guideline for the regulator. The trajectory between x_k and x_{k+1} is shaped by the dynamical system (3.1) and the regulator (3.2). This can be viewed clearly in Figure 3.2 (d_1, d_2), where the blue circle represents the $\{x_k\}$ and the red dashed line represents the state trajectory.

The convergence speed to the global minimum depends largely on the performance of the optimization algorithm and the regulation time. Thus, we have seen it is much faster for the system with quadratic function (3.4) than the Rosenbrock's function (3.5). And the choice of δ_k is rather heuristic in this example, which is actually a very important design factor. We can see that the smaller δ_k is, the larger control force we need to fulfill the regulation. Thus, δ_k should be chosen appropriately to not exceed the practical limit on the control force. There is always a tradeoff between the extremum seeking time and the control gain. Moreover, notice that in this example, $[1, 1]^T$ is the global minimizer. And we know that in order to stabilize the system at a reference r we require $Ar = 0$, thus, $r = [1, 1]^T$ does not satisfy Assumption 3.1.4 since there is no controller able to stabilize the system at $[1, 1]^T$. However, the controller will ensure that the state crosses $[1, 1]^T$ every δ_k , that is why we see the state is oscillating in the Figure 3.2 (b_1) and therefore the steady state performance output is oscillating as well.

3.3 State Feedback Linearizable Systems

Now, consider a SISO nonlinear affine system

$$\dot{x} = f_x(x) + g_x(x)u \quad (3.7)$$

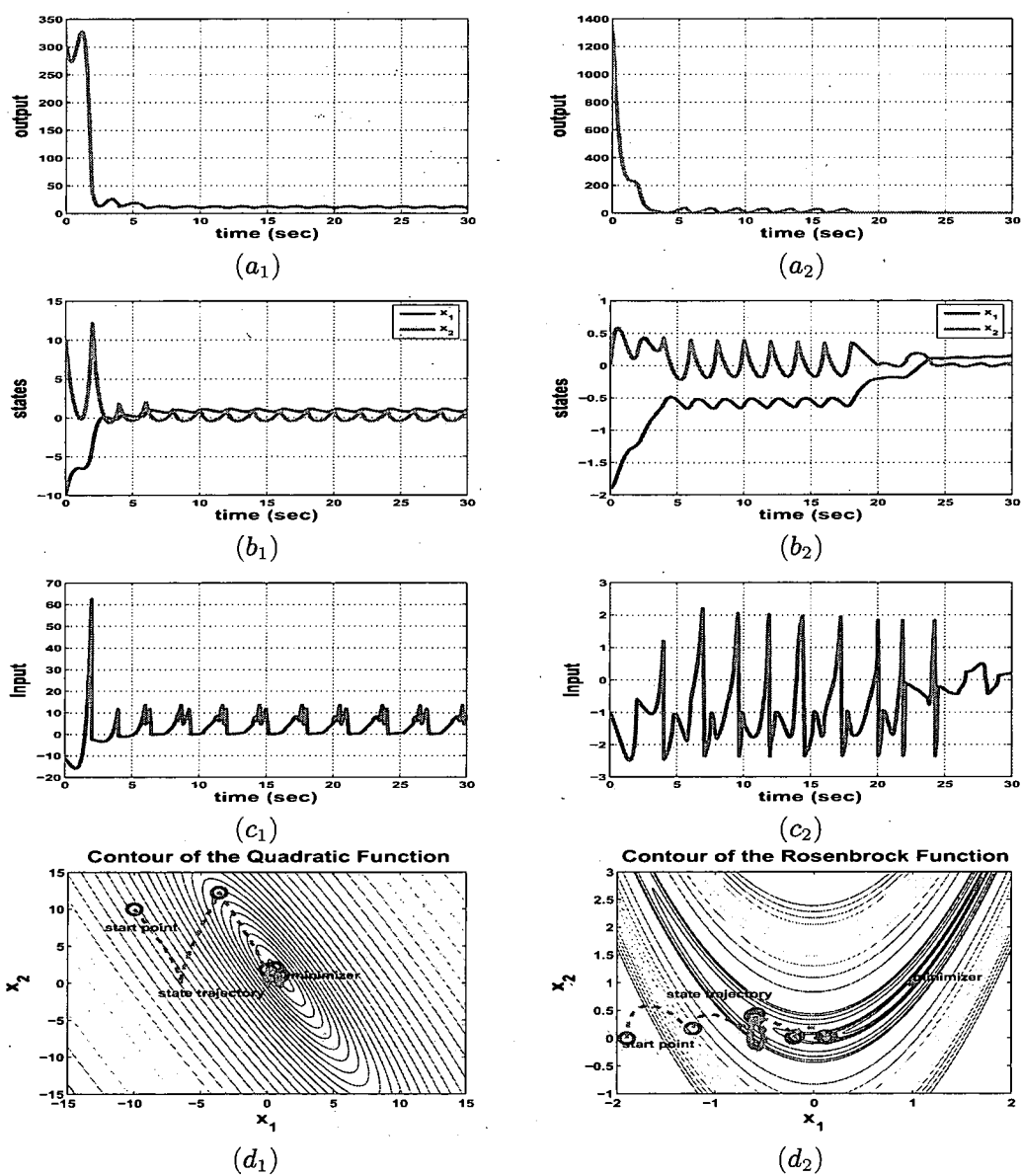


Figure 3.2: Line search-based extremum seeking control for controllable and stable LTI system: (a_1, a_2) performance function; (b_1, b_2) state; (c_1, c_2) control input; (d_1, d_2) phase portrait.

with the performance function defined in (2.2), where $x \in \mathbb{R}^n$ is the state, $u \in \mathbb{R}$ is the input, $f_x, g_x : D \rightarrow \mathbb{R}^n$ are sufficiently smooth on a domain $D \subseteq \mathbb{R}^n$. We have the following assumption for the nonlinear affine system:

ASSUMPTION 3.3.1 *The nonlinear affine system (3.7) is state feedback linearizable on D .*

Upon this assumption, we can always put the system in the controllable canonical form [131]. That is, there exists a diffeomorphism $T : D \rightarrow \mathbb{R}^n$ such that $D_z = T(D)$ contains the origin and the change of variables $z = T(x)$ transforms the system (3.7) into the form

$$\dot{z} = Az + B[f(z) + g(z)u] \quad (3.8)$$

with (A, B) controllable and $g(z) = g(T(x))$ nonsingular for all $x \in D$.

3.3.1 Line Search-based Extremum Seeking Control

We can easily extend the results on LTI systems to outline the line search-based extremum seeking control scheme for state feedback linearizable systems (3.7):

Line Search-based ESC for State Feedback Linearizable Systems

Step 0 Given t_0 , let $x_0 = x(t_0)$ and $k = 0$.

Step 1 Use a line search algorithm with first order convergence to produce x_{k+1} based on current state $x(t_k)$, the measurement of $J(x(t_k))$ and $\nabla J(x(t_k))$. Denote

$$x_{k+1} = x(t_k) + \alpha_k p_k.$$

Step 2 Compute $z_{k+1} = T(x_{k+1})$.

Step 3 Choose regulation time δ_k , let $t_{k+1} = t_k + \delta_k$, and design the control input during $t_k \leq t \leq t_{k+1}$ to be

$$u(t) = g^{-1}(z)(-f(z) + v(t)), \quad (3.9)$$

where

$$v(t) = -B^T e^{A^T(t_{k+1}-t)} W_c^{-1}(\delta_k) [e^{A\delta_k} T(x(t_k)) - z_{k+1}], \quad (3.10)$$

and $W_c(\delta_k)$ is in (3.3).

Step 4 Set $k \leftarrow k + 1$. Go to step 1.

THEOREM 3.3.2 *If the nonlinear affine system (3.7) satisfies Assumption 3.3.1 and the performance function (2.2) satisfies Assumptions 3.1.1 – 3.1.4, and ∇J is Lipschitz continuous with constant L ; moreover, if the above line search-based extremum seeking control scheme is applied, then the state x will globally asymptotically converge to the global minimizer x^* of $J(x)$.*

The proof mainly follows the proof of Theorem 3.2.4. The feasibility of the controller defined in (3.9) is guaranteed by the Assumption 3.3.1. Similar remarks like 3.2.5 – 3.2.7 also apply here.

Consider a second order nonlinear system

$$\begin{aligned} \dot{x}_1 &= -x_1 + x_2, \\ \dot{x}_2 &= x_1 x_2 + u. \end{aligned} \quad (3.11)$$

Now let $z_1 = x_1$, $z_2 = -x_1 + x_2$, the transformed system in the new coordinates (z_1, z_2) is

$$\begin{aligned} \dot{z}_1 &= z_2 \\ \dot{z}_2 &= -(-x_1 + x_2) + x_1 x_2 + u \end{aligned}$$

Then, by choosing the feedback linearizing controller

$$u = (-x_1 + x_2) - x_1x_2 - 2z_1 - 3z_2 = (-x_1 + x_2) - x_1x_2 - 2x_1 - 3(-x_1 + x_2) + v$$

will result in an LTI system as in (3.6). The simulation results can be found in Figures 3.3 and 3.4.

After coordination change and feedback linearization, the nonlinear affine system (3.11) will be the same as the LTI system (3.6) in the z coordinate. Therefore, in order to compare the simulation results of the linear system with the performance functions (3.4) and (3.5) in Section 3.2, we need to ensure performance functions in the z coordinate of the same form as (3.4) and (3.5). Then, we need the quadratic performance function in z coordinate to be

$$J(z) = 5z_1^2 + z_2^2 + 4z_1z_2 - 14z_1 - 6z_2 + 20,$$

and $J(z)$ has its minimizer at $z^* = [1, 1]^\top$ and $J(1, 1) = 10$, exact line search in the z coordinate requires the search direction

$$p_k^z = -\nabla J(z_1^k, z_2^k) = [-10z_1^k - 4z_2^k + 14, -2z_2^k - 4z_1^k + 6]^\top = [p_{k_1}^z, p_{k_2}^z]^\top.$$

and optimal step length

$$\alpha_k^z = \operatorname{argmin}_\alpha f(z_k + \alpha p_k^z) = \frac{(p_{k_1}^z)^2 + (p_{k_2}^z)^2}{2(5(p_{k_1}^z)^2 + (p_{k_2}^z)^2 + 4p_{k_1}^z p_{k_2}^z)}.$$

Now, the quadratic function in x -coordinates becomes,

$$\begin{aligned} J(x) &= 5x_1^2 + (-x_1 + x_2)^2 + 4x_1(-x_1 + x_2) - 14x_1 - 6(-x_1 + x_2) + 20 \\ &= 2x_1^2 + x_2^2 + 2x_1x_2 - 8x_1 - 6x_2 + 20, \end{aligned}$$

which has its minimizer at $x^* = [1, 2]^\top$ and $J(1, 2) = 10$. Exact line search in x -coordinates requires the search direction

$$p_k^x = -\nabla J(x_1^k, x_2^k) = [-4x_1^k - 2x_2^k + 8, -2x_2^k - 2x_1^k + 6]^\top = [p_{k_1}^x, p_{k_2}^x]^\top$$

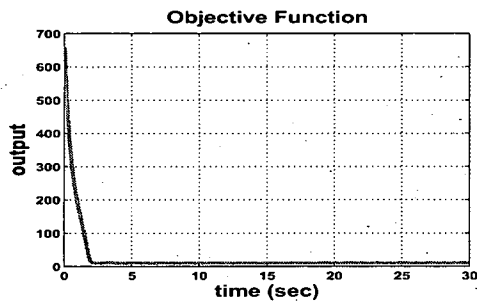
and the optimal step length

$$\alpha_k^x = \operatorname{argmin}_{\alpha} f(x_k + \alpha p_k) = \frac{(p_{k_1}^x)^2 + (p_{k_2}^x)^2}{2(2(p_{k_1}^x)^2 + (p_{k_2}^x)^2 + 2p_{k_1}^x p_{k_2}^x)}.$$

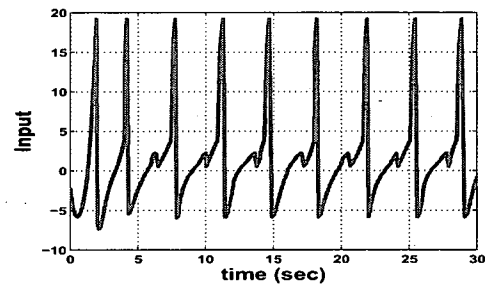
We can verify that even though the transformation between z and x is a diffeomorphism, it is not the same diffeomorphism between the resulting search sequence $z_{k+1} = z_k + \alpha_k^z p_k^z$ and $x_{k+1} = x_k + \alpha_k^x p_k^x$. If given the same guided sequence, the resulting performance as in the previous x -coordinates of the stable LTI system and the z -coordinates of the nonlinear system is the same. However, since the optimization is performed in x -coordinates, after the coordinate transformation, the resulting guided search sequence $\{z_k\}$ for the regulator design is not the same as if the optimization is conducted in the z -coordinates, and therefore we observe different search paths in the simulation. Please compare Figure 3.3 (f) with Figure 3.2 (d_1), and 3.4 (f) with Figure 3.2 (d_2). Moreover, since the minimizer $[1, 2]^T$ cannot be stabilized by any controller, we also observe steady state oscillation in the simulation results in Figure 3.3.

3.3.2 Trust Region-based Extremum Seeking Control

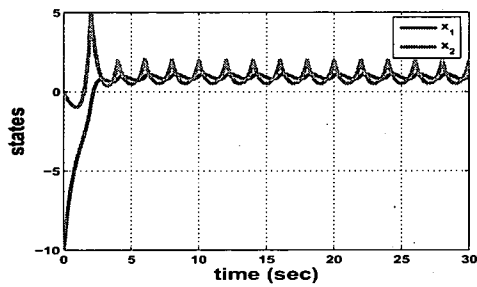
Unlike line search methods, with the derivative free trust region methods [122, 123, 124] the next set point x_{k+1} is based only on the current measurements. The main difficulty of using trust region method is that the ratio ρ_k in (1.6) is required to determine x_{k+1} . Let's call the k^{th} step a successful step if $\rho_k \geq \eta_1$, and unsuccessful step if $\rho_k < \eta_1$. Therefore, in order to obtain ρ_k , we need the measurement of $J(x_k + p_k)$, which means we need to drive the state x to $x_k + p_k$ even if it is not the next iterate x_{k+1} . For example, at the k^{th} step, the controller $u(t)$ needs to regulate x to $x_k + p_k$ to obtain the ratio ρ_k . If $\rho_k \geq \eta_1$, then it is a successful step and we are done. However,



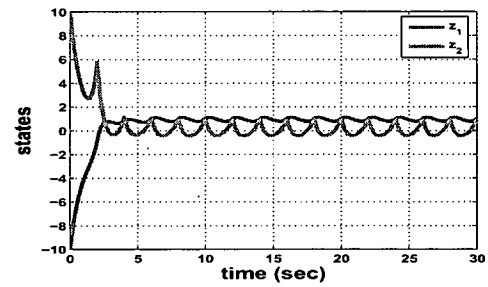
(a)



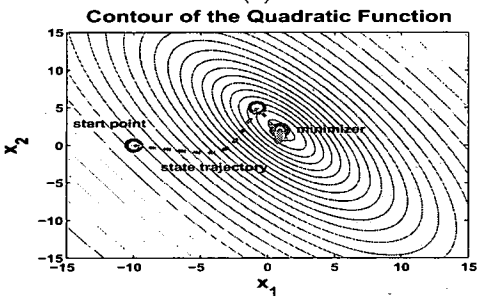
(b)



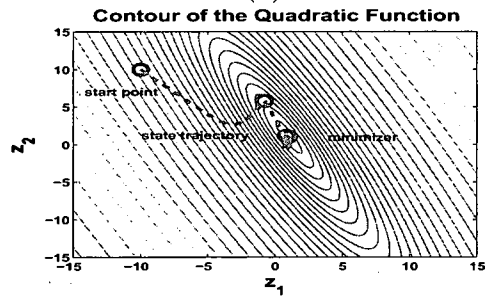
(c)



(d)

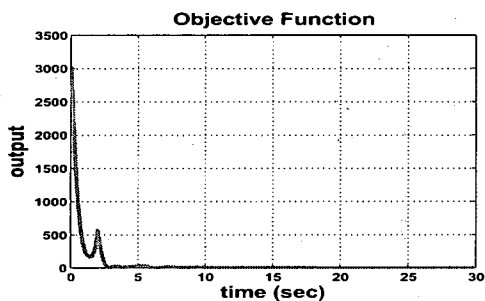


(e)

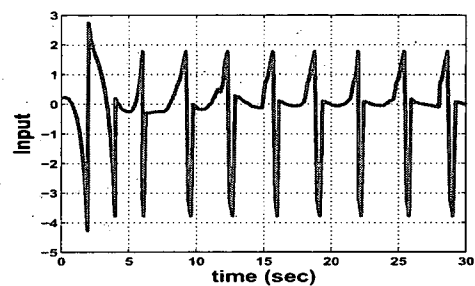


(f)

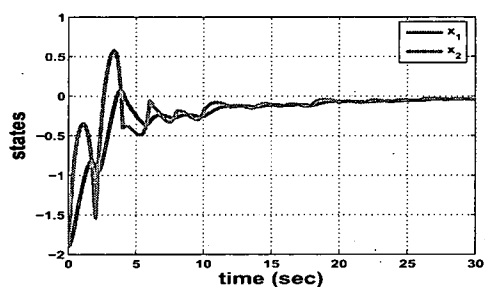
Figure 3.3: Line search-based extremum seeking control for state feedback linearizable nonlinear system with quadratic performance function: (a) performance function; (b) control input; (c) state x ; (d) state z ; (e) phase portrait in x -coordinates; (f) phase portrait in z -coordinates.



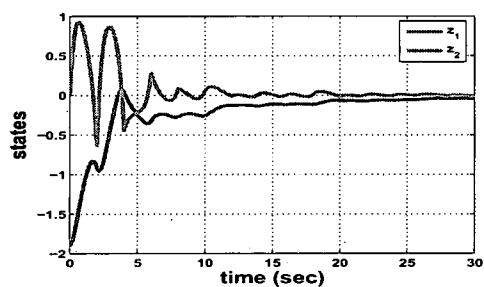
(a)



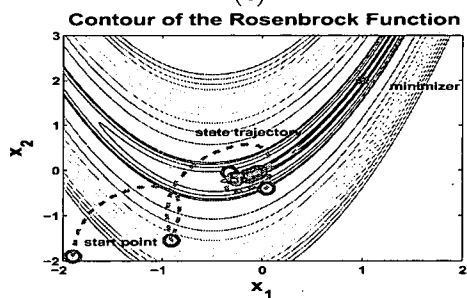
(b)



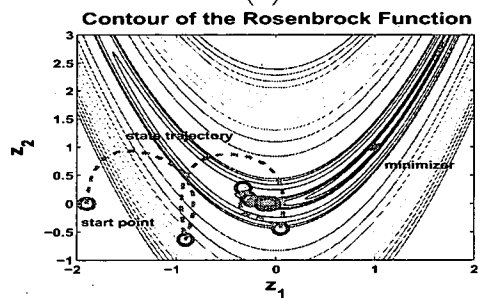
(c)



(d)



(e)



(f)

Figure 3.4: Line search-based extremum seeking control for state feedback linearizable nonlinear system with Rosenbrock's performance function: (a) performance function; (b) control input; (c) state x ; (d) state z ; (e) phase portrait in x -coordinates; (f) phase portrait in z -coordinates.

if $\rho_k < \eta_1$, then we need the controller $u(t)$ to transfer the state back to x_k since it is an unsuccessful step. In particular, we use the derivative free trust region method [123] in the extremum seeking control scheme, where the quadratic approximation m_k is constructed based on the Newton fundamental polynomials and the interpolation set Y containing the interpolation node and corresponding measurements of function values only.

Trust Region-based ESC for State Feedback Linearizable Systems

Step 0 Given $t_0 = 0$, let $x_0 = x(t_0)$, set $\Delta_0 > 0, 0 < \eta_1 \leq \eta_2 < 1, 0 < \gamma_1 \leq \gamma_2 < 1$ and

$k = 0$. Create an initial well poised interpolation set Y containing x_0 and initial basis of Newton fundamental polynomials.

Step 1 Using the interpolation set Y and the basis of Newton fundamental polynomials, build an interpolation model $m_k(x(t_k) + p)$.

Step 2 Approximately minimize $m_k(x(t_k) + p)$ within the trust region $\|p\| \leq \Delta_k$ to obtain p_k .

Step 3 Compute $z_{k+1}^1 = T(x(t_k) + p_k)$.

Step 4 Choose regulation time δ_k^1 , let $t_{k+1}^1 = t_k + \delta_k^1$, and let the control input during $t_k \leq t \leq t_{k+1}^1$ be

$$u(t) = g^{-1}(z)(-f(z) + v(t)), \quad (3.12)$$

where

$$v(t) = -B^\top e^{A^\top(t_{k+1}^1 - t)} W_c^{-1}(\delta_k^1) [e^{A\delta_k^1} T(x(t_k)) - z_{k+1}^1], \quad (3.13)$$

$$W_c(\delta_k^1) = \int_0^{\delta_k^1} e^{A\tau} B B^\top e^{A^\top \tau} d\tau. \quad (3.14)$$

Step 5 At $t = t_{k+1}^1$, measure $J(x(t_{k+1}^1))$, evaluate the ratio

$$\rho_k = \frac{J(x(t_k)) - J(x(t_{k+1}^1))}{m_k(x(t_k)) - m_k(x(t_k) + p_k)}.$$

Step 6 Update the interpolation set Y and the basis of Newton fundamental polynomials.

Step 8 If $\rho_k \geq \eta_1$, let $t_{k+1} = t_{k+1}^1$, $x_{k+1} = x(t_{k+1}^1)$ and go to step 10. Else let $x_{k+1} = x(t_k)$ and $z_{k+1}^2 = T(x(t_k))$.

Step 9 Choose regulation time δ_k^2 , let $t_{k+1} = t_{k+1}^1 + \delta_k^2$, and let the control input during $t_{k+1}^1 \leq t \leq t_{k+1}$ be

$$u(t) = g^{-1}(z)(-f(z) + v(t)), \quad (3.15)$$

where

$$v(t) = -B^\top e^{A^\top(t_{k+1}-t)} W_c^{-1}(\delta_k^2) [e^{A\delta_k^2} T(x(t_{k+1}^1)) - z_{k+1}^2], \quad (3.16)$$

$$W_c(\delta_k^2) = \int_0^{\delta_k^2} e^{A\tau} B B^\top e^{A^\top \tau} d\tau. \quad (3.17)$$

Step 10 Set $k = k + 1$ and go to step 1.

The details of how to update the interpolation set and the basis of Newton fundamental polynomials can be found in [123]. The employment of derivative free trust region algorithm means the extremum seeking control scheme is gradient free. Now, we have the following theorem for the convergence of the trust region-based extremum seeking control:

THEOREM 3.3.3 *If the nonlinear affine system (3.7) satisfies Assumption 3.3.1, and $\|B_k\| \leq \beta$ for some constant β and the performance function (2.2) is Lipschitz continuous differentiable and bounded below on the level set $\{x | J(x) \leq J(x_0)\}$ and satisfies*

Assumptions 3.1.2 – 3.1.4; moreover, if the trust region-based extremum seeking control above is applied, and all approximated solutions p_k satisfy (1.8) and $\|p_k\| \leq \gamma \Delta_k$ for some constant $\gamma \geq 1$, then the state x will globally asymptotically converge to the global minimizer x^* of $J(x)$.

Proof: Given the controller (3.12), the closed loop system (3.7) becomes

$$\dot{z} = Az + B[f(z) + g(z)g^{-1}(z)(-f(z) + v(t))] = Az + Bv(t),$$

where $v(t)$ is defined in (3.13). Now at $t = t_k$, we have the current state $x(t_k) = x_k$, then for time $t = t_{k+1}^1$, we will have

$$\begin{aligned} x(t_{k+1}^1) &= T^{-1}(z(t_{k+1}^1)) \\ &= T^{-1}\left(e^{A\delta_k}T(x(t_k)) + \int_{t_k}^{t_{k+1}^1} e^{A(t_{k+1}^1 - \tau)} Bv(\tau) d\tau\right) \\ &= T^{-1}(z_{k+1}^1) = x(t_k) + p_k = x_k + p_k. \end{aligned}$$

So if we have $\rho_k \geq \eta_1$, we ready to have $x(t_{k+1}) = x(t_{k+1}^1) = x_k + p_k = x_{k+1}$. Otherwise, if we have $\rho_k < \eta_1$, we can obtain at $x(t_{k+1}) = T^{-1}(z_{k+1}^2) = x(t_k) = x_{k+1}$ by applying controller (3.15) and letting $t_{k+1} = t_{k+1}^1 + \delta_k^2$. Also we know that the sequence $\{x_k\}$ is a convergent sequence to the minimizer x^* according to Theorem 1.5.14, and the control input defined in (3.12) and (3.15) regulates the current state $x(t_k)$ to the desired set point x_{k+1} in a finite time.

According to Assumptions 3.1.3 and 3.1.4, we suppose that the unknown global minimizer is an isolated equilibrium point. Therefore, let $e = x - x^*$, we choose the Lyapunov candidate

$$V(e, k) = J(x(t_k)) - J(x^*),$$

which is positive whenever $e \neq 0$ and zero for $e = 0$. Since we know the trust region sequence $\{x_k\}$ will have a descent subsequence $\{x_{s_i}\}$ containing all the successful

steps, i.e., $J(x_{s_{i+1}}) < J(x_{s_i})$, which converges to x^* as $k \rightarrow \infty$; and we know that the controller will ensure that the state x crosses x_{s_i} and $x_{s_{i+1}}$ at times $t = t_i$ and t_{i+1} , respectively, we conclude that

$$\begin{aligned}\Delta V &= V(e, i+1) - V(e, i) = J(x(t_{i+1})) - J(x(t_i)) \\ &= J(x_{s_{i+1}}) - J(x_{s_i}) < 0\end{aligned}$$

Thus, we will have $e = 0$ as a globally uniformly asymptotically stable equilibrium point. That is, the performance function achieves its global minimum as $t \rightarrow \infty$ and the state x will globally asymptotically converge to the global minimizer x^* . ■

Now, we can apply the trust region-based extremum seeking control to the second order nonlinear system (3.11), the simulation results can be found in Figure 3.5.

3.4 Input-Output Feedback Linearizable Systems

Now, we may extend the previous results on state feedback linearizable system to the input-output feedback linearizable system given some minor modifications. We will have the following assumption:

ASSUMPTION 3.4.1 *The nonlinear affine system (3.7) is input-output feedback linearizable from input u to output \bar{y} on D .*

We define a new output

$$\bar{y} = h(x), \tag{3.18}$$

where $h : D \rightarrow \mathbb{R}$ is sufficiently smooth in the domain D . The motivation of defining a new output \bar{y} is to retain the claim that in the extremum seeking control we do not have the knowledge of the performance function, while we do need the knowledge of some suitable output $\bar{y} = h(x)$ to perform input-output linearization. Let d be

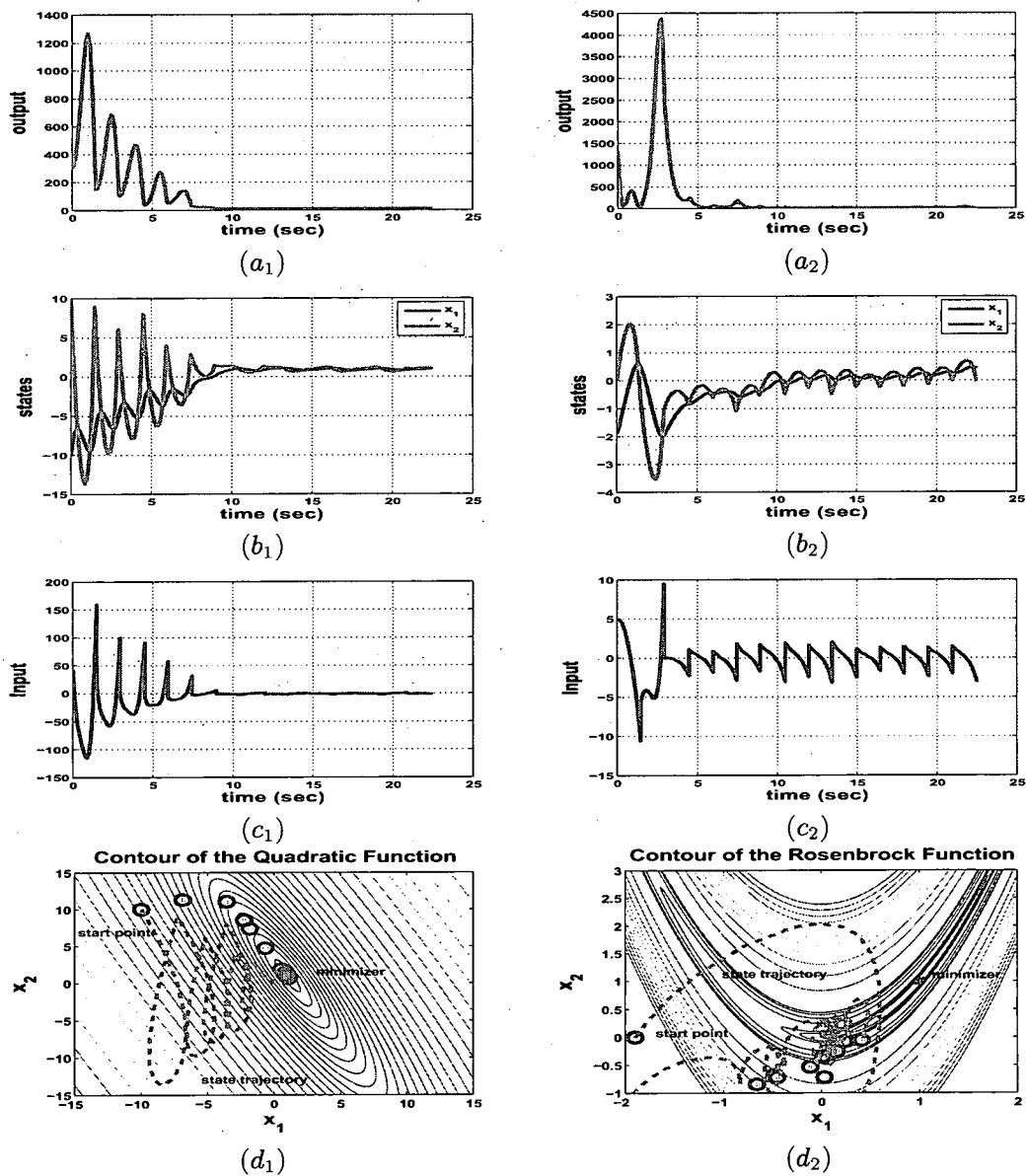


Figure 3.5: Trust region-based extremum seeking control for the feedback linearizable nonlinear system: (a₁, a₂) performance function; (b₁, b₂) state; (c₁, c₂) control input; (d₁, d₂) phase portrait.

the relative degree of the nonlinear system (2.1) with output (3.18), then for every $x_0 \in D$, a neighborhood N of x_0 and a diffeomorphism $T : N \rightarrow \mathbb{R}^n$ exist such that the change of variables $z = [\eta, \xi] = T(x)$, transforms the system (3.18) into the form

$$\dot{\eta} = f_0(\eta, \xi), \quad (3.19)$$

$$\dot{\xi} = A\xi + B[f(z) + g(z)u], \quad (3.20)$$

$$\bar{y} = C\xi, \quad (3.21)$$

where $\xi \in \mathbb{R}^d, \eta \in \mathbb{R}^{n-d}, (A, B)$ is controllable and $g(z) = g(T(x))$ is nonsingular in N . Since η is uncontrollable, therefore, in order to fulfill the extremum seeking of the performance function, two more assumptions are proposed:

ASSUMPTION 3.4.2 The performance function is not dependent on the state η of the internal dynamics.

ASSUMPTION 3.4.3 The state η of the internal dynamics (3.19) will be bounded given bounded ξ and any initial state.

The Assumption 3.4.2 puts the performance function as

$$y = J(x) = J_d(\xi), \quad (3.22)$$

It is a reasonable assumption since variables of the performance function need to be controllable and therefore we can achieve the extremum seeking by tuning them via the control input. Moreover, Assumption 3.4.3 simply means that the internal dynamics are well behaved such that the state regulator will not grow unbounded due to the existence of the uncontrollable state η . Note that assuming input-to-state stability of the internal dynamics is stronger than Assumption 3.4.2. However, simply

assuming the zero dynamics are asymptotically stable is also not enough since the state η may grow unbounded given bounded input ξ . Thus, the same analysis for the state feedback linearizable systems holds here given the extremum seeking schemes in Section 3.3 with minor modifications. For example, we need to change (3.13) to

$$v(t) = -B^T e^{A^T(t_{k+1}^1 - t)} W_c^{-1}(\delta_k^1) [e^{A\delta_k^1} T(\xi(t_k)) - T(\xi(t_k) + p_k)].$$

Then we will have the state ξ asymptotically converge to global minimizer ξ^* of $J_d(\xi)$. The following theorem is a straightforward extension of Theorems 3.3.2 and 3.3.3.

THEOREM 3.4.4 *If the nonlinear affine system (3.7) satisfies Assumptions 3.4.1 and 3.4.3, the performance function (2.2) satisfies Assumptions 3.1.1 – 3.1.4 and 3.4.2; moreover, if extremum seeking schemes of Section 3.3 is applied with z replaced by ξ , then the states ξ will globally asymptotically converge to the global minimizer x^* of $J(x)$.*

The proof mainly follows the proof of Theorem 3.2.4. The feasibility of the controller defined in (3.9) is guaranteed by Assumption 3.4.1 and 3.4.3. Other assumptions for the performance function such as the Lipschitz continuous gradient and approximated solution p_k satisfying inequality (1.8) are needed to guaranteed the convergence of an optimization algorithm used in the extremum seeking control, similar remarks like 3.2.5 – 3.2.7 also apply here. Now, consider a third order nonlinear

system

$$\begin{aligned}
 \dot{x}_1 &= -x_1 + x_2 \\
 \dot{x}_2 &= x_1 x_2 + u \\
 \dot{x}_3 &= -x_3 + x_2^2 \\
 \bar{y} &= x_1
 \end{aligned} \tag{3.23}$$

with performance function (3.4) or (3.5). Now let $q = x_3, \xi_1 = x_1, \xi_2 = -x_1 + x_2$, the transformed system in the new coordinates (q, ξ_1, ξ_2) is

$$\begin{aligned}
 \dot{q} &= -q + (\xi_1 + \xi_2)^2 \\
 \dot{\xi}_1 &= \xi_2 \\
 \dot{\xi}_2 &= -(-x_1 + x_2) + x_1 x_2 + u
 \end{aligned}$$

It is easy to verify the system satisfies Assumptions 3.4.1 and 3.4.3 and the performance function satisfies Assumptions 3.1.1–3.1.4 and 3.4.2. Then, by choosing the feedback linearizing controller

$$u = (-x_1 + x_2) - x_1 x_2 - 2\xi_1 - 3\xi_2 = (-x_1 + x_2) - x_1 x_2 - 2x_1 - 3(-x_1 + x_2)$$

results in an LTI system as seen in (3.6) and therefore same state regulator is used. The simulation starts from $[-10, 10]^\top$ or $[-1.9, 0]^\top$, and regulation time $\delta_k = 2$, line search with steepest descent direction for 25 steps; refer to Figure 3.6. In this example, the global minimizer $[1, 1]^\top$ can be stabilized by a control input $u = -1$; however, by rendering the nonlinear system equivalent to the LTI system (3.6) via the feedback linearizing controller, we cannot stabilize $[1, 1]^\top$ via a controller design for the LTI system. Thus, we still see the steady state performance output $J(x)$

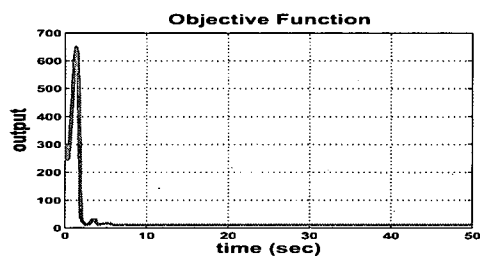
oscillating in Figure 3.6. Of course, we can design a stabilizing controller to render the nonlinear system to an equivalent linear system which can be stabilizable at ξ^* as long as $A\xi^* = 0$; however, since we do not know the minimizer ξ^* , there is no way we can obtain a desired A in advance. Later, we can see a different design of state regulator to ensure that the system will be eventually stabilized at ξ^* .

3.5 Robustness Issues

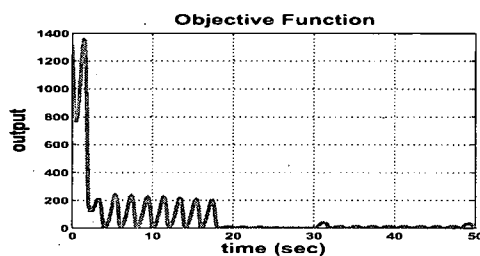
The main restriction of Theorems 3.2.4, 3.3.2, 3.3.3 and 3.4.4 is the requirement of perfect regulation to guarantee the convergence. That is, at each iteration, the controller needs to regulate the state precisely to the desired set point x_{k+1} , which is offered by the iterative optimization algorithm based on the current state $x(t_k)$, function value and gradient measurements. In practical applications, noisy output or state measurements, input disturbance, saturation and time delay, nonlinear system dynamics and computational error will be detrimental to the theoretical result. In fact, in practice we are only able to regulate the state to a neighborhood of the set point x_{k+1} .

For example, let us consider a LTI system with input disturbance. Let $\hat{u}(t) = u(t) + \Delta u(t)$, where $u(t)$ is given as in (3.2), then at time $t = t_0$, the state $x(t_0) = x_0$ and $x_1 = x_0 + \alpha_0 \nabla J(x_0)$, then at time $t_1 = t_0 + \delta_0$, we will have

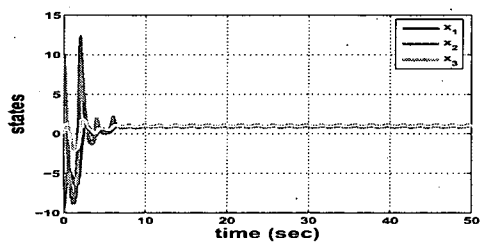
$$\begin{aligned} x(t_1) &= e^{A\delta_0}x_0 + \int_{t_0}^{t_1} e^{A(t_1-\tau)}B\hat{u}(\tau)d\tau \\ &= e^{A\delta_0}x_0 + \int_{t_0}^{t_1} e^{A(t_1-\tau)}Bu(\tau)d\tau + \int_{t_0}^{t_1} e^{A(t_1-\tau)}B\Delta u(\tau)d\tau \\ &= x_1 + e_1, \end{aligned}$$



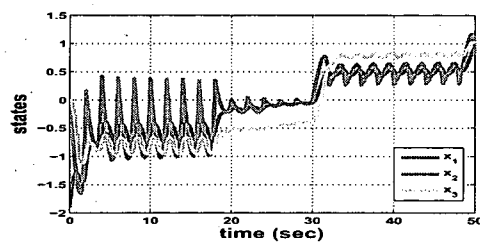
(a₁)



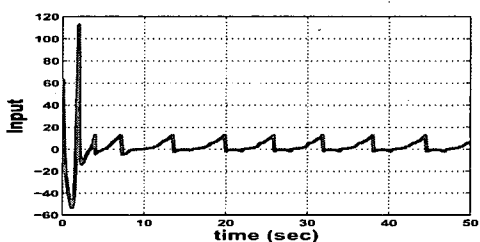
(a₂)



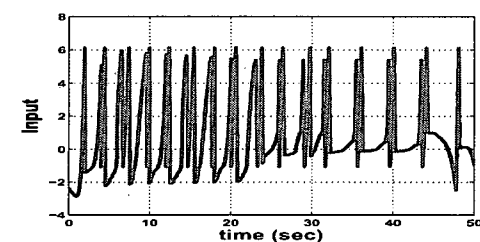
(b₁)



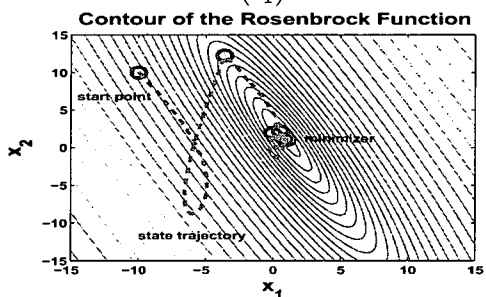
(b₂)



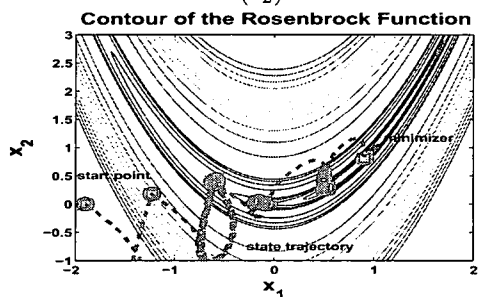
(c₁)



(c₂)



(d₁)



(d₂)

Figure 3.6: Line search-based extremum seeking control for input-output feedback linearizable nonlinear system: (a₁, a₂) performance function; (b₁, b₂) state; (c₁, c₂) control input; (d₁, d₂) phase portrait.

where $e_1 = \int_{t_0}^{t_1} e^{A(t_1-\tau)} B \Delta u(\tau) d\tau$. The error will not accumulate since the optimization algorithm will generate the next destination based on the current state $x(t_1)$. That is, by including the numerical optimizer in the extremum seeking loop, it offers a mechanism to robustify the extremum seeking scheme. Denote $\hat{x}_0 = x(t_0) = x_0$, $\hat{x}_1 = x(t_1) = x_1 + e_1$, then by induction, we will have that the controller interpolates precisely between the sequence $\{\hat{x}_k\}$, where

$$\begin{aligned}\hat{x}_{k+1} &= x(t_{k+1}) = x_{k+1} + e_{k+1}, \\ x_{k+1} &= \hat{x}_k + \alpha_k p_k, \\ e_{k+1} &= \int_{t_k}^{t_{k+1}} e^{A(t_{k+1}-\tau)} B \Delta u(\tau) d\tau.\end{aligned}$$

e_{k+1} will be bounded if δ_k is bounded and the input disturbance is bounded. Moreover, for stable systems, we will have matrix A with negative eigenvalues and therefore the transition matrix has exponentials with negative real parts. Then, e_{k+1} will asymptotically converge to some constant as t_{k+1} approaches infinity, that is, even if δ_k is unbounded, we will still have a bounded e_{k+1} given bounded input disturbance. Moreover, the more negative the eigenvalues are, the smaller e_{k+1} will be. On the contrary, for unstable systems, the transition matrix will amplify even a small input disturbance Δu , and e_{k+1} will grow unbounded as δ_k increases. Therefore, we wish to have a stable LTI system and a short regulation time. The latter amounts to the need of high gain controller to deal with disturbance. In case the LTI system is controllable, we can always perform pole placement to transform the unstable LTI system to a desired stable LTI system, and then apply the extremum seeking control on the stabilized LTI system. We will illustrate these ideas in the simulation shown later.

Similarly, for state feedback linearizable systems to deal with unknown plant dynamics, given the knowledge of A, B which are in the controllable canonical form, the controller will be implementing functions $\hat{f}(z)$ and $\hat{g}(z)$, approximations of $f(z)$ and $g(z)$, that is,

$$u(t) = \hat{g}^{-1}(z)(-\hat{f}(z) + v(t)),$$

where $v(t)$ is defined in (3.10). Then the closed loop systems becomes

$$\dot{z} = Az + B[f(z) + g(z)\hat{g}^{-1}(z)(-\hat{f}(z) + v(t))] = Az + B[v(t) + \Delta v(t)],$$

where $\Delta v(t) = -g^{-1}(z)(-f(z) + v(t)) + \hat{g}^{-1}(z)(-\hat{f}(z) + v)$. Thus, the closed loop system appears to have an input disturbance. The controller will not be able to drive the current state $x(t_k)$ to the desired the set point x_{k+1} at time $t = t_{k+1}$:

$$x(t_{k+1}) = T^{-1}(z(t_{k+1})) = T^{-1}\left(z_{k+1} + \int_{t_k}^{t_{k+1}} e^{A(t_{k+1}-\tau)} B \Delta v(\tau) d\tau\right) = x_{k+1} + e_{k+1},$$

where e_{k+1} is the remainder of Taylor series expansion at z_{k+1} . And e_{k+1} is bounded given bounded disturbance since $\int_{t_k}^{t_{k+1}} e^{A(t_{k+1}-\tau)} B \Delta v(\tau) d\tau$ is bounded and T^{-1} is a diffeomorphism.

Now, recall the LTI system (3.6), a random disturbance uniformly distributed with amplitude 1 is introduced to the input. The simulation results are shown in Figure 3.7 with $x_0 = [-1.9, 0]^\top$, $\delta_k = 2$. The controller (3.2) is not able to regulate the state to the desired set point precisely. For example, at the first step with the Rosenbrock's function (3.5), the controller cannot transfer the state to the desired destination $x_1 = [-1.2288, 0.1763]^\top$, instead it arrives at $\hat{x}_1 = [-1.0809, 0.2216]^\top$ due to the input disturbance. Then $x_2 = \hat{x}_1 - \alpha_k \nabla J(\hat{x}_1) = [-0.6771, 0.4065]^\top$, that is, the line search method tries to amend the deviated path towards the minimum. Again, the state only arrives at $\hat{x}_2 = [-0.5289, 0.4529]^\top$, therefore, eventually we will still

have a descent sequence $\{\hat{x}_k\}$ as long as the error e_k satisfies a bound, shown later in (3.24). The comparison of $\{x_k\}$ and $\{\hat{x}_k\}$ can be seen in Figure 3.7 (d_1, d_2), where the blue circle represents the $\{x_k\}$, magenta square denotes the $\{\hat{x}_k\}$ and the red dashed line is the state trajectory. Interestingly, we find that the disturbance actually helps the algorithm to achieve more reduction in function values in the first 15 steps by comparing with the ideal case (refer to Figure 3.2 (d_2) and Figure 3.7 (d_2)). The performance output again approaches its minimum but with a longer oscillation due to the disturbance, as shown in to Figure 3.7 (a_1).

Further we can see the effect of regulation time for the regulation error e_{k+1} , we have $\delta_k = 2$, $x_0 = [-1.9, 0]^\top$, the amplitude of random disturbance is 1, $\delta_k = 0.5$ and 5. The simulation results can be found in Figure 3.8. Thus, the state of the stable LTI system converges to a neighborhood of the global minimizer given certain amount of input disturbance. However, this is not the case for unstable LTI systems. Consider a second order unstable LTI system in its controllable canonical form.

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ -2 & 3 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$$

The eigenvalues of matrix A are $-1, 2$. That is, the LTI system is controllable but unstable. The transition matrix is given by

$$e^{At} = \begin{bmatrix} -e^{2t} + 2e^t & e^{2t} - e^t \\ -2e^{2t} + 2e^t & 2e^{2t} - e^t \end{bmatrix},$$

and the Grammian matrix is

$$\begin{aligned} W_c(t_{k+1}) &= \int_0^{t_{k+1}-t_k} e^{At} B B^\top e^{A^\top t} dt \\ &= \int_0^{\delta_k} \begin{bmatrix} (e^{2t} - e^t)^2 & (e^{2t} - e^t)(2e^{2t} - e^t) \\ (2e^{2t} - e^t)(e^{2t} - e^t) & (2e^{2t} - e^t)^2 \end{bmatrix} dt. \end{aligned}$$

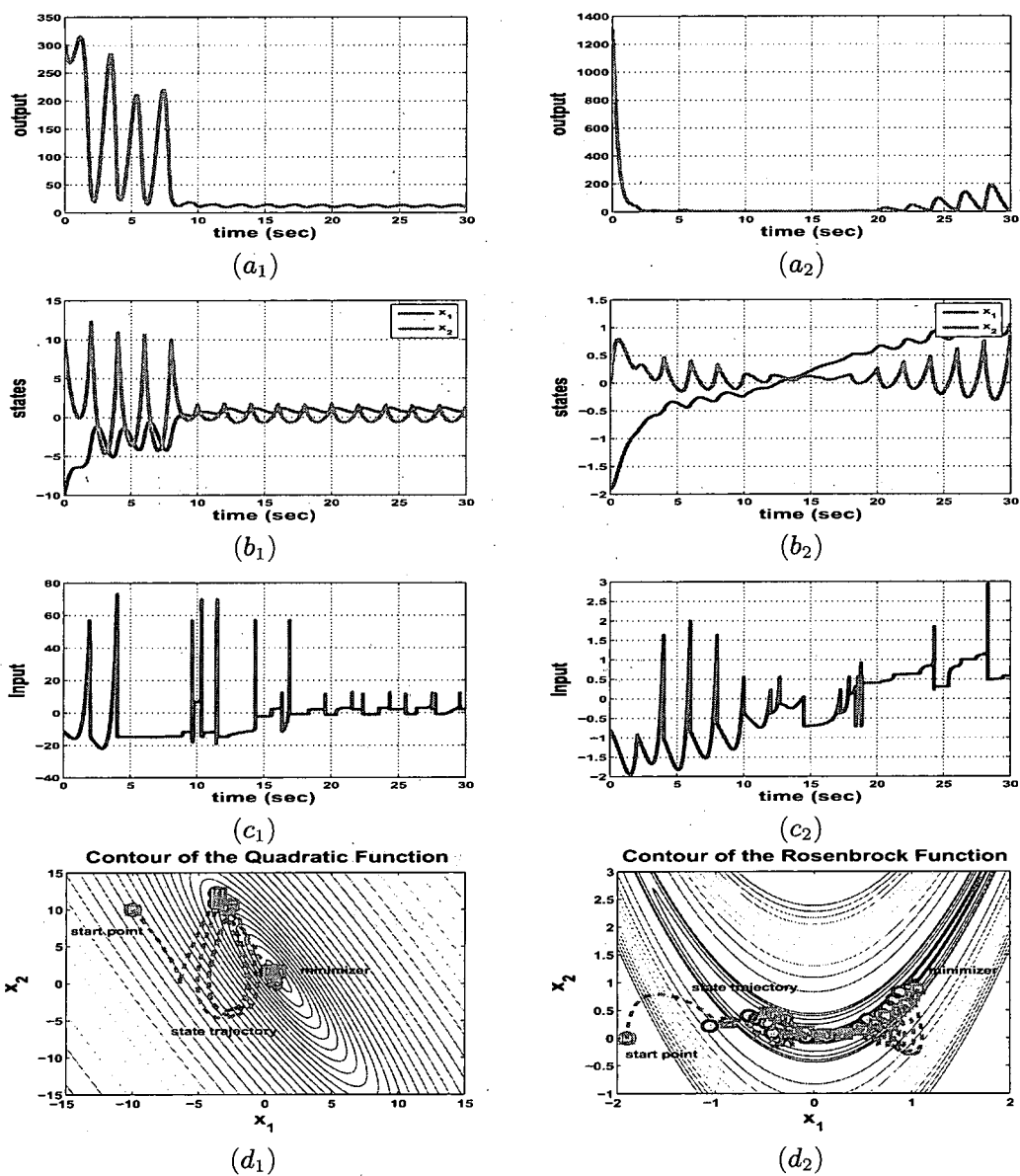


Figure 3.7: Line search-based extremum seeking control for controllable and stable LTI system with input disturbance: (a_1, a_2) performance function; (b_1, b_2) state; (c_1, c_2) control input; (d_1, d_2) phase portrait.

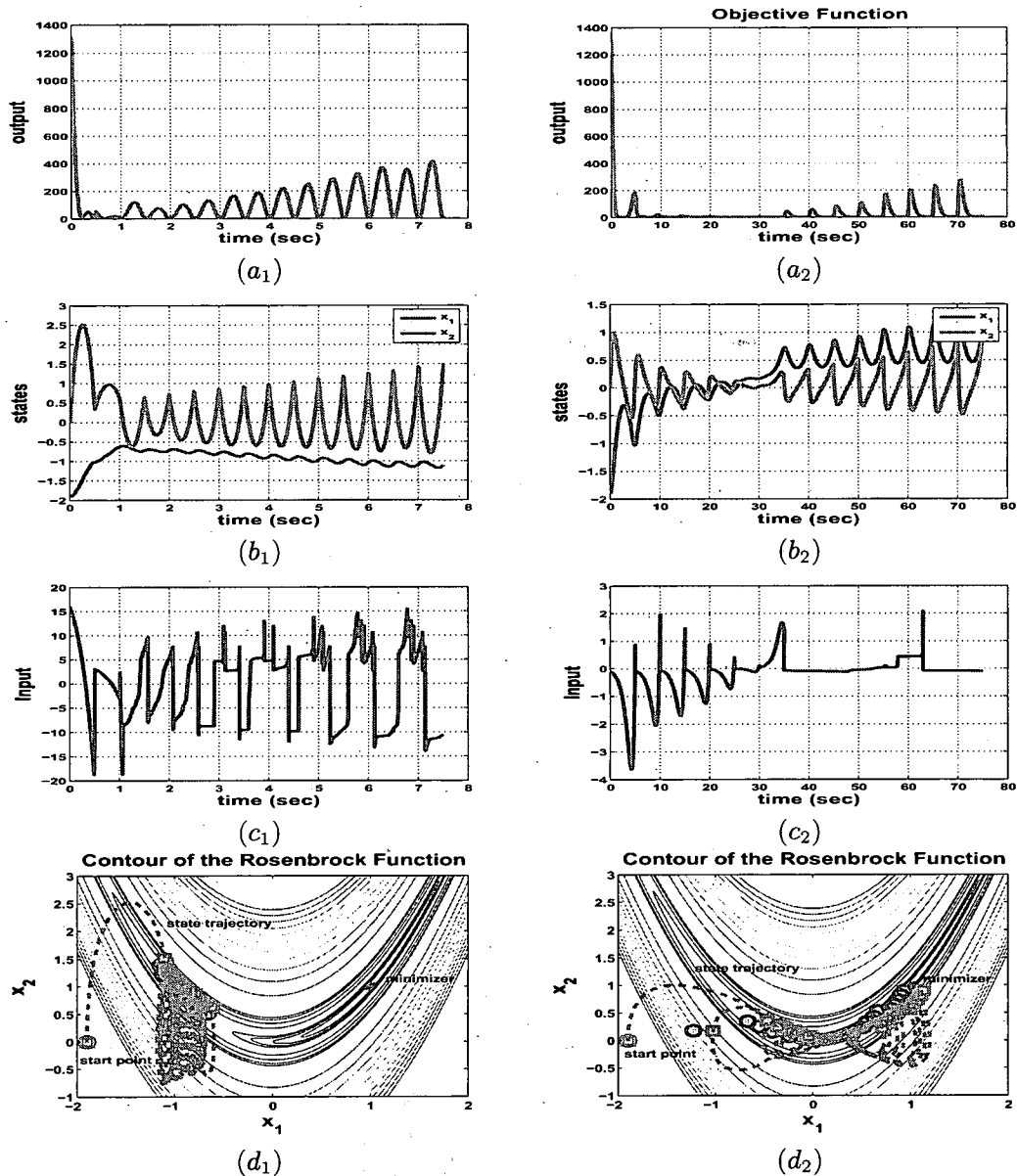


Figure 3.8: Extremum seeking control for controllable and stable LTI system with input disturbance, given different regulation time: (a_1, a_2) performance function; (b_1, b_2) state; (c_1, c_2) control input; (d_1, d_2) phase portrait.

Now we apply the line search-based extremum seeking control scheme with steepest descent direction. Given $\delta_k = 2$, $x_0 = [-10, 10]^\top$, the amplitude of the disturbance is 0.02. The simulation results are shown in Figure 3.9. And we can see the effect of regulation time in amplifying the disturbance in the case of unstable LTI system. We change the $\delta_k = 0.5$ and $\delta_k = 3$, the results are shown in Figure 3.10. Furthermore, we apply a pole placement controller to stabilize the unstable system, and the new poles are placed at $(-1, -2)$. Then the simulation results with $\delta_k = 2$, $x_0 = [-1.9, 0]^\top$, and the amplitude of the disturbance is 0.02 are shown in Figure 3.11.

Overall, we will hope that a well designed optimization algorithm will convey its robustness to the extremum seeking control schemes. That is, if the new sequence $\{\hat{x}_k\}$ is able to converge to the minimum or its small neighborhood given the error e_k is bounded, then the extremum seeking control schemes will enable the state converge to the minimum or its small neighborhood as well. In the following, we will present the robustness analysis of optimization algorithms.

3.5.1 Robustness of Line Search Methods

The following lemmas will be used in the robustness analysis for line search methods.

LEMMA 3.5.1 (DESCENT LEMMA [126]) *Let $J : \mathbb{R}^n \rightarrow \mathbb{R}$ be continuously differentiable on \mathbb{R}^n . And suppose that ∇J is Lipschitz continuous with constant L . Then for $x, y \in \mathbb{R}^n$,*

$$J(x + y) \leq J(x) + y^\top \nabla J(x) + \frac{L}{2} \|y\|^2.$$

LEMMA 3.5.2 *Let $J : \mathbb{R}^n \rightarrow \mathbb{R}$ be continuously differentiable on \mathbb{R}^n . And suppose that ∇J is Lipschitz continuous with constant L . Let α_k, p_k be the step length and*

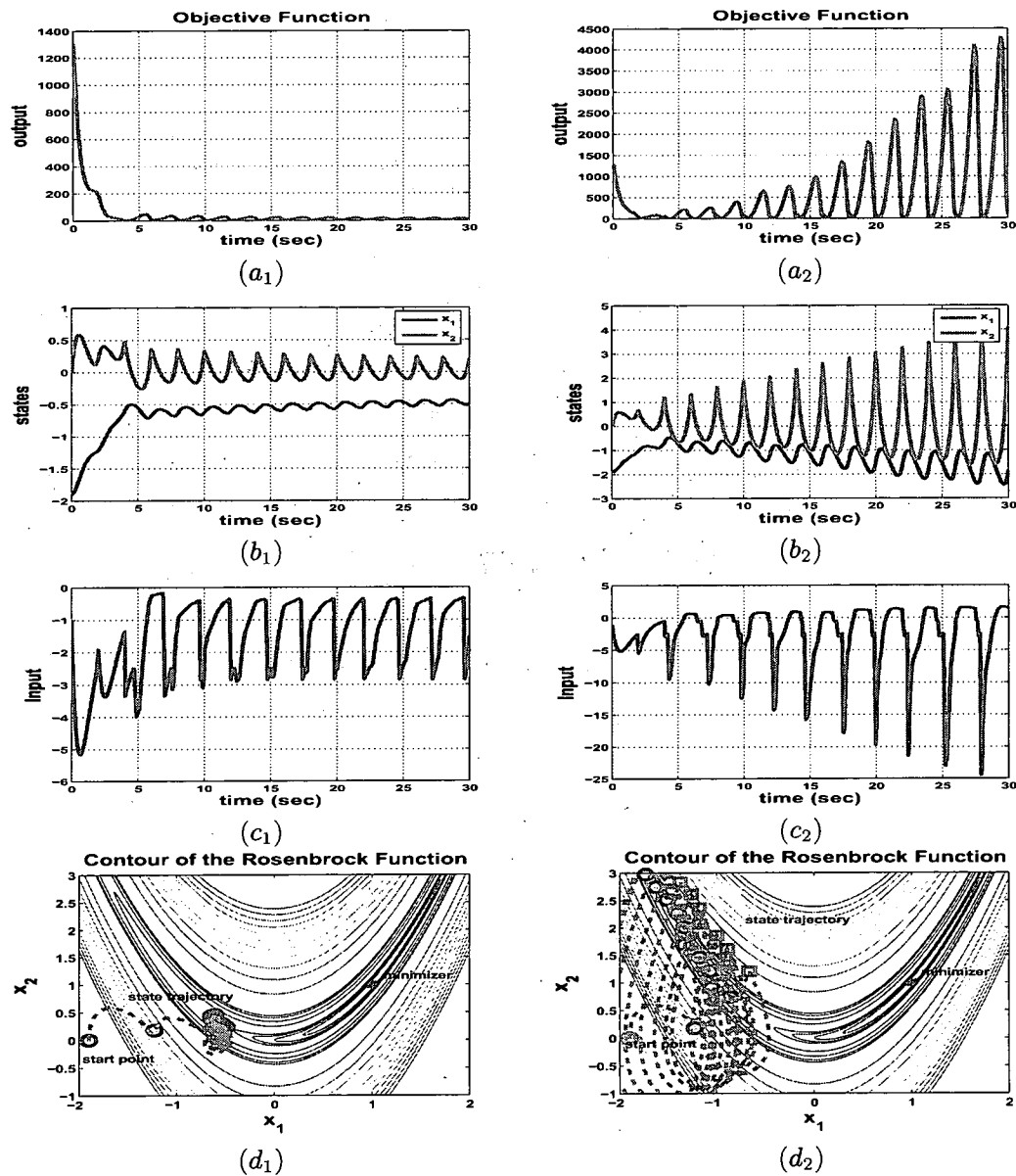


Figure 3.9: Extremum seeking control for controllable but unstable LTI system without and with input disturbance: (a₁, a₂) performance function; (b₁, b₂) state; (c₁, c₂) control input; (d₁, d₂) phase portrait.

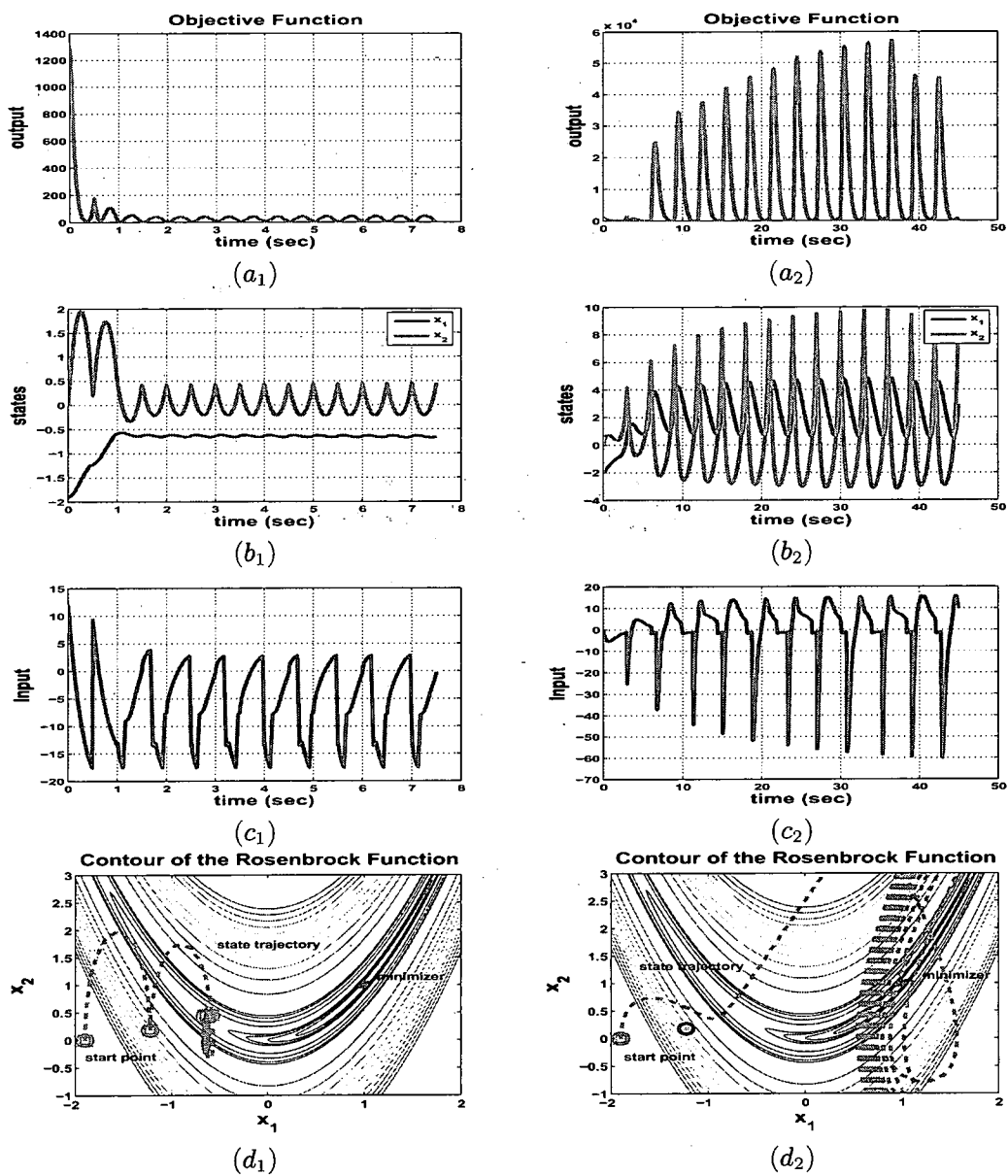


Figure 3.10: Extremum seeking control for controllable but unstable LTI system with input disturbance, given different regulation time: (a_1, a_2) performance function; (b_1, b_2) state; (c_1, c_2) control input; (d_1, d_2) phase portrait.

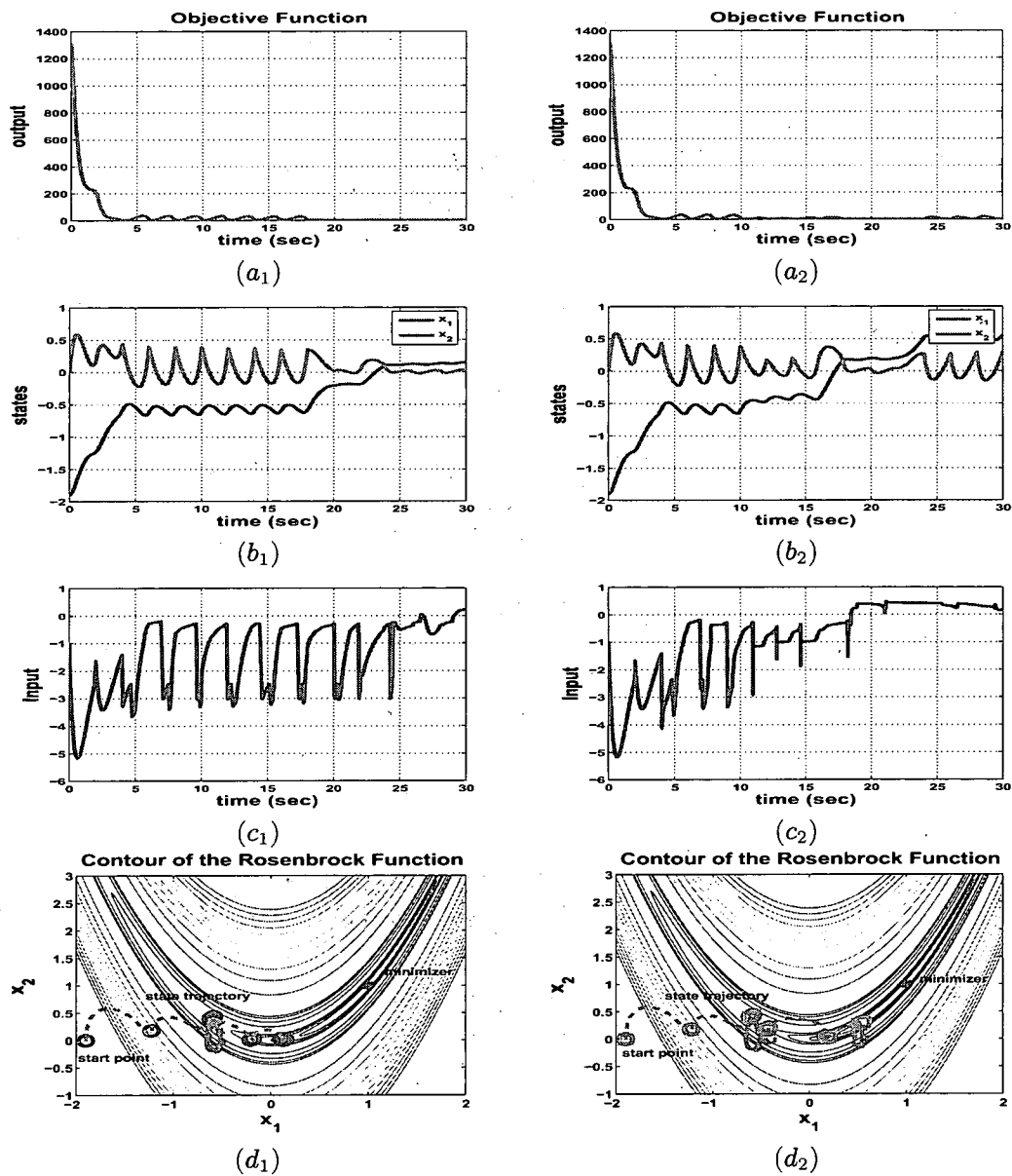


Figure 3.11: Extremum seeking control for controllable but unstable LTI system with and without input disturbance, with stabilized pole placement controller: (a_1, a_2) performance function; (b_1, b_2) state; (c_1, c_2) control input; (d_1, d_2) phase portrait.

descent direction, then

$$J(x_k + \alpha_k p_k) - J(x_k) \leq -\frac{c}{2L} \|\nabla J(x_k)\|^2 \cos^2 \theta_k,$$

where $c = 1$ for exact line search, and $c = 2c_1(1 - c_2)$ for inexact line search satisfying conditions (1.3) and (1.4), θ_k represents the angle between vector p_k and $-\nabla J(x_k)$.

Proof: First, for exact line search, α_k is the solution of Equation (1.2). From the Descent Lemma 3.5.1, we have $J(x_k + \alpha p_k) \leq J(x_k) + \alpha p_k^\top \nabla J(x_k) + \frac{\alpha^2}{2} L \|p_k\|^2$ valid for all $\alpha > 0$. Letting $\bar{\alpha} = -\frac{p_k^\top \nabla J(x_k)}{L \|p_k\|^2} > 0$, then

$$\begin{aligned} & J(x_k + \alpha_k p_k) - J(x_k) \\ & \leq J(x_k + \bar{\alpha} p_k) - J(x_k) \\ & \leq \bar{\alpha} p_k^\top \nabla J(x_k) + \frac{\bar{\alpha}^2}{2} L \|p_k\|^2 \\ & = -\frac{(p_k^\top \nabla J(x_k))^2}{L \|p_k\|^2} + \frac{L \|p_k\|^2}{2} \frac{(p_k^\top \nabla J(x_k))^2}{(L \|p_k\|^2)^2} \\ & = -\frac{1}{2L} \|\nabla J(x_k)\|^2 \cos^2 \theta_k. \end{aligned}$$

Second, for inexact line search, α_k satisfies conditions (1.3) and (1.4). From Lipschitz condition, we have $p_k^\top [\nabla J(x_k + \alpha_k p_k) - \nabla J(x_k)] \leq \|p_k\| \|\nabla J(x_k + \alpha_k p_k) - \nabla J(x_k)\| \leq \alpha_k L \|p_k\|^2$. Then from (1.4), we have $-\alpha_k L \|p_k\|^2 \leq p_k^\top [\nabla J(x_k) - \nabla J(x_k + \alpha_k p_k)] \leq (1 - c_2) p_k^\top \nabla J(x_k)$. That is, $-\alpha_k \|p_k\| \leq -\frac{1 - c_2}{L} \|\nabla J(x_k)\| \cos \theta_k$. Finally from (1.3),

$$\begin{aligned} J(x_k + \alpha_k p_k) - J(x_k) & \leq c_1 \alpha_k p_k^\top \nabla J(x_k) \\ & = -c_1 \alpha_k \|p_k\| \|\nabla J(x_k)\| \cos \theta_k \\ & \leq -\frac{c}{2L} \|\nabla J(x_k)\|^2 \cos^2 \theta_k. \end{aligned}$$

■

Since $0 < c_1 < c_2 < 1$ is required to ensure the feasibility of inexact line search, we will have $c = 2c_1(1 - c_2) < 1$. This observation is consistent for the upper bound results in the above lemma. That is, we always expect the exact line search to have more decrease along the search direction than the inexact line search.

For line search-based extremum seeking control, the control input with disturbance or unknown plant dynamics will result in the state trajectory interpolating among the sequence $\{\hat{x}_k\}$, where $\hat{x}_0 = x_0, \hat{x}_{k+1} = \hat{x}_k + \alpha_k p_k + e_{k+1}$.

THEOREM 3.5.3 *Suppose the performance function $J(x)$ satisfies Assumption 3.1.1 and ∇J is Lipschitz continuous with constant L . A line search algorithm starting from $\hat{x}_0 = x_0$ is used but with bounded error⁵ at each iteration, i.e., $\hat{x}_{k+1} = \hat{x}_k + \alpha_k p_k + e_{k+1}$. The new sequence $\{\hat{x}_k\}$ is a descent sequence, that is*

$$J(\hat{x}_{k+1}) < J(\hat{x}_k),$$

given

$$\|e_{k+1}\| \leq \frac{(c\|\nabla J(\hat{x}_k)\|^2 \cos^2 \theta_k)/L}{\sqrt{\|\nabla J(x_{k+1})\|^2 + c\|\nabla J(\hat{x}_k)\|^2 \cos^2 \theta_k} + \|\nabla J(x_{k+1})\|}, \quad (3.24)$$

where $c = 1$ for exact line search and $c = 2c_1(1 - c_2)$ for inexact line search satisfying conditions (1.3) and (1.4).

⁵due to the inability of the controller to precisely regulate to $\hat{x}_k + \alpha_k p_k$.

Proof: Now, for line search method at step $k+1$ and from Lemmas 3.5.1 and 3.5.2 we have

$$\begin{aligned}
J(\hat{x}_{k+1}) - J(\hat{x}_k) &= J(x_{k+1} + e_{k+1}) - J(\hat{x}_k) \\
&\leq J(x_{k+1}) + \nabla J(x_{k+1})^\top e_{k+1} + \frac{L}{2} \|e_{k+1}\|^2 - J(\hat{x}_k), \\
&\leq \nabla J(x_{k+1})^\top e_{k+1} + \frac{L}{2} \|e_{k+1}\|^2 - \frac{c}{2L} \|\nabla J(\hat{x}_k)\|^2 \cos^2 \theta_k \\
&\leq \frac{L}{2} \left[\|e_{k+1}\|^2 + 2 \frac{\|\nabla J(x_{k+1})\|}{L} \|e_{k+1}\| - \frac{c}{L^2} \|\nabla J(\hat{x}_k)\|^2 \cos^2 \theta_k \right] \\
&= \frac{L}{2} \left[\left(\|e_{k+1}\| + \frac{\|\nabla J(x_{k+1})\|}{L} \right)^2 - \frac{1}{L^2} \left(\|\nabla J(x_{k+1})\|^2 + c \|\nabla J(\hat{x}_k)\|^2 \cos^2 \theta_k \right) \right].
\end{aligned}$$

That is, if we have

$$\begin{aligned}
\|e_{k+1}\| &< \frac{\sqrt{\|\nabla J(x_{k+1})\|^2 + c \|\nabla J(\hat{x}_k)\|^2 \cos^2 \theta_k} - \|\nabla J(x_{k+1})\|}{L} \\
&= \frac{(c \|\nabla J(\hat{x}_k)\|^2 \cos^2 \theta_k) / L}{\sqrt{\|\nabla J(x_{k+1})\|^2 + c \|\nabla J(\hat{x}_k)\|^2 \cos^2 \theta_k} + \|\nabla J(x_{k+1})\|},
\end{aligned}$$

we can obtain $J(\hat{x}_{k+1}) - J(\hat{x}_k) < 0$. ■

The error bound (3.24) one needs to satisfy to guarantee a descent search sequence is proportional to the gradient, in which sense is quite similar to the inequality (2.3) obtained by Bertsekas and Tsitsiklis in [136]. Although the bound (3.24) is very conservative, it can give us some insights into the robustness of line search methods. First, the exact line search allows a larger error bound than the inexact line search. Second, we can see that the bound is an increasing function of $\|\nabla J(\hat{x}_k)\|$. That is, when \hat{x}_k is far away from the minimizer of the performance function, we will expect the gradient to be large and thus the error the algorithm can tolerate is also large. This observation implies that line search methods will be very robust until the gradient converges to some invariant set, which is illustrated in the following corollary.

COROLLARY 3.5.4 Suppose the performance function $J(x)$ satisfies Assumption 3.1.1 and ∇J is Lipschitz continuous with constant L . A steepest descent algorithm is used with bounded error at each iteration. Assuming $\|e_k\| \leq e_L$ for some constant e_L , then we will have the gradient of the sequence $\{\hat{x}_k\}$ converges to the invariant set

$$\|\nabla J(\hat{x}_k)\| \leq \frac{Le_L}{c} [\sqrt{(1 + \alpha_k L)^2 + c} + (1 + \alpha_k L)], \quad (3.25)$$

where $c = 1$ for exact line search and $c = 2c_1(1 - c_2)$ for inexact line search satisfying conditions (1.3) and (1.4).

Proof: We have $\cos \theta_k = 1$ for steepest descent algorithm, and from inequality (3.24), as long as

$$\frac{1}{L} (\sqrt{\|\nabla J(x_{k+1})\|^2 + c\|\nabla J(\hat{x}_k)\|^2} - \|\nabla J(x_{k+1})\|) > e_L,$$

we will always have $J(\hat{x}_{k+1}) < J(\hat{x}_k)$. So we can find a conservative bound on $\nabla J(\hat{x}_k)$ given the error bound e_L .

For steepest descent method, we have $x_{k+1} = \hat{x}_k - \alpha_k \nabla J(\hat{x}_k)$. Then from Lipschitz condition,

$$\begin{aligned} \|\nabla J(x_{k+1})\| &\leq \|\nabla J(x_{k+1}) - \nabla J(\hat{x}_k)\| + \|\nabla J(\hat{x}_k)\| \\ &\leq (1 + \alpha_k L) \|\nabla J(\hat{x}_k)\|. \end{aligned}$$

Now the bound can be found via

$$\begin{aligned}
& \frac{1}{L}(\sqrt{\|\nabla J(x_{k+1})\|^2 + c\|\nabla J(\hat{x}_k)\|^2} - \|\nabla J(x_{k+1})\|) \leq e_L \\
& \Leftrightarrow \sqrt{\|\nabla J(x_{k+1})\|^2 + c\|\nabla J(\hat{x}_k)\|^2} \leq Le_L + \|\nabla J(x_{k+1})\| \\
& \Leftrightarrow c\|\nabla J(\hat{x}_k)\|^2 \leq 2Le_L(1 + \alpha_k L)\|\nabla J(\hat{x}_k)\| + L^2 e_L^2 \\
& \Leftrightarrow [\sqrt{c}\|\nabla J(\hat{x}_k)\| - \frac{Le_L}{\sqrt{c}}(1 + \alpha_k L)]^2 \leq L^2 e_L^2 [(1 + \alpha_k L)^2/c + 1] \\
& \Leftrightarrow \|\nabla J(\hat{x}_k)\| - \frac{Le_L}{c}(1 + \alpha_k L) \leq \frac{Le_L}{c} \sqrt{(1 + \alpha_k L)^2 + c}
\end{aligned}$$

Thus, we have that the gradient of the sequence $\{\hat{x}_k\}$ converges to the invariant set

$$\|\nabla J(\hat{x}_k)\| \leq \frac{Le_L}{c} [\sqrt{(1 + \alpha_k L)^2 + c} + (1 + \alpha_k L)].$$

■

Observed from inequality (3.25), a diminishing step length α_k is preferred later on to decrease the bound of the invariant set. As $\alpha_k \rightarrow 0$, the bound (3.25) converges to $(\frac{1}{c} + \sqrt{\frac{1}{c^2} + \frac{1}{c}})Le_L$. This is again coincident with the theory of numerical optimization, where generally a diminishing step length is required for the algorithms to converge to the minimum. And if there is no error between \hat{x}_k and x_k , we will see that the gradient converges to zero. Moreover, exact line search can achieve a smaller bound than the inexact line search.

3.5.2 Robustness of Trust Region Methods

For trust region-based extremum seeking control, the control input with disturbance or unknown plant dynamics will result in the state trajectory interpolating among the sequence $\{\hat{x}_k\}$, where $\hat{x}_0 = x_0$, $\hat{x}_{k+1} = \hat{x}_k + p_k + e_{k+1}$ for a successful step, $\hat{x}_{k+1} = \hat{x}_k + e_{k+1}$ for an unsuccessful step and e_{k+1} will be bounded given bounded

disturbance. The following theorem provides one possible quantitative analysis about the convergence of trust region methods under certain bounded error.

THEOREM 3.5.5 *Let $J : \mathbb{R}^n \rightarrow \mathbb{R}$ be Lipschitz continuously differentiable and bounded below on the level set $\{x | J(x) \leq J(x_0)\}$. Suppose that $\|B_k\| \leq \beta$ for some constant β , where B_k is the approximation or the Hessian matrix of J itself, and that all trial steps p_k satisfy the inequality (1.8) for some constant c_t , and $\|p_k\| \leq \gamma \Delta_k$ for some constant $\gamma \geq 1$. A trust region algorithm starting from $\hat{x}_0 = x_0$ is used but with bounded error at each iteration, that is, $\hat{x}_{k+1} = \hat{x}_k + p_k + e_{k+1}$ for a successful step or $\hat{x}_{k+1} = \hat{x}_k + e_{k+1}$ for an unsuccessful step. If, for every successful step*

$$\|e_{k+1}\| \leq -\frac{2}{L} \|\nabla J(\hat{x}_k + p_k)\| \cos \alpha_k, \quad (3.26)$$

and for every unsuccessful step,

$$\|e_{k+1}\| \leq -\frac{2}{L} \|\nabla J(\hat{x}_k)\| \cos \beta_k, \quad (3.27)$$

where $\cos \alpha_k = \frac{\nabla J(\hat{x}_k + p_k)^\top e_{k+1}}{\|\nabla J(\hat{x}_k + p_k)\| \|e_{k+1}\|}$ and $\cos \beta_k = \frac{\nabla J(\hat{x}_k)^\top e_{k+1}}{\|\nabla J(\hat{x}_k)\| \|e_{k+1}\|}$, then there exists a subsequence of $\{\hat{x}_k\}$ that converges to the first order stationary point of $J(x)$.

Proof: At iteration k and $t = t_k$, we obtain p_k by approximately solving the quadratic approximation $m_k(x(t_k) + p)$ within the trust region $\|p\| \leq \Delta_k$. And ideally we would have a controller (3.12) to regulate the state from $x(t_k) = \hat{x}_k$ to $\hat{x}_k + p_k$ to obtain an ideal ratio (1.6). However, due to the input disturbance or model uncertainty, such controller can only regulate the state to $\hat{x}_k + p_k + e_{k+1}$ at $t = t_{k+1}^1$, and therefore we can only obtain a practical ratio

$$\hat{\rho}_k = \frac{J(\hat{x}_k) - J(\hat{x}_k + p_k + e_{k+1})}{m_k(\hat{x}_k) - m_k(\hat{x}_k + p_k)}. \quad (3.28)$$

Two cases need to be analyzed to guarantee the global convergence of the sequence $\{\hat{x}_k\}$. Let $S_k = \{s_1, s_2, \dots, s_i\}$ be a subsequence of $\{1, 2, \dots, k\}$ such that S_k represents the index set of successful steps $\{\hat{x}_{s_i}\}$ up to $t = t_k$, that is $s_i \leq k$ and

$$\frac{J(\hat{x}_{s_{i-1}}) - J(\hat{x}_k)}{m_{s_{i-1}}(\hat{x}_{s_{i-1}}) - m_{s_{i-1}}(\hat{x}_{s_{i-1}} + p_{s_{i-1}})} \geq \eta_1, \quad (3.29)$$

which means \hat{x}_k is a successful step if started from $\hat{x}_{s_{i-1}}$.

Case I: for a successful step, $\rho_k \geq \eta_1$, then we want $\hat{\rho}_k \geq \eta_1$ to guarantee a successful step in the presence of e_{k+1} , that is we wish

$$\begin{aligned} & \frac{J(\hat{x}_k) - J(\hat{x}_k + p_k + e_{k+1})}{m_k(\hat{x}_k) - m_k(\hat{x}_k + p_k)} \\ & \geq \frac{J(\hat{x}_k) - J(\hat{x}_k + p_k) - \nabla J(\hat{x}_k + p_k)^\top e_{k+1} - \frac{L}{2} \|e_{k+1}\|^2}{m_k(\hat{x}_k) - m_k(\hat{x}_k + p_k)} \\ & = \frac{J(\hat{x}_k) - J(\hat{x}_k + p_k)}{m_k(\hat{x}_k) - m_k(\hat{x}_k + p_k)} - \frac{\nabla J(\hat{x}_k + p_k)^\top e_{k+1} + \frac{L}{2} \|e_{k+1}\|^2}{m_k(\hat{x}_k) - m_k(\hat{x}_k + p_k)} \\ & \geq \eta_1 - \frac{\nabla J(\hat{x}_k + p_k)^\top e_{k+1} + \frac{L}{2} \|e_{k+1}\|^2}{m_k(\hat{x}_k) - m_k(\hat{x}_k + p_k)} \geq \eta_1. \end{aligned}$$

Since $m_k(\hat{x}_k) - m_k(\hat{x}_k + p_k)$ is always positive, we need

$$\nabla J(\hat{x}_k + p_k)^\top e_{k+1} + \frac{L}{2} \|e_{k+1}\|^2 \leq 0.$$

Now define α_k to be the angle between $\nabla J(\hat{x}_k + p_k)$ and e_{k+1} , that is $\cos \alpha_k = \frac{\nabla J(\hat{x}_k + p_k)^\top e_{k+1}}{\|\nabla J(\hat{x}_k + p_k)\| \|e_{k+1}\|}$. Then,

$$\begin{aligned} & \nabla J(\hat{x}_k + p_k)^\top e_{k+1} + \frac{L}{2} \|e_{k+1}\|^2 = \|\nabla J(\hat{x}_k + p_k)\| \|e_{k+1}\| \cos \alpha_k + \frac{L}{2} \|e_{k+1}\|^2 \leq 0 \\ & \Leftrightarrow \|e_{k+1}\| \leq -\frac{2}{L} \|\nabla J(\hat{x}_k + p_k)\| \cos \alpha_k. \end{aligned}$$

Case II: for an unsuccessful step, $\rho_k < \eta_1$, then we do not really need to care about the regulation error due to $u(t)$ from $t_k \leq t \leq t_{k+1}^1$. If the resulting $\hat{\rho}_k \geq \eta_1$, then it actually drives the state to a place with lower performance function value, which

helps the extremum seeking process. If $\hat{\rho}_k < \eta_1$, then we need to have $u(t)$ again to transfer the state x back to \hat{x}_k . Here, we redefine the transfer error e_{k+1} as an error from the second transition that drives the state back to \hat{x}_k , then at time $t = t_{k+1}^1 + \delta_k^2$, the state $x = \hat{x}_k + e_{k+1}$. Therefore, we wish $\hat{x}_k + e_{k+1}$ to be a successful step compared with the previous one $\hat{x}_{s_{i-1}}$, i.e.

$$\begin{aligned}
& \frac{J(\hat{x}_{s_{i-1}}) - J(\hat{x}_k + e_{k+1})}{m_{s_{i-1}}(\hat{x}_{s_{i-1}}) - m_{s_{i-1}}(\hat{x}_{s_{i-1}} + p_{s_{i-1}})} \\
& \geq \frac{J(\hat{x}_{s_{i-1}}) - J(\hat{x}_k) - \nabla J(\hat{x}_k)^\top e_{k+1} - \frac{L}{2} \|e_{k+1}\|^2}{m_{s_{i-1}}(\hat{x}_{s_{i-1}}) - m_{s_{i-1}}(\hat{x}_{s_{i-1}} + p_{s_{i-1}})} \\
& = \frac{J(\hat{x}_{s_{i-1}}) - J(\hat{x}_k)}{m_{s_{i-1}}(\hat{x}_{s_{i-1}}) - m_{s_{k-1}}(\hat{x}_{s_{i-1}} + p_{s_{k-1}})} \\
& \quad - \frac{\nabla J(\hat{x}_k)^\top e_{k+1} + \frac{L}{2} \|e_{k+1}\|^2}{m_{s_{i-1}}(\hat{x}_{s_{i-1}}) - m_{s_{i-1}}(\hat{x}_{s_{i-1}} + p_{s_{k-1}})} \\
& \geq \eta_1 - \frac{\nabla J(\hat{x}_k)^\top e_{k+1} + \frac{L}{2} \|e_{k+1}\|^2}{m_{s_{i-1}}(\hat{x}_{s_{i-1}}) - m_{s_{i-1}}(\hat{x}_{s_{i-1}} + p_{s_{k-1}})} \geq \eta_1
\end{aligned}$$

Now define β_k to be the angle between $\nabla J(\hat{x}_k)$ and e_{k+1} , that is, $\cos \beta_k = \frac{\nabla J(\hat{x}_k)^\top e_{k+1}}{\|\nabla J(\hat{x}_k)\| \|e_{k+1}\|}$.

Then,

$$\begin{aligned}
& \nabla J(\hat{x}_k)^\top e_{k+1} + \frac{L}{2} \|e_{k+1}\|^2 = \|\nabla J(\hat{x}_k)\| \|e_{k+1}\| \cos \beta_k + \frac{L}{2} \|e_{k+1}\|^2 \leq 0 \\
& \Leftrightarrow \|e_{k+1}\| \leq -\frac{2}{L} \|\nabla J(\hat{x}_k)\| \cos \beta_k.
\end{aligned}$$

Thus, if (3.26) and (3.27) are satisfied, we will have a descent subsequence $\{\hat{x}_{s_k}\}$ of $\{\hat{x}_k\}$ that converges to the first order stationary point of $J(x)$. ■

3.5.3 Design of Robust Extremum Seeking Control Scheme

The importance of Theorems 3.5.3 and 3.5.5 is that they present design criteria for a robust regulator. For extremum seeking control based on line search methods, as long as we can design a regulator satisfying (3.24), then the extremum seeking will

continue to decrease the performance output. However, how to design such robust regulator is not the topic of this chapter and will be pursued later in Chapter 4, which can be done via adaptive control [60], sliding mode control and so on.

For extremum seeking control based on trust region methods, at each step we do not know whether it will be a successful one or not in advance since we do not have the knowledge of ρ_k , then, the controller $u(t)$ from $t_k \leq t \leq t_k + \delta_k^1$ needs to be designed to satisfy (3.26), if it is a successful step, then the design will guarantee the step still to be a successful step since $\hat{\rho}_k \geq \eta_1$. If it is not a successful step, that is $\rho_k < \eta_1$, then there is no guarantee that $\hat{\rho}_k$ will be larger than η_1 . So if $\hat{\rho}_k \geq \eta_1$, then we are in luck in the sense that we are on the right way due to the disturbance; if not, then we need a robust controller satisfying (3.27) from $t_{k+1}^1 \leq t \leq t_{k+1}^1 + \delta_k^2$ to transfer the current state back to the neighborhood of \hat{x}_k , which means we can return to the right track as inferred from previous successful steps. Both inequalities (3.26) and (3.27) define similar criteria for the design of robust extremum seeking controllers, which is still very conservative from the analysis since the ideal ρ_k in general is larger than η_1 . A pictorial illustration of the tuning criteria in \mathbb{R}^2 is shown in Figure 3.12, where we let the destination, $\hat{x}_k + p_k$ in (3.26) and \hat{x}_k in (3.27), to be $[0, 0]^T$ and its gradient to be $[10, 10]^T$, and the Lipschitz constant $L = 1$. The blue star is the desired destination, and the dotted area depicts the acceptable region satisfying inequalities (3.26) or (3.27). The size of the region is proportional to the gradient of the destination position, and the reciprocal of the Lipschitz constant. Therefore, we will be able to estimate the robust region given the estimation of gradient and Lipschitz constant.

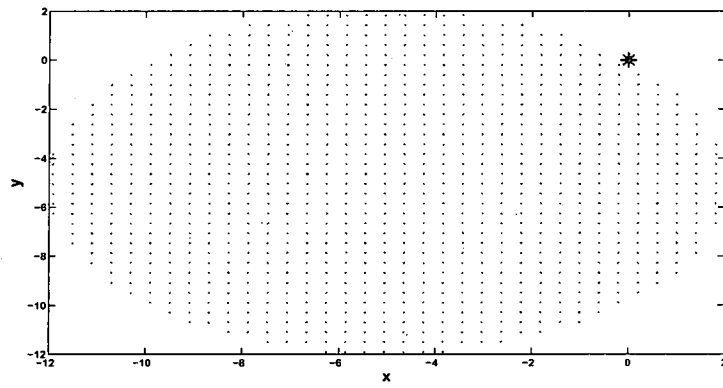


Figure 3.12: Robust region (3.26) and (3.27) of the trust region method.

CHAPTER 4

ROBUST AND ADAPTIVE DESIGN OF EXTREMUM SEEKING CONTROL

We continue the minimum seeking control design of nonlinear system (2.1) with performance function (2.2) in this chapter. The numerical optimization-based extremum seeking control proposed in Chapter 3 contains a state regulator that ensures the state to travel along a search sequence provided by a convergent numerical optimization algorithm and therefore the state eventually converges to the optimizer of the performance function. One advantage of such design is to achieve extremum seeking purpose without separating the dynamics from extremum seeking loop. Compared with the analog optimization-based extremum seeking control schemes to tune the control arguments to optimize the performance function, this can be viewed as a direct extremum seeking control scheme by tuning the state via the control input. All we need is the existence of some state regulator to ensure the state to visit the desired set point and thus we can measure the necessary performance output or its derivatives to continue the optimization process. The global convergence property of the optimization algorithm and the state regulator guarantee the global convergence of the extremum seeking control scheme. Most important, the numerical optimization-based extremum seeking control scheme provides large design flexibility, such as the

incorporation of gradient free optimization algorithm, or robust and adaptive control design. In this chapter, robust and adaptive design of extremum seeking control for state feedback linearizable systems are studied. A new design of the state regulator is proposed via output tracking, where we trade off the finite time state regulation to obtain the flexibility in designing a robust state regulator to deal with input disturbance and unknown plant dynamics. At the same time, the robustness of the optimization algorithm guarantees that we can use the asymptotic state regulator and still ensure the convergence of the extremum seeking control scheme.

4.1 An Output Tracking Framework for Numerical Optimization-based Extremum Seeking Control

Consider the minimum seeking control design of a SISO nonlinear affine model in (3.7) with the performance function defined in (2.2). By assuming the nonlinear affine system (3.7) is state feedback linearizable on the domain D , we can always put the system in the controllable canonical form (3.8) via coordinate transformation. We consider the state feedback linearizable system to outline the output tracking framework for the numerical optimization-based extremum seeking control. The basic concept and framework applies to more general nonlinear systems as long as one can fulfill the output tracking purpose.

4.1.1 Construction of a Reference Signal

As seen in the step 2 of the general framework of the extremum seeking control in Section 3.1 of Chapter 3, we need to design a state regulator u that drives the state to x_{k+1} , a set point candidate generated by the numerical optimization algorithm. We have shown in Chapter 3 [72] a design of state regulator obtained from the proof

of controllability for LTI systems. Even though it has the advantage of finite time regulation, it is very difficult to robustify that control law. Also, the finite time state regulator based on controllability can only guarantee that the state will arrive at the desired set point at the prescribed time given that the system is controllable, however, it does not guarantee the system will stay at the set point after that. For example, consider a point mass model

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u.$$

It is easy to verify that the point mass model is controllable, and therefore we can design a finite time state regulator to drive the state to an arbitrary point in the state space. For example, we start the point mass at $x_0 = [0, 0]^T$, and we want to drive the point mass to $[1, 1]^T$. The finite time state regulator can guarantee the state trajectory will pass $[1, 1]^T$, however, the point mass cannot stay there for ever due to the velocity state is 1 and therefore $[1, 1]^T$ is not an equilibrium point. The point mass can only stay at any equilibrium point such as $[c, 0]^T$ for any $c \in \mathbb{R}^n$. Of course, for certain class of systems, any point in the state space is a stabilizable equilibrium. Then this is equivalent to the Assumptions 2.0.15 and 2.0.16, where the performance function will reduce to a function of the parameter of the control law, and then assume such argument will produce a stabilizable equilibrium point.

However, all we need is to design a controller that enables the state to visit the desired set point to obtain the necessary measurement of the performance output or its gradient at $x = x_{k+1}$ to continue the optimization process. Thus we no longer need to separate the dynamics from the extremum seeking loop (there is no need to stabilize the system at x_{k+1}). Now, we can achieve the state regulation via output tracking as well, which leads to large flexibility in the robust design.

Now, let x_{k+1} be a desired set point and denote $z_{k+1} = T(x_{k+1})$. We want to construct a bounded periodic reference signal $r_k(t) \in \mathbb{R}$ with periodic T_r that satisfies the following conditions:

$$r_k(T_r) = z_{k+1}(1), \dot{r}_k(T_r) = z_{k+1}(2), \dots, r_k^{(n-1)}(T_r) = z_{k+1}(n) \quad (4.1)$$

Then if we can design a controller such that z_1 asymptotically tracks $r_k(t)$, then we will have z_2 asymptotically track $\dot{r}_k(t)$ and so on. Therefore, we will be able to ensure the state x approaches x_{k+1} asymptotically.

We first assume that the number of states n is odd, and let $p = (n - 1)/2$. The following is one possible design for the reference signal satisfying conditions (4.1):

$$\begin{aligned} r_k(t) = & a_1 \sin(\omega_1 t) + b_1 \cos(\omega_1 t) + a_2 \sin(\omega_2 t) + b_2 \cos(\omega_2 t) \\ & + \dots + a_p \sin(\omega_p t) + b_p \cos(\omega_p t) + r_{k0}, \end{aligned} \quad (4.2)$$

where a_i, b_i, ω_i and r_{k0} for $i = 1, \dots, p$ are parameters to be determined based on (4.1). We have

$$\begin{aligned} \dot{r}_k(t) = & a_1 \omega_1 \cos(\omega_1 t) - b_1 \omega_1 \sin(\omega_1 t) + \dots + a_p \omega_p \cos(\omega_p t) - b_p \omega_p \sin(\omega_p t), \\ \ddot{r}_k(t) = & -a_1 \omega_1^2 \sin(\omega_1 t) - b_1 \omega_1^2 \cos(\omega_1 t) + \dots - a_p \omega_p^2 \sin(\omega_p t) - b_p \omega_p^2 \cos(\omega_p t), \\ & \vdots \\ r_k^{(2p-1)}(t) = & (-1)^{p-1} a_1 \omega_1^{2p-1} \cos(\omega_1 t) - (-1)^{p-1} b_1 \omega_1^{2p-1} \sin(\omega_1 t) + \dots \\ & + (-1)^{p-1} a_p \omega_p^{2p-1} \cos(\omega_p t) - (-1)^{p-1} b_p \omega_p^{2p-1} \sin(\omega_p t), \\ r_k^{(2p)}(t) = & (-1)^p a_1 \omega_1^{2p} \sin(\omega_1 t) + (-1)^p b_1 \omega_1^{2p} \cos(\omega_1 t) + \dots \\ & + (-1)^p a_p \omega_p^{2p} \sin(\omega_p t) + (-1)^p b_p \omega_p^{2p} \cos(\omega_p t). \end{aligned} \quad (4.3)$$

Then, since T_r is the period of the $r_k(t)$, such that $\omega_i T_r = 2m_i\pi$ for some positive integer $m_i, i = 1, \dots, p$, we have

$$r_k(T_r) = b_1 + b_2 + \dots b_p + r_{k0} = z_{k+1}(1)$$

$$\dot{r}_k(T_r) = a_1\omega_1 + a_2\omega_2 + \dots + a_p\omega_p = z_{k+1}(2)$$

$$\ddot{r}_k(T_r) = -b_1\omega_1^2 - b_2\omega_2^2 - \dots - b_p\omega_p^2 = z_{k+1}(3)$$

⋮

$$r_k^{(2p-1)}(T_r) = (-1)^{p-1}[a_1\omega_1^{2p-1} + \dots + a_p\omega_p^{2p-1}] = z_{k+1}(2p)$$

$$r_k^{(2p)}(T_r) = (-1)^p[b_1\omega_1^{2p} + \dots + b_p\omega_p^{2p}] = z_{k+1}(2p+1)$$

Thus, we have the following linear equations

$$\underbrace{\begin{bmatrix} \omega_1 & \omega_2 & \dots & \omega_p \\ \omega_1^3 & \omega_2^3 & \dots & \omega_p^3 \\ \vdots & \vdots & \ddots & \vdots \\ \omega_1^{2p-1} & \omega_2^{2p-1} & \dots & \omega_p^{2p-1} \end{bmatrix}}_{\Omega_a} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix} = \begin{bmatrix} (-1)^{1-1}z_{k+1}(2) \\ (-1)^{2-1}z_{k+1}(4) \\ \vdots \\ (-1)^{p-1}z_{k+1}(2p) \end{bmatrix}, \quad (4.4)$$

$$\underbrace{\begin{bmatrix} \omega_1^2 & \omega_2^2 & \dots & \omega_p^2 \\ \omega_1^4 & \omega_2^4 & \dots & \omega_p^4 \\ \vdots & \vdots & \ddots & \vdots \\ \omega_1^{2p} & \omega_2^{2p} & \dots & \omega_p^{2p} \end{bmatrix}}_{\Omega_b} \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_p \end{bmatrix} = \begin{bmatrix} (-1)^1z_{k+1}(3) \\ (-1)^2z_{k+1}(5) \\ \vdots \\ (-1)^pz_{k+1}(2p+1) \end{bmatrix}. \quad (4.5)$$

Therefore, we can first choose different frequencies as $0 < \omega_1 < \omega_2 < \dots < \omega_n$, then the generalized Vandermonde matrices Ω_a and Ω_b will have positive determinants and hence are non-singular, so that we can obtain a_i, b_i for $i = 1, \dots, p$ by solving Equations (4.4) and (4.5). Finally, we can solve $r_{k0} = z_{k+1}(1) - \sum_{i=1}^p b_i$. If the number of states is even and great than 2, we can let $p = n/2$ and $r_k(t)$ be (4.2) with $r_{k0} = 0$, then we will have a very similar result as n is odd.

4.1.2 Design of a State Regulator

Now, the design of state regulator becomes the standard output tracking problem [60] for a state feedback linearizable system defined by (3.7). Given the desired set point x_{k+1} , we can construct a reference signal $r_k(t)$ as in Section 4.1.1. Then, we define a stable manifold

$$e_z = k_1(z_1 - r_k) + \dots + k_{n-1}(z_{n-1} - r_k^{(n-2)}) + z_n - r_k^{(n-1)}, \quad (4.6)$$

and the polynomial $s^{n-1} + k_{n-1}s^{n-2} + \dots + k_1$ is Hurwitz. Taking the derivative of e_z we find

$$\dot{e}_z = \chi(z) + f(z) + g(z)u, \quad (4.7)$$

where $\chi(z) = k_1(z_2 - \dot{r}_k) + \dots + k_{n-1}(z_n - r_k^{(n-1)}) - r_k^{(n)}$. Now consider the Lyapunov candidate $V = \frac{1}{2}e_z^2$ and

$$u = \frac{-\chi(z) - f(z) - ke_z}{g(z)}, \quad k > 0, \quad (4.8)$$

we find that $\dot{V} = e_z \dot{e}_z = -ke_z^2 = -2kV$ so $e_z = 0$ is an exponentially stable equilibrium point (the state trajectory converges to the manifold $S_z(t) = \{z \in \mathbb{R}^n, e_z = 0\}$). Moreover, we have $V(t) = V(0)e^{-2kt}$ and hence

$$e_z(t) = \sqrt{2V(0)}e^{-kt} = e_z(0)e^{-kt}, \quad (4.9)$$

where $e_z(0) = \sum_{i=1}^{n-1} k_i(z_i(0) - r_k^{i-1}(0)) + z_n(0) - r_k^{(n-1)}(0)$.

We will now show that $e_z \rightarrow 0$ means $z_1 \rightarrow r_k(t)$ and bounding $|e_z|$ implies that $\|z\|$ is bounded. By taking Laplace form on both sides of Equation (4.6), we obtain

$$\frac{Z_1(s) - R_k(s)}{E_z(s)} = \frac{1}{s^{n-1} + k_{n-1}s^{n-2} + \dots + k_2s + k_1},$$

where $Z_1(s)$, $R_k(s)$ and $E_z(s)$ are Laplace transforms of z_1 , r_k and e_z . Since denominator poles are designed to be in the left half plane, it is straightforward to see that z_1 converges to r_k asymptotically as e_z converges to zero. Therefore, we have $z_1 \rightarrow r_k$, $z_2 \rightarrow \dot{r}_k$ and so on. Moreover, let $\mu = [z_1 - r_k, \dots, z_{n-1} - r_k^{(n-2)}]^\top$, so $\dot{\mu} = L\mu + be_z$, where

$$L = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & & \ddots & \\ 0 & 0 & \dots & \dots & 1 \\ -k_1 & \dots & \dots & \dots & -k_{n-1} \end{bmatrix}$$

and $b = [0, \dots, 0, 1]^\top \in \mathbb{R}^{n-1}$. Since L is Hurwitz, we can show that $\|e^{Lt}\| \leq d_1 e^{-d_2 t}$ for some positive scalars d_1 and d_2 ⁶. Now, we can further conclude that $\|\mu(t)\| \leq \phi_\mu(t, |e_z|)$ where

$$\begin{aligned} \phi_\mu(t, |e_z|) &= d_1 e^{-d_2 t} \|\mu(0)\| + \int_0^t d_1 e^{-d_2(t-\tau)} |e_z(\tau)| d\tau \\ &= d_1 e^{-d_2 t} \|\mu(0)\| + \int_0^t d_1 e^{-d_2(t-\tau)} |e_z(0)| e^{-k\tau} d\tau \\ &= d_1 e^{-d_2 t} \|\mu(0)\| + d_1 e^{-d_2 t} |e_z(0)| \int_0^t e^{(d_2-k)\tau} d\tau \\ &= d_1 e^{-d_2 t} \|\mu(0)\| + \frac{d_1 |e_z(0)|}{d_2 - k} (e^{-kt} - e^{-d_2 t}). \end{aligned} \quad (4.10)$$

Then $\phi_\mu(t, |e_z|)$ is bounded for any bounded e_z , nondecreasing with respect to $|e_z| \in \mathbb{R}^+$ for each fixed t , and decreasing to zero as $t \rightarrow \infty$. Using the definition of the error

⁶The following analysis provides a simple method to find values of d_1 and d_2 . Consider a system defined by $\dot{\mu} = L\mu$ where L is Hurwitz, the solution is $\mu(t) = e^{Lt}\mu(0)$. If we let $V_\mu = \mu^\top P \mu$ where P is a symmetric positive definite matrix, then $\dot{V}_\mu = \mu^\top (PL + L^\top P)\mu$. Now choose P to satisfy the Lyapunov matrix equation $PL + L^\top P = -I$ so that $\dot{V}_\mu \leq -\mu^\top \mu \leq -V/\lambda_{\max}(P)$. Solving the above differential inequality, we find that $V_\mu \leq V_\mu(0)e^{-t/\lambda_{\max}(P)}$. Since

$$\|\mu(t)\|^2 \leq \frac{1}{\lambda_{\min}(P)} V_\mu \leq \frac{V_\mu(0)}{\lambda_{\min}(P)} e^{-t/\lambda_{\max}(P)},$$

we conclude that $\|e^{Lt}\| \leq d_1 e^{-d_2 t}$ for L is Hurwitz and $d_1 = \sqrt{\mu^\top(0)P\mu(0)/\lambda_{\min}(P)}/\|\mu(0)\|$ and $d_2 = 1/(2\lambda_{\max}(P))$.

system, we find

$$\begin{bmatrix} z_1 \\ \vdots \\ z_n \end{bmatrix} = \begin{bmatrix} r_k \\ \vdots \\ r_k^{(n-1)} \end{bmatrix} + \begin{bmatrix} \mu \\ \varsigma \end{bmatrix}, \quad (4.11)$$

where $\varsigma = e_z - k_1(z_1 - r_k) - \dots - k_{n-1}(z_{n-1} - r_k^{(n-2)})$ so we have

$$\|z\| \leq (1 + |k_1| + \dots + |k_{n-1}|)\phi_\mu(t, |e_z|) + |e_z| + \|\bar{r}_k\|,$$

where $\bar{r}_k = [r_k, \dot{r}_k, \dots, r_k^{(n-1)}]^\top$. Therefore, bounded $|e_z|$ implies bounded $\|z\|$ and hence bounded $\|x\|$ given tracking a bounded reference signal $r_k(t)$. Since the reference signal $r_k(t)$ is also periodic, we can even apply iterative learning control [137] or repetitive control to improve the tracking performance.

4.1.3 Determination of the Regulation Time

The controller (4.8) implies z_1 will asymptotically track r_k . Since T is a diffeomorphism on D , then T^{-1} is continuously differentiable and we further assume $\left\| \frac{\partial T^{-1}}{\partial x} \right\|$ is bounded on D , then T^{-1} is Lipschitz continuous on D for some Lipschitz constant L_T . Now from Equation (4.11), we have

$$\begin{aligned} \|x - x_{k+1}\| &= \|T^{-1}(z) - T^{-1}(z_{k+1})\| \\ &\leq L_T \|z - z_{k+1}\| \\ &\leq L_T (\|z - \bar{r}_k\| + \|\bar{r}_k - z_{k+1}\|) \\ &\leq L_T \left(|e_z| + \left(1 + \sum_{i=1}^{n-1} |k_i| \right) \phi_\mu(t, |e_z|) + \|\bar{r}_k - z_{k+1}\| \right), \end{aligned} \quad (4.12)$$

where e_z and $\phi_\mu(t, |e_z|)$ will converge to zero as $t \rightarrow \infty$ as inferred from (4.10) and (4.9), and we know that $z_{k+1} = \bar{r}_k(t)$ when $t = mT_r$ for some positive integer m . Therefore, theoretically it may take infinite time for the state x to reach to the required set point x_{k+1} and this is not a property we want. However, the robustness

of optimization algorithms can relax the requirement for perfect regulation, since the optimization algorithm is still functional as long as we can regulate the state to a neighborhood of the set point x_{k+1} . Therefore, we can still implement the controller (4.8) for a finite time to carry on the optimization process. In particular, the robustness theorem (3.5.3) for line search method implies that given the current state $x = \hat{x}_k$, if we use the line search method to generate the new set point $x_{k+1} = \hat{x}_k + \alpha_k p_k$, as long as the control (4.8) drives the state into the neighborhood of x_{k+1} given by (3.24), we can still ensure the sequence $\{\hat{x}_k\}$ is a descent sequence. Therefore, we can use the error bound (3.24) as a stop criterion of the regulation, and we can further determine an upper bound of the regulation time δ_k in advance. Since we have no measurement of $\nabla J(x_{k+1})$ unless we drive the state to x_{k+1} , the right hand side of inequality (3.24) is not known yet. The following presents an estimation of the error bound based on available information.

From the assumption that ∇J is Lipschitz continuous with constant L , we will have $\|\nabla J(x_{k+1})\| \leq \|\nabla J(x_{k+1}) - \nabla J(\hat{x}_k)\| + \|\nabla J(\hat{x}_k)\| \leq (1 + \alpha_k L)\|\nabla J(\hat{x}_k)\|$, and $1 \geq \cos \theta_k \geq c_3 > 0$ ($c_3 = 1$ for steepest descent method) to satisfy the angle condition required for the global convergence of a line search method, therefore

$$\begin{aligned} & \frac{(c\|\nabla J(\hat{x}_k)\|^2 \cos^2 \theta_k)/L}{\sqrt{\|\nabla J(x_{k+1})\|^2 + c\|\nabla J(\hat{x}_k)\|^2 \cos^2 \theta_k} + \|\nabla J(x_{k+1})\|} \\ & \geq \frac{(c\|\nabla J(\hat{x}_k)\|^2 c_3^2)/L}{\sqrt{(1 + \alpha_k L)^2 \|\nabla J(\hat{x}_k)\|^2 + c\|\nabla J(\hat{x}_k)\|^2} + (1 + \alpha_k L)\|\nabla J(\hat{x}_k)\|} \\ & = \frac{cc_3^2 \|\nabla J(\hat{x}_k)\|}{L(\sqrt{(1 + \alpha_k L)^2 + c} + (1 + \alpha_k L))}. \end{aligned}$$

Now, let $b_k = \frac{cc_3^2}{L(\sqrt{(1 + \alpha_k L)^2 + c} + (1 + \alpha_k L))}$, and then if the state x enters the region given by

$$\|x - x_{k+1}\| \leq b_k \|\nabla J(\hat{x}_k)\| \quad (4.13)$$

then it satisfies the inequality (3.24), and we can stop the current regulation, get the measurement of the performance output or its gradient, and continue the iteration. The estimated bound $b_k \|\nabla J(\hat{x}_k)\|$ is proportional to the last gradient measurement and can be computed in the beginning of the current regulation. Moreover, when \hat{x}_k is away from the minimizer of the performance function, we will expect the gradient to be large and therefore the error bound (4.13) is large. Also since the step length α_k generally converges to zero as $k \rightarrow \infty$, then we have $b_k \rightarrow \frac{cc_3^2}{L(\sqrt{1+c}+1)}$.

Now, we can solve the following inequality (4.14) to obtain the upper bound of the regulation time $\bar{\delta}_k$ for $\|\nabla J(\hat{x}_k)\| \neq 0$:

$$|e_z| + \left(1 + \sum_{i=1}^{n-1} |k_i|\right) \phi_\mu(t, |e_z|) + \|\bar{r}_k - z_{k+1}\| \leq \frac{b_k}{L_T} \|\nabla J(\hat{x}_k)\|. \quad (4.14)$$

The smallest t satisfies the above inequality (4.14) will be a good choice of the regulation time upper bound $\bar{\delta}_k$. Moreover, from the above analysis, we can have some guidelines in choosing control parameters. If we want to decrease the regulation time δ_k , we would like to have relatively small constants k_1, \dots, k_n , large control gain k , small d_1 , large d_2 , and a small period T_r for the reference signal $r_k(t)$. For practical application, such upper bound $\bar{\delta}_k$ is generally needed to avoid the search getting stuck in somewhere. Also, a lower bound of the regulation time $\underline{\delta}_k$ is needed, for example, it can be the amount of time one need to collect enough performance output measurements to estimate the gradient.

4.1.4 Algorithm and Convergence

Now we are ready to present the extremum seeking control algorithm for state feedback linearizable systems based on line search and output tracking.

Line Search and Output Tracking-based ESC for State Feedback Linearizable Systems

Step 0 Set $t_0 = 0$, $\hat{x}_0 = x_0 = x(t_0)$, set $t_0 = 0$ and $k = 0$. Measure $\nabla J(x(t_0))$.

Step 1 Use a line search method with global convergence to produce

$$x_{k+1} = x(t_k) + \alpha_k p_k, \quad (4.15)$$

where α_k is the step length, p_k is the search direction.

Step 2 Compute $z_{k+1} = T(x_{k+1})$.

Step 3 Construct a reference signal $r_k(t)$ as in Section 4.1.1.

Step 4 Choose parameters $k, k_i, 1 \leq i \leq n - 1$, apply the state regulator as in (4.8).

Step 5 Compute or estimate b_k , if the state satisfies inequality

$$\|x - x_{k+1}\| \leq b_k \|\nabla J(x(t_k))\|,$$

then stop the current regulation and record current time to be t_{k+1} , $\delta_k = t_{k+1} - t_k$

and $\hat{x}_{k+1} = x(t_{k+1})$. Measure $\nabla J(x(t_{k+1}))$.

Step 6 Set $k \leftarrow k + 1$. Go to step 1.

We have the following theorem for the convergence of this extremum seeking control algorithm.

THEOREM 4.1.1 *If the affine system (2.1) satisfies Assumption 3.3.1 and the performance function $J(x)$ (2.2) satisfies Assumptions 3.1.1 - 3.1.4, and ∇J is Lipschitz*

continuous with constant L ; moreover, if the above line search and output tracking-based extremum seeking control is applied, then the state x will globally asymptotically converge to the global minimizer x^* of $J(x)$.

Proof: Now at time $t = t_k$, we have the current state $x = x(t_k) = \hat{x}_k$ and obtain x_{k+1} as in (4.15). Then we apply controller (4.8) to perform output tracking of reference $r_k(t)$. In the Section 4.1.3, we have shown how to obtain the upper bound $\bar{\delta}_k$ of the regulation time δ_k by solving inequality (4.14). Thus we claim we can enter the desired robust region in finite time, and therefore ensure the state trajectory interpolates among the descent sequence $\{\hat{x}_k\}$ every δ_k time.

According to Assumptions 3.1.3 and 3.1.4, we suppose that the unknown global minimizer is an isolated equilibrium point. Therefore, let $e = x - x^*$, and we choose the Lyapunov candidate

$$V(e, k) = J(x(t_k)) - J(x^*),$$

which is positive whenever $e \neq 0$ and zero for $e = 0$. And we know that the controller will ensure that the state x crosses \hat{x}_k and \hat{x}_{k+1} (where $\hat{x}_{k+1} = \hat{x}_k + \alpha_k p_k + e_{k+1}$) at time $t = t_k$ and t_{k+1} respectively. Then, according to Theorem 3.5.3

$$\begin{aligned} \Delta V &= V(e, k+1) - V(e, k) = J(x(t_{k+1})) - J(x(t_k)) \\ &= J(\hat{x}_{k+1}) - J(\hat{x}_k) < 0 \end{aligned}$$

Thus, we will have the $e = 0$ is a globally uniformly asymptotically stable equilibrium point. That is, the performance function achieves its global minimum as $t \rightarrow \infty$ and the state x will globally asymptotically converge to the global minimizer x^* . ■

Moreover, we can also use the robustness result of trust region method stated in Theorem 3.5.5. As seen from the illustration in Figure 3.12, we can transform such

robust region into a similar region defined as in (4.13). First, we need to estimate the gradient at x_{k+1} and therefore obtain the size of the robust region, then we can shift the regulation set point x_{k+1} to the center of the robust region shown in Figure 3.12. Now we can use the output tracking framework to design a asymptotic state regulator and the stopping criterion for trust region method is very similar to (4.13).

4.1.5 Simulations

Consider the second order nonlinear system (3.11) with the performance function (3.5). Let $z_1 = x_1, z_2 = -x_1 + x_2$, the transformed system in the new coordinate (z_1, z_2) is

$$\begin{aligned}\dot{z}_1 &= z_2, \\ \dot{z}_2 &= \underbrace{-(-x_1 + x_2) + x_1 x_2}_{f(z)} + u,\end{aligned}$$

The minimizer $x^* = [1, 1]^T$ can be a stabilizable equilibrium by letting $u = -1$. The explicit form of the performance function and its minimizer are both unknown to the designer. In the extremum seeking control scheme, we only need to measure the function value and its gradient value every δ_k time. Then, at iteration $k + 1$, we use the line search method to obtain candidate $x_{k+1} = x(t_k) - \alpha_k \nabla J(x(t_k))$, we then compute $z_{k+1} = T(x_{k+1})$, that is $z_{k+1}(1) = x_{k+1}(1), z_{k+1}(2) = -x_{k+1}(1) + x_{k+1}(2)$. The bounded reference signal is chosen to have period T_r as

$$r_k(t) = a_1 \sin(\omega_1 t) + a_2,$$

where $\omega_1 = 2\pi/T_r, a_1 = z_{k+1}(2)/\omega_1$ and $a_2 = z_{k+1}(1)$. It is easy to verify such design satisfies conditions (4.1). The error manifold is then defined as $e_z = k_1(z_1 - r_k) + z_2 - \dot{r}_k$, then the controller (4.8) becomes $u = -k_1(z_2 - \dot{r}_k) + \ddot{r}_k + (-x_1 + x_2) - x_1 x_2 - k e_z$.

We now implement the extremum seeking control algorithm in Section 4.1.4, where the initial condition is $t_0 = 0, x_0 = [-1.9, 0]^\top$, and line search method with $p_k = -\nabla J(x(t_k))$. The designer chooses step length α_k , gains k_1, k , and estimates the b_k in (4.13). The simulation results can be found in Figure 4.1, where the performance output (3.5) (Figure 4.1 (a)) approaches its minimum at $J(1, 1) = 0$ and the state (Figure 4.1 (b)) accordingly converges to the minimizer $[1, 1]^\top$. The control input can be seen in Figure 4.1 (c). The steepest descent algorithm produces a sequence $\{x_k\}$ as a guideline for the state regulation. The trajectory between x_k and x_{k+1} is shaped by the dynamic system (3.11) and the state regulator (4.8). This can be viewed clearly in Figure 4.1 (d), where the blue circle represents the $\{x_k\}$ and the red dashed line represents the state trajectory. It is also worth to notice that the control only regulates the state to the neighborhood of the set point as seen from the Figure 4.1 (d). Moreover, in order to accelerate the extremum seeking loop, we can choose a large control gain k , small period of the reference signal T_r . This simulation result can be found in Figure 4.2, where we see the decrease of convergence time with a large control gain and large amplitude of the oscillating transient compared with relevant results in Chapter 3. Finally, we note that the $[1, 1]^\top$ is stabilized by $u = -1$ and therefore steady state is not oscillating by using the asymptotic state regulator.

4.2 Robust Design for Input Disturbance

Consider input disturbance satisfying the matching condition

$$\dot{e}_z = \chi(z) + f(z) + g(z)(u + \Delta(t, z)), \quad (4.16)$$

where $\Delta(t, z)$ is the unknown input disturbance (we postulate it as a function of z for notation convenience, mathematically it is equivalent to put it as a function of

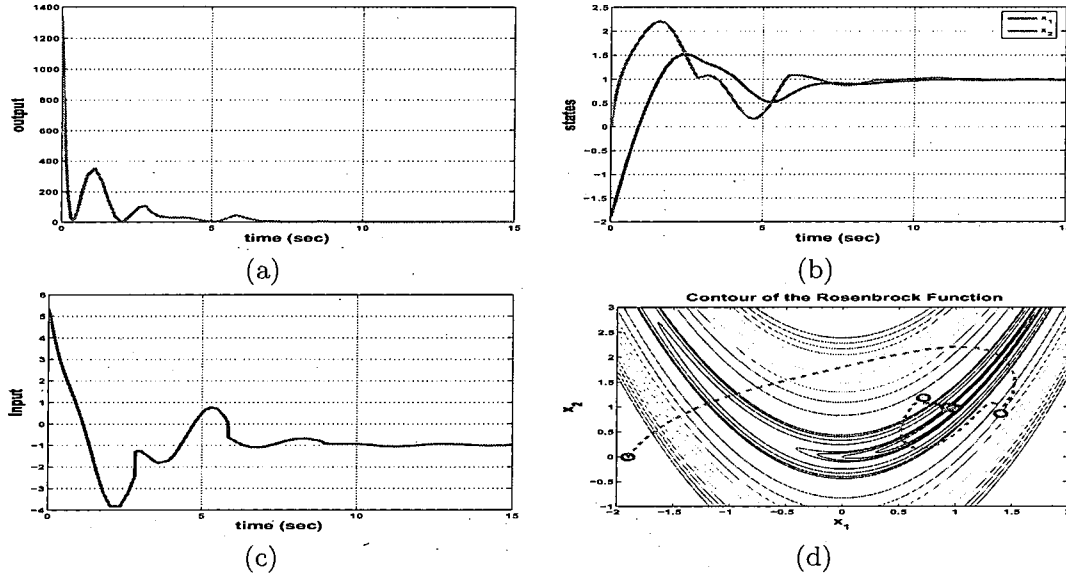


Figure 4.1: Extremum seeking control based on line search and output tracking, $k_1 = 2, k = 2, T_r = 3, \alpha_k = 0.0012$: (a) performance output; (b) state; (c) control input; (d) phase portrait.

x). We wish to add a stabilizing term to overcome the disturbance, that is, the new controller will be

$$u = \frac{-\chi(z) - f(z) - ke_z}{g(z)} + u_s, \quad k > 0, \quad (4.17)$$

where u_s is the stabilizing term designed based on the properties of the disturbance.

4.2.1 Bounded Input Disturbance

First, we assume $|\Delta(t, z)| \leq \beta$ represents some bounded uncertainty with $\beta > 0$ a known constant. Now consider the Lyapunov candidate $V = \frac{1}{2}e_z^2$ and we find that

$$\dot{V} = e_z \dot{e}_z = -2kV + e_z g(z)(u_s + \Delta(t, z)),$$

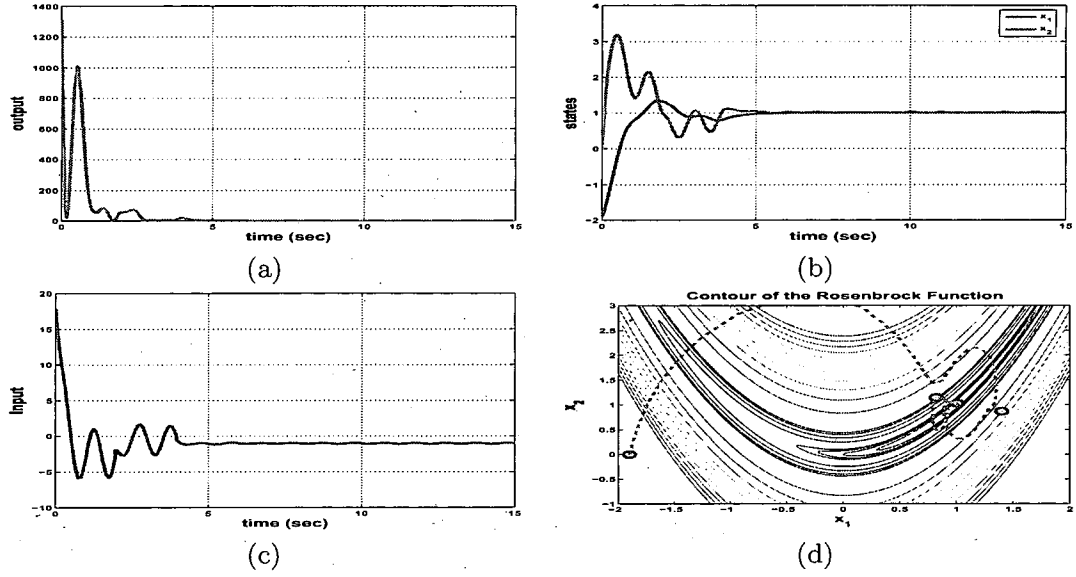


Figure 4.2: Extremum seeking control based on line search and output tracking, $k_1 = 2, k = 5, T_r = 1, \alpha_k = 0.0012$: (a) performance output; (b) state; (c) control input; (d) phase portrait.

Now, we can first choose

$$u_s = -\beta \text{sgn}(e_z g(z)), \quad (4.18)$$

where $\text{sgn}(\cdot)$ is the signum function. Then we have

$$\dot{V} \leq -2kV - |e_z g(z)|\beta + |e_z g(z)||\Delta(t, z)| \leq -2kV.$$

Since u_s in (4.18) is discontinuous, one may not be able to guarantee existence and uniqueness of the solutions. To avoid this issue, it is possible to choose a smoothed approximation of the nonlinear damping term

$$u_s = -\beta \frac{e_z g(z)}{|e_z g(z)| + c}, \quad (4.19)$$

with $c > 0$ a small constant. Then we find

$$\begin{aligned}
\dot{V} &\leq -2kV + e_z g(z) \left(-\beta \frac{e_z g(z)}{|e_z g(z)| + c} + \Delta(t, z) \frac{|e_z g(z)| + c}{|e_z g(z)| + c} \right) \\
&\leq -2kV + \frac{\beta}{|e_z g(z)| + c} (-|e_z g(z)|^2 + |e_z g(z)|^2 + c|e_z g(z)|) \\
&= -2kV + \frac{c\beta |e_z g(z)|}{|e_z g(z)| + c} \\
&\leq -2kV + c\beta.
\end{aligned} \tag{4.20}$$

We find $\dot{V} < 0$ whenever $V > c\beta/2k$ or $|e_z| > \sqrt{c\beta/k}$. Instead of having e_z exponentially stable, we achieved convergence of e_z to a neighborhood of zero. We now conclude that e_z is uniformly ultimately bounded. The size of the neighborhood can be made arbitrary small given large enough k and small enough c . There are other approximations of the signum function available and will arrive at similar results.

Thus, for extremum seeking control based on line search and output tracking, as stated from Theorem 3.2 in [75], the convergence relies on the controller to drive the state to the desired set point x_{k+1} or within the required neighborhood in finite time. That is, we should be able to find a $\bar{\delta}_k$ such that for $t \geq \bar{\delta}_k$, the following inequality is valid:

$$|e_z| + \left(1 + \sum_{i=1}^{n-1} |k_i| \right) \phi_\mu(t, |e_z|) + \|\bar{r}_k - z_{k+1}\| \leq \frac{b_k}{L_T} \|\nabla J(\hat{x}_k)\|. \tag{4.21}$$

Given e_z asymptotically converging to zero, we will have $\phi_\mu(t, |e_z|) \rightarrow 0$ as $t \rightarrow \infty$, and we know that $z_{k+1} = \bar{r}_k(t)$ when $t = mT_r$ for some positive integer m . Therefore, we always can find such $\bar{\delta}_k$ given e_z is asymptotically convergent. Now, consider e_z is uniformly ultimately bounded as seen above, that is,

$$|e_z(t)| \leq \sqrt{\frac{c\beta}{k} + \left(2V(0) - \frac{c\beta}{k} \right) e^{-2kt}},$$

therefore, we have $\lim_{t \rightarrow \infty} |e_z(t)| = \sqrt{c\beta/k}$ and denote $|e_z(t)| \leq \bar{e}_z = \max \left\{ \sqrt{2V(0)}, \sqrt{c\beta/k} \right\}$.

Then

$$\begin{aligned} \phi_\mu(t, |e_z|) &= d_1 e^{-d_2 t} \|\mu(0)\| + \int_0^t d_1 e^{-d_2(t-\tau)} |e_z(\tau)| d\tau \\ &\leq d_1 e^{-d_2 t} \|\mu(0)\| + d_1 \bar{e}_z e^{-d_2 t} \int_0^t e^{d_2 \tau} d\tau \\ &\leq d_1 e^{-d_2 t} \|\mu(0)\| + d_1 \bar{e}_z (1 - e^{-d_2 t}) / d_2 \end{aligned}$$

Thus $\lim_{t \rightarrow \infty} \phi_\mu(t, |e_z|) \leq d_1 \bar{e}_z / d_2 \neq 0$, and

$$\begin{aligned} &\lim_{t=mT_r, m \rightarrow \infty} |e_z| + \left(1 + \sum_{i=1}^{n-1} |k_i| \right) \phi_\mu(t, |e_z|) + \|\bar{r}_k - z_{k+1}\| \\ &\leq \sqrt{c\beta/k} + \left(1 + \sum_{i=1}^{n-1} |k_i| \right) d_1 \bar{e}_z / d_2 \end{aligned} \quad (4.22)$$

Now, we may not have a solution for the inequality (4.21) given e_z is only uniformly ultimately bounded. Therefore, instead of concluding that x will be asymptotically converging to the global minimizer $\|\nabla J(x)\| = 0$, we can only conclude that the closed loop system is stable and x will converge to a neighborhood of the global minimizer, that is,

$$\|\nabla J(x)\| \leq \left[\sqrt{c\beta/k} + \left(1 + \sum_{i=1}^{n-1} |k_i| \right) d_1 \bar{e}_z / d_2 \right] L_T / b_k \quad (4.23)$$

Thus we want to have large k, d_2 , small $d_1, k_1, k_2, \dots, k_{n-1}$ to have a small size neighborhood.

4.2.2 Unbounded Input Disturbance

Now, we assume $|\Delta(t, z)| \leq \beta \psi(z)$ with β an unknown constant and $\psi : \mathbb{R}^n \rightarrow \mathbb{R}$ is a known nonnegative function. It is assumed that ψ is bounded for any bounded $z \in \mathbb{R}^n$, in this case, we have the disturbance Δ may grow unbounded if $z \rightarrow \infty$.

Here, we consider the stabilizing term

$$u_s = -\eta e_z g(z) \psi^2(z), \quad \eta > 0. \quad (4.24)$$

The time derivative of the Lyapunov candidate $V = \frac{1}{2} e_z^2$ becomes

$$\begin{aligned} \dot{V} &= -2kV + e_z g(z) (-\eta e_z g(z) \psi^2(z) + \Delta(t, z)) \\ &\leq -2kV - \eta |e_z g(z)|^2 \psi^2(z) + \beta |e_z g(z)| |\psi(z)| \\ &\leq -2kV + \frac{\beta^2}{4\eta} \end{aligned} \quad (4.25)$$

We find $\dot{V} < 0$ whenever $V > \beta^2/(8k\eta)$ or $|e_z| > \beta/\sqrt{4k\eta}$. This guarantees that e_z is uniformly ultimately bounded and hence the close loop system is stable. The size of the neighborhood of e_z can be made arbitrarily small given large enough k and η . Therefore, we want to choose suitable parameters $\beta, \eta, d_1, d_2, k_1, k_2, \dots, k_{n-1}$ to ensure a small size of neighborhood similar as in (4.23).

4.2.3 Simulations

Consider the second order nonlinear system (3.11) with the performance function (3.5). In the simulation results shown below, we have $p_k = -\nabla J(x(t_k))$ for the line search method, and initial condition $x_0 = [-1.9, 0]^\top$. First, we consider the input disturbance case. Let the bounded input disturbance be a uniformly distributed random noise with amplitude 2 and the unbounded disturbance be $\Delta(t, z) = \frac{1}{|z_1| + \text{rand}(t)} + 3(\cos(t) + 1)z_2$, where $\text{rand}(t)$ is uniformly distributed noise with amplitude 1 and therefore $|\Delta(t, z)| \leq \frac{1}{|z_1| + 1} + 6z_2$. The simulation results of the nominal controller (4.8) given input disturbances are shown in Figure 4.3 and Figure 4.4, where we have the state converge to a neighborhood of x^* due to the bounded disturbance, and we have the system unstable given the unbounded disturbance. Now,

robust extremum seeking controllers are introduced to deal with input disturbance. The simulation results for stabilizing controller (4.18) can be found in Figure 4.5, where the control input can be seen in Figure 4.5 (c). The steepest descent algorithm produces a sequence $\{x_k\}$ as a guideline for the state regulation. The state trajectory eventually converges to the minimizer $[1, 1]^T$, and the transient between x_k and x_{k+1} is shaped by the dynamical system (3.11) and the state regulator (4.18). This can be viewed clearly in Figure 4.5 (d), where the blue circle represents the $\{x_k\}$ and the red dashed line represents the state trajectory. Even though the stabilizing controller with signum function achieves a good performance, it produces chattering in the control input as seen in Figure 4.5 (c). The approximated version (4.19) implements a smoothed control law and achieves comparable results, refer to Figure 4.6. For unbounded disturbance, the stabilizing controller (4.24) did overcome the unbounded disturbance and achieve the extremum seeking purpose as seen in Figure 4.7.

4.3 Robust Design for Unknown Plant Dynamics

The state regulator (4.8) is based on exact mathematical cancellation of the nonlinear terms $f(z)$ and $g(z)$. This is generally impossible in practice for several reasons such as model simplification, parameter uncertainty and computational errors. Generally we will implement the feedback control law $u = \frac{-\chi(z) - \hat{f}(z) - ke_z}{\hat{g}(z)}$, where $\hat{f}(z), \hat{g}(z)$ are approximations of $f(z)$ and $g(z)$. One method to deal with the approximation error is to treat it as an input disturbance, then we can design static (non-adaptive) stabilizing controllers to deal with input disturbance as seen in the Section 4.2. Here, we remove the assumption of exact knowledge of plant dynamics,

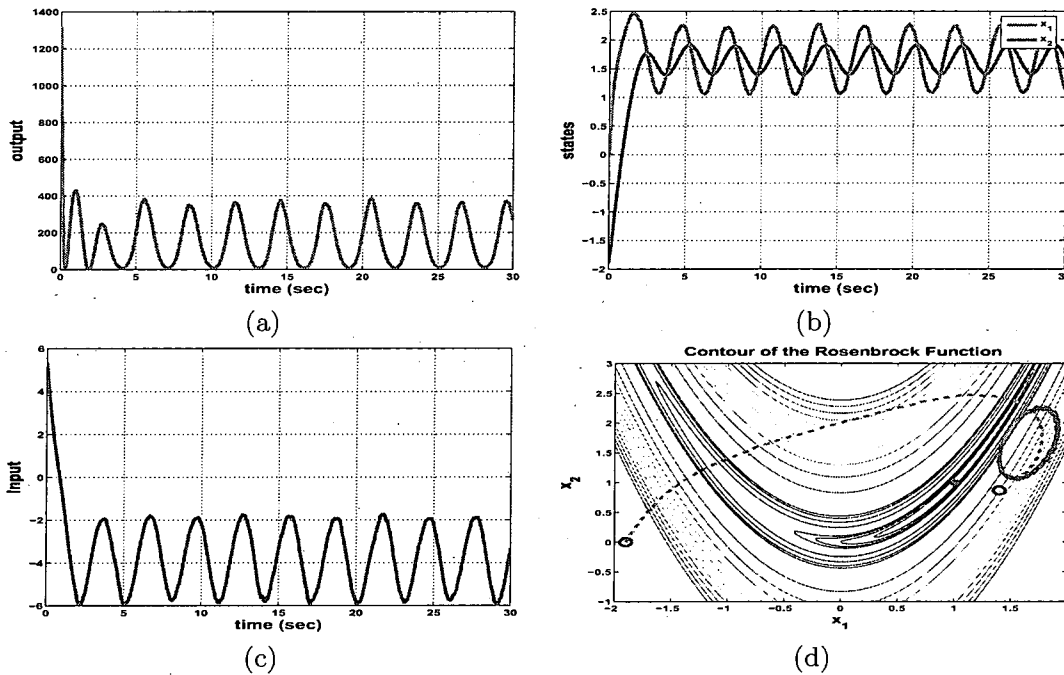


Figure 4.3: Extremum seeking control based on line search and output tracking, bounded input disturbance, $k_1 = 2, k = 2, \alpha_k = 0.0012$: (a) performance output; (b) state; (c) control input; (d) phase portrait.

only assuming that $g(z) \geq g_0 > 0$. Then we will consider two different approaches in developing adaptive control laws to deal with unknown plant dynamics.

4.3.1 Indirect Adaptive Control

First we will approximate the unknown plant dynamics $f(z)$ and $g(z)$ using two function approximators, and then use them to construct an adaptive controller. Now, we assume that the function approximator will approximate the plant dynamics

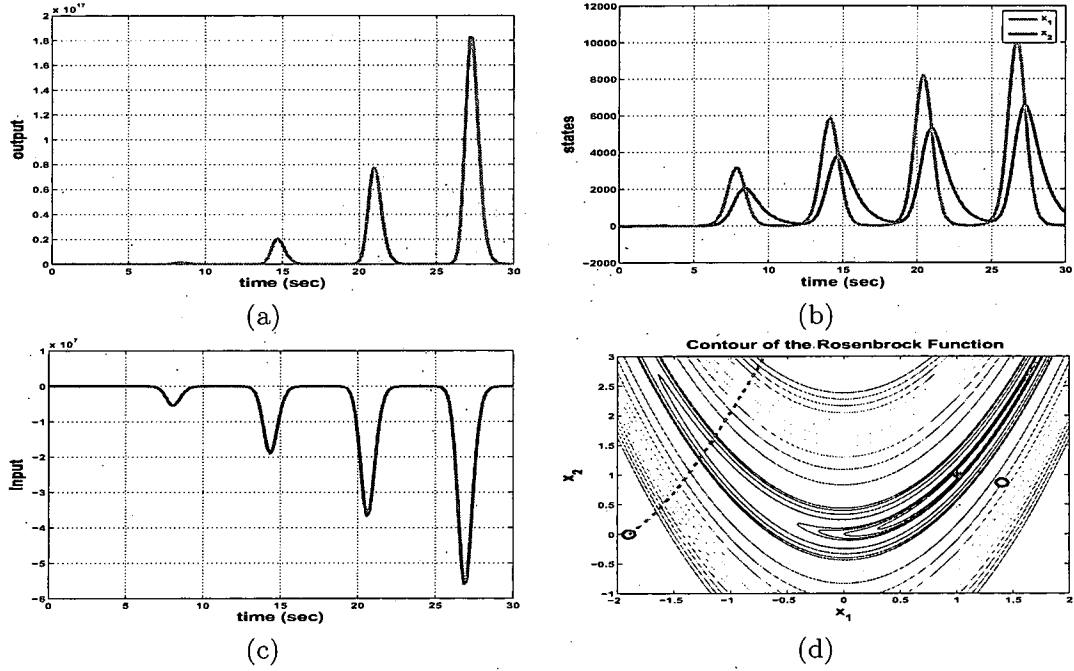


Figure 4.4: Extremum seeking control based on line search and output tracking, unbounded input disturbance, $k_1 = 2, k = 2, \alpha_k = 0.0012$: (a) performance output; (b) state; (c) control input; (d) phase portrait.

within a compact set $S \subset \mathbb{R}^n$, that is,

$$f(z) = \mathcal{F}_1(z, \theta_1^*) + \omega_1(z), \quad (4.26)$$

$$g(z) = \mathcal{F}_2(z, \theta_2^*) + \omega_2(z), \quad (4.27)$$

where $\mathcal{F}_1(z, \theta_1^*) = \theta_1^{*\top} \xi_1(z)$, $\mathcal{F}_2(z, \theta_2^*) = \theta_2^{*\top} \xi_2(z)$ are function approximators using basis functions $\xi_1(z), \xi_2(z)$. Moreover, $\theta_1^* \in \mathbb{R}^{p_1}$, $\theta_2^* \in \mathbb{R}^{p_2}$ are unknown optimal parameters such that for arbitrary $z \in S$ we have $|\omega_1(z)| \leq W_1$ and $|\omega_2(z)| \leq W_2$ for some known constants W_1 and W_2 , which are the smallest possible given p_1, p_2 and S . Let $\hat{\theta}_1, \hat{\theta}_2$ be the estimates of θ_1^*, θ_2^* , with parameter error vectors $\bar{\theta}_1 = \hat{\theta}_1 - \theta_1^*$ and

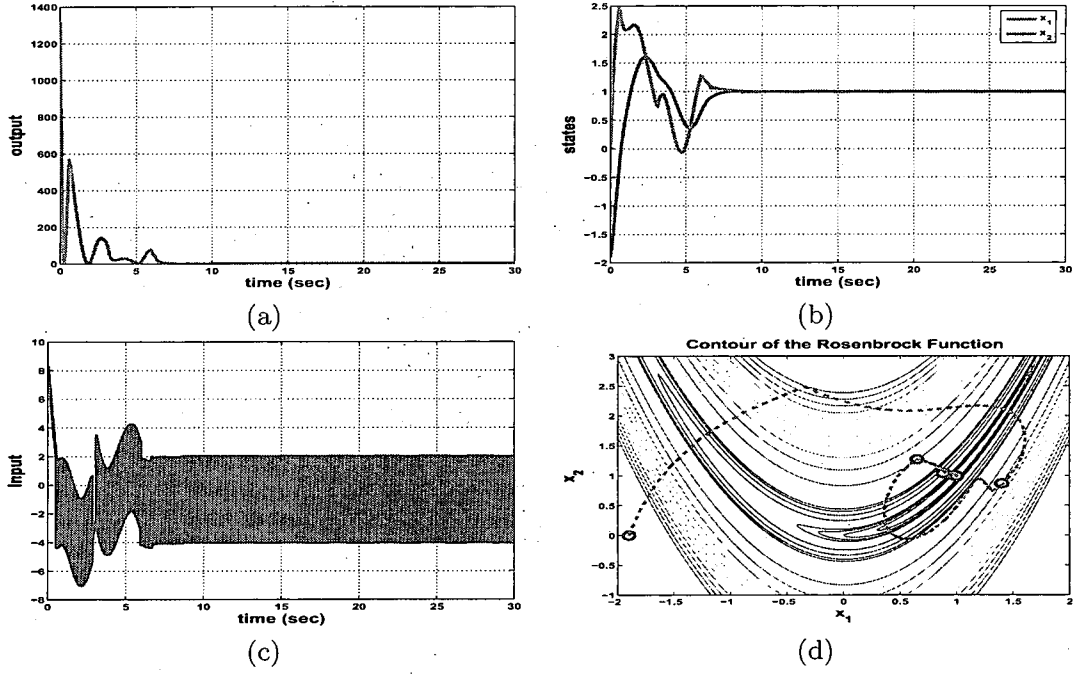


Figure 4.5: Stabilizing controller using signum function for bounded input disturbance, $k_1 = 2, k = 2, 1 \leq \delta_k \leq 3, \alpha_k = 0.001, \beta = 3$: (a) performance output; (b) state; (c) control input; (d) phase portrait.

$\tilde{\theta}_2 = \hat{\theta}_2 - \theta_2^*$. Now we redefine the controller as

$$u = \underbrace{\frac{1}{\mathcal{F}_2(z, \hat{\theta}_2)} \left(-\chi(z) - \mathcal{F}_1(z, \hat{\theta}_1) - k e_z \right)}_{u_{FL}} + u_s, \quad (4.28)$$

where u_s is a stabilizing term defined later. Now, we choose the Lyapunov candidate

$$V = \frac{1}{2} e_z^2 + \frac{1}{2\gamma_1} \tilde{\theta}_1^\top \tilde{\theta}_1 + \frac{1}{2\gamma_2} \tilde{\theta}_2^\top \tilde{\theta}_2,$$

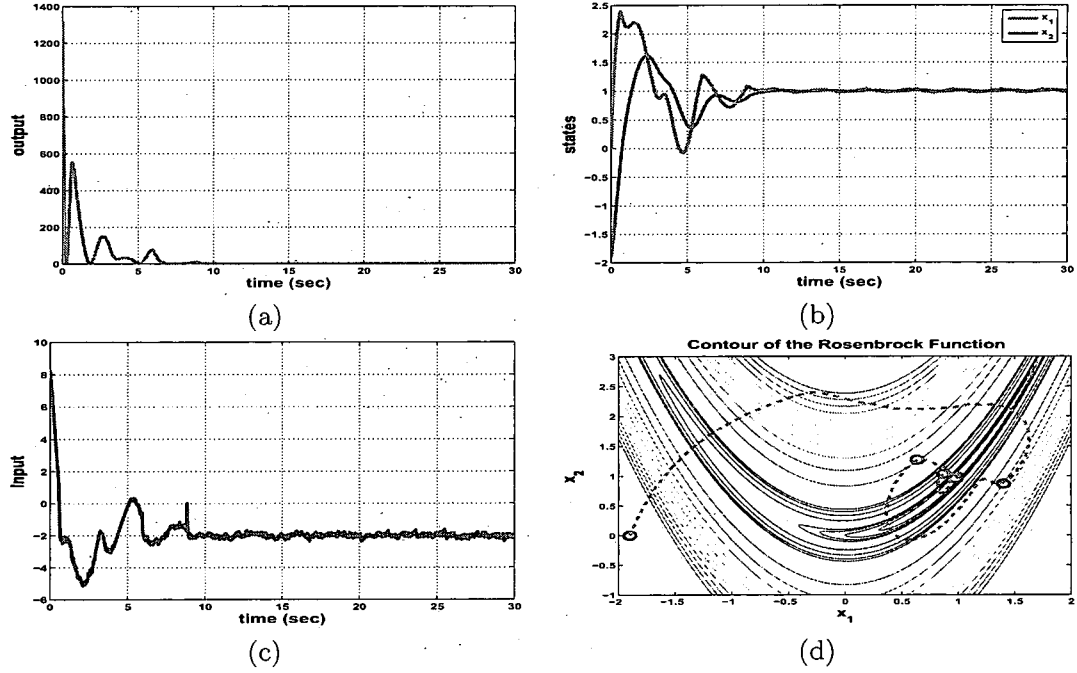


Figure 4.6: Smoothed stabilizing controller for bounded input disturbance, $k_1 = 2, k = 2, \alpha_k = 0.0012, \beta = 3, c = 0.1$: (a) performance output; (b) state; (c) control input; (d) phase portrait.

where γ_1, γ_2 are some positive constants. Note that

$$\begin{aligned}
 \dot{e}_z &= \chi(z) + f(z) + g(z)(u_{FL} + u_s) \\
 &= \chi(z) + f(z) + g(z)(u_{FL} + u_s) + \mathcal{F}_2(z, \hat{\theta}_2)(u_{FL} - u_{FL}) \\
 &= (\mathcal{F}_1(z, \theta_1^*) + \omega_1(z)) + (\mathcal{F}_2(z, \theta_2^*) + \omega_2(z))(u_{FL} + u_s) - \mathcal{F}_1(z, \hat{\theta}_1) - ke_z - \mathcal{F}_2(z, \hat{\theta}_2)u_{FL} \\
 &= -ke_z - \mathcal{F}_1(z, \tilde{\theta}_1) - \mathcal{F}_2(z, \tilde{\theta}_2)u_{FL} + \omega_1(z) + \omega_2(z)u_{FL} + g(z)u_s.
 \end{aligned}$$

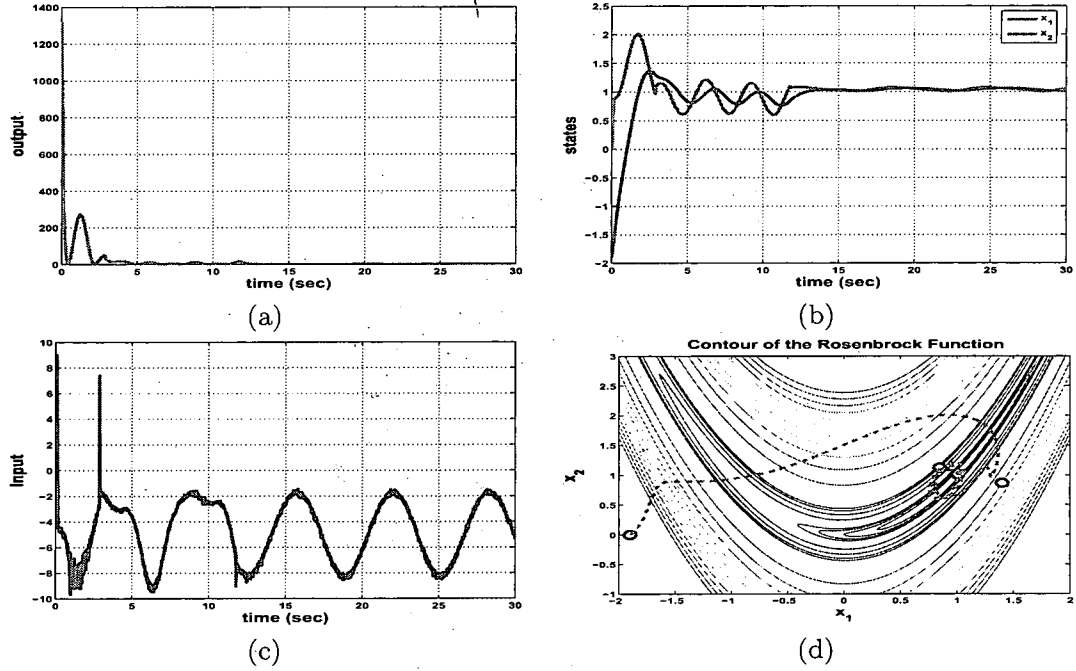


Figure 4.7: Stabilizing controller for unbounded input disturbance, $k_1 = 1, k = 3, \alpha_k = 0.0012, \eta = 2$: (a) performance output; (b) states; (c) control input; (d) phase portrait.

Then,

$$\begin{aligned} \dot{V} &= e_z \dot{e}_z + \frac{1}{\gamma_1} \tilde{\theta}_1^\top \dot{\tilde{\theta}}_1 + \frac{1}{\gamma_2} \tilde{\theta}_2^\top \dot{\tilde{\theta}}_2 \\ &= -k e_z^2 - \tilde{\theta}_1^\top \xi_1(z) e_z - \tilde{\theta}_2^\top \xi_2(z) u_{FL} e_z + \frac{1}{\gamma_1} \tilde{\theta}_1^\top \dot{\tilde{\theta}}_1 + \frac{1}{\gamma_2} \tilde{\theta}_2^\top \dot{\tilde{\theta}}_2 + \omega_1(z) e_z + \omega_2(z) u_{FL} e_z + g(z) u_s e_z \end{aligned}$$

Choose parameter update laws and the stabilizing term to be

$$\dot{\tilde{\theta}}_1 = \gamma_1 \xi_1(z) e_z, \quad (4.29)$$

$$\dot{\tilde{\theta}}_2 = \gamma_2 \xi_2(z) u_{FL} e_z, \quad (4.30)$$

$$u_s = \frac{1}{g_0} (-W_1 - W_2 |u_{FL}|) \text{sgn}(e_z). \quad (4.31)$$

Now, we have

$$\begin{aligned}
\dot{V} &\leq -ke_z^2 + |\omega_1(z)e_z| + |\omega_2(z)u_{FL}e_z| + g(z)u_s e_z \\
&\leq -ke_z^2 + W_1|e_z| + W_2|u_{FL}||e_z| + \frac{g(z)}{g_0}(-W_1 - W_2|u_{FL}|)|e_z| \\
&\leq -ke_z^2
\end{aligned}$$

Therefore, we can first see that the system is stable and $e_z, \tilde{\theta}_1, \tilde{\theta}_2$ are bounded. We can further conclude that e_z asymptotically converges to zero by LaSalle-Yoshizawa theorem [131].

4.3.2 Direct Adaptive Control

Secondly, instead of approximating the plant dynamics we can directly approximate the controller (4.8). In order to do this, we need two more assumptions: $0 < g_0 \leq g(z) < g_1 \leq \infty$ and $|\dot{g}(z)| < B < \infty$ for $z \in S$, where g_0 and B are known constants. Now let u_{FL}^* represents the nominal controller in (4.8), and we assume that the function approximator will approximate it within a compact set $S \subset \mathbb{R}^n$, that is,

$$u_{FL}^* = \mathcal{F}_u(z, \theta_u^*) + \omega_u(z), \quad (4.32)$$

where $\mathcal{F}_u(z, \theta_u^*) = \theta_u^{*\top} \xi_u(z)$ is a function approximator using basis function $\xi_u(z)$. Moreover, $\theta_u^* \in \mathbb{R}^{p_u}$ are unknown optimal parameters such that for arbitrary $z \in S$ we have $|\omega_u(z)| \leq W_u$ for some known constant W_u , which is the smallest possible given p_u, S . Let $\hat{\theta}_u$ be the estimate of θ_u^* and $\tilde{\theta}_u = \hat{\theta}_u - \theta_u^*$. We design the controller as

$$u = \mathcal{F}_u(z, \hat{\theta}_u) + u_s, \quad (4.33)$$

where u_s is a stabilizing term defined later. Now, we choose the Lyapunov candidate

$$V = \frac{1}{2g(z)}e_z^2 + \frac{1}{2\gamma_u}\tilde{\theta}_u^\top \tilde{\theta}_u,$$

where γ_u is a positive constant. Note that

$$\begin{aligned}
\dot{e}_z &= \chi(z) + f(z) + g(z)u \\
&= \chi(z) + f(z) + g(z)(\mathcal{F}_u(z, \hat{\theta}_u) + u_s + \mathcal{F}_u(z, \theta_u^*) - \mathcal{F}_u(z, \theta_u^*)) \\
&= \chi(z) + f(z) + g(z)(\tilde{\theta}_u^\top \xi_u(z) + u_s + u_{FL}^* - \omega_u(z)) \\
&= -ke_z + g(z) \left(\tilde{\theta}_u^\top \xi_u(z) + u_s - \omega_u(z) \right).
\end{aligned}$$

Then,

$$\begin{aligned}
\dot{V} &= \frac{1}{g(z)} e_z \dot{e}_z - \frac{\dot{g}(z) e_z^2}{2g(z)^2} + \frac{1}{\gamma_u} \tilde{\theta}_u^\top \dot{\hat{\theta}}_u \\
&= -\frac{k}{g(z)} e_z^2 + \tilde{\theta}_u^\top \xi_u(z) e_z + (u_s - \omega_u(z)) e_z - \frac{\dot{g}(z) e_z^2}{2g(z)^2} + \frac{1}{\gamma_u} \tilde{\theta}_u^\top \dot{\hat{\theta}}_u
\end{aligned}$$

Choose the parameter update law and stabilizing term to be

$$\dot{\hat{\theta}}_u = -\gamma_u \xi_u(z) e_z \quad (4.34)$$

$$u_s = - \left(W_u + \frac{B}{2g_0^2} |e_z| \right) \text{sgn}(e_z). \quad (4.35)$$

Now, we have

$$\begin{aligned}
\dot{V} &\leq -\frac{k}{g(z)} e_z^2 + |w_u(z) e_z| + \left| \frac{\dot{g}(z) e_z^2}{2g(z)^2} \right| + u_s e_z \\
&\leq -\frac{k}{g(z)} e_z^2 + W_u |e_z| + \frac{B}{2g_0^2} |e_z|^2 - \left(W_u + \frac{B}{2g_0^2} |e_z| \right) |e_z| \\
&\leq -\frac{k}{g(z)} e_z^2 \leq -\frac{k}{g_1} e_z^2.
\end{aligned} \quad (4.36)$$

Therefore, we can first see that the system is stable and $e_z, \tilde{\theta}_u$ are bounded. We can further conclude that e_z asymptotically converges to zero by LaSalle-Yoshizawa theorem [131].

Moreover, we can relax the assumption of knowing W_1, W_2 and W_u , which can be estimated online as well. Also both stabilizing controllers (4.31) and (4.35) use

the singum function, but we can choose a continuous stabilizing term to approximate the signum function similarly to (4.19). In this case, we will have that the error e_z is uniformly ultimately bounded instead of asymptotically convergent to zero. Therefore, again we will have the closed loop system is stable and the state x will converge to a neighborhood of the global minimizer x^* .

4.3.3 Simulations

Without exact knowledge of the plant dynamics, we can use indirect or direct adaptive control to robustify the extremum seeking design. Simulation results for indirect adaptive controller and direct adaptive control can be found in Figure 4.8 and Figure 4.9, respectively. In both cases, we use saturation function to approximate the signum in the stabilizing controller term. We can see that such adaptive controllers can not only deal with the unknown plant dynamics, but also the unknown input disturbance. In the indirect adaptive control case, the effect of input disturbance will be taken account into $f(z)$ and the function approximator will try to approximate $f(z) + g(z)\Delta(z)$ together. Similarly, in the direct adaptive control case, the function approximator will approximate the robust controller such as (4.17) to deal with the input disturbance implicitly. Therefore, we have seen that advanced control techniques can be incorporated in the robust design of extremum seeking control.

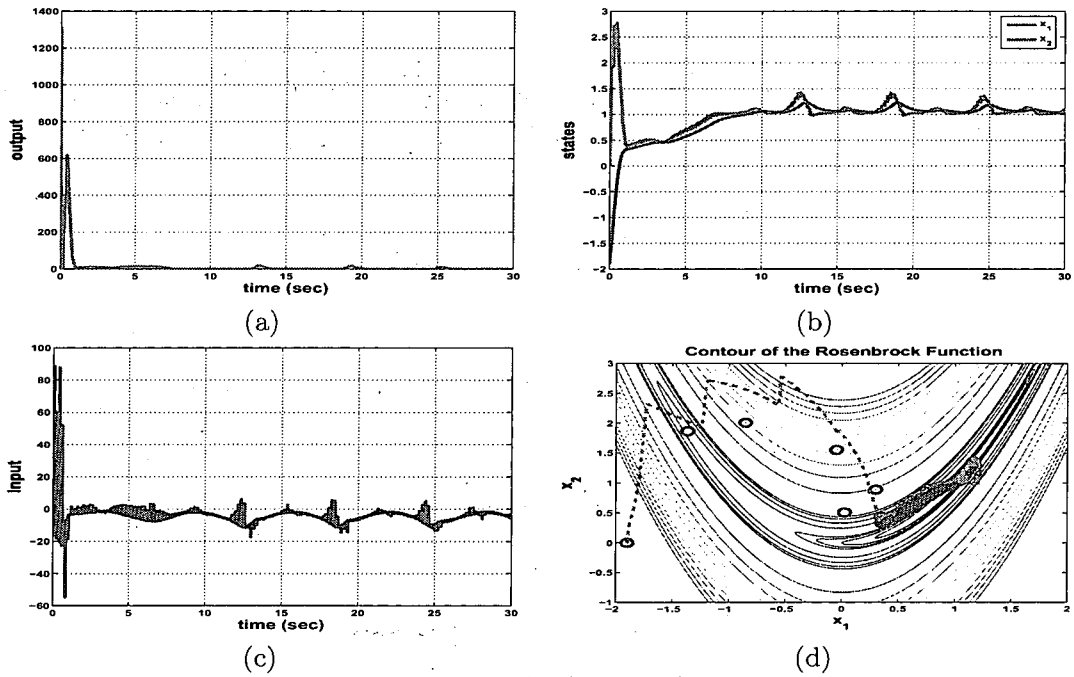


Figure 4.8: Indirect adaptive control for unknown plant dynamics with unknown unbounded input disturbance, $k_1 = 2, k = 2, p_1 = p_2 = 3, \alpha_k = 0.0007, g_0 = 0.9, W_1 = 10, W_2 = 10, \gamma_1 = 2, \gamma_2 = 2$: (a) performance output; (b) state; (c) control input; (d) phase portrait.

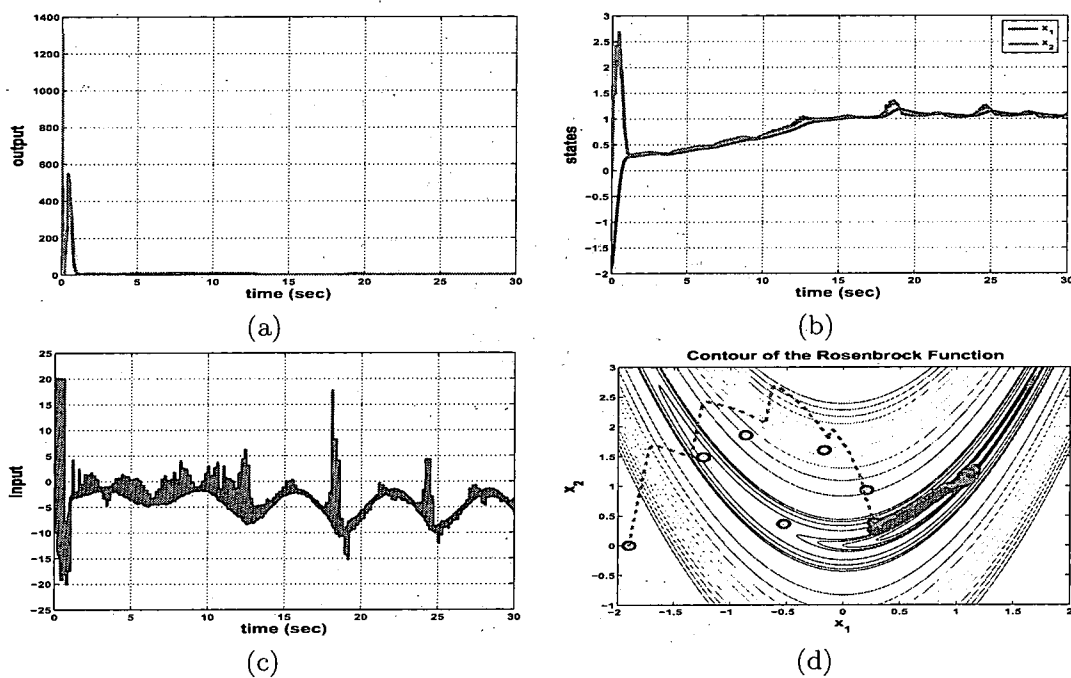


Figure 4.9: Direct adaptive control for unknown plant dynamics with unknown unbounded input disturbance, $k_1 = 2, k = 2, p_u = 3, \alpha_k = 0.001, g_0 = 0.9, B = 0, W_u = 20, \gamma_u = 5$: (a) performance output; (b) state; (c) control input; (d) phase portrait.

CHAPTER 5

ANTILOCK BRAKING SYSTEMS

Antilock braking systems (ABS) are an important tool in the automotive industry. They stop the vehicle faster and make safer turns when the wheels are prevented from locking. ABS design was proposed to deal with braking on slippery surfaces, i.e., to prevent the wheels from locking and skidding. Due to the nonlinearity and uncertainty in the braking systems, the design of ABS is difficult. The character of the friction force acting on the tires has a maximum for a low (nonzero) wheel slip and decreases as the slip increases. Standard ABS systems apply braking pressure in a rapid intermittent fashion. In some of them, the purpose of the intermittent action is to “seek” the maximum of the friction characteristic. In this chapter, we study the ABS design via different extremum seeking control schemes; our goal is to design a control algorithm for the braking torque to achieve maximal friction force without prior knowledge of the optimal slip. The wheel model and the perturbation-based extremum seeking design are taken from Chapter 7 of [3].

5.1 Model of a Slipping Wheel

Consider the single wheel model depicted in Figure 5.1. The tire dynamics are

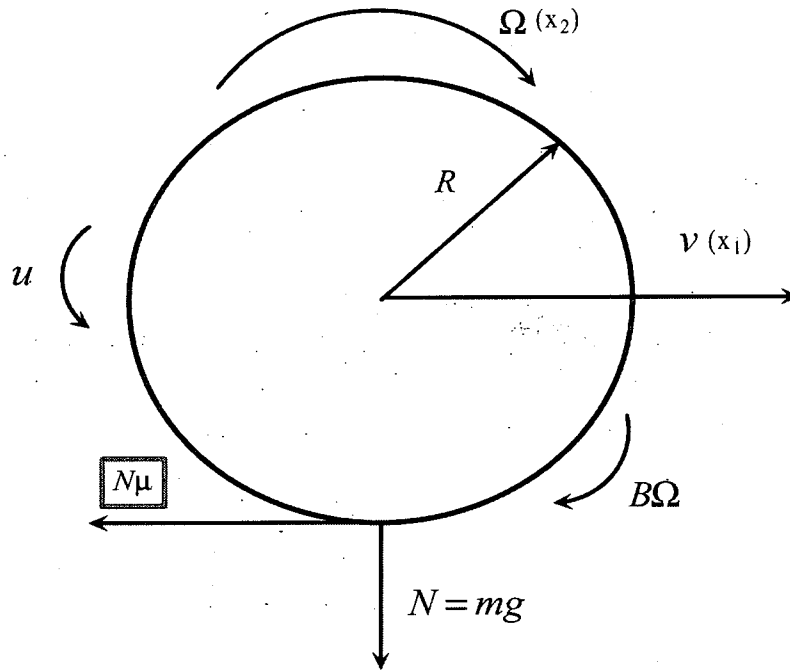


Figure 5.1: The wheel forces.

described as in [3]

$$m\dot{x}_1 = -N\mu(\lambda) \quad (5.1)$$

$$I\dot{x}_2 = -Bx_2 + NR\mu(\lambda) - u, \quad (5.2)$$

where x_1 is the linear velocity v and x_2 is the angular velocity Ω of the wheel, m is the mass, $N = mg$ is the weight of the wheel, R is the radius of the wheel, I is the moment of inertia of the wheel, Bx_2 is the braking friction torque, u is the braking torque, $\mu(\lambda)$ is the friction force coefficient and the wheel slip λ is defined as

$$\lambda = \frac{x_1 - Rx_2}{x_1} \quad (5.3)$$

for $Rx_2 \leq x_1$. There exists a maximum μ^* for the friction force coefficient $\mu(\lambda)$ at λ^* , but λ^*, μ^* will change as the road conditions change. The friction force coefficient $\mu(\lambda)$ is shown in Figure 5.2 for three road conditions. Now the purpose of the ABS design is to generate a control input u such that the friction force coefficient $\mu(\lambda)$ is maximized regardless of the road conditions. Moreover, even though the knowledge of $\mu(\lambda)$ is not available, we are able to obtain the measurement of $\mu(\lambda)$ from (5.1) given the linear acceleration \ddot{x}_1 is measured via an accelerometer.

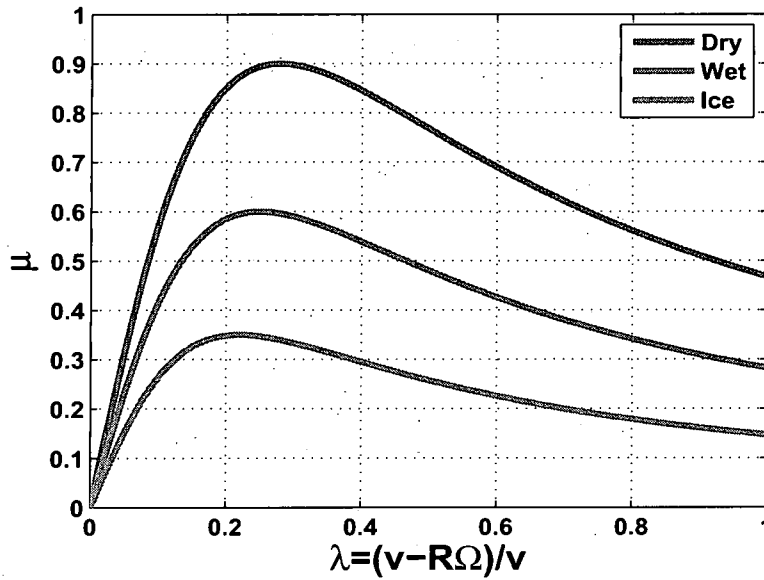


Figure 5.2: Friction force coefficient.

5.2 Perturbation-based Extremum Seeking Control Design

In order to formulate the problem into the perturbation-based extremum seeking setting, let us introduce a constant λ_0 (which is unknown) and define $\tilde{\lambda} = \lambda - \lambda_0$. The governing equation for $\tilde{\lambda}$ is

$$\dot{\tilde{\lambda}} = \dot{\lambda} = \left(\frac{Rx_2}{x_1^2} + \frac{mR^2}{Ix_1} \right) \dot{x}_1 + \frac{RB}{Ix_1} x_2 + \frac{R}{Ix_1} u \quad (5.4)$$

Since \dot{x}_1 is measurable via an accelerometer, it is easy to see that the simple feedback linearizing controller

$$u = -\frac{cIx_1}{R}(\lambda - \lambda_0) - Bx_2 - \frac{Ix_2}{x_1}\dot{x}_1 - mR\dot{x}_1, \quad (5.5)$$

where c is a positive constant, makes the equilibrium λ_0 of the system (5.4) exponentially stable, giving $\dot{\tilde{\lambda}} = -c\tilde{\lambda}$. Note that in the control u (5.5) we do not require the knowledge of the unknown function $\mu(\lambda)$. Then the wheel model under feedback controller (5.5) can be written as a cascade of input dynamics and a static map:

$$\begin{aligned} \frac{1}{c}\dot{\tilde{\lambda}} &= -(\lambda - \lambda_0), \\ y &= \mu(\lambda). \end{aligned} \quad (5.6)$$

We can apply the perturbation-based extremum seeking control scheme given in Figure 5.3 with

$$\lambda_0 = \hat{\lambda}_0 + \alpha \sin(\omega t).$$

For simulation purpose, we postulate a simple function that qualitatively matches $\mu(\lambda)$ as in [3]:

$$\mu(\lambda) = 2\mu^* \frac{\lambda^* \lambda}{\lambda^{*2} + \lambda^2}. \quad (5.7)$$

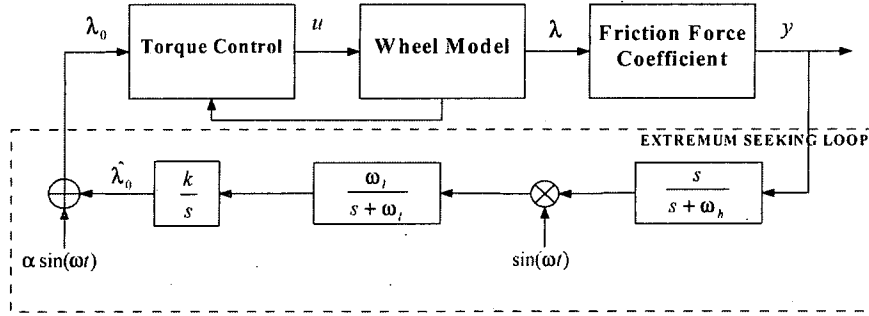


Figure 5.3: ABS design via perturbation-based extremum seeking control.

This function has a maximum at $\lambda = \lambda^*$, whose value is $\mu(\lambda^*) = \mu^*$. We run simulation for $\lambda^* = 0.25$ and $\mu^* = 0.6$. The parameters of the wheel are chosen as: $m = 400\text{kg}$, $B = 0.01$, $R = 0.3\text{m}$. Initial conditions are linear velocity $x_1(0) = 33.33\text{m/s}$; angular velocity $x_2(0) = 400/3.6$, which makes $\lambda(0) = 0$. The simulation employs perturbation-based extremum seeking scheme with $\alpha = 0.01$, $\omega = 3$, $\omega_h = 0.6$, $\omega_l = 0.8$, $c = 20$ and $k = 1.5$. For $\lambda_0 = 0.1$, the simulation results are shown in Figure 5.4. It is seen that during braking, maximum friction force is reached and the car is stopped within the shortest time and distance. The low pass filter in the design can be removed without loss of stability, i.e., $\omega_l = 0$. Its purpose is to attenuate noise in the loop.

5.3 Sliding Mode-based Extremum Seeking Control Design

We can easily use sliding mode extremum seeking to replace the sinusoidal perturbation-based extremum seeking one. By using the same torque controller (5.5), we apply the

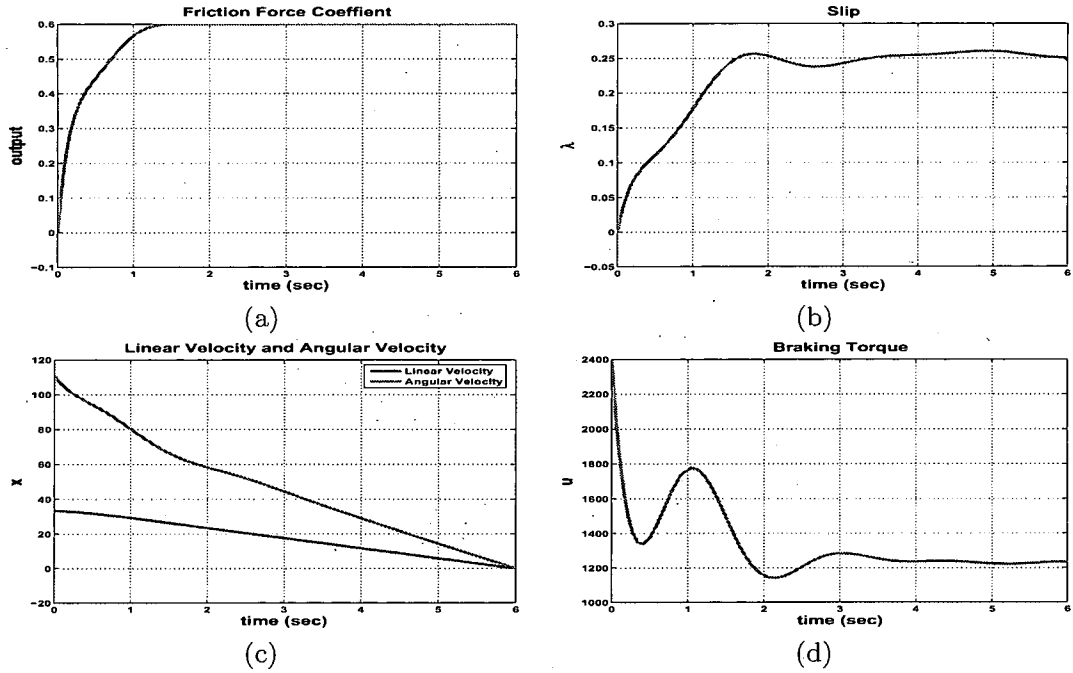


Figure 5.4: ABS design via perturbation-based extremum seeking control: (a) friction force coefficient; (b) slip; (c) linear velocity and angular velocity; (d) braking torque.

sliding mode-based extremum seeking scheme given in Figure 5.5 with

$$\dot{\lambda}_0 = k \text{sgn} \sin(\pi s / \alpha),$$

where α is a positive constant, s is a switching function defined as $s(t) = y - g(t)$ and $g(t)$ is an increasing function satisfying $\dot{g}(t) = \rho > 0$.

In the simulation, $\mu(\lambda)$ is postulated as (5.7), the simulation employs sliding mode-based extremum seeking scheme with $\alpha = 0.1$, $\rho = 1$, $c = 20$, and $k = 1.5$. For the same wheel parameters and initial conditions as in Section 5.2, the simulation results are shown in Figure 5.6. It is seen that during braking, maximum friction force is reached and the car is stopped within the shortest time and distance.

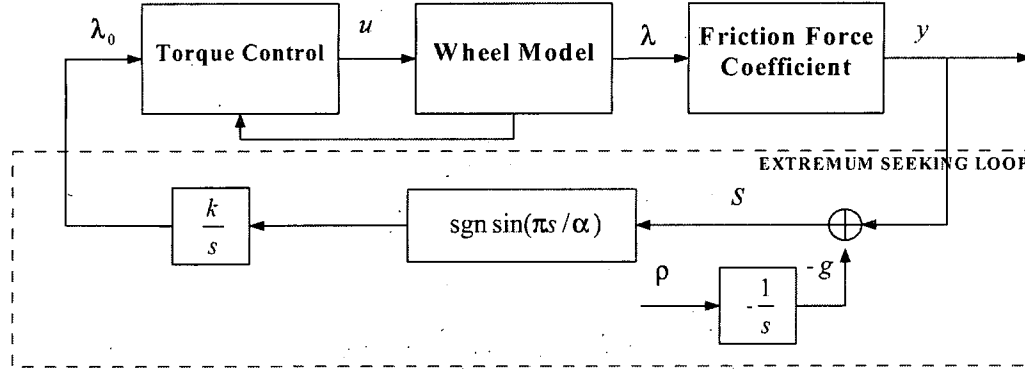


Figure 5.5: ABS design via sliding mode-based extremum seeking control.

5.4 Numerical Optimization-based Extremum Seeking Control Design

By observing (5.1), we find that x_1 is not controllable from u given $\mu(\lambda)$ fixed. Fortunately, the friction force coefficient μ is only dependent on λ , which is a function of x_1 and x_2 , and it maybe controllable by u . Therefore, we try to linearize the system from the input u to output λ . We define the change of variables $\eta = x_1$, $\lambda = (x_1 - Rx_2)/x_1$ that transforms the system (5.1), (5.2) into the form

$$\dot{\eta} = -N\mu(\lambda)/m \quad (5.8)$$

$$\dot{\lambda} = -\left(\frac{R}{Ix_1} + \frac{x_2}{mx_1^2}\right)NR\mu(\lambda) + \frac{R}{Ix_1}(u + Bx_2). \quad (5.9)$$

Since η is the linear velocity, it will be bounded all the time due to physical restrictions. Then, given $x_1(0) > 0$, let the braking torque be

$$u = \frac{Ix_1}{R}(-c\lambda + v) - Bx_2 + \left(\frac{R}{Ix_1} + \frac{x_2}{mx_1^2}\right)IN\mu(\lambda)x_1. \quad (5.10)$$

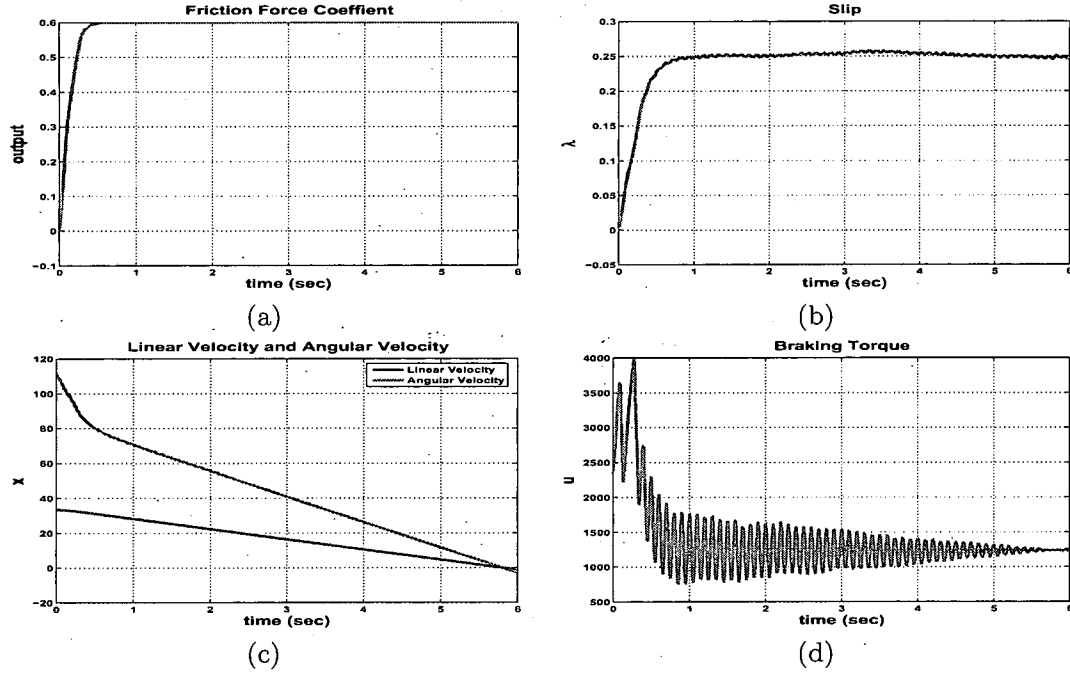


Figure 5.6: ABS design via sliding mode-based extremum seeking control: (a) friction force coefficient; (b) slip; (c) linear velocity and angular velocity; (d) braking torque.

Then (5.9) becomes

$$\dot{\lambda} = -c\lambda + v, \quad (5.11)$$

where v is the regulator defined as in (3.10). A block diagram of extremum seeking scheme for the wheel model can be found in Figure 5.7. That is, by designing the control torque as in (5.10), we are able to adjust the slip λ to maximize the friction force coefficient.

In the simulation, $\mu(\lambda)$ is postulated as (5.7), same wheel parameters and initial conditions as in Section 5.2 are used in the simulations. The simulation results for line search-based extremum seeking control are shown in Figure 5.8, where $c = 1$, $\delta_k = 0.5$.

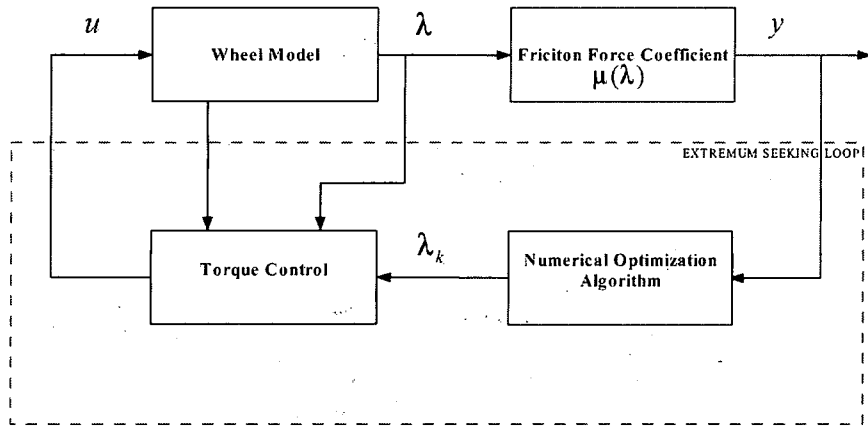


Figure 5.7: ABS design via numerical optimization-based extremum seeking control.

However, since the finite time state regulator can not render the λ^* as a equilibrium point of the closed loop system, we can observe the steady state slip oscillates. The simulation results based on trust region method are shown in Figure 5.9, where no gradient measurements of the $\mu(\lambda)$ is needed.

We further apply the output tracking design for the state regulator. This time, there is no need to put the design into the output tracking framework since we only have one state to control via one control input. We can render any λ_k to be an equilibrium point by choosing the following asymptotic controller

$$v = c\lambda_k$$

for linearized system in (5.11). Let $e = \lambda - \lambda_k$, we have

$$\dot{e} = \dot{\lambda} = c\lambda + v = -c\lambda + c\lambda_k = -ce,$$

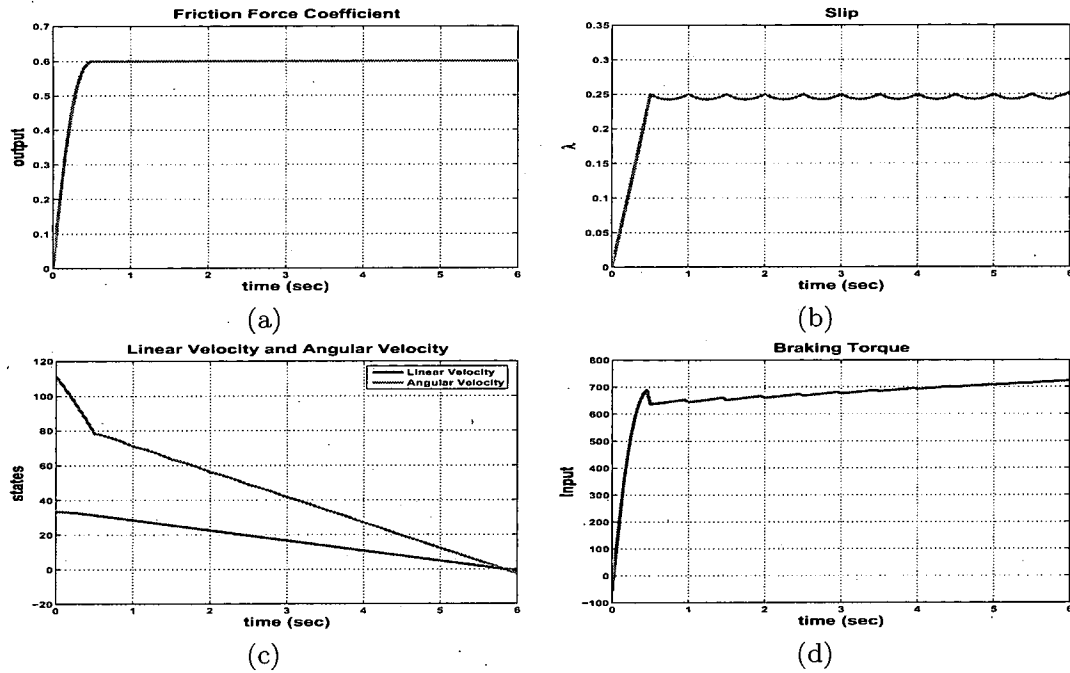


Figure 5.8: ABS design via line search-based extremum seeking control: (a) friction force coefficient; (b) slip; (c) linear velocity and angular velocity; (d) braking torque.

which implies e asymptotically converges to zero as long as $c > 0$. Now, the simulation results for line search or trust region-based extremum seeking design can be found in Figure 5.10 and Figure 5.11.

5.5 Comparison

1. Given only one parameter needs to be tuned among different extremum seeking control schemes, the perturbation-based and sliding mode-based design turn out to be easy to tune.

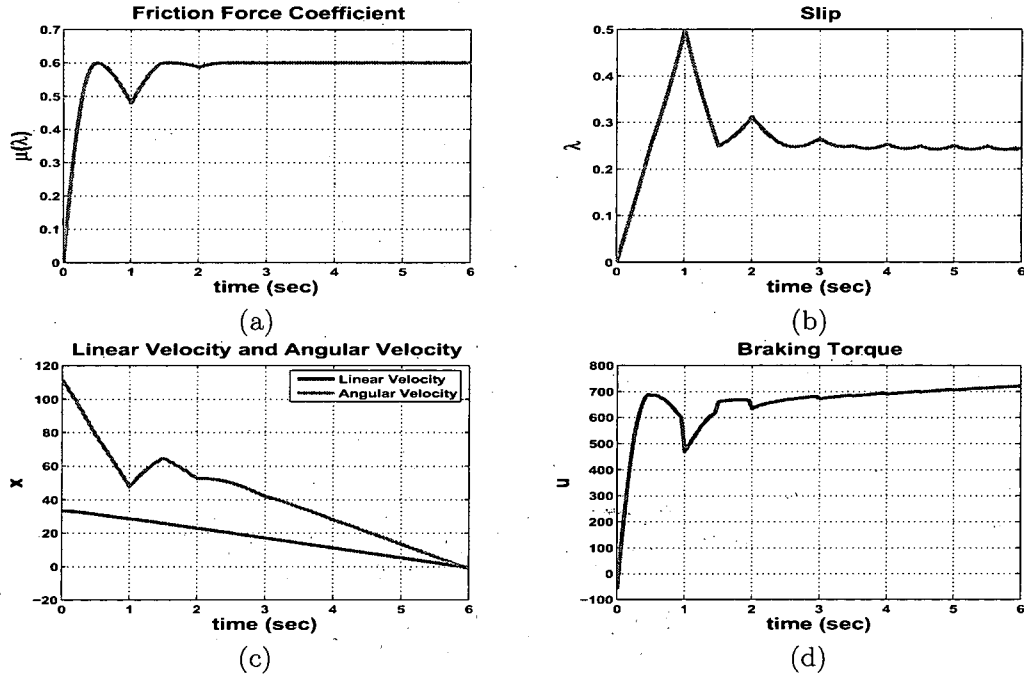


Figure 5.9: ABS design via trust region-based extremum seeking control: (a) friction force coefficient; (b) slip; (c) linear velocity and angular velocity; (d) braking torque.

2. The design via perturbation and sliding mode-based extremum seeking control do not need the gradient, however, they do bring additionally oscillations due to the perturbation signal and the sliding mode, respectively.
3. Further research to deal with oscillations can be found in [49], time delay can be found in [55].

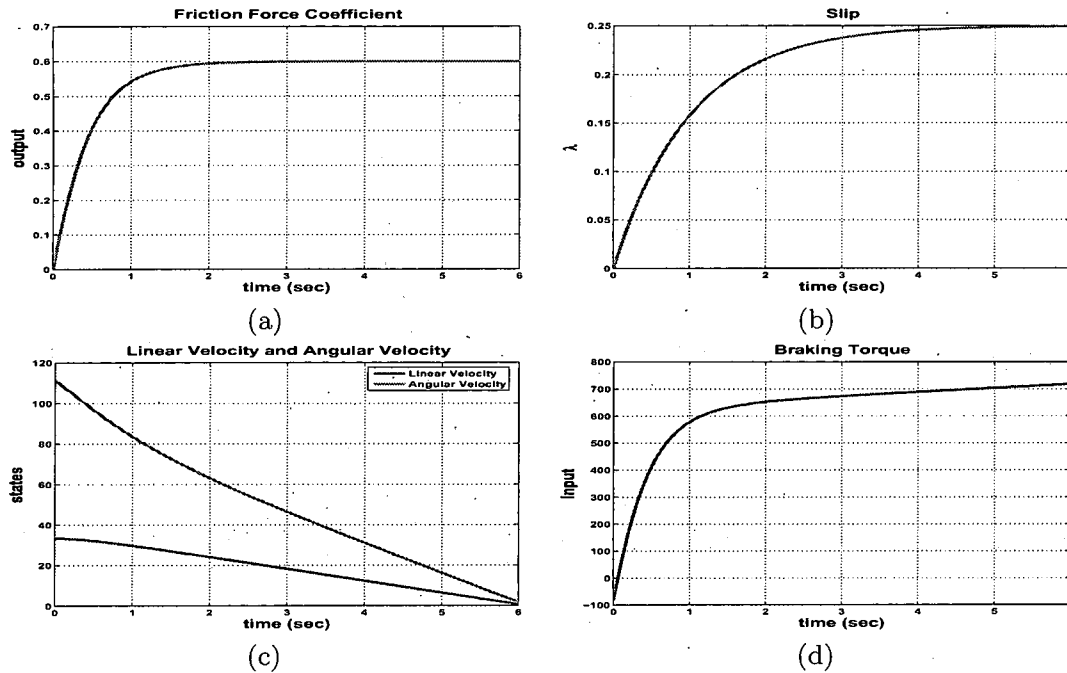


Figure 5.10: ABS design via extremum seeking control based on line search and output tracking: (a) friction force coefficient; (b) slip; (c) linear velocity and angular velocity; (d) braking torque.

4. The design via numerical optimization-based extremum seeking can be made gradient free as well via derivative free trust region method, however, the convergence will be relative slow due to the slow convergence of the optimization method.
5. Oscillation is successfully avoided in the numerical optimization-based extremum seeking control because no perturbation signal is used or no sliding mode function is introduced.

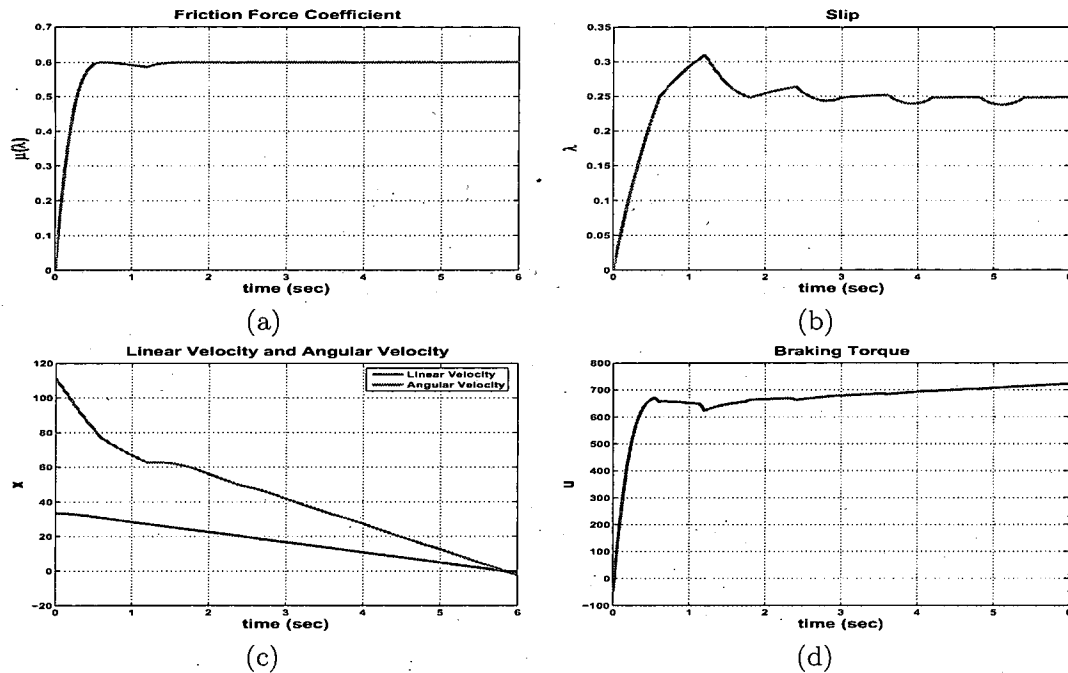


Figure 5.11: ABS design via extremum seeking control based on trust region and output tracking: (a) friction force coefficient; (b) slip; (c) linear velocity and angular velocity; (d) braking torque.

6. Additional robust term can be easily added to the numerical optimization-based design to deal with input disturbance and unmodeled plant dynamics.

CHAPTER 6

SOURCE SEEKING

6.1 Introduction

In the rapidly growing literature on coordinated motion control and autonomous agents, "autonomy" never means deprivation of position information. The vehicles are always assumed to have GPS and/or INS on board. There is however interest in developing vehicles with full autonomy. The reasons are two-fold: (1) applications under water, under ice, or in caves where GPS is unavailable, and (2) the high cost of INS systems that remain accurate over longer periods of time.

Control of autonomous vehicles is an immensely active area of research dealing with a multitude of objectives, under a great variety of constraints. Typically, autonomous agents are allowed some amount of information sharing and are supplied with at least their position measurements. In this chapter, we consider the problem of seeking the source of a scalar signal. One can imagine such signal to be the concentration of a chemical or biological agent, an electromagnetic, acoustic, or even thermal signal. The strength of the signal is assumed to decay away from the source (though not necessarily in a symmetric/concentric pattern), however, the shape of the signal field is not available to the seeking vehicle. The seeking vehicle has access

only to the *value* of the signal at its location. Traditional target tracking problem can be putting into the framework of source seeking by establishing an attraction potential function between the vehicle and the target. We use extremum seeking to address a problem with complete autonomy—a vehicle, without any position or velocity information, tracks the source of a scalar valued signal, extracting the implicit position information through gradient estimation that extremum seeking performs in a non-model based manner. Motions that allow gradient estimation are much harder to achieve with the unicycle than with a point mass, however we do succeed in developing a simple and provably stable scheme that drives our kinematically constrained vehicle towards the signal source. We mainly focus on the design of source seeking via perturbation-based extremum seeking controller, the application of numerical optimization-based extremum seeking controller to source seeking is straightforward and will be mentioned in Chapter 7. The utilization of sliding mode-based design is in need of future research.

The classical perturbation-based extremum seeking scheme is modified for the stated task by observing that the integrator, a key adaptation element, is already present in vehicle models where the primary forces or moments acting on the vehicle are those that provide thrust/propulsion, i.e., for vehicles that act primarily in the $m\ddot{x} = F$ manner, where F is the motion-generating input and \ddot{x} is the acceleration vector. In this chapter, we present results for a point mass model in the plane. An extension to 3D is trivial, except that one has to employ separate probing frequencies in the extremum seeking algorithm for the individual axes of motion. The extension to point mass models with extensive losses (due for example to drag) is straightforward by noting that the input-output relationship drops in relative degree,

making the problem actually easier. Drift-inducing forces like gravity or buoyancy are automatically accommodated by extremum seeking which auto-tunes the input to compensate for such constant disturbances. An extension to underactuated or nonholonomic vehicle models is not straightforward and will be present here as well. The novelty of our result here is in simultaneously solving a (nonholonomic) steering problem and an adaptive optimization problem. The stability results we prove are local. The techniques introduced by Tan, Nesic, and Mareels [39] can be used to achieve semi-global versions of our results.

Somewhat related problems have been considered in the past. Porat and Ne-horai [138] considered the problem of localizing vapor-emitting sources in spatio-temporal fields modeled by the heat equation PDE, however their vehicle had position information and could move arbitrarily fast from one point in space to another. Ogren, Fiorelli, and Leonard [139] considered coordination problems with groups of vehicles where each vehicle carries a single sensor and gradient climbing is performed. Gradient estimation in that work is a group effort among vehicles that have position information and communicate amongst themselves.

Unicycle models of autonomous vehicles have been employed in several previous studies of coordinated motion control—by Justh and Krishnaprasad [140] for convergent vehicle formations, by Klein and Morgansen [141] for trajectory tracking, and by Marshall, Broucke, and Francis [142] for the cyclic pursuit problem.

6.2 A Velocity-Actuated Point Mass (Single Integrators)

In the plane, an autonomous vehicle is modeled as a point mass:

$$\begin{aligned}\dot{x} &= v_x \\ \dot{y} &= v_y,\end{aligned}\tag{6.1}$$

where $[x, y]$ is the position of the point mass and v_x, v_y are the velocity inputs. Our method is extended in next section to the case where the inputs are forces, however for clarity in introducing the new concept, we consider the simplest case of a velocity-actuated point mass first.

A block diagram of extremum seeking is shown in Figure 6.1. The nonlinear map

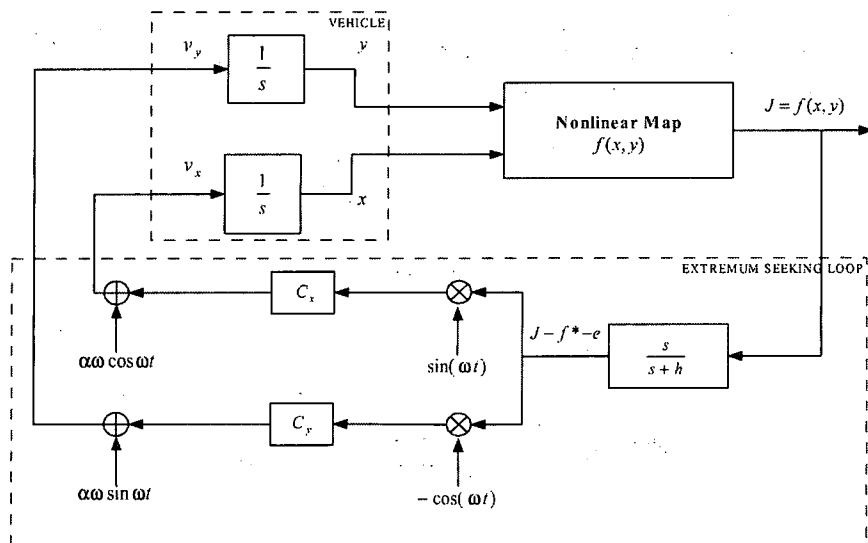


Figure 6.1: Extremum seeking for velocity-actuated point mass.

represents the distribution of the signal being tracked, whose strength will typically

decay away from the origin, thus we assume that the nonlinear map $J = f(x, y)$ has a local maximum and pursue local tracking of that maximum. For clarity of our presentation we assume that the nonlinear map is quadratic and that its Hessian is diagonal, viz.,

$$J = f(x, y) = f^* - q_x(x - x^*)^2 - q_y(y - y^*)^2, \quad (6.2)$$

where (x^*, y^*) is the maximizer, $f^* = f(x^*, y^*)$ is the maximum and q_x, q_y are some unknown positive constants (since the Hessian is negative). General non-quadratic maps with non-diagonal Hessians are equally amenable to analysis, using the same technique as in [30, 3]. We show next that extremum seeking drives the autonomous vehicle to (x^*, y^*) without employing any knowledge of $f(x, y)$ or the measurements of (x, y) , only the measurement of the output J of the nonlinear map $f(x, y)$. The designer chooses the parameters $\alpha, \omega, h, c_x, c_y$ in the block diagram (Figure 6.1), whereas the extremum seeking automatically tunes v_x, v_y to lead the vehicle to the peak of $f(x, y)$.

The analysis that follows employs the method of averaging. Let

$$e = \frac{h}{s + h}[J] - f^*, \quad (6.3)$$

then the signal after the washout filter can be expressed as

$$\frac{s}{s + h}[J] = J - \frac{h}{s + h}[J] = J - f^* - e.$$

Now, let us introduce the new coordinates

$$\tilde{x} = x - x^* - \alpha \sin(\omega t) \quad (6.4)$$

$$\tilde{y} = y - y^* + \alpha \cos(\omega t). \quad (6.5)$$

Then, in the time scale $\tau = \omega t$, we define:

$$\Delta = (J - f^* - e) = -[q_x(\tilde{x} + \alpha \sin \tau)^2 + q_y(\tilde{y} - \alpha \cos \tau)^2 + e]. \quad (6.6)$$

So we summarize the system in Figure 6.1 as

$$\frac{d\tilde{x}}{d\tau} = +\frac{1}{\omega} c_x \Delta \sin \tau \quad (6.7)$$

$$\frac{d\tilde{y}}{d\tau} = -\frac{1}{\omega} c_y \Delta \cos \tau \quad (6.8)$$

$$\frac{de}{d\tau} = +\frac{h}{\omega} \Delta. \quad (6.9)$$

The system (6.7)–(6.9) is in the form to which the averaging method is applicable, provided $1/\omega$ is small, i.e., provided ω is large (relative to the other parameters in the extremum seeking scheme and relative to the parameters in the nonlinear map).

The average model of (6.7)–(6.9) is

$$\frac{d\tilde{x}_{avg}}{d\tau} = -\frac{1}{\omega} \alpha c_x q_x \tilde{x}_{avg} \quad (6.10)$$

$$\frac{d\tilde{y}_{avg}}{d\tau} = -\frac{1}{\omega} \alpha c_y q_y \tilde{y}_{avg} \quad (6.11)$$

$$\frac{de_{avg}}{d\tau} = -\frac{1}{\omega} h \left[q_x \tilde{x}_{avg}^2 + q_y \tilde{y}_{avg}^2 + e_{avg} + \frac{\alpha^2}{2} (q_x + q_y) \right]. \quad (6.12)$$

Then the equilibrium of the average model (6.10)–(6.12) is

$$\tilde{x}_{avg}^e = 0, \quad \tilde{y}_{avg}^e = 0, \quad e_{avg}^e = -\frac{\alpha^2}{2} (q_x + q_y). \quad (6.13)$$

The Jacobian of (6.10)–(6.12) at $(\tilde{x}_{avg}^e, \tilde{y}_{avg}^e, e_{avg}^e)$ is

$$J_{avg} = \frac{1}{\omega} \begin{bmatrix} -\alpha c_x q_x & 0 & 0 \\ 0 & -\alpha c_y q_y & 0 \\ 0 & 0 & -h \end{bmatrix}. \quad (6.14)$$

Given the knowledge that the extremum is a maximum, it follows that q_x, q_y are known to be positive, though their actual values are unknown. Therefore, if we

choose $\alpha > 0, c_x > 0, c_y > 0$ and $h > 0$, the Jacobian (6.14) is Hurwitz and the equilibrium (6.10)–(6.13) of the average system (6.12) is locally exponentially stable. Then according to the averaging theorem [131], we have the following result.

THEOREM 6.2.1 *There exists $\bar{\omega}$ such that for all $\frac{1}{\omega} \in (0, \frac{1}{\bar{\omega}})$ the system in the Figure 6.1 with the nonlinear map of the form (6.2) has a unique exponentially stable periodic solution $(\tilde{x}^{2\pi/\omega}, \tilde{y}^{2\pi/\omega}, e^{2\pi/\omega})$ of period $\frac{2\pi}{\omega}$ and this solution satisfies*

$$\left\| \begin{bmatrix} \tilde{x}^{2\pi/\omega} \\ \tilde{y}^{2\pi/\omega} \\ e^{2\pi/\omega} + \frac{\alpha^2}{2}(q_x + q_y) \end{bmatrix} \right\| \leq O(1/\omega), \quad \forall \tau \geq 0. \quad (6.15)$$

Since

$$x - x^* = \tilde{x} + \alpha \sin(\omega t) = (\tilde{x} - \tilde{x}^{2\pi/\omega}) + (\tilde{x}^{2\pi/\omega} - 0) + \alpha \sin \tau.$$

The above theorem implies that the first term converges to zero, the second term is $O(1/\omega)$, and the third term is $O(\alpha)$. Thus

$$\limsup_{\tau \rightarrow \infty} |x - x^*| = O(\alpha + 1/\omega).$$

Similarly, we can obtain

$$\limsup_{\tau \rightarrow \infty} |y - y^*| = O(\alpha + 1/\omega).$$

Hence, we get

$$\limsup_{\tau \rightarrow \infty} |f - f^*| = O(\alpha^2 + (1/\omega)^2), \quad (6.16)$$

which characterizes the asymptotic performance of the extremum seeking loop in Figure 6.1. The vehicle converges to a neighborhood of the maximum, the size of which is proportional to the square of the amplitude of the periodic perturbation,

and the reciprocal of the square of the perturbation frequency. Since we choose α small and ω large, the tracking error is very small.

It is very important to see that the extremum seeking scheme can be used for tracking of slowly time-varying trajectories, i.e., for tracking moving signal sources/targets. When the trajectories are periodic our stability proof extends with very minor modifications which we don't present here in the interest of space. For example, consider a target motion that is in the shape of the number eight (8),

$$\begin{aligned}x^* &= a_m \sin(\omega_m t) \\ y^* &= a_m \cos(2\omega_m t + \phi_m),\end{aligned}\tag{6.17}$$

where $\omega_m \ll \omega$. If ω and ω_m are commensurate, i.e., if there exist natural numbers N and N_m such that

$$\frac{\omega}{\omega_m} = \frac{N}{N_m},\tag{6.18}$$

then our proof extends, with averaging applied over a period of $2\pi N$ in the τ -time scale to account for the presence of an additional periodic terms on the right hand sides of (6.7) and (6.8). If, however, ω and ω_m are incommensurate (for example, $\omega = 4\pi\omega_m$ or $\omega = 3\sqrt{23}\omega_m$), the technique of general averaging for quasiperiodic systems leads to the same stability conclusions.

We first illustrate the simulation results of seeking a stationary target. The point mass model (6.1) and the quadratic map (6.2) are used in the simulation. We set parameters of the target as $(x^*, y^*) = (-1, -1)$, $f^* = 1$, $q_x = 1$ and $q_y = 0.5$. The parameters of the extremum seeking loop are chosen as $\omega = 30$, $\alpha = 0.08$, $c_x = c_y = 10$ and $h = 1$. The start position of the autonomous vehicle is $(x(0), y(0)) = (1, 1)$. As shown in Figure 6.2 (b), the autonomous vehicle starts at (1, 1) by probing around

to climb the gradient of the unknown map, eventually circling around the maximizer $(-1, -1)$, the output of the unknown signal J is shown in Figure 6.2 (a), while the control inputs are shown in Figures 6.2 (c) and (d).

For the slow time varying target (6.17), the simulation results are shown in Figure 6.3, where we let $a_m = 1, \omega_m = 0.1, \phi_m = 3, f^* = 1, q_x = 1, q_y = 0.5$, and $\omega = 30, \alpha = 0.05, c_x = c_y = 15, h = 1$. The starting position of the autonomous vehicle is still $(x(0), y(0)) = (1, 1)$. Moreover, we found that an increase in the adaptation gain will improve the tracking performance as shown in Figure 6.4 (a), where we increase $c_x = c_y = 30$ while keeping the other settings. Similarly, it is much easier to track a slower maximizer as shown in Figure 6.4 (b), where we set $\omega_m = 0.05$ while keeping the other parameter values.

6.3 A Force-Actuated Point Mass (Double Integrators)

In this section we present a modified scheme for force-actuated point mass models, which instead of single integrators include double integrators. Both the vehicle model and the modified ES scheme are shown in Figure 6.5. One can observe the double integrators in the vehicle model and the presence of phase lead compensators of the form $G(s) = k_c \frac{s - z_0}{s - p_0}$ whose role is to recover some of the phase in the feedback loop lost due to the addition of the second integrator. Four new states are introduced due to the PD compensators w_x, w_y and the additional integrators v_x, v_y . Again, we introduce the new coordinates

$$\tilde{v}_x = v_x - \alpha \omega \cos(\omega t),$$

$$\tilde{v}_y = v_y - \alpha \omega \sin(\omega t).$$

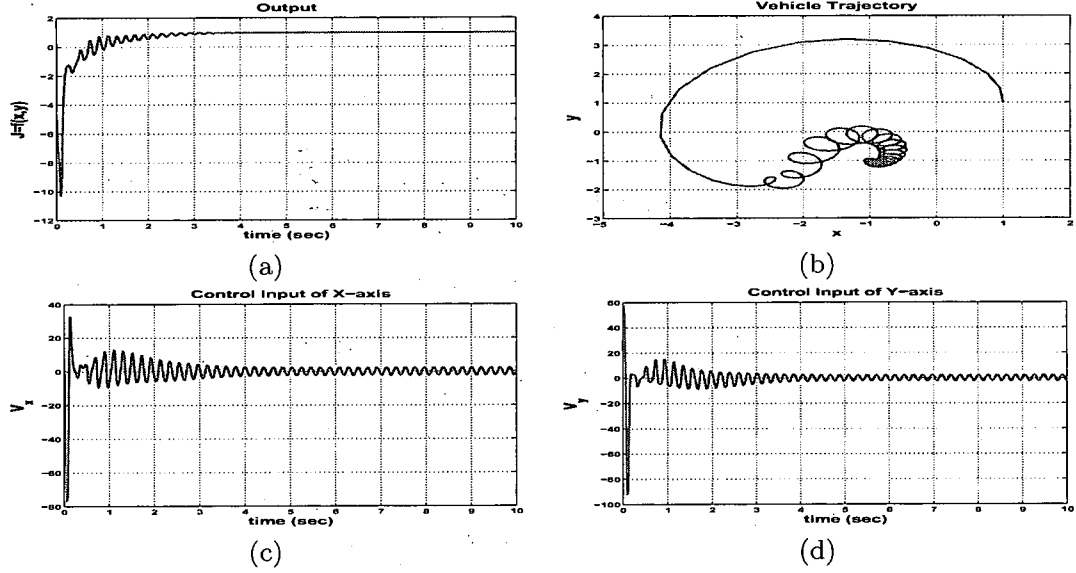


Figure 6.2: Extremum seeking for velocity-actuated point mass, stationary case: (a) output; (b) vehicle trajectory starts from (1,1); (c) control input of x -axis; (d) control input of y -axis.

Then, in the time scale $\tau = \omega t$, we summarize the system in Figure 6.5 as

$$\begin{aligned}
 \frac{d\tilde{x}}{d\tau} &= \frac{1}{\omega} \tilde{v}_x \\
 \frac{d\tilde{y}}{d\tau} &= \frac{1}{\omega} \tilde{v}_y \\
 \frac{de}{d\tau} &= \frac{h}{\omega} \Delta \\
 \frac{d\tilde{v}_x}{d\tau} &= \frac{1}{\omega} w_x \\
 \frac{d\tilde{v}_y}{d\tau} &= \frac{1}{\omega} w_y \\
 \frac{dw_x}{d\tau} &= \frac{1}{\omega} \left[p_x w_x - c_x k_x z_x \Delta \sin \tau + c_x k_x \omega \Delta \cos \tau + c_x k_x \frac{d\Delta}{dt} \sin \tau \right] \\
 \frac{dw_y}{d\tau} &= \frac{1}{\omega} \left[p_y w_y + c_y k_y z_y \Delta \cos \tau + c_y k_y \omega \Delta \sin \tau - c_y k_y \frac{d\Delta}{dt} \cos \tau \right]
 \end{aligned} \tag{6.19}$$

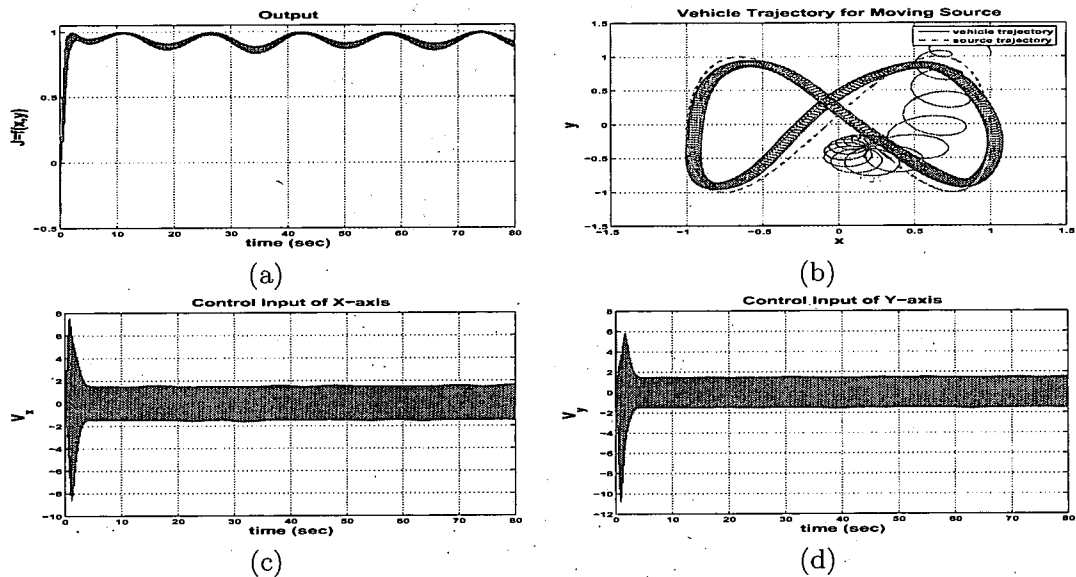


Figure 6.3: Extremum seeking for velocity-actuated point mass, slowly time-varying case: (a) output; (b) vehicle trajectory starts from (1,1) and source trajectory starts from (0,0); (c) control input of x -axis; (d) control input of y -axis.

where Δ is defined in (6.6), and then

$$\begin{aligned} \frac{d\Delta}{dt} = & -2q_x(\tilde{x} + \alpha \sin(\omega t))(\tilde{v}_x + \alpha \omega \cos(\omega t)) \\ & - 2q_y(\tilde{y} - \alpha \cos(\omega t))(\tilde{v}_y + \alpha \omega \sin(\omega t)) - h\Delta. \end{aligned}$$

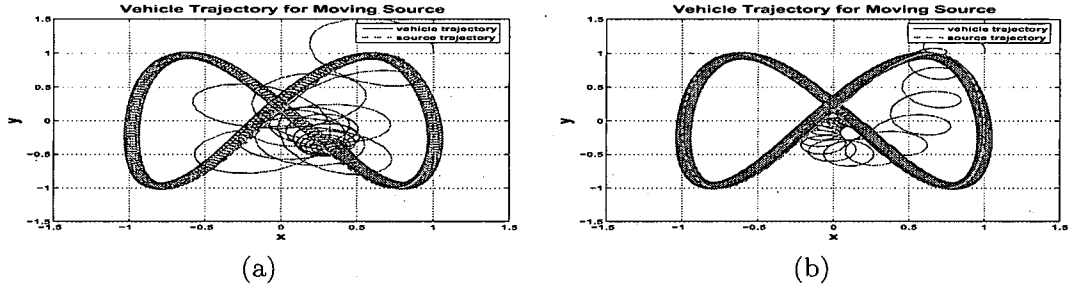


Figure 6.4: Extremum seeking for velocity-actuated point mass, slowly time-varying case: (a) large adaption gain, (b) slower time varying maximizer.

The average model of (6.19) is

$$\begin{aligned}
 \frac{d\tilde{x}_{avg}}{d\tau} &= \frac{1}{\omega} \tilde{v}_{xavg} \\
 \frac{d\tilde{y}_{avg}}{d\tau} &= \frac{1}{\omega} \tilde{v}_{yavg} \\
 \frac{de_{avg}}{d\tau} &= \frac{1}{\omega} (-h) \left[q_x \tilde{x}_{avg}^2 + q_y \tilde{y}_{avg}^2 + e_{avg} + \frac{\alpha^2}{2} (q_x + q_y) \right] \\
 \frac{d\tilde{v}_{xavg}}{d\tau} &= \frac{1}{\omega} w_{xavg} \\
 \frac{d\tilde{v}_{yavg}}{d\tau} &= \frac{1}{\omega} w_{yavg} \\
 \frac{dw_{xavg}}{d\tau} &= \frac{1}{\omega} \left[p_x w_{xavg} + \alpha c_x k_x q_x (z_x + h) \tilde{x}_{avg} - \alpha c_x k_x q_x \tilde{v}_{xavg} \right] \\
 \frac{dw_{yavg}}{d\tau} &= \frac{1}{\omega} \left[p_y w_{yavg} + \alpha c_y k_y q_y (z_y + h) \tilde{y}_{avg} - \alpha c_y k_y q_y \tilde{v}_{yavg} \right]
 \end{aligned} \tag{6.20}$$

and its equilibrium is

$$\tilde{x}_{avg}^e = \tilde{y}_{avg}^e = \tilde{v}_{xavg}^e = \tilde{v}_{yavg}^e = w_{xavg}^e = w_{yavg}^e = 0, e_{avg}^e = -\frac{\alpha^2}{2} (q_x + q_y). \tag{6.21}$$

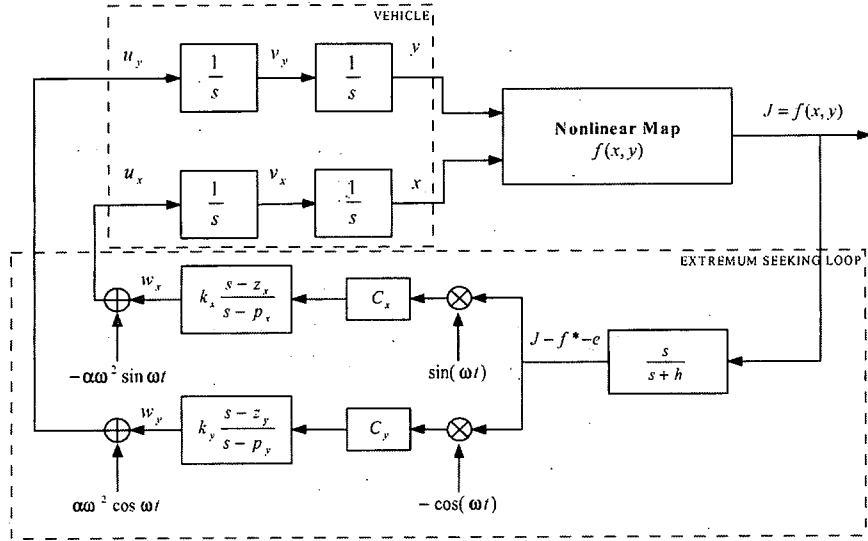


Figure 6.5: Extremum seeking for force-actuated point mass.

The Jacobian of (6.20) at the equilibrium $(\tilde{x}_{avg}^e, \tilde{v}_{xavg}^e, w_{xavg}^e, \tilde{y}_{avg}^e, \tilde{v}_{yavg}^e, w_{yavg}^e, e_{avg}^e)$

is

$$J_{avg} = \frac{1}{\omega} \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ -a_3 & -a_2 & -a_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -b_3 & -b_2 & -b_1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -h \end{bmatrix}, \quad (6.22)$$

where

$$a_1 = -p_x, a_2 = \alpha c_x k_x q_x, a_3 = -\alpha c_x k_x q_x (z_x + h),$$

$$b_1 = -p_y, b_2 = \alpha c_y k_y q_y, b_3 = -\alpha c_y k_y q_y (z_y + h).$$

Therefore the characteristic function of J_{avg} is

$$D(\lambda) = (\lambda + h)(\lambda^3 + a_1\lambda^2 + a_2\lambda + a_3)(\lambda^3 + b_1\lambda^2 + b_2\lambda + b_3),$$

Since the sufficient and necessary condition for a third order polynomial to have positive roots is $a_1, a_2, a_3 > 0$ and $a_1 a_2 - a_3 > 0$, then J_{avg} will be Hurwitz if and only if the following inequalities hold

$$\begin{aligned}
& -p_x > 0 \\
& \alpha c_x k_x q_x > 0 \\
& -\alpha c_x k_x q_x (z_x + h) > 0 \\
& -\alpha c_x k_x p_x q_x + \alpha c_x k_x q_x (z_x + h) > 0 \\
& -p_y > 0 \\
& \alpha c_y k_y q_y > 0 \\
& -\alpha c_y k_y q_y (z_y + h) > 0 \\
& -\alpha c_y k_y p_y q_y + \alpha c_y k_y q_y (z_y + h) > 0 \\
& h > 0
\end{aligned} \tag{6.23}$$

One possible design to satisfy these inequalities (6.23) of the x loop is

1. Choose $\alpha > 0$ to be small, $h > 0$.
2. Choose $c_x > 0, k_x > 0, z_x < -h, p_x < z_x + h$.
3. Choose $c_y > 0, k_y > 0, z_y < -h, p_y < z_y + h$.

Then according to the averaging theorem [131], we have the following result.

THEOREM 6.3.1 *Consider the system in the Figure 6.5, where the nonlinear map has the form of (6.2). There exists $\bar{\omega}$ such that for all $\frac{1}{\omega} \in (0, \frac{1}{\bar{\omega}})$ the system has a unique exponentially stable periodic solution $(\tilde{x}^{2\pi/\omega}, \tilde{y}^{2\pi/\omega}, e^{2\pi/\omega})$ of period $\frac{2\pi}{\omega}$ and*

this solution satisfies

$$\left\| \begin{bmatrix} \tilde{x}^{2\pi/\omega} \\ \tilde{y}^{2\pi/\omega} \\ \tilde{v}_x^{2\pi/\omega} \\ \tilde{v}_y^{2\pi/\omega} \\ e^{2\pi/\omega} + \frac{\alpha^2}{2}(q_x + q_y) \\ w_x^{2\pi/\omega} \\ w_y^{2\pi/\omega} \end{bmatrix} \right\| \leq O(1/\omega), \quad \forall \tau \geq 0. \quad (6.24)$$

Hence

$$\limsup_{\tau \rightarrow \infty} |f - f^*| = O(\alpha^2 + (1/\omega)^2).$$

Simulation results for a stationary target can be found in Figure 6.6 for $\omega = 30$, $\alpha = 0.05$, $c_x = c_y = 20$, $h = 1$. The parameters of the PD compensator are designed to satisfy the inequalities (6.23), where $k_c = 2$, $z_0 = -2$, $p_0 = -5$. The start position of the autonomous vehicle is $(x(0), y(0)) = (1, 1)$. Simulation results for a slowly time varying target (6.17) are shown in Figure 6.7 for $a_m = 1$, $\omega_m = 0.1$, $\phi_m = 3$, $f^* = 1$, $q_x = 1$, $q_y = 0.5$, and $\omega = 30$, $\alpha = 0.05$, $c_x = c_y = 20$, $h = 1$. Also same PD compensator and starting position for stationary target is used. As expected, tracking with a vehicle that has a double integrator in its input-output relation is harder than with a vehicle with a single integrator, but the use of a phase lead compensator helps achieve comparable performance.

6.4 Nonholonomic Unicycle

We consider a unicycle model of a mobile robot with a sensor that is either collocated at the center of the vehicle or mounted some distance r away from the center. A diagram depicting the position, heading, angular and forward velocities, and the sensor location on the autonomous vehicle is shown in Figure 6.8. According to the

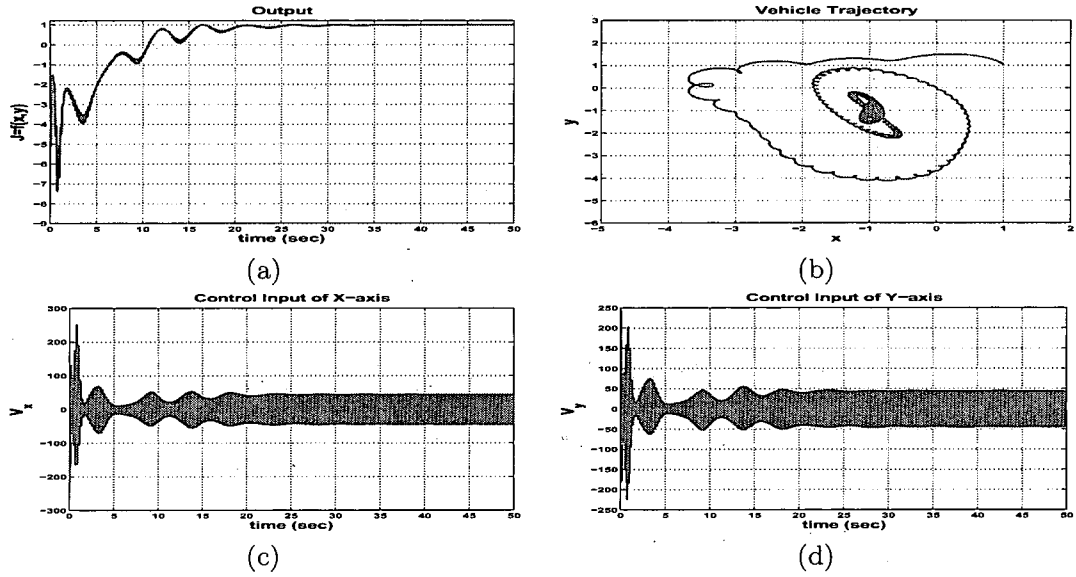


Figure 6.6: Extremum seeking for force-actuated point mass, stationary case: (a) output; (b) vehicle trajectory starts from (1,1); (c) control input of x -axis; (d) control input of y -axis.

diagram, the equations of motion for the vehicle center are:

$$\dot{x}_c = v \cos \theta, \quad (6.25)$$

$$\dot{y}_c = v \sin \theta, \quad (6.26)$$

$$\dot{\theta} = \omega_0, \quad (6.27)$$

where $[x_c, y_c]$ is the center of the vehicle, θ is the orientation, and v, ω_0 are the forward and angular velocity inputs. We point out that our extremum seeking algorithm will be tuning only the forward velocity input v , while keeping the angular velocity input

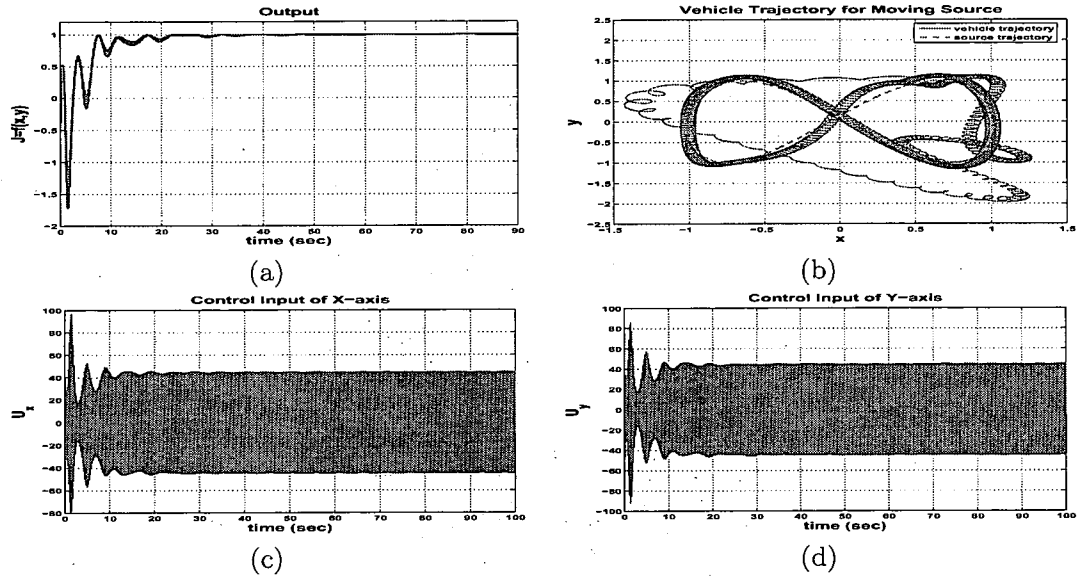


Figure 6.7: Extremum seeking for force-actuated point mass, slowly time-varying case: (a) output; (b) vehicle trajectory starts from (1,1) and source trajectory starts from (0,0); (c) control input of x -axis; (d) control input of y -axis.

ω_0 constant. The sensor position is governed by the equations

$$\dot{x}_s = v \cos \theta - r \omega_0 \sin \theta, \quad (6.28)$$

$$\dot{y}_s = v \sin \theta + r \omega_0 \cos \theta, \quad (6.29)$$

where $[x_s, y_s]$ is the position of the sensor and r is the distance between the sensor and center of the vehicle. We observe that the relationship between the center of the vehicle and the sensor is

$$x_s = x_c + r \cos \theta \quad (6.30)$$

$$y_s = y_c + r \sin \theta. \quad (6.31)$$

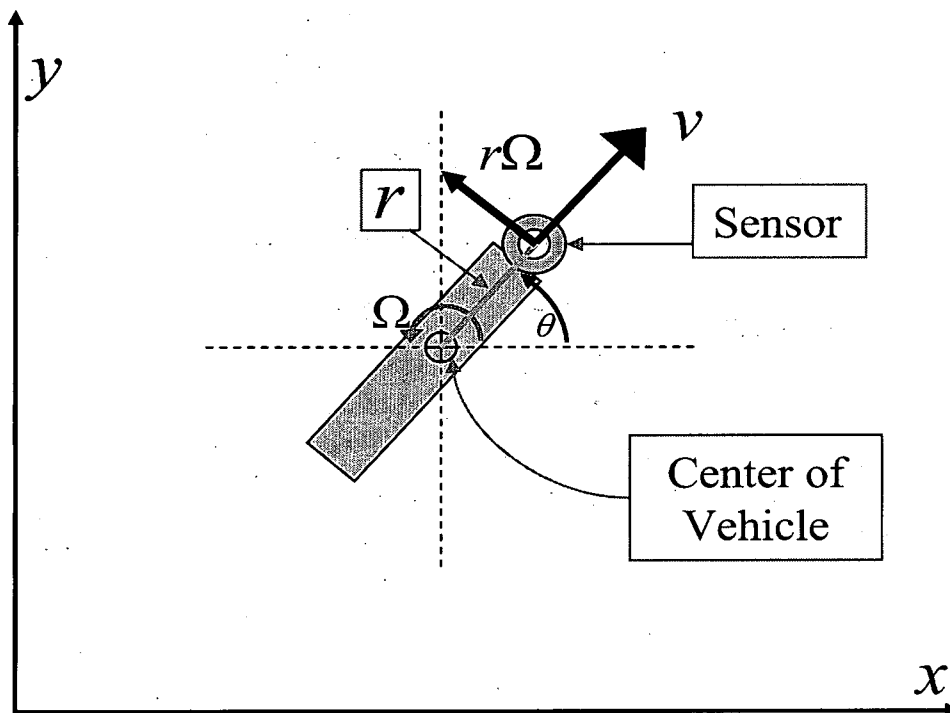


Figure 6.8: Unicycle model with non-collocated sensor.

The distance r from the sensor to the center of the vehicle is allowed to be zero in the search approach via extremum seeking. However, in some applications the sensor may be mounted off-center for design reasons. Additionally, the placement of the sensor with $r > 0$ tends to improve the convergence rate because one is “sweeping” the concentration field more broadly with lesser movement of the vehicle itself, yielding better estimation of the gradient. For all the above reasons, we consider the general case $r \geq 0$ in our analysis.

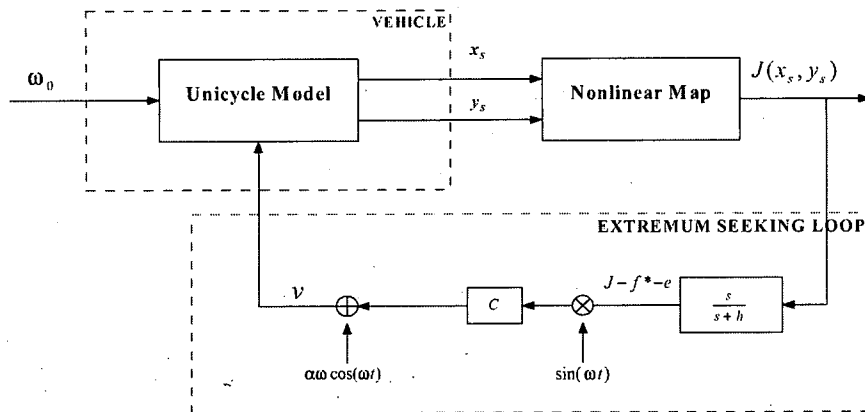


Figure 6.9: Extremum seeking for unicycle model.

We assume that the signal source being tracked is distributed according to an *unknown* nonlinear map $J = f(x, y)$, which has an isolated local maximum $f^* = f(x^*, y^*)$ at (x^*, y^*) . Our purpose is to control the autonomous vehicle to achieve local convergence to the maximizer (x^*, y^*) without the knowledge of the shape of $f(x, y)$ and using only the measurements of its value during the motion of the vehicle.

If we know the nonlinear function $f(x, y)$ and if we can directly actuate (and measure) the position of the vehicle, then we can design a control law to force the vehicle's motion to evolve according to the gradient dynamical system

$$[\dot{x}_c, \dot{y}_c]^\top = -\nabla f(x_c, y_c).$$

In that case the trajectory of $[x_c, y_c]$ will asymptotically converge to the set of stationary points of f where $\nabla f(x^*, y^*) = 0$. Even finite time tracking can be obtained with the gradient system

$$[\dot{x}_c, \dot{y}_c]^\top = -\frac{\nabla f(x_c, y_c)}{\|\nabla f(x_c, y_c)\|_2},$$

as explained by Cortes [129].

In the absence of the knowledge of function $f(x, y)$ and of the vehicle's position, we have to employ techniques of non-model-based optimization. In addition, in the absence of direct actuation of the vehicle's position, namely, for a nonholonomic vehicle that cannot be directly steered sideways and all of its motion has to be produced using forward and angular velocity inputs, the task of source seeking becomes even more challenging.

In this section, given only the measurement of the values of the function $J = f(x_s, y_s)$, we employ extremum seeking to tune the forward velocity v (with fixed angular velocity ω_0) to ensure that $[x_c(t), y_c(t)]$ asymptotically converges towards $[x^*, y^*]$. A block diagram of the extremum seeking scheme is shown in Figure 6.9. The designer can affect the seeking performance using the parameters α, ω, ω_0 , and c .

By fixing the angular velocity to be ω_0 , we have $\theta = \omega_0 t$. Moreover, we set the perturbation frequency $\omega = k\omega_0$ for a positive integer $k > 3$. The analysis that follows employs the method of averaging. The signal after the washout filter $\frac{s}{s+h}$ in Figure 6.9 can be expressed as

$$\Delta = \frac{s}{s+h}[J] = J - \frac{h}{s+h}[J] = J - f^* - e.$$

Now, let us introduce new coordinates

$$\tilde{x} = x_s - x^* - \alpha \sin(\omega t) \cos(\omega_0 t) - r \cos(\omega_0 t), \quad (6.32)$$

$$\tilde{y} = y_s - y^* - \alpha \sin(\omega t) \sin(\omega_0 t) - r \sin(\omega_0 t). \quad (6.33)$$

Then, in the time scale $\tau = \omega t$, we have

$$\begin{aligned}
\frac{d\tilde{x}}{d\tau} &= \frac{1}{\omega} \frac{d\tilde{x}}{dt} \\
&= \frac{1}{\omega} [v \cos(\omega_0 t) - r\omega_0 \sin(\omega_0 t) - \alpha\omega \cos(\omega t) \cos(\omega_0 t) + \alpha\omega_0 \sin(\omega t) \sin(\omega_0 t) + r\omega_0 \sin(\omega_0 t)] \\
&= \frac{1}{\omega} [c \sin(\omega t) \cos(\omega_0 t) \Delta + \alpha\omega \cos(\omega t) \cos(\omega_0 t) - \alpha\omega \cos(\omega t) \cos(\omega_0 t) + \alpha\omega_0 \sin(\omega t) \sin(\omega_0 t)] \\
&= \frac{1}{\omega} [c \sin \tau \cos(\tau/k) \Delta + \alpha\omega_0 \sin \tau \sin(\tau/k)], \\
\frac{d\tilde{y}}{d\tau} &= \frac{1}{\omega} \frac{d\tilde{y}}{dt} \\
&= \frac{1}{\omega} [v \sin(\omega_0 t) + r\omega_0 \cos(\omega_0 t) - \alpha\omega \cos(\omega t) \sin(\omega_0 t) - \alpha\omega_0 \sin(\omega t) \cos(\omega_0 t) - r\omega_0 \cos(\omega_0 t)] \\
&= \frac{1}{\omega} [c \sin(\omega t) \sin(\omega_0 t) \Delta + \alpha\omega \cos(\omega t) \sin(\omega_0 t) - \alpha\omega \cos(\omega t) \sin(\omega_0 t) - \alpha\omega_0 \sin(\omega t) \cos(\omega_0 t)] \\
&= \frac{1}{\omega} [c \sin \tau \sin(\tau/k) \Delta - \alpha\omega_0 \sin \tau \cos(\tau/k)], \\
\frac{de}{d\tau} &= \frac{1}{\omega} \frac{de}{dt} = \frac{h}{\omega} \Delta,
\end{aligned}$$

where

$$\Delta = -q_x(\tilde{x} + \alpha \sin \tau \cos(\tau/k) + r \cos(\tau/k))^2 - q_y(\tilde{y} + \alpha \sin \tau \sin(\tau/k) + r \sin(\tau/k))^2 - e. \quad (6.34)$$

So we summarize the system in Figure 6.9 for the unicycle with non-collocated sensor as

$$\frac{d\tilde{x}}{d\tau} = \frac{1}{\omega} [c \sin \tau \cos(\tau/k) \Delta + \alpha\omega_0 \sin \tau \sin(\tau/k)], \quad (6.35)$$

$$\frac{d\tilde{y}}{d\tau} = \frac{1}{\omega} [c \sin \tau \sin(\tau/k) \Delta - \alpha\omega_0 \sin \tau \cos(\tau/k)], \quad (6.36)$$

$$\frac{de}{d\tau} = \frac{h}{\omega} \Delta. \quad (6.37)$$

The system (6.35)–(6.37) is in the form to which the averaging method is applicable, provided $1/\omega$ is small, i.e., provided ω is large (relative to the other parameters in the extremum seeking scheme and relative to the parameters in the nonlinear map).

Now, averaging equations (6.35)–(6.37) over the larger period $2\pi k$, we have

$$\begin{aligned}
\frac{d\tilde{x}_{avg}}{d\tau} &= \frac{1}{\omega} \frac{1}{2k\pi} \int_0^{2k\pi} \frac{d\tilde{x}}{dt} d\tau \\
&= \frac{1}{\omega} \frac{1}{2k\pi} \int_0^{2k\pi} [c \sin \tau \cos(\tau/k) \Delta + \alpha \omega_0 \sin \tau \sin(\tau/k)] d\tau \\
&= \frac{1}{\omega} \frac{1}{2\pi} \int_0^{2\pi} [c \sin(k\tau) \cos \tau \Delta + \alpha \omega_0 \sin(k\tau) \sin \tau] d\tau \\
&= \frac{1}{\omega} \frac{1}{2\pi} \left\{ - \int_0^{2\pi} c \sin(k\tau) \cos \tau (q_x \tilde{x}^2 + q_y \tilde{y}^2 + e) d\tau \right. \\
&\quad - \int_0^{2\pi} c \sin(k\tau) \cos \tau [2q_x \tilde{x} \alpha \sin(k\tau) \cos \tau + 2q_x \tilde{x} r \cos \tau + 2q_x \alpha r \sin(k\tau) \cos^2 \tau \\
&\quad + 2q_y \tilde{y} \alpha \sin(k\tau) \sin \tau + 2q_y \tilde{y} r \sin \tau + 2q_y \alpha r \sin(k\tau) \sin^2 \tau] d\tau \\
&\quad - \int_0^{2\pi} c \sin(k\tau) \cos \tau [q_x \alpha^2 \sin^2(k\tau) \cos^2 \tau + q_x r^2 \cos^2 \tau + q_y \alpha^2 \sin^2(k\tau) \sin^2 \tau \\
&\quad + q_y r^2 \sin^2 \tau] d\tau \\
&\quad \left. + \int_0^{2\pi} \alpha \omega_0 \sin(k\tau) \sin \tau d\tau \right\} \\
&= -\frac{1}{\omega} \frac{1}{2\pi} \frac{\pi}{2} c 2q_x \tilde{x}_{avg} \alpha \\
&= -\frac{1}{2\omega} \alpha c q_x \tilde{x}_{avg}
\end{aligned}$$

$$\begin{aligned}
\frac{d\tilde{y}_{avg}}{d\tau} &= \frac{1}{\omega} \frac{1}{2k\pi} \int_0^{2k\pi} \frac{d\tilde{y}}{dt} d\tau \\
&= \frac{1}{\omega} \frac{1}{2k\pi} \int_0^{2k\pi} [c \sin \tau \sin(\tau/k) \Delta - \alpha \omega_0 \sin \tau \cos(\tau/k)] d\tau \\
&= \frac{1}{\omega} \frac{1}{2\pi} \int_0^{2\pi} [c \sin(k\tau) \sin \tau \Delta - \alpha \omega_0 \sin(k\tau) \cos \tau] d\tau \\
&= \frac{1}{\omega} \frac{1}{2\pi} \left\{ - \int_0^{2\pi} c \sin(k\tau) \sin \tau (q_x \tilde{x}^2 + q_y \tilde{y}^2 + e) d\tau \right. \\
&\quad - \int_0^{2\pi} c \sin(k\tau) \sin \tau [2q_x \tilde{x} \alpha \sin(k\tau) \cos \tau + 2q_x \tilde{x} r \cos \tau + 2q_x \alpha r \sin(k\tau) \cos^2 \tau \\
&\quad + 2q_y \tilde{y} \alpha \sin(k\tau) \sin \tau + 2q_y \tilde{y} r \sin \tau + 2q_y \alpha r \sin(k\tau) \sin^2 \tau] d\tau \\
&\quad - \int_0^{2\pi} c \sin(k\tau) \sin \tau [q_x \alpha^2 \sin^2(k\tau) \cos^2 \tau + q_x r^2 \cos^2 \tau + q_y \alpha^2 \sin^2(k\tau) \sin^2 \tau \\
&\quad + q_y r^2 \sin^2 \tau] d\tau \\
&\quad \left. - \int_0^{2\pi} \alpha \omega_0 \sin(k\tau) \cos \tau d\tau \right\} \\
&= -\frac{1}{\omega} \frac{1}{2\pi} \frac{\pi}{2} c 2q_y \tilde{y}_{avg} \alpha \\
&= -\frac{1}{2\omega} \alpha c q_y \tilde{y}_{avg}
\end{aligned}$$

$$\begin{aligned}
\frac{d\tilde{e}_{avg}}{d\tau} &= \frac{1}{\omega} \frac{1}{2k\pi} \int_0^{2k\pi} \frac{d\tilde{e}}{dt} d\tau \\
&= \frac{h}{\omega} \frac{1}{2k\pi} \int_0^{2k\pi} \Delta d\tau \\
&= \frac{h}{\omega} \frac{1}{2\pi} \left\{ - \int_0^{2\pi} (q_x \tilde{x}^2 + q_y \tilde{y}^2 + e) d\tau \right. \\
&\quad - \int_0^{2\pi} [2q_x \tilde{x} \alpha \sin(k\tau) \cos \tau + 2q_x \tilde{x} r \cos \tau + 2q_x \alpha r \sin(k\tau) \cos^2 \tau \\
&\quad + 2q_y \tilde{y} \alpha \sin(k\tau) \sin \tau + 2q_y \tilde{y} r \sin \tau + 2q_y \alpha r \sin(k\tau) \sin^2 \tau] d\tau \\
&\quad - \int_0^{2\pi} [q_x \alpha^2 \sin^2(k\tau) \cos^2 \tau + q_x r^2 \cos^2 \tau + q_y \alpha^2 \sin^2(k\tau) \sin^2 \tau + q_y r^2 \sin^2 \tau] d\tau \\
&= -\frac{h}{\omega} [q_x \tilde{x}_{avg}^2 + q_y \tilde{y}_{avg}^2 + e_{avg} + \frac{1}{2\pi} \frac{\pi}{2} q_x \alpha^2 + \frac{1}{2\pi} \pi q_x r^2 + \frac{1}{2\pi} \frac{\pi}{2} q_y \alpha^2 + \frac{1}{2\pi} \pi q_y r^2] \\
&= -\frac{h}{\omega} \left[q_x \tilde{x}_{avg}^2 + q_y \tilde{y}_{avg}^2 + e_{avg} + \left(\frac{\alpha^2}{4} + \frac{r^2}{2} \right) (q_x + q_y) \right]
\end{aligned}$$

Then, summarizing the average model we have

$$\frac{d\tilde{x}_{avg}}{d\tau} = -\frac{1}{2\omega}\alpha c q_x \tilde{x}_{avg}, \quad (6.38)$$

$$\frac{d\tilde{y}_{avg}}{d\tau} = -\frac{1}{2\omega}\alpha c q_y \tilde{y}_{avg}, \quad (6.39)$$

$$\frac{de_{avg}}{d\tau} = -\frac{h}{\omega} \left[q_x \tilde{x}_{avg}^2 + q_y \tilde{y}_{avg}^2 + e_{avg} + \left(\frac{\alpha^2}{4} + \frac{r^2}{2} \right) (q_x + q_y) \right]. \quad (6.40)$$

The equilibrium of the average model (6.38)–(6.40) is

$$\tilde{x}_{avg}^e = 0, \quad \tilde{y}_{avg}^e = 0, \quad e_{avg}^e = -\left(\frac{\alpha^2}{4} + \frac{r^2}{2} \right) (q_x + q_y). \quad (6.41)$$

The Jacobian of (6.38)–(6.40) at $(\tilde{x}_{avg}^e, \tilde{y}_{avg}^e, e_{avg}^e)$ is

$$J_{avg} = \frac{1}{\omega} \begin{bmatrix} -\alpha c q_x / 2 & 0 & 0 \\ 0 & -\alpha c q_y / 2 & 0 \\ 0 & 0 & -h \end{bmatrix}. \quad (6.42)$$

Given the knowledge that the extremum is a maximum, it follows that q_x, q_y are known to be positive, though their actual values are unknown. Therefore, if we choose $\alpha > 0, c > 0$, and $h > 0$, the Jacobian (6.42) is Hurwitz and the equilibrium (6.41) of the average system (6.38)–(6.40) is locally exponentially stable. Then according to the averaging theorem [131], we have the following result.

THEOREM 6.4.1 *There exists $\bar{\omega}$ such that for all $\frac{1}{\omega} \in (0, \frac{1}{\bar{\omega}})$ the system in the Figure 6.9 has a unique exponentially stable periodic solution $(\tilde{x}^{2\pi/\omega}, \tilde{y}^{2\pi/\omega}, e^{2\pi/\omega})$ of period $\frac{2\pi}{\omega}$ and this solution satisfies*

$$\left\| \begin{bmatrix} \tilde{x}^{2\pi/\omega} \\ \tilde{y}^{2\pi/\omega} \\ e^{2\pi/\omega} + \left(\frac{\alpha^2}{4} + \frac{r^2}{2} \right) (q_x + q_y) \end{bmatrix} \right\| \leq O(1/\omega), \quad \forall \tau \geq 0. \quad (6.43)$$

Since

$$\begin{aligned} x_s - x^* &= \tilde{x} + \alpha \sin(\omega t) \cos(\omega_0 t) + r \cos(\omega_0 t) \\ &= (\tilde{x} - \tilde{x}^{2\pi/\omega}) + \tilde{x}^{2\pi/\omega} + \alpha \sin(\omega t) \cos(\omega_0 t) + r \cos(\omega_0 t), \end{aligned}$$

the above theorem implies that the first term converges to zero, the second term is $O(1/\omega)$, the third term is $O(\alpha)$ and the forth term is $O(r)$. Thus

$$\limsup_{\tau \rightarrow \infty} |x_s - x^*| = O(\alpha + 1/\omega + r).$$

Similarly, we obtain

$$\limsup_{\tau \rightarrow \infty} |y_s - y^*| = O(\alpha + 1/\omega + r).$$

Hence, we have

$$\limsup_{\tau \rightarrow \infty} |f(x_s, y_s) - f^*| = O(\alpha^2 + (1/\omega)^2 + r^2).$$

The above limits characterize the asymptotic performance of the extremum seeking loop in Figure 6.9. The vehicle converges to a neighborhood of the maximizer $[x^*, y^*]$, the size of which is proportional to the amplitude of the periodic perturbation, the reciprocal of the perturbation frequency and the vehicle radius. Since we choose α and r small and ω large, the tracking error between the vehicle and the source is small. A direct trade-off between the tracking error and the speed of convergence is evident from the fact that two of the eigenvalues of the Jacobian become more negative as α increases.

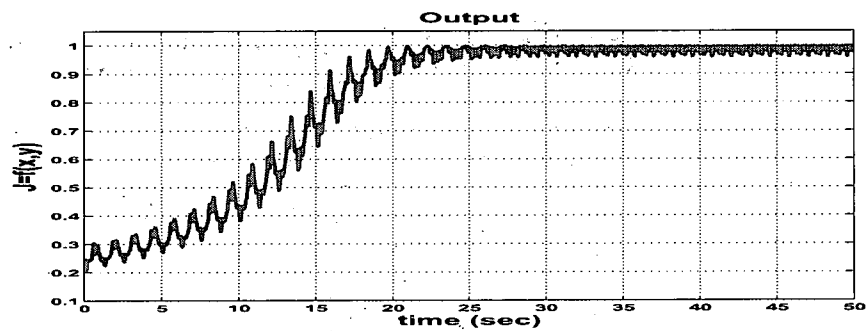
In the following simulations, we set the parameters of the stationary source as $[x^*, y^*] = [0, 0]$, $f^* = 1$, $q_x = 0.5$ and $q_y = 0.25$. The parameters of the extremum seeking loop are chosen as $\omega = 25$, $\alpha = 0.1$, $c = 20$, $h = 1$, and $\omega_0 = \omega/5$. The starting position of the autonomous vehicle is $[x_s(0), y_s(0)] = [2, 2]$, $r = 0.2$, $\theta_0 = \pi$, and therefore $[x_c, y_c] = [1.8, 2]$. As shown in Figure 6.10 (b), the autonomous vehicle starts at $[1.8, 2]$ by probing around to climb the gradient of the unknown map in a star pattern, eventually circling around the maximizer $[0, 0]$. The output of the unknown signal $J = f(x_s, y_s)$ is shown in Figure 6.10 (a), while the forward velocity before

modulation v is shown in Figure 6.10 (c). The trajectories of the vehicle center and sensor are compared in Figure 6.10 (b). Because of the vehicle radius, the steady state position of the sensor forms a star pattern with the vertex nearly 0.2 away from the maximizer $[0, 0]$. Therefore, the observations from the simulation coincide with the theoretical analysis presented above.

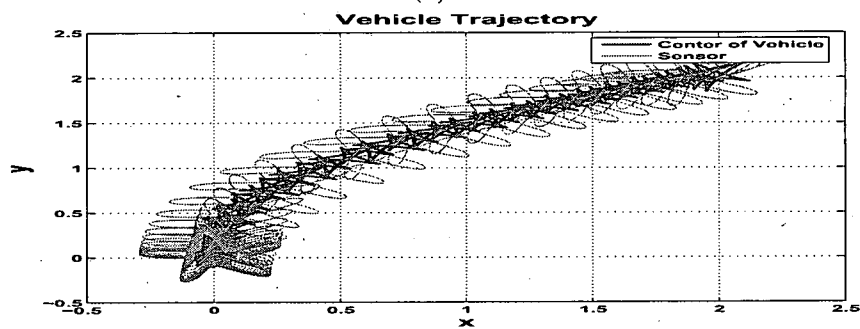
The reason we observe a star pattern with five vertices in the vehicle trajectory (Figure 6.10 (b)) is due to the scaling constant k . In this case $\omega_0 = \omega/5$. If we change $\omega_0 = \omega/3$, we observe a triangular pattern in the trajectory of the vehicle center, and a star pattern with three vertices in the trajectory of the sensor, refer to Figure 6.11.

Finally, we consider a slow moving source whose trajectory is in the shape of the number eight 8, that is, $x^* = a_m \sin(\omega_m t)$, $y^* = a_m \sin(2\omega_m t)$, where $\omega_m \ll \omega$, the initial position of the target is $[0, 0]$, and $a_m = 1$, $\omega_m = 0.03$. Here, we maintain the same parameter settings as in the above simulations, except $\alpha = 0.04$, $c = 50$, and $\omega_0 = \omega/5$. The simulation results in Figure 6.12 show successful tracking in a non-stationary case.

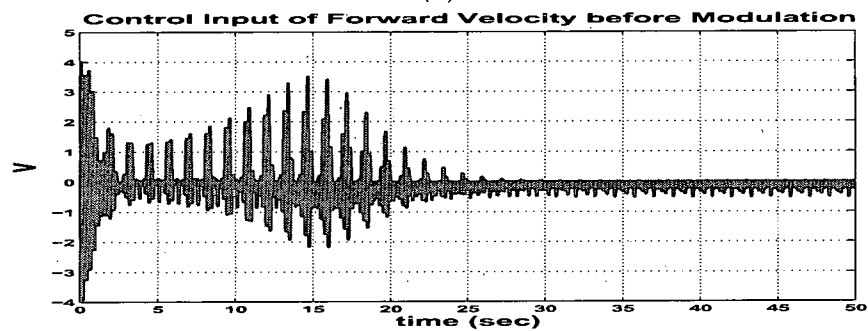
The star or triangle-patterned motion of our vehicle is perhaps the most interesting part of our result. The gradient is being estimated due to the local exploration of space in a pattern that our kinematically constrained vehicle can execute. Though such motion may appear a little awkward, it should not be entirely surprising because it has come up in motion planning problems for nonholonomic systems such as the "snakeboard" (see the work by Lewis, Ostrowski, Murray, and Burdick [143] and by Bullo and Lewis [144]).



(a)

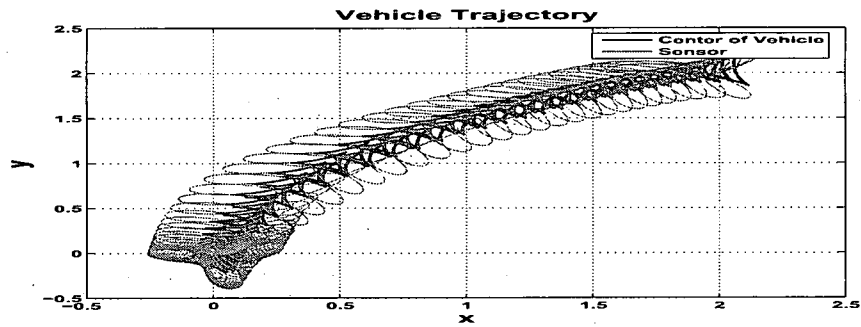


(b)

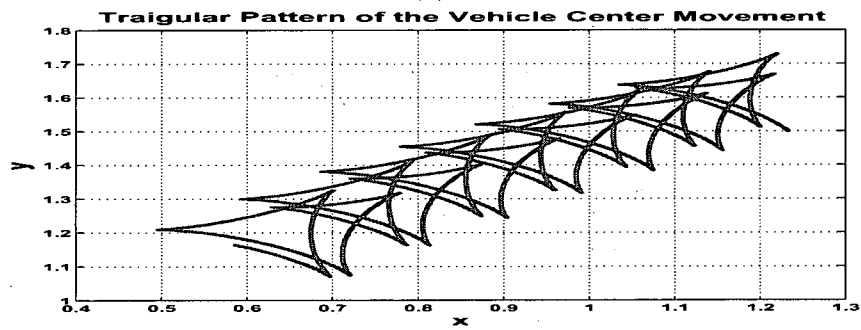


(c)

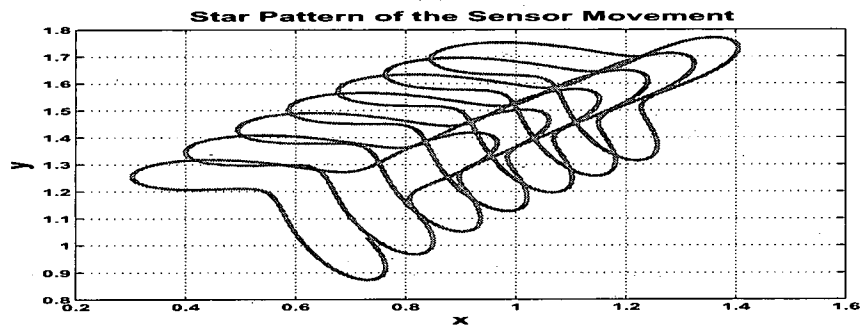
Figure 6.10: Extremum seeking of unicycle with non-collocated sensor, stationary target case, $\omega_0 = \omega/5$: (a) output of nonlinear map; (b) vehicle center and sensor trajectories; (c) forward velocity before modulation



(a)



(b)



(c)

Figure 6.11: Extremum seeking of unicycle with non-collocated sensor, stationary target case, $\omega_0 = \omega/3$: (a) vehicle center and sensor trajectories; (b) triangular pattern of the vehicle center movement (c) sensor position trajectory

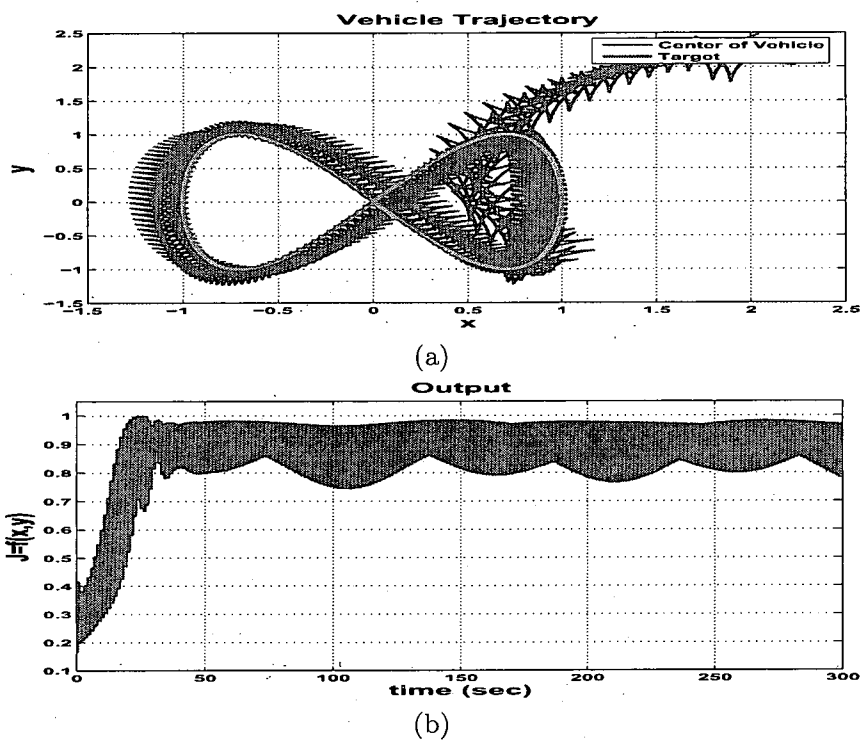


Figure 6.12: Extremum seeking of unicycle with non-collocated sensor, moving target case, $\omega_0 = \omega/5$: (a) vehicle center and target trajectories; (b) output of the nonlinear map.

CHAPTER 7

SWARM SOURCE SEEKING

The problem of coordination and control of autonomous vehicles (agents) has been receiving an extraordinary amount of attention during the past few years due to its critical importance for military, as well as civilian applications. Research on vehicle autonomy will open up the door to a myriad of applications. For instance, the use of robots for navigating and inspecting dangerous areas could be made cheap, reliable and wide-spread. Other similar uses include search over wide areas and automated geographical and topological surveying. Moreover, providing vehicles and robots with autonomy would enormously facilitate planetary exploration, where automated construction of structures and surveying of areas would help reduce the amount of the much more costly and risky alternative, that of direct human intervention.

In recent years it has become popular to study the biological world and apply the derived principles to the design of engineering. The topic of distributed coordination and control of multiple autonomous agents has gained lots of attention [145, 146, 147]. Cooperative agents can often be used to perform tasks that are too difficult for a single one to perform. Instead of the traditional trajectory tracking problem, people began to study coordinated tracking [148, 149]. Tasks can be performed more efficiently by controlling the group to move in formation. Possible applications could range from

autonomous robot assembly to UAVs scout and counterwork. Compared to individuals, swarms, flocks, and schools can have remarkable group-level characteristics, which may allow them to perform complicated task efficiently.

The swarm source seeking problem is to find a coordinated control scheme for a group of agents to make them achieve and maintain some given geometrical formation, at the same time, the agents need to seek a source of a scalar signal. Thus, there is a trade-off between maintaining formation and arriving at the final goal. Possible approaches for formation control include leader-following and the virtual structure approaches [150, 151]. In both approaches, one agent, which could be real or virtual, is designated as leader to perform the tracking task without considering the followers, and the remaining agents only have to stay from the leader within a desired offset, without considering the target. However, these approaches are centralized and therefore not robust – if the leader fails, the task would also fail. In that case, decentralized formation control is preferred as discussed in [152], where feedback control laws are used to keep the formation during tracking process.

In this chapter, formation control, obstacle avoidance and collision avoidance are achieved via artificial potential functions. Artificial potential functions have been first widely used for robot navigation and control including multi-agent coordination [150, 153, 154, 155]. The potential function is created to encode the interaction rule for the group. In order to perform the task in a decentralized way, in this chapter we consider a potential function composed of several parts. The inter-connection part makes the agent constrained by its neighbor to maintain a group structure. This part includes functions of the relative distance between each pair of neighbors. In addition, the scalar signal is added to direct the group behavior to seek the source. The specific

form of potential function is defined according to the desired geometric formation. The choice of a potential function is important because different potentials might result in different performance even with the same control algorithm. In particular, existence of multiple local minima in the potential function leads to guaranteeing only local convergence to the desired formation. Nevertheless, we show that by appropriate choice of the potential function one can always guarantee that eventually the target will be surrounded or “enclosed” by the tracking agents. Later, we will even include the potential function between the agents and the obstacle to achieve obstacle avoidance.

7.1 Problem Statement

The problem statement and the next section are mainly extended from the work by Yao, Ordóñez and Gazi in [156], it appears that their research turns out to one special case of swarm source seeking via gradient-based extremum seeking control. Consider a multi-agent system (i.e., a swarm) consisting of N individuals in an n -dimensional Euclidean space. We assume synchronous motion and no time delays. Let $x \in \mathbb{R}^n$ denotes a vector n -dimensional Euclidean space, and $x^i \in \mathbb{R}^n$ denotes the position vector of individual agent i , whose motion is governed by the following velocity actuated point mass “kinematic model,”

$$\dot{x}^i = u^i \tag{7.1}$$

where $u^i \in \mathbb{R}^n$ is the control input for the i^{th} agent. Furthermore, we assume there is a scalar signal $J_t(x, x_t)$ to be tracked by the swarming agents, which has an unknown isolated global minimum at $x_t \in \mathbb{R}^n$. The strength of the scalar signal can be measured

by the agent. Our purpose is to design a control law for each agent such that we can achieve source seeking, formation control, collision avoidance and obstacle avoidance.

First, consider a potential of the form

$$J_{at}(x^1, x^2, \dots, x^N, x_t) = \sum_{i=1}^N J_{it}(\|x^i - x_t\|), \quad (7.2)$$

where J_{it} is constructed based on the scalar signal J_t such that it satisfies

$$\nabla_{x^i} J_{it}(\|x^i - x_t\|) = 0 \text{ if and only if } \|x^i - x_t\| = \delta_{it}. \quad (7.3)$$

By doing so, we actually shift the unique minimum at $x^i = x_t$ of J_t to $\|x^i - x_t\| = \delta_{it}$ of J_{it} , that is, we require each vehicle tracking x_t by a prescribed distance of δ_{it} . We do not want δ_{it} to be zero since it will result in collision of the vehicle. Also if the vehicle is modeled as a unicycle with non-collocated sensor, we can have the sensor position of each vehicle tracking the target, while the vehicle itself is at a distance r away from the target, where r is the distance between the vehicle and sensor footprint. Similar case can happen in the case of air vehicles tracking a ground target, where we might want the sensor footprint of each air vehicle focusing on the target but avoiding collision.

Second, consider the potential function of the form

$$J_{aa}(x^1, x^2, \dots, x^N) = \sum_{i=1}^{N-1} \sum_{j=i+1}^N J_{ij}(\|x^i - x^j\|), \quad (7.4)$$

where $J_{ij}(\|x^i - x^j\|)$ is the potential between i^{th} and j^{th} agent. Moreover, we assume it satisfies

(A1) The potential $J_{ij}(\|x^i - x^j\|)$ are symmetric and satisfy

$$\nabla_{x^i} J_{ij}(\|x^i - x^j\|) = -\nabla_{x^j} J_{ij}(\|x^i - x^j\|).$$

(A2) There exist corresponding functions $g_{ar}^{ij} : \mathbb{R}^+ \rightarrow \mathbb{R}^+$ such that

$$\nabla_x J_{ij}(\|x\|) = x g_{ar}^{ij}(\|x\|).$$

(A3) There exist unique distances δ_{ij} at which we have

$$g_{ar}^{ij}(\|x\|) \begin{cases} > 0, & \|x\| > \delta_{ij}, \\ = 0, & \|x\| = \delta_{ij}, \\ < 0, & \|x\| < \delta_{ij} \end{cases}$$

Potential functions satisfying these conditions are odd functions that are attractive on distances $\|x\| > \delta_{ij}$ and repulsive on distances $\|x\| < \delta_{ij}$. The term $g_{ar}^{ij}(\|x\|)$ determines the attraction-repulsion relationship between the individuals and usually is of the form

$$g_{ar}^{ij}(\|x\|) = g_a^{ij}(\|x\|) + g_r^{ij}(\|x\|),$$

where $g_a^{ij}(\|x\|)$ represents the attraction and $g_r^{ij}(\|x\|)$ represents the repulsion. The distance δ_{ij} is the equilibrium distance at which the attraction and the repulsion balance. That is, the potential function $J_{ij}(\|x^i - x^j\|)$ has a minimum at $\|x^i - x^j\| = \delta_{ij}$. Later, we will see that if we can construct a potential function J_a encoding a desired formation of the agents, then the formation control will be realized if such potential function is minimized. And in the mean time, by involving a repulsive part in the potential function, we can achieve collision avoidance as well. Similarly, we can set up a repulsive potential function J_{ao} between obstacle and agents, the potential function is inverse proportional to the distance between obstacle and agents, therefore, we can achieve obstacle avoidance by minimizing J_{ao} .

Let

$$\begin{aligned}
y &= J(x^1, x^2, \dots, x^N, x_t) \\
&= J(\|x^i - x_t\|, \|x^i - x^j\|), \quad 1 \leq i, j \leq N \\
&= K_{at}J_{at}(x^1, x^2, \dots, x^N, x_t) + K_{aa}J_{aa}(x^1, x^2, \dots, x^N) \\
&= K_{at} \sum_{i=1}^N J_{it}(\|x^i - x_t\|) + K_{aa} \sum_{i=1}^{N-1} \sum_{j=i+1}^N J_{ij}(\|x^i - x^j\|), \quad (7.5)
\end{aligned}$$

be the performance function of the system (7.1) for $i = 1, \dots, N$. Our objective is to make the entire group aggregate around the target x_t and move together with it, possibly in a specific formation regardless of the target's movement, since we have assumed the potential function will have a unique minimum at $\|x^i - x_t\| = \delta_{it}$ and $\|x^i - x^j\| = \delta_{ij}$. Now, if we can design an extremum seeking controller to minimize the performance function (7.5), then we can achieve our objective of source seeking, formation control and collision avoidance. Similarly, we can obtain obstacle avoidance by adding J_{ao} into (7.5), however, for simplicity we will address the source seeking, formation control and collision avoidance problem first. A block diagram can be found in Figure 7.1.

7.2 Gradient-based Extremum Seeking Control Design

We have seen the gradient-based extremum seeking control is able to seek a stationary source in Section 2.1. Since the knowledge of gradient is a strong assumption, therefore, we expect that it can achieve a strong result such as seeking a moving target. Consider the i^{th} vehicle, the scalar signal $J_{it}(\|x^i - x_t\|)$ has a unique minimum at $\|x^i - x_t\| = \delta_{it}$ and we assume additionally

$$\nabla_{x^i} J_{it}(\|x^i - x_t\|) = -\nabla_{x_t} J_{it}(\|x^i - x_t\|). \quad (7.6)$$

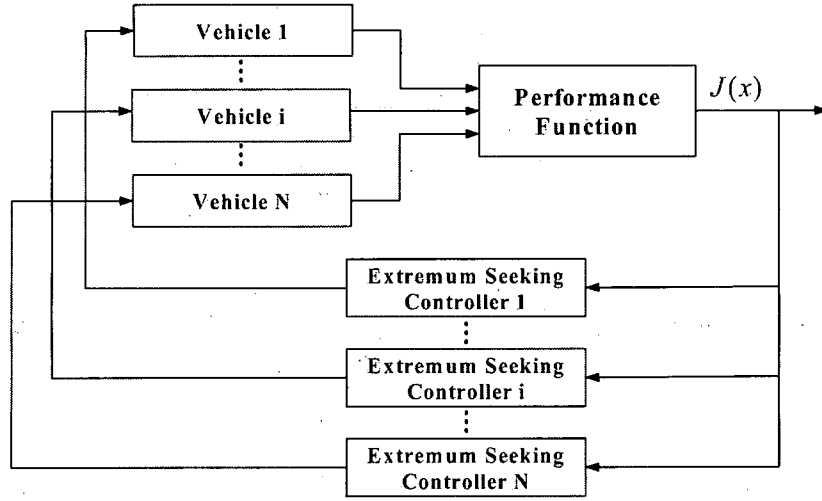


Figure 7.1: Swarm source seeking.

Now choosing a Lyapunov candidate $V = J_{it}(\|x^i - x_t\|) - J_{it}(\delta_{it})$, and

$$\dot{V} = \nabla_{x^i} J_{it}^\top(\|x^i - x_t\|) \dot{x}^i + \nabla_{x_t} J_{it}^\top(\|x^i - x_t\|) \dot{x}_t = \nabla_{x^i} J_{it}^\top(\|x^i - x_t\|) (\dot{x}^i - \dot{x}_t).$$

so for the i^{th} vehicle, if we let

$$u^i = \dot{x}_t - k \nabla_{x^i} J_{it}(\|x^i - x_t\|), k > 0. \quad (7.7)$$

Then

$$\dot{V} = -k \|\nabla_{x^i} J_{it}^\top(\|x^i - x_t\|)\|^2 \leq 0.$$

By Lasalle-Yoshizawa theorem, we can further conclude that the trajectory x^i asymptotically converges to $\|\nabla_{x^i} J_t^\top(x^i, x_t)\| = 0$, that is, x^i asymptotically converges to a cycle centered at x_t with radius δ_{it} given Euclidean norm is used since $\|x^i - x_t\| = \delta_{it}$ is the unique minimum. However, assuming that \dot{x}_t is known is a strong assumption

since usually it is not possible for one to know the velocity of the source, especially since one does not even know its position x_t . Thus, we will only assume that $\|\dot{x}_t\| \leq \gamma_t$ for some known $\gamma_t > 0$. Note that this constitutes a more realistic assumption since any moving target has a bounded velocity. With this assumption, we choose the controller as

$$u^i = -k\nabla_{x^i} J_{it}(\|x^i - x_t\|) - \beta \text{sgn}(\nabla_{x^i} J_{it}(\|x^i - x_t\|)), k > 0, \beta > \gamma_t. \quad (7.8)$$

Then,

$$\begin{aligned} \dot{V} &= -k\|\nabla_{x^i} J_{it}^\top(\|x^i - x_t\|)\|^2 - \beta\|\nabla_{x^i} J_{it}^\top(\|x^i - x_t\|)\| - \nabla_{x^i} J_{it}^\top(\|x^i - x_t\|)\dot{x}_t \\ &\leq -k\|\nabla_{x^i} J_{it}^\top(\|x^i - x_t\|)\|^2 - \beta\|\nabla_{x^i} J_{it}^\top(\|x^i - x_t\|)\| + \gamma_t\|\nabla_{x^i} J_{it}^\top(\|x^i - x_t\|)\| \\ &\leq -k\|\nabla_{x^i} J_{it}^\top(\|x^i - x_t\|)\|^2, \end{aligned}$$

We can further relax the controller design to

$$u^i = -(k + \beta)\text{sgn}(\nabla_{x^i} J_{it}(\|x^i - x_t\|)), k > 0, \beta > \gamma_t. \quad (7.9)$$

Then,

$$\begin{aligned} \dot{V} &= -(k + \beta)\|\nabla_{x^i} J_{it}^\top(\|x^i - x_t\|)\| - \nabla_{x^i} J_{it}(\|x^i - x_t\|)\dot{x}_t \\ &\leq -(k + \beta)\|\nabla_{x^i} J_{it}^\top(\|x^i - x_t\|)\| + \gamma_t\|\nabla_{x^i} J_{it}(\|x^i - x_t\|)\| \\ &\leq -k\|\nabla_{x^i} J_{it}^\top(\|x^i - x_t\|)\|, \end{aligned}$$

Therefore, these controllers (7.8) and (7.9) again result x^i tracking x_t from δ_{it} distance away asymptotically, which requires the knowledge of the gradient of the scalar signal, or at least the sign of the gradient and a bound on the target speed. Then, with the help of a switching term, the controllers guarantee asymptotic tracking of the target. Intuitively, the signum function allows for the detection of the changes in the

direction of the motion of the target and helps redirect the vehicle in that direction. All the three controllers (7.7) – (7.9) belong to the gradient-based extremum seeking controller type, and we can apply such design to all the vehicles in Figure 7.1 with the performance function J in (7.5) to replace J_{it} in (7.7) – (7.9).

For simulation purposes, we consider a two dimensional case where $n = 2$ and three vehicles $N = 3$. The scalar signal is postulated as a quadratic function:

$$J_t(x, x_t) = \|x - x_t\|^2, \quad (7.10)$$

where the dynamics of the target x_t satisfy

$$\dot{x}_{t1} = 0.25, \quad \dot{x}_{t2} = \sin(0.25t). \quad (7.11)$$

Then we construct the potential function J_{at} as

$$J_{at}(x^1, x^2, x^3, x_t) = \sum_{i=1}^3 J_{it}(\|x^i - x_t\|) = \frac{1}{2} \sum_{i=1}^3 (\|x^i - x_t\|^2 - \delta_{it}^2)^2. \quad (7.12)$$

From which we see that $J_{it}(\|x^i - x_t\|) = \frac{1}{2}(\|x^i - x_t\|^2 - \delta_{it}^2)^2$, and we can verify that $\nabla_{x^i} J_{it}(\|x^i - x_t\|) = 2(\|x^i - x_t\|^2 - \delta_{it}^2)(x^i - x_t) = 0$ if and only if $\|x^i - x_t\| = \delta_{it}$, which exactly satisfies (7.3). Moreover, we choose the potential function between agents

$$\begin{aligned} J_{aa}(x^1, x^2, x^3) &= \sum_{i=1}^2 \sum_{j=2}^3 J_{ij}(\|x^i - x^j\|) \\ &= \frac{1}{2} [(\|x^1 - x^2\|^2 - \delta_{12}^2)^2 + (\|x^1 - x^3\|^2 - \delta_{13}^2)^2 + (\|x^2 - x^3\|^2 - \delta_{23}^2)^2] \end{aligned} \quad (7.13)$$

where $J_{ij}(\|x^i - x^j\|) = \frac{1}{2}(\|x^i - x^j\|^2 - \delta_{ij}^2)^2$, and we can also verify $\nabla_{x^i} J_{ij}(\|x^i - x^j\|) = 2(\|x^i - x^j\|^2 - \delta_{ij}^2)(x^i - x^j)$ and $\nabla_{x^j} J_{ij}(\|x^i - x^j\|) = -2(\|x^i - x^j\|^2 - \delta_{ij}^2)(x^i - x^j)$, which satisfies Assumption A1. Moreover, we see that $\nabla_x J_{ij}(\|x\|) = 2(\|x\|^2 - \delta_{ij}^2)x = 0$ if and only if $\|x\| = \delta_{ij}$. We can further verify it satisfies Assumptions A2 and A3. Such

design of potential function J_{aa} prescribes a triangular formation for the vehicles to form with each lateral length equal to δ_{ij} . Now, we propose the performance function

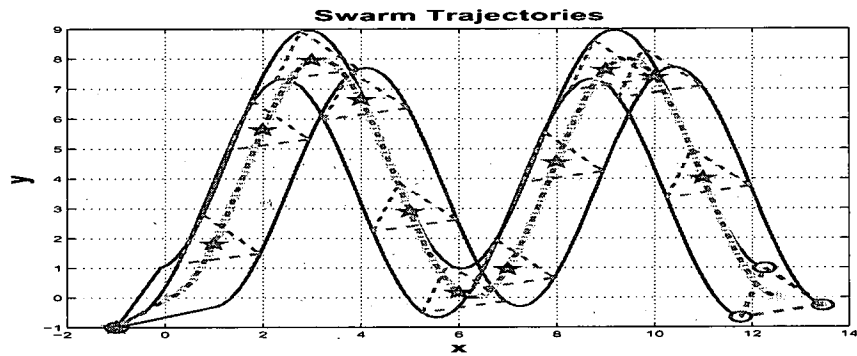
$$y = J(x^1, x^2, x^3, x^t) = K_{at}J_{at}(x^1, x^2, x^3, x^t) + K_{aa}J_{aa}(x^1, x^2, x^3), \quad (7.14)$$

where J_{at} and J_{aa} can be found in (7.12) and (7.13), respectively, and K_{at}, K_{aa} are the weights of the potential component, which is actually very important in balancing the priority in source seeking and formation keeping. In the simulation, we choose $\delta_{it} = 1, \delta_{ij} = \sqrt{3}$ for $i, j = 1, 2, 3$, $K_{at} = 1, K_{aa} = 0.1$, and $k = 10, \beta = 2$. We apply the controller (7.8) as

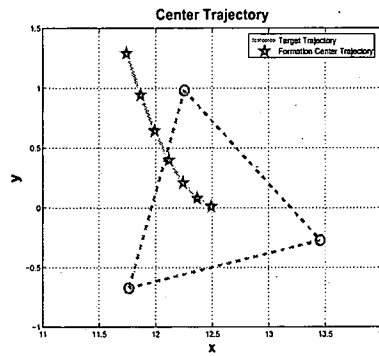
$$u^i = -k\nabla_{x^i}J - \beta\text{sgn}(\nabla_{x^i}J),$$

where $i = 1, 2, 3$ and J can be found in (7.14), the three vehicles start randomly inside a ball centered at $(-1, -1)$ with radius 0.01, and the target is started from $(0, 0)$. The simulation results can be found in Figure 7.2. The entire movements of the vehicles and the target can be found in Figure 7.2 (a), where source seeking, formation control and collision avoidance are successfully achieved. The final formation can be found in Figure 7.2 (b), and the performance functions are minimized as seen in Figure 7.2 (c).

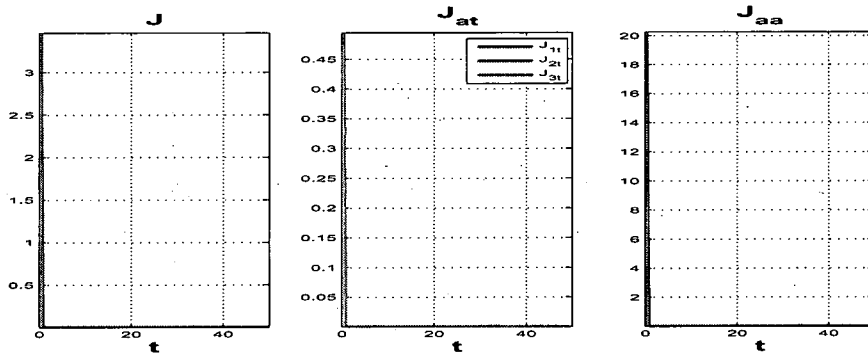
Consider a force-actuated point mass model, we can follow the controller design for a single vehicle as in [157], a sliding mode is used to force the sliding mode occur on the gradient flow. The simulation results can be found Figure 7.3.



(a)

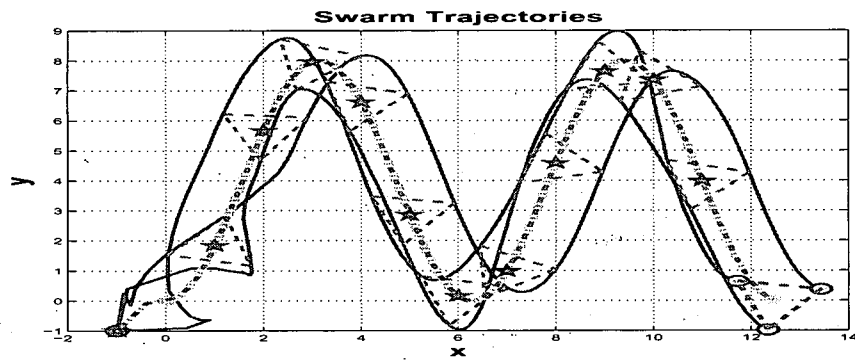


(b)

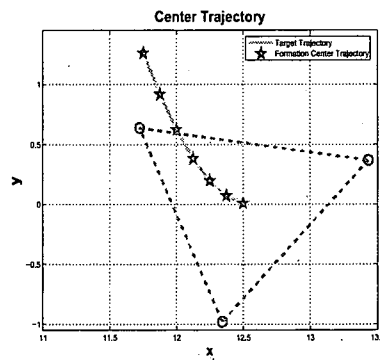


(c)

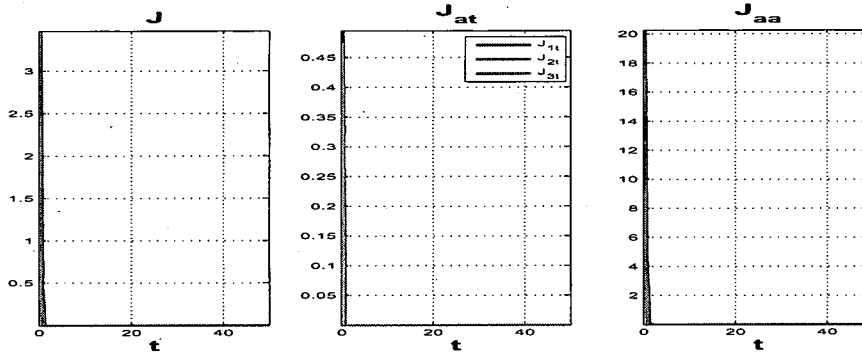
Figure 7.2: Swarm source seeking via gradient-based extremum seeking control, velocity-actuated point mass: (a) trajectories of the swarm vehicles; (b) final formation; (c) potential functions.



(a)



(b)



(c)

Figure 7.3: Swarm source seeking via gradient-based extremum seeking control, force-actuated point mass: (a) trajectories of the swarm vehicles; (b) final formation; (c) potential functions.

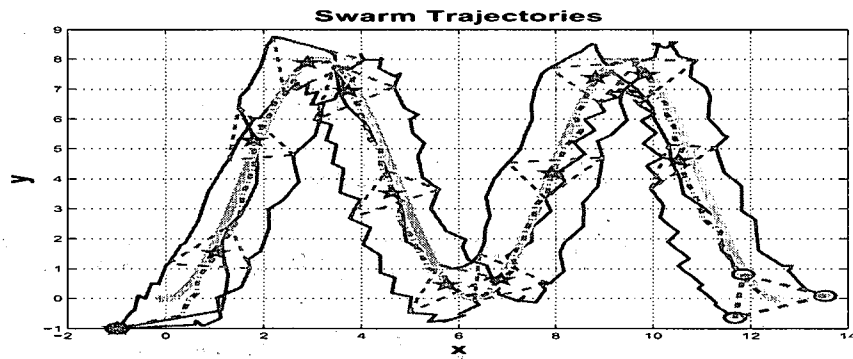
7.3 Perturbation-based Extremum Seeking Control Design

We achieved very nice simulation results by using the gradient-based extremum seeking control. However, such controller relies on the knowledge of gradient measurements, which is a very strong assumption. In this section, we apply the perturbation-based extremum seeking control for each vehicle as seen in Figure 6.1. Now, consider the performance function (7.14): the 1st vehicle is only able to influence the potential functions J_{1t} , J_{12} and J_{13} , therefore, instead of giving each vehicle the same performance function, we present the performance J_i for the i^{th} vehicle, where

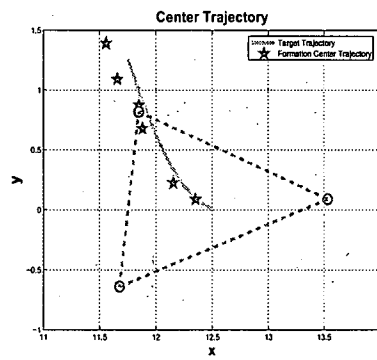
$$\begin{aligned} J_i(x^i, x_t) &= K_{at} J_{it}(\|x^i - x_t\|) + K_{aa} \sum_{j=1, j \neq i}^3 J_{ij}(\|x^i - x^j\|) \\ &= \frac{K_{at}}{2} (\|x^i - x_t\|^2 - \delta_{it}^2)^2 + \frac{K_{aa}}{2} \sum_{j=1, j \neq i}^3 J_{ij}(\|x^i - x^j\|^2 - \delta_{ij}^2). \end{aligned} \quad (7.15)$$

The simulation results can be found in Figure 7.4, where the setting of the potential function and initial conditions are the same as above. The parameters of the perturbation-based extremum seeking controller are $\omega_1 = 200, \omega_2 = 215, \omega_3 = 230, \alpha_i = 0.15$, and $c_{x_i} = c_{y_i} = 25, i = 1, 2, 3$. The weight of the potential function (7.15) are $K_{at} = 1$ and $K_{aa} = 0.2$. It is important to notice that three different perturbation frequencies are needed for three vehicles in a two dimensional plane, and such choice of distinctive frequencies is suggested from the stability analysis of multiple parameters perturbation-based extremum seeking control in [11].

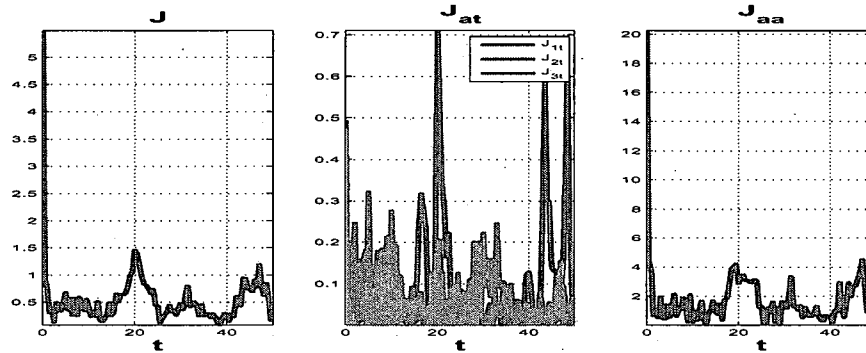
The main advantage of the perturbation-based extremum seeking control is that no gradient information and absolute position measurements are needed. The latter means no GPS or self-localization is required, only relative position between agent and agent is needed to calculate the potentials, which is easy to obtain through infrared or acoustic sensors. The former property is very appealing in incorporating new



(a)



(b)



(c)

Figure 7.4: Swarm source seeking via perturbation-based extremum seeking control, velocity-actuated point mass: (a) trajectories of the swarm vehicles ; (b) final formation; (c) potential functions.

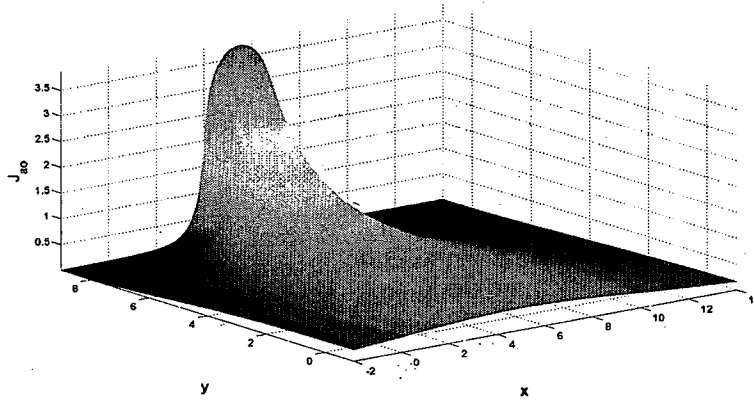


Figure 7.5: Potential function of a rectangular obstacle.

constraints or design requirement through potential function, such as, we can easily incorporate obstacle avoidance. For example, assume there is a rectangular obstacle from with its coordinate as $[o_1, o_2] \times [o_3, o_4]$, where $[o_1, o_2]$ is the x axis coverage and $[o_3, o_4]$ is the y axis coverage. We can calculate a potential between a vehicle (x^i, y^i) outside the rectangular obstacle like:

$$J_{io}(x^i, o) = \left| x_{o_1} \log \left(\frac{(x_{o_1}^2 + y_{o_1}^2)^{0.5} + y_{o_1}}{(x_{o_1}^2 + y_{o_1}^2)^{0.5} - y_{o_1}} \right) - x_{o_1} \log \left(\frac{(x_{o_1}^2 + y_{o_2}^2)^{0.5} + y_{o_2}}{(x_{o_1}^2 + y_{o_2}^2)^{0.5} - y_{o_2}} \right) \right. \\ \left. + x_{o_2} \log \left(\frac{(x_{o_2}^2 + y_{o_2}^2)^{0.5} + y_{o_2}}{(x_{o_2}^2 + y_{o_2}^2)^{0.5} - y_{o_2}} \right) - x_{o_2} \log \left(\frac{(x_{o_2}^2 + y_{o_1}^2)^{0.5} + y_{o_1}}{(x_{o_2}^2 + y_{o_1}^2)^{0.5} - y_{o_1}} \right) \right|, \quad (7.16)$$

where $o = [o_1, o_2, o_3, o_4]$, $x_{o_1} = o_1 - x^i$, $x_{o_2} = o_2 - x^i$, $y_{o_1} = o_3 - y^i$ and $y_{o_2} = o_4 - y^i$. An illustration for a rectangular obstacle ranged from $[4, 6]$ in x axis and $[8, 9]$ in y axis can be found in Figure 7.5.

Let

$$J_{ao}(x^1, x^2, x^3) = J_{1o} + J_{2o} + J_{3o},$$

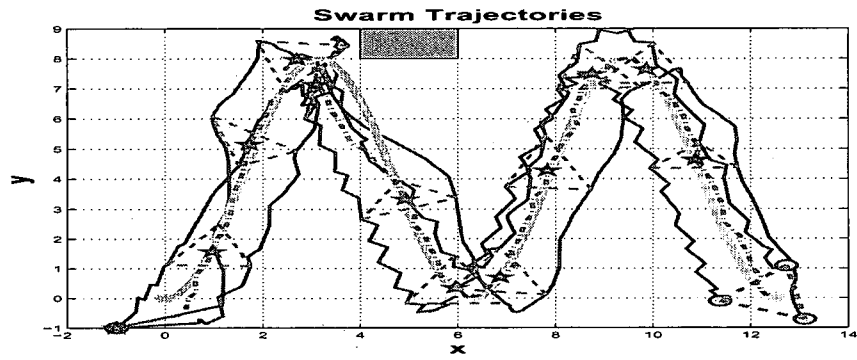
then the whole potential function (7.14) is augmented as

$$y = J(x^1, x^2, x^3, x^t) = K_{at}J_{at}(x^1, x^2, x^3, x^t) + K_{aa}J_{aa}(x^1, x^2, x^3) + K_{ao}J_{ao}(x^1, x^2, x^3), \quad (7.17)$$

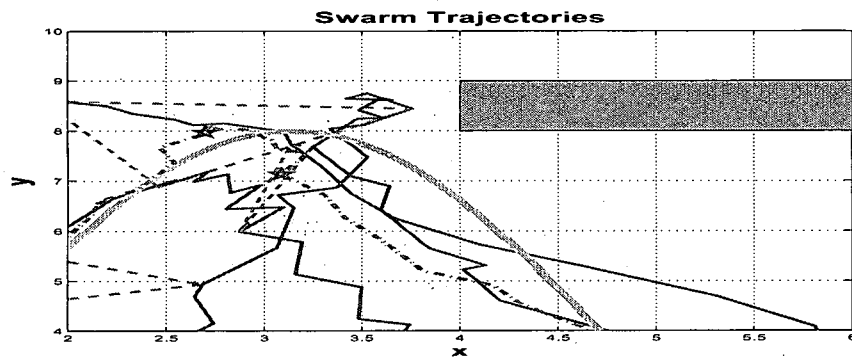
where K_{ao} is the weight of the potential component generated by the obstacle. In general, we do not know where the obstacle is, however, if the vehicle is equipped with infrared or other sensors, it is possible to sense the relative distance to the obstacle and calculate a repulsive force similar to J_{io} . In such cases, it is very difficult or impossible to calculate the gradient of the potential function J_{io} and apply the gradient-based extremum seeking control, however, we can easily incorporate such repulsive force into the performance function and apply perturbation-based extremum seeking control, which is non-gradient based. Thus, we change the performance function J_i for the i^{th} vehicle to

$$J_i(x^i, x_t, o) = K_{at}J_{it}(\|x^i - x_t\|) + K_{aa} \sum_{j=1, j \neq i}^3 J_{ij}(\|x^i - x^j\|) + K_{ao}J_{io}(x^i, o).$$

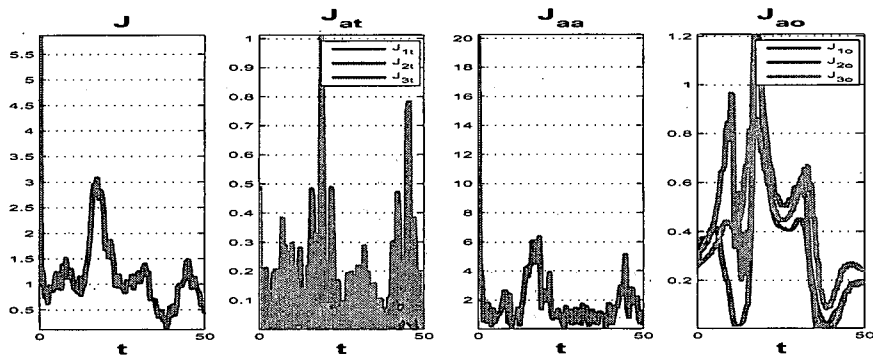
Now the simulation results by including the obstacle $o = [4, 6, 8, 9]$ can be found in Figure 7.6, where we changed $k_{aa} = 0.2$ and $k_{ao} = 0.5$. The obstacle is successfully avoided as seen in Figure 7.6 (b). Given the vehicle is modeled by a force-actuated point mass, we can use the perturbation-based extremum seeking design in Figure 6.5. The simulation results can be found in Figure 7.7, where the settings of the potential functions and initial conditions are the same as above. The parameters of the perturbation-based extremum seeking controller are $\omega_1 = 80, \omega_2 = 95, \omega_3 = 110, \alpha_i = 0.15$, and $c_{x_i} = c_{y_i} = 35, i = 1, 2, 3$. The zeros of the PD compensator are $z_{x_i} = -2$, poles are $p_{x_i} = -5$ and the gains are $k_{x_i} = 1$ for $i = 1, 2, 3$. The weights of



(a)



(b)



(c)

Figure 7.6: Swarm source seeking via perturbation-based extremum seeking control, velocity-actuated point mass, obstacle avoidance: (a) trajectories of the swarm vehicles; (b) obstacle avoidance; (c) potential functions.

the potential function (7.15) are $K_{at} = 1$ and $K_{aa} = 0.2$. The simulation results for obstacle avoidance can be found in Figure 7.8, where we add $K_{ao} = 0.5$.

7.4 Numerical Optimization-based Extremum Seeking Control Design

We can also design a numerical optimization-based extremum seeking control for the swarm tracking problem. In this case, the vehicle is modeled by (7.1), which we can write as

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

where $x, u \in \mathbb{R}^n$, $A = 0$ and $B = [1, 1, \dots, 1]^\top$. Therefore, we can apply numerical optimization-based extremum seeking control, where the set point candidate x_{k+1} is generated by the numerical optimization algorithm and the state regulator can be designed as

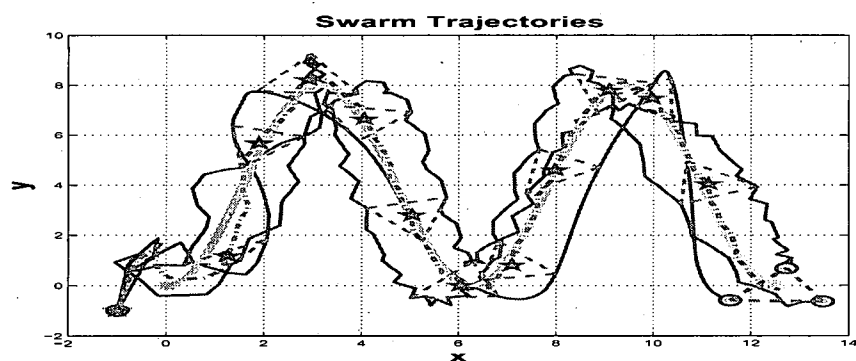
$$u = -B^\top e^{A^\top(t_{k+1}-t)} W_c^{-1}(\delta_k) [e^{A\delta_k} x(t_k) - x_{k+1}] = -\frac{1}{\delta_k} (x(t_k) - x_{k+1}), \quad (7.18)$$

which will regulate the vehicle to x_{k+1} in δ_k time, or

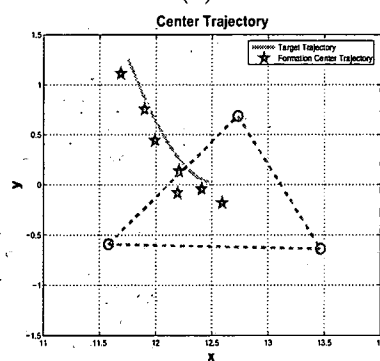
$$u = -k(x - x_{k+1}), \quad (7.19)$$

which will regulate the vehicle to x_{k+1} asymptotically.

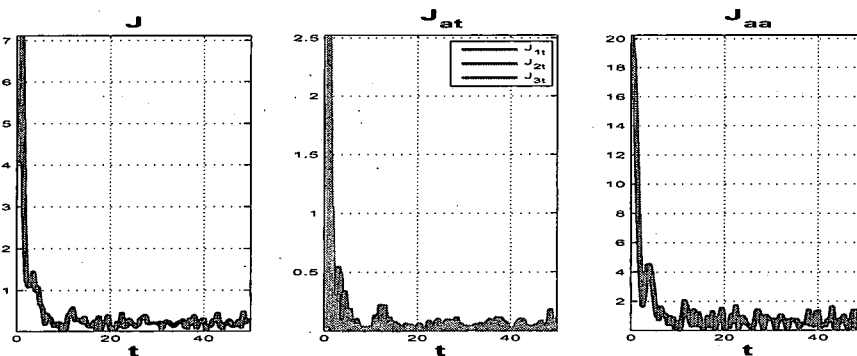
We use direct search to generate the set point candidate and the asymptotic regulator (7.19) in the simulation, where the setting of the potential functions and initial conditions are the same as before. The additional requirement for direct search is to have four performance output measurements in the same time on the corner of the rectangular structure centered at the vehicle's current position, which can be realized if the vehicle is equipped with four sensors to measure the environments.



(a)



(b)



(c)

Figure 7.7: Swarm source seeking via perturbation-based extremum seeking control, force-actuated point mass: (a) trajectories of the swarm vehicles; (b) final formation; (c) potential functions.

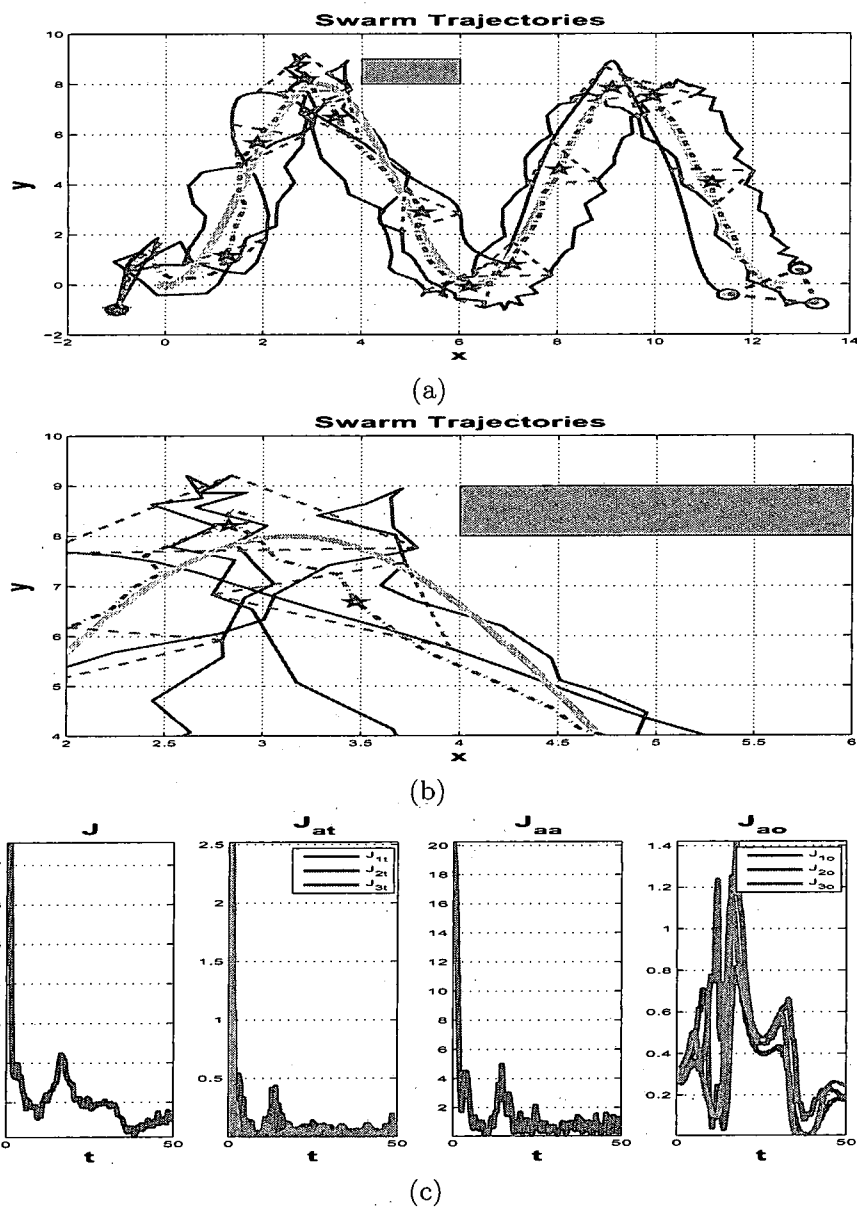


Figure 7.8: Swarm source seeking via perturbation-based extremum seeking control, force-actuated point mass, obstacle avoidance: (a) trajectories of the swarm vehicles; (b) obstacle avoidance; (c) potential functions.

For the obstacle free case, the simulation results can be found in Figure 7.9, where $\delta_k = 0.01, k = 5, K_{at} = 1$ and $K_{aa} = 0.2$. The simulation results for the obstacle at $o = [4, 6, 8, 9]$ can be found in Figure 7.10, where $\delta_k = 0.005, k = 30, K_{at} = 1, k_{aa} = 0.2$ and $k_{ao} = 0.05$.

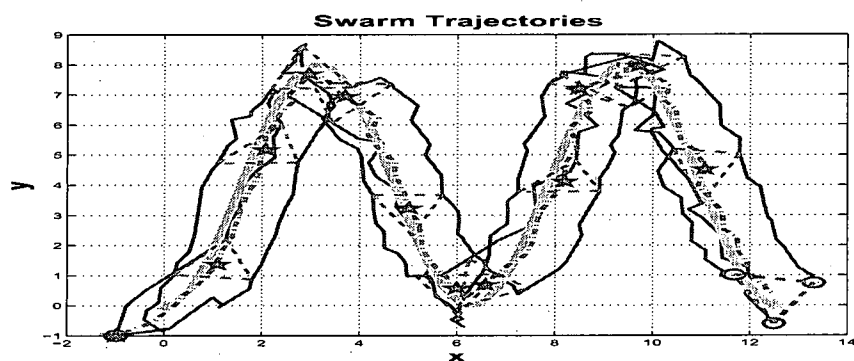
Consider a force-actuated point mass model in the two dimensional plane, let $x = [x_p, y_p, x_v, y_v]$, note this time the performance is only a function of the position, thus we need to design $u = [u_x, u_y]$ to regulate the position state $[x_p, y_p]$ to the set point x_{k+1} . we can let $e_x = k_1(x_p - x_{k+1}(1)) + v_x$ and $e_y = k_1(y_p - y_{k+1}(1)) + v_y$, then $\dot{e}_x = k_1\dot{x}_p + \dot{v}_x = k_1v_x + u_x$ and $\dot{e}_y = k_1\dot{y}_p + \dot{v}_y = k_1v_y + u_y$, we can design the state regulator

$$u = [u_x, u_y]^T = [-k_1v_x - ke_x, -k_1v_y - ke_y], \quad (7.20)$$

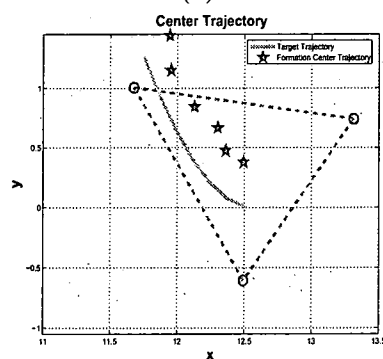
In the simulation results shown in Figure 7.11, where $\delta_k = 0.01, K_{at} = 1, K_{aa} = 0.2, k_1 = 10$ and $k = 20$. Then in the obstacle case as seen in Figure 7.12, we have $\delta_k = 0.005, K_{at} = 1, K_{aa} = 0.3, K_{ao} = 0.05, k_1 = 10$ and $k = 10$. Moreover, such controller design (7.20) is robust to the bounded input disturbance due to the damping term $-ke_x$ and $-ke_y$, simulation results for a force-actuated point mass model (double integrator) for the obstacle avoidance case with unknown input disturbance $5 \sin(0.2t)$ can be found in Figure 7.13,

7.5 Comparison

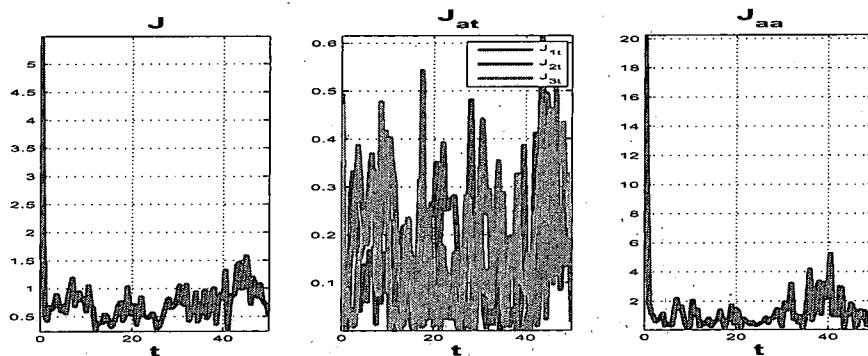
1. All the three extremum seeking control designs achieve source seeking, formation control and collision avoidance.
2. The gradient-based extremum seeking control design obtains the best performance based on its requirement of gradient information, which amounts to



(a)



(b)



(c)

Figure 7.9: Swarm source seeking via numerical optimization-based extremum seeking control, velocity-actuated point mass: (a) trajectories of the swarm vehicles; (b) final formation; (c) potential functions.

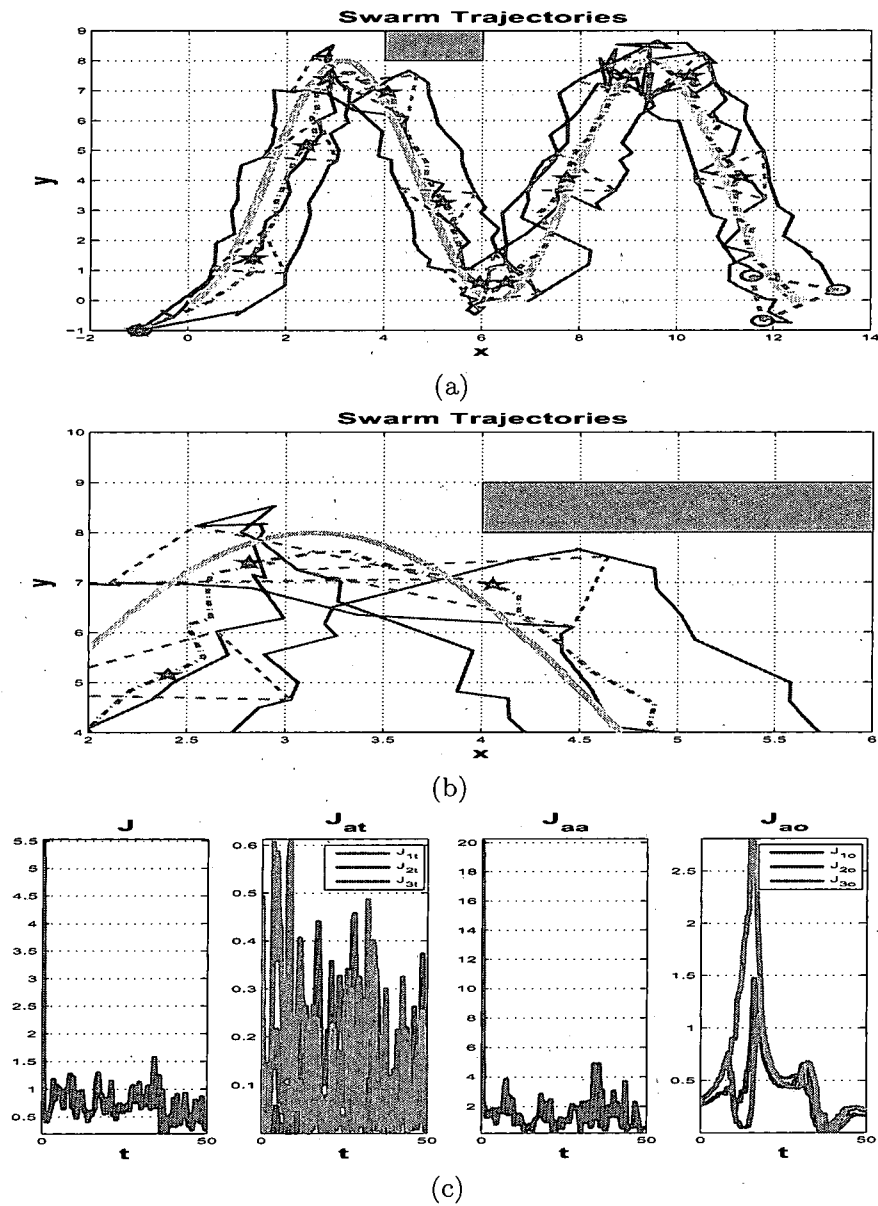


Figure 7.10: Swarm source seeking via numerical optimization-based extremum seeking control, velocity-actuated point mass, obstacle avoidance: (a) trajectories of the swarm vehicles; (b) obstacle avoidance; (c) potential functions.

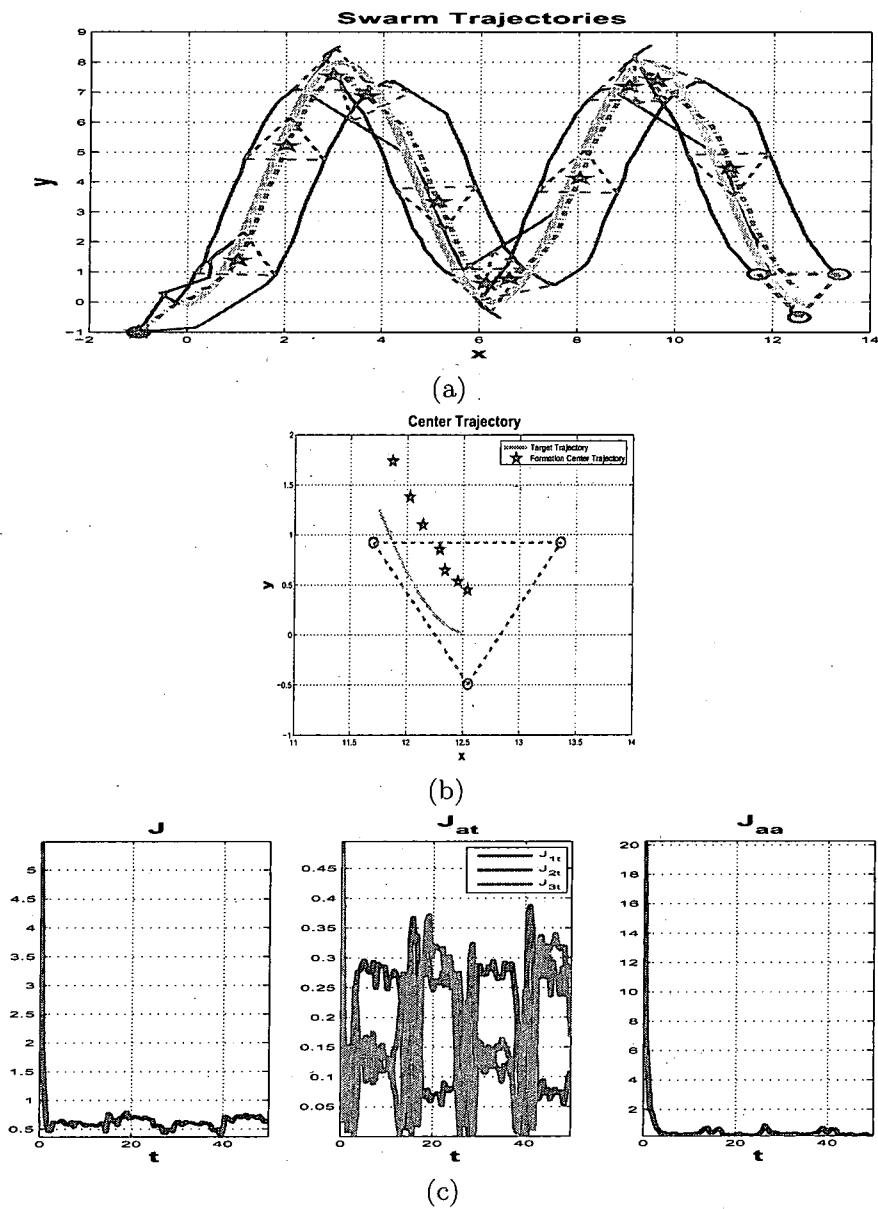


Figure 7.11: Swarm source seeking via numerical optimization-based extremum seeking control, force-actuated point mass: (a) trajectories of the swarm vehicles; (b) final formation; (c) potential functions.

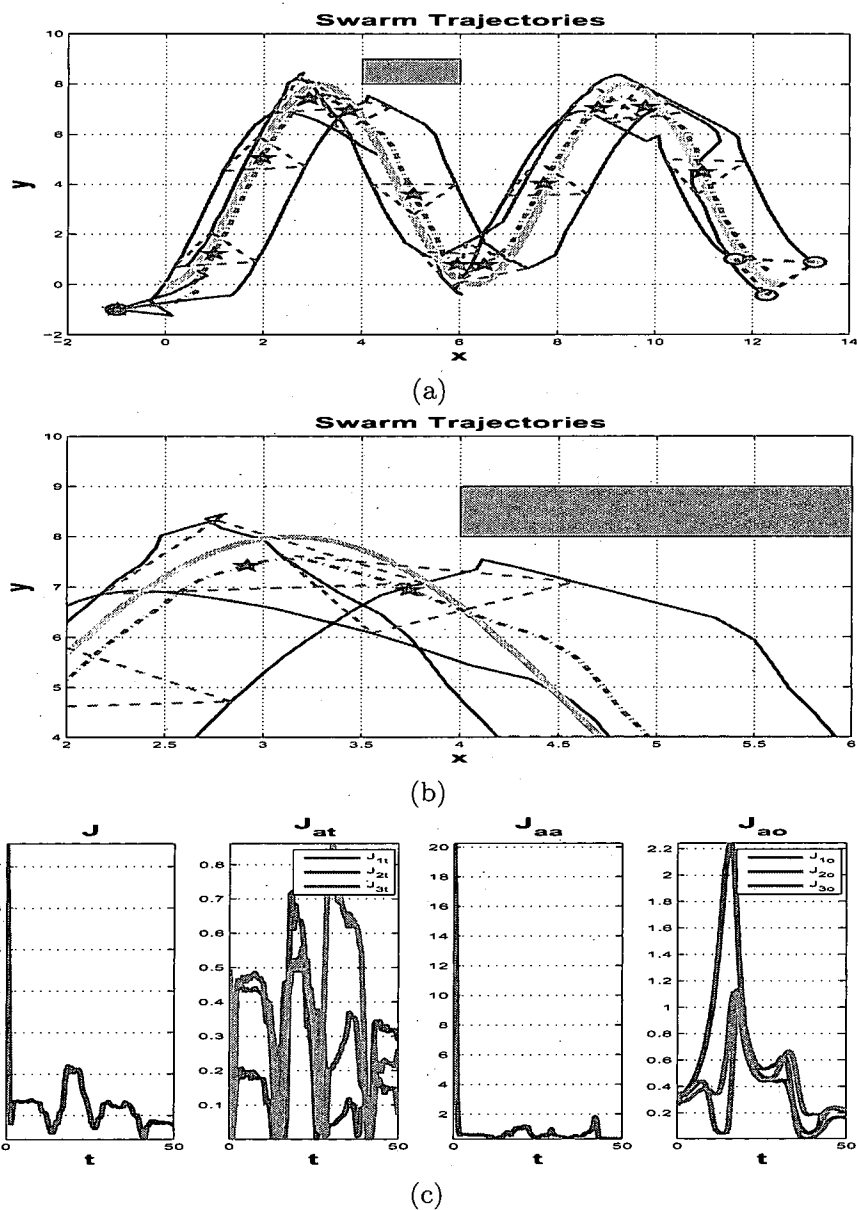
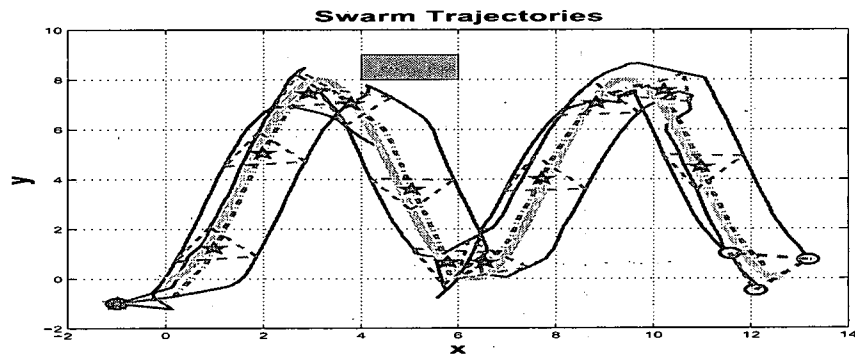
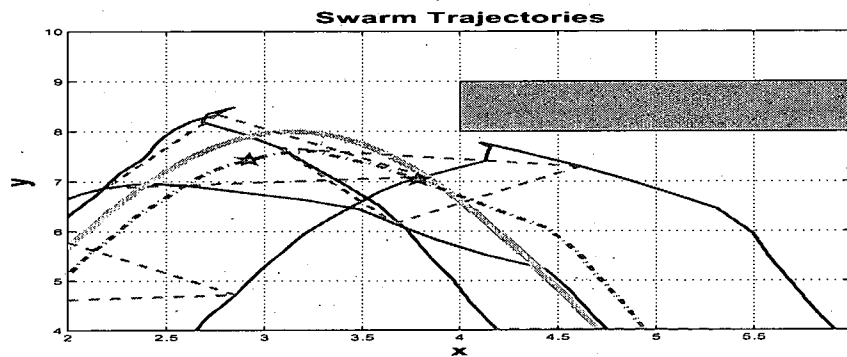


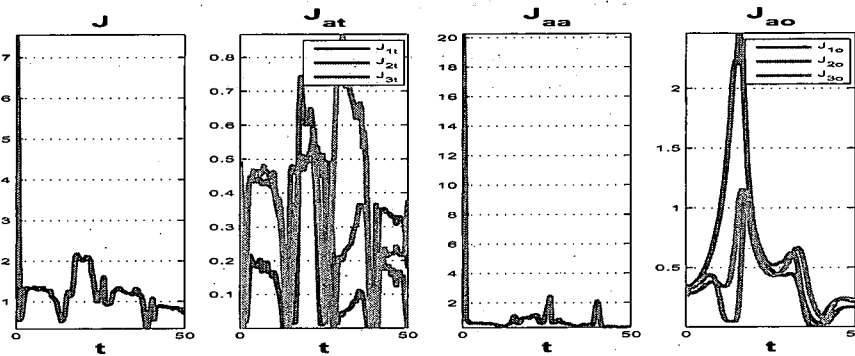
Figure 7.12: Swarm source seeking via numerical optimization-based extremum seeking control, force-actuated point mass, obstacle avoidance: (a) trajectories of the swarm vehicles; (b) obstacle avoidance; (c) potential functions.



(a)



(b)



(c)

Figure 7.13: Swarm source seeking via numerical optimization-based extremum seeking control, force-actuated point mass, obstacle avoidance with input disturbance: (a) trajectories of the swarm vehicles; (b) obstacle avoidance; (c) potential functions.

knowing the target position. It can be used to avoid obstacles as long as we can calculate the gradient field of the potential between the obstacle and the autonomous vehicles. However, it is generally impossible for irregular and unknown obstacles.

3. The perturbation-based and numerical optimization-based designs both achieve source seeking, formation control, and collision avoidance with good performance. The obstacle avoidance is also easy to obtain via the incorporation of additional potential measurements between the obstacle and the autonomous vehicles due to the non-gradient extremum seeking ability. And such design allows for unknown and irregular obstacle as long as one can obtain the potential value via sensor measurements.
4. The extension of perturbation-based extremum seeking control design to more complicated vehicle dynamics is not straightforward, and additional robust design issues are not addressed. The numerical optimization-based design is easy to fit for more general model of autonomous vehicle, the robust and adaptive design techniques for numerical optimization-based extremum seeking provide large space to accommodate input disturbance, and unmodeled plant dynamics.
5. By observing the simulation results, we found that the control gain of the perturbation-based design tends to be high due to the tracking moving target, moreover, similar things also happen for the numerical optimization-based design to improve seeking speed by reducing the regulation time, which also means an increase of control gain. Generally, we need to have an estimation of the velocity of the target to choose a suitable regulation time δ_k .

6. In the obstacle avoidance case, unlike the continuous optimization running in the gradient-based or perturbation-based design, the numerical optimization-based design does not guarantee that the vehicle will avoid the obstacle during the transient due to the fact that no potential force is fed back in the regulation period.

CHAPTER 8

CONCLUSIONS AND FUTURE WORK

8.1 Conclusions

The motivation for the research on extremum seeking control arises from its practical interest, since even small improvements in the performance can lead to large savings in raw material and energy consumption. The main purpose of this dissertation was to develop new theory for perturbation-based extremum seeking control, numerical optimization-based extremum seeking control and their applications.

We begin with a brief introduction of the history and existing literature on extremum seeking control. Then, we group the gradient-based, sliding mode-based and perturbation-based extremum seeking control into the analog optimization-based extremum seeking control. We analyze the robustness of the perturbation-based extremum seeking control given a slightly unstable system and then introduce a new idea based on phase lead compensators to extend the applicability of perturbation-based extremum seeking control to moderately unstable systems.

In the mean time to explore the continuous implementation of optimization algorithms to fit the analog optimization-based extremum seeking framework, we propose

the novel numerical optimization-based extremum seeking control based on numerical optimization algorithms and state regulation. The convergence of this extremum seeking control scheme is guaranteed by the global convergence of the optimization algorithm and state regulation. We also analyze the robustness of line search methods and trust region methods, which provide design criteria for a robust extremum seeking control scheme. Numerical examples are given to illustrate the analysis results. Further design flexibility is achieved via the robustness analysis of the optimization algorithm and output tracking framework, then the robust and adaptive design of extremum seeking control is studied, where nonlinear damping and nonlinear adaptive control are used to deal with input disturbance and unmodeled plant dynamics.

We then perform a comparative study on ABS design via different extremum seeking control schemes. A more interesting and promising application studied here is the autonomous vehicle source seeking problem. The design via gradient and numerical optimization-based extremum seeking is straightforward. We especially extend the perturbation-based extremum seeking control for the design of autonomous vehicle control law in performing source seeking, where stability analysis is conducted for different models of autonomous vehicle. Finally, we make further progress on the swarm seeking problem, where source seeking, formation control, collision avoidance and obstacle avoidance of a group of autonomous vehicles are achieved via extremum seeking control and potential field.

8.2 Future Work

The application of perturbation-based extremum seeking control can be pursued in much greater generality than in the present study, allowing additional stable and fast

dynamics, combined with unstable poles. Such robust design will be of future interests. Another interesting problem is the stability analysis for the multiple parameter case $\theta \in \mathbb{R}^p, p > 1$ for the perturbation-based extremum seeking control of nonlinear systems. Increasing the chance to find a global minimum in the perturbation-based or sliding mode-based optimization, and other continuous implementations of optimization algorithms will be interesting topics to pursue.

The setting of numerical optimization-based extremum seeking is one attractive way to use numerical optimization algorithms for the purpose of real time optimization, as it retains the global convergence property of the numerical optimization algorithm. It allows us to utilize the current research results from the optimization society, and it further expedites us to extend the extremum seeking under certain constraints. However, it does waste a lot of information by allowing the numerical optimizer to work discretely. In the future, the design of the regulation time δ_k needs to be further studied to deal with various requirements originated from practical applications. And the exploration of more robust numerical optimization algorithms and the design of robust state regulators will be two ways to enhance the robustness of the extremum seeking control schemes.

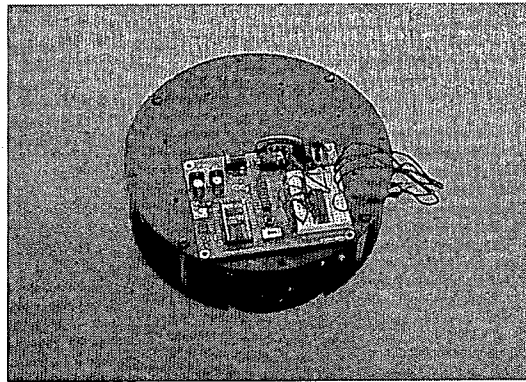
Moreover, we have assumed in this dissertation that the nonlinear system (2.1) is globally feedback linearizable by letting $D = \mathbb{R}^n$, which is a strong assumption and can be relaxed by considering a nonlinear system that is only feedback linearizable in $D \subset \mathbb{R}^n$. Furthermore, we may want to put additional algebraic constraints on the performance function by minimizing function $J(x)$ where, for example, the state needs to satisfy certain inequality constraints, $a_i \leq x_i \leq b_i, i \in \{1, \dots, n\}$, which can be simplified as $\min_x J(x)$ subject to $x \in E \subset \mathbb{R}^n$. Also in Section 4.3, we use

function approximators to approximate either the dynamics or the nominal controller, for which the approximation is only valid on the subset S ; the controller design has not yet been addressed to guarantee that the state will not exit the subset S (although this can be easily achieved using, e.g., high gain bounding control terms or via analysis techniques similar to those in [60]). Thus, extremum seeking control with state constraints will be an important topic to study. Some basic ideas can be the following: let $S_x \subseteq D \cap E \cap T^{-1}(S)$, if we can choose controller parameters to ensure the state x stays within the set S_x , then above requirements are all satisfied. Of course, we first would need a constrained optimization algorithm (e.g., penalty methods or barrier methods [125, 116, 126]) to generate the set point candidate x_{k+1} inside S_x , then we can use Lyapunov method to choose parameters for the state regulator such that x will not only converge to the required vicinity of x_{k+1} , but also not violate the constraints and stay within the require set S_x during the transient.

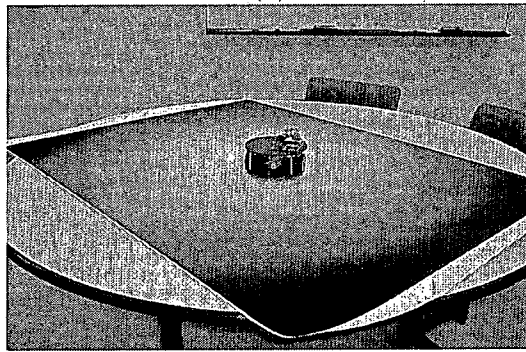
In the autonomous vehicle control design, the star patterned motion of the nonholonomic unicycle in seeking the source shown does not seem like the most energy efficient way to move around in space; however, an even bigger problem is that it requires motion both forward and in reverse. As such, it is implementable by mobile robots, ground vehicles, and some underwater vehicles, but not by aircraft. The future research of nonholonomic unicycle will present a result dual to the result in Section 6.4 – the forward velocity will be held constant and the angular velocity will be tuned. We have developed an extremum seeking configuration which achieves source seeking in simulations. However, the analysis is considerably more challenging because the system has one extra state (θ) and the system equations involve sinusoids of sinusoids. The stability analysis of swarm seeking is another important problem

to study. Moreover, high level coordination algorithms [158, 159] can be made more practical via extremum seeking control.

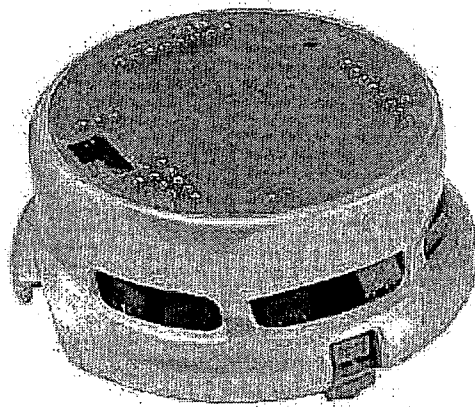
As with the research group at University of California San Diego, we plan on performing experiments with the source seeking problem studied in Chapter 6. The mobile robot in Figure 8.1 (a) and (b) is equipped with a light sensor and will be made to seek the extremum on a large printed paper surface, produced on a plotter, and blending smoothly in color from black to gray to white. Our mobile robot is not a unicycle. The motion of its two wheels can be scheduled (in the same or in the opposite direction) to produce arbitrary forward and angular velocities, similar to what is done for the "Caltech hovercraft" [160]. In the research group at University of Dayton, three Khepera robots may be used for the swarm seeking problem in Chapter 7. A single Khepera robot can be seen in Figure 8.1 (c).



(a)



(b)



(c)

Figure 8.1: (a) A two-wheel autonomous vehicle for future experiments, (b) Test bed: a vehicle with a source field represented with a gray scale image; (c) Khepera robot.

BIBLIOGRAPHY

- [1] S. V. Drakunov, Ümit Özgüner, P. Dix, and B. Ashrafi, "Abs control using optimum search via sliding modes," *IEEE Transactions on Control Systems Technology*, no. 3, pp. 79–85, 1995.
- [2] İlker Tunay, "Antiskid control for aircraft via extremum-seeking," in *Proceedings of the American Control Conference*, vol. 2, pp. 665–670, 2001.
- [3] K. B. Ariyur and M. Krstić, *Real-Time Optimization by Extremum-Seeking Control*. Hoboken, NJ: Wiley-Interscience, 2003.
- [4] K. J. Astrom and B. Wittenmark, *Adaptive Control*. Boston, MA: Wiley, 1994.
- [5] H.-H. Wang, M. Krstić, and G. Bastin, "Optimizing bioreactors by extremum seeking," *International Journal of adaptive Control and Signal Processing*, vol. 13, no. 8, pp. 651–669, 1999.
- [6] T. Zhang, M. Guay, and D. Dochain, "Adaptive extremum seeking control of continuous stirred-tank bioreactors," *AIChE Journal*, vol. 17, no. 10, pp. 113–123, 2003.
- [7] P. Binetti, K. B. Ariyur, and M. Krstić, "Control of formation flight via extremum seeking," in *Proceedings of the American Control Conference*, vol. 4, pp. 2848–2853, 2002.
- [8] N. Killingsworth and M. Krstić, "Pid tuning using extremum seeking," *Control Systems Magazine*, vol. 26, pp. 70–79, 2006.
- [9] E. Lavretsky, N. Hovakimyan, and A. Calise, "Adaptive extremum seeking control design," in *Proceedings of the American Control Conference*, vol. 1, pp. 567–572, 2003.
- [10] Nusawardhana and S. H. Žak, "Simultaneous perturbation extremum seeking method for dynamic optimization problems," in *Proceedings of the American Control Conference*, pp. 2085–2810, 2004.

- [11] M. A. Rotea, "Analysis of multivariable extremum seeking algorithms," in *Proceedings of the American Control Conference*, vol. 1, pp. 433-437, 2000.
- [12] R. N. Banavar, D. F. Chichka, and J. L. Speyer, "Convergence and synthesis issues in extremum seeking control," in *Proceedings of the American Control Conference*, vol. 1, pp. 438-443, 2000.
- [13] M. H. Hamza, "Extremum control of continuous systems," *IEEE Transactions on Automatic Control*, vol. 11, no. 2, pp. 182-189, 1966.
- [14] I. I. Ostrovskii, "Extremum regulation," *Automatic & Remote Control*, no. 18, pp. 900-907, 1957.
- [15] I. S. Morosanov, "Method of extremum control," *Automatic & Remote Control*, no. 18, pp. 1077-1092, 1957.
- [16] A. A. Pervozvanskii, "Continuous extremum control system in the presence of random noise," *Automation and Remote Control*, no. 21, pp. 673-677, 1960.
- [17] P. F. Blackman, "Extremum-seeking control," *J. H. Westcott Ed., An Exposition of Adaptive Control*, The Macmillan Company, New York, NY, 1962.
- [18] C. S. Drapper and Y. T. Li, "Principles of optimizing control systems and an application to the internal combustion engine," *R. Oldenburger Ed., Optimal and self-Optimizing Control*, MIT Press, Boston, MA, 1962.
- [19] S. M. Meerkov, "Asymptotic methods for investigating quasistationary states in continuous systems of automatic optimization," *Automation and Remote Control*, no. 11, pp. 1726-1743, 1967.
- [20] S. M. Meerkov, "Asymptotic methods for investigating a class of forced states in extremal systems," *Automation and Remote Control*, no. 12, pp. 1916-1920, 1967.
- [21] S. M. Meerkov, "Asymptotic methods for investigating stability of continuous systems of automatic optimization subject to disturbance action," *Automatika i Telemekhanika*, no. 12, pp. 14-24, (in Russian) 1967.
- [22] O. L. R. Jacobs and G. C. Shering, "Design of a single-input sinusoidal-perturbation extremum-control system," in *Proceedings of the IEE*, vol. 115, pp. 212-217, 1968.
- [23] H. S. Tsien, *Engineering Cybernetics*. New York, NY: McGraw-Hill, 1954.
- [24] A. A. Feldbaum, *Computers in Automatic Control Systems*. 1959, in Russian.

- [25] A. A. Krasovskii, *Dynamics of Continuous Self-Tuning Systems*. 1963, in Russian.
- [26] D. J. Wilde, *Optimum Seeking Methods*. Prentice Hall, 1964.
- [27] E. P. I. Chinaev, *Self-Tuning Systems Handbook*. 1969, in Russian.
- [28] M. Leblanc, "Sur l'électrification des chemins de fer au moyen de courants alternatifs de fréquence élevée," *Revue generale de l'Electricite*, 1922.
- [29] J. Sternby, "Extremum control systems: An area for adaptive control?," in *Preprints of the Joint American Control Conference*, 1980.
- [30] M. Krstić and H.-H. Wang, "Design and stability analysis of extremum seeking feedback for general nonlinear systems," *Automatica*, vol. 36, no. 2, pp. 595–601, 2000.
- [31] M. Krstić, "Performance improvement and limitations in extremum seeking control," *Systems and Control Letters*, vol. 39, no. 5, pp. 313–326, 2000.
- [32] Y. Pan, Ümit Özgüner, and T. Acarman, "Stability and performance improvement of extremum seeking control with sliding mode," *International Journal of Control*, vol. 76, no. 9/10, pp. 968–985, 2003.
- [33] Nusawardhana and S. H. Žak, "Extremum seeking using analog nonderivative optimizers," in *Proceedings of the American Control Conference*, vol. 4, pp. 3242–3247, 2003.
- [34] J.-Y. Choi, M. Krstić, K. B. Ariyur, and J. S. Lee, "Extremum seeking control for discrete-time systems," *IEEE Transactions on Automatic Control*, vol. 47, no. 2, pp. 318–323, 2002.
- [35] H. Takata, T. Hachino, R. Tamura, and K. Komatsu, "Design of extremum seeking control with accelerator," *IEICE Transactions on Fundamentals*, vol. E88-A, no. 10, pp. 2535–2540, 2005.
- [36] G. C. Walsh, "On the application of multi-parameter extremum seeking control," in *Proceedings of the American Control Conference*, vol. 1, pp. 411–415, 2000.
- [37] K. B. Ariyur and M. Krstić, "Analysis and design of multivariable extremum seeking," in *Proceedings of the American Control Conference*, vol. 4, pp. 2903–2908, 2002.
- [38] K. B. Ariyur and M. Krstić, "Sloping seeking: a generalization of extremum seeking," *International Journal of adaptive Control and Signal Processing*, vol. 18, no. 2, pp. 1–22, 2002.

- [39] Y. Tan, D. Nesic, and I. Mareels, "On non-local stability properties of extremum seeking control," in *Proceedings of the 16th IFAC World Congress*, 2005.
- [40] C. Zhang, A. Siranosian, and M. Krstić, "Extremum seeking for moderately unstable systems and for autonomous target tracking without position measurements," in *Proceedings of the American Control Conference*, 2006.
- [41] S. K. Korovin and V. I. Utkin, "Use of the sliding mode in problems of static optimization," *Automation and Remote Control*, no. 33, pp. 570–579, 1972.
- [42] S. K. Korovin and V. I. Utkin, "Using sliding modes in static optimization and nonlinear programming," *Automatica*, vol. 10, pp. 525–532, 1974.
- [43] S. V. Drakunov and Ümit Özgüner, "Optimization of nonlinear system output via sliding mode approach," in *Proceedings of the IEEE International Workshop on Variable Structure and Lyapunov Control of Uncertain Dynamical Systems, Sheffield, UK*, pp. 61–62, 1992.
- [44] J. Winkelman, I. Haskara, and Ümit Özgüner, "Tuning for dynamic spark advance control," in *Proceedings of the American Control Conference*, pp. 163–164, 1999.
- [45] I. Haskara, Ümit Özgüner, and J. Winkelman, "Extremum control for optimal operating point determination and set point optimizing via sliding modes," *Transaction of the ASME, Journal of Dynamical Systems, Measurement, and Control*, no. 122,4, pp. 719–724, 2000.
- [46] I. Haskara, C. Hatipoglu, and Ümit Özgüner, "Sliding mode compensation, estimation and optimization methods in automotive control," *Variable Structure Systems: Towards the 21st Century, Lecture Notes in Control and Information Sciences, Xinghuo Yu and Jian-Xin Xu (Eds.)*, vol. 274, pp. 155–174, 2002.
- [47] Y. Pan, T. Acarman, and Ümit Özgüner, "Nash solution by extremum seeking control approach," in *Proceedings of the Conference on Decision and Control*, vol. 1, pp. 329–334, 2002.
- [48] H. Yu and Ümit Özgüner, "Extremum-seeking control via sliding mode with periodic search signals," in *Proceedings of the Conference on Decision and Control*, vol. 1, pp. 323–328, 2002.
- [49] H. Yu and Ümit Özgüner, "Smooth extremum-seeking control via second order sliding mode," in *Proceedings of the American Control Conference*, vol. 4, pp. 3248–3253, 2003.

- [50] Y. Pan and Ümit Özgüner, "Sliding mode extremum seeking control for linear quadratic dynamic game," in *Proceedings of the American Control Conference*, pp. 614–619, 2004.
- [51] H. S. Tan and M. Tomizuka, "An adaptive sliding mode vehicle traction controller design," in *Proceedings of the American Control Conference*, pp. 1035–1058, 1989.
- [52] S. Yamanka and H. Ohmori, "Nonlinear adaptive extremum seeking control for time delayed index in the presence of deterministic disturbance," in *Proceedings of the PhysCon 2005*, pp. 121–125, 2005.
- [53] Y. K. C. et al, "Sliding mode abs wheel slip control," in *Proceedings of the American Control Conference*, pp. 1–6, 1992.
- [54] I. Haskara, Ümit Özgüner, and J. Winkelman, "Wheel slip control for antispin acceleration via dynamic spark advance," *IFAC Journal of Control Engineering Practice*, no. 8,10, pp. 1135–1148, 2000.
- [55] H. Yu and Ümit Özgüner, "Extremum-seeking control strategy for ABS system with time delay," in *Proceedings of the American Control Conference*, vol. 5, pp. 3753–3758, 2002.
- [56] J. L. Speyer, R. N. Banavar, D. F. Chichka, and I. Rhee, "Extremum seeking loops with assumed functions," in *Proceedings of the 39th Conference on Decision and Control*, vol. 1, pp. 142–147, 2000.
- [57] R. N. Banavar, D. F. Chichka, and J. L. Speyer, "Functional feedback in an extremum seeking loop," in *Proceedings of the 40th Conference on Decision and Control*, vol. 2, pp. 1316–1321, 2001.
- [58] R. N. Banavar, "Extremum seeking loops with assumed functions: estimation and control," in *Proceedings of the American Control Conference*, vol. 4, pp. 3159–3164, 2002.
- [59] R. N. Banavar, D. F. Chichka, and J. L. Speyer, "Convergence and synthesis issues in extremum seeking control," *International Journal of adaptive Control and Signal Processing*, vol. 17, pp. 751–762, 2002.
- [60] J. T. Spooner, M. Maggiore, R. Ordóñez, and K. M. Passino, *Stable Adaptive Control and Estimation for Nonlinear Systems, Neural and Fuzzy Approximator Techniques*. New York, NY: John Wiley and Sons, 2002.
- [61] P. Y. Li and R. Horowitz, "Self-optimizing control," in *Proceedings of the Conference on Decision and Control*, pp. 1228–1233, 1997.

- [62] P. Y. Li and R. Horowitz, "Control of smart exercise machine – part ii: Self-optimizing control," *IEEE Transactions on Mechatronics*, vol. 2, no. 4, pp. 248–258, 1997.
- [63] J. Shields and R. Horowitz, "Controller design and experimental verification of a self-optimizing motion controller for smart exercise machines," in *Proceedings of the American Control Conference*, pp. 2736–2742, 1997.
- [64] M. Guay and T. Zhang, "Adaptive extremum seeking control of nonlinear dynamic systems with parametric uncertainties," *Automatica*, vol. 39, no. 7, pp. 1283–1293, 2003.
- [65] V. Adetola, D. Dehaan, and M. Guay, "Adaptive extremum seeking receding horizon control of nonlinear system," in *Proceedings of the American Control Conference*, pp. 2937–2942, 2004.
- [66] V. Adetola and M. Guay, "Adaptive output feedback extremum seeking receding horizon control of linear systems," *Journal of Process Control*, vol. 16, pp. 521–533, 2006.
- [67] D. Dochain and M. Guay, "Extremum seeking control of state-constrained nonlinear systems," *Automatica*, vol. 41, pp. 1567–1574, 2005.
- [68] M. Guay, "Real-time dynamical optimization of nonlinear systems: A flatness-based approach," in *Proceedings of the Conference on Decision and Control*, pp. 5842–5847, 2005.
- [69] M. C. M. Teixeira and S. H. Zak, "Analog neural nonderivative optimizers," *IEEE Transactions on Neural Networks*, vol. 9, no. 4, pp. 629–638, 1998.
- [70] Y. Zhang, "Stability and performance tradeoff with discrete time triangular search minimum seeking," in *Proceedings of the American Control Conference*, vol. 1, pp. 423–427, 2000.
- [71] A. R. Teel and D. Popović, "Solving smooth and nonsmooth multivariable extremum seeking problems by the methods of nonlinear programming," in *Proceedings of the American Control Conference*, vol. 3, pp. 2394–2399, 2001.
- [72] C. Zhang and R. Ordóñez, "Numerical optimization-based extremum seeking control of lti systems," in *Proceedings of the Conference on Decision and Control*, 2005.
- [73] C. Zhang and R. Ordóñez, "Non-gradient extremum seeking control of feedback linearizable systems with application to abs design," in *submitted to the Conference Decision and Control*, 2006.

- [74] C. Zhang and R. Ordóñez, "Numerical optimization-based extremum seeking control with application to abs design," *IEEE Transactions on Automatic Control*, accepted for publication.
- [75] C. Zhang and R. Ordóñez, "Extremum seeking control based on numerical optimization and state regulation – part i: Theory and framework," in *submitted to the Conference Decision and Control*, 2006.
- [76] C. Zhang and R. Ordóñez, "Extremum seeking control based on numerical optimization and state regulation – part ii: Robust and adaptive control design," in *submitted to the Conference Decision and Control*, 2006.
- [77] A. G. Ivakhnenko, "On constructing an extremum controller without hunting oscillations," *IEEE Transactions on Automatic Control*, vol. 12, no. 2, pp. 144–153, 1967.
- [78] P. Wellstead and P. Scotson, "Self-tuning extremum control," *IEE Proceedings D Control Theory and Applications*, vol. 137, no. 3, pp. 165–175, 1990.
- [79] P. Wellstead and P. Scotson, "Self-tuning optimization of spark ignition automotive engines," *IEEE Control Systems Magazine*, vol. 10, no. 3, pp. 94–101, 1990.
- [80] J. J. Floretin, "An approximately optimal extremal regulator," *Journal of Electronics and Control*, vol. 17, no. 2, pp. 211–310, 1964.
- [81] C. L. Nascimento, M. B. Zarrop, and A. Muir, "A neural network extremum controller," in *Proceedings of the European Control Conference, Groningen, The Netherlands*, 1993.
- [82] M. B. Zarrop and M. Rommens, "Convergence of a multi-input adaptive extremum controller," in *IEE Proceedings-D*, vol. 140, March 1993.
- [83] W. Bamberger and R. Isermann, "Adaptive on-line steady-state optimization of slow dynamic processes," *Automatica*, vol. 14, pp. 223–232, 1978.
- [84] L. B. Navarro and M. B. Zarrop, "Dynamic extremum control," in *Proceedings of the IEE Colloquium on Adaptive Control, London*, 1996.
- [85] L. B. Navarro and M. B. Zarrop, "Adaptive implicit control using extremum control methods," in *Proceedings of the UKACC International Conference on Control*, 1996.
- [86] B. Wittenmark and R. J. Evans, "Extremal control of wiener model processes," in *Proceedings of the Conference on Decision and Control*, pp. 4637–4642, 2002.

- [87] S. Velut and P. Hagander, "Analysis of probing control strategy," in *Proceedings of the American Control Conference*, vol. 4, pp. 609–614, 2003.
- [88] S. Velut and P. Hagander, "A probing control strategy: stability and performance," in *Proceedings of the Conference on Decision and Control*, pp. 4607–4612, 2004.
- [89] Y. Pan and Ümit Özgüner, "Discrete-time extremum seeking algorithms," in *Proceedings of the American Control Conference*, vol. 4, pp. 3147–3152, 2002.
- [90] B. Egardt and S. Larsson, "On a parameter adaptive extremum controller," in *Proceedings of the Conference on Decision and Control*, pp. 4809–4814, 2005.
- [91] X. T. Zhang, D. M. Dawson, W. E. Dixon, and B. Xian, "Extremum seeking nonlinear controllers for a human exercise machine," in *Proceedings of the 43th Conference on Decision and Control*, pp. 3950–3955, 2004.
- [92] N. I. Marcos, M. Guay, D. Dochain, and T. Zhang, "Adaptive extremum-seeking control of a continuous stirred tank bioreactor with haldane's kinetics," *Journal of Process Control*, vol. 14, no. 3, pp. 317–328, 2004.
- [93] M. Guay, D. Dochain, and M. Perrier, "Adaptive extremum seeking control of continuous stirred tank bioreactors with unknown growth kinetics," *Automatica*, vol. 40, no. 5, pp. 881–888, 2004.
- [94] N. I. Marcos, M. Guay, and D. Dochain, "Output feedback adaptive extremum seeking control of a continuous stirred tank bioreactor with monod's kinetics," *Journal of Process Control*, vol. 14, no. 11, pp. 807–818, 2004.
- [95] N. Hudon, M. Perrier, M. Guay, and D. Dochain, "Adaptive extremum seeking control of a non-isothermal tubular reactor with unknown kinetics," *Computer and Chemical Engineering*, vol. 29, pp. 839–849, 2005.
- [96] D. F. Chichka, J. L. Speyer, and C. G. Park, "Peaking-seeking control with application to formation flight," in *Proceedings of the 38th Conference on Decision and Control*, vol. 5, pp. 2463–2470, 1999.
- [97] F. Xie, X. Zhang, R. Fierro, and M. Motter, "Autopilot-based nonlinear uav formation controller with extremum-seeking," in *Proceedings of the Conference on Decision and Control*, pp. 4933–4938, 2005.
- [98] H.-H. Wang and M. Krstić, "Extremum seeking for limit cycle minimization," *IEEE Transactions on Automatic Control*, vol. 45, no. 12, pp. 2432–2436, 2000.

- [99] H.-H. Wang, S. Yeung, and M. Krstić, "Experimental application of extremum seeking on an axial-flow compressor," *IEEE Transactions on Control Systems Technology*, vol. 8, no. 2, pp. 300–309, 2000.
- [100] G. Schneider, K. B. Ariyur, and M. Krstić, "Tuning of a combustion controller by extremum seeking: a simulation study," in *Proceedings of the 39th Conference on Decision and Control*, vol. 5, pp. 5219–5223, 2000.
- [101] A. Banaszuk, Y. Zhang, and C. A. Jacobson, "Adaptive control of combustion instability using extremum-seeking," in *Proceedings of the American Control Conference*, vol. 1, pp. 416–422, 2000.
- [102] A. Banaszuk, K. B. Ariyur, M. Krstić, and C. A. Jacobson, "An adaptive algorithm for control of combustion instability and application to compressor instability control," *Automatica*, vol. 40, no. 11, pp. 1965–1972, 2004.
- [103] S. K. Nguang and X. D. Cheng, "Extremum seeking scheme for continuous fermentation processes described by an unstructured fermentation model," *Bio-processing Engineering*, vol. 23, no. 5, pp. 417–420, 2000.
- [104] M. Gafvert, K.-E. Arzen, and L. M. Pederson, "Simple linear feedback and extremum control of gdi engines," in *Seoul 2000 FISITA World Automotive Congress*, 2000.
- [105] D. Popović, M. Janković, S. Magner, and A. Teel, "Extremum seeking methods for optimization of variable cam timing engine operation," in *Proceedings of the American Control Conference*, vol. 4, pp. 3136–3141, 2003.
- [106] M. Janković and S. Magner, "Optimization and scheduling for automotive powertrains," in *Proceedings of the American Control Conference*, pp. 4054–4059, 2004.
- [107] M. Komatsu, H. Miyamoto, H. Ohmori, and A. Sano, "Output maximization control of wind turbine based on extremum control strategy," in *Proceedings of the American Control Conference*, vol. 2, pp. 1739–1740, 2001.
- [108] A. Banaszuk, S. Narayanan, and Y. Zhang, "Adaptive control of flow separation in a planar diffuser," in *41st Aerospace Sciences Meeting & Exhibit*, 2003.
- [109] N. Hudon, M. Guay, M. Perrier, and D. Dochain, "Adaptive extremum seeking control of tubular reactor with limited actuation," in *Proceedings of the American Control Conference*, pp. 4563–4568, 2005.
- [110] K. S. Peterson and A. G. Stefanopoulou, "Extremum seeking control for soft landing of an electromechanical valve actuator," *Automatica*, vol. 40, no. 6, pp. 1063–1069, 2004.

- [111] Y. Li, M. A. Rotea, G. T.-C. Chiu, L. G. Mongeau, and I.-S. Paek, "Extremum seeking control of tunable thermoacoustic cooler," in *Proceedings of the American Control Conference*, pp. 2033–2038, 2004.
- [112] T. A. Johansen and D. Sbárbaro, "Lyapunov-based optimizing control of non-linear blending processes," *IEEE Transactions on Control Systems Technology*, vol. 13, no. 4, pp. 631–638, 2005.
- [113] C. Centioli, F. Iannone, G. Mazza, M. Panella, L. Pangione, S. Podda, A. Tuccillo, V. Vitale, and L. Zaccarian, "Extremum seeking applied to the plasma control system of the Frascati tokamak upgrade," in *Proceedings of the Conference on Decision and Control*, pp. 8227–8232, 2005.
- [114] R. Leyva, C. Alonso, I. Queinnec, A. Cid-pastor, D. Lagrange, and L. Martínez-salamero, "Mppt of photovoltaic systems using extremum-seeking control," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 42, no. 1, pp. 249–258, 2005.
- [115] C. Zhang, D. Arnold, N. Ghods, A. Siranosian, and M. Krstić, "Source seeking with nonholonomic unicycle without position measurement – part i: Tuning of forward velocity," in *submitted to the Conference Decision and Control*, 2006.
- [116] J. Nocedal and S. Wright, *Numerical Optimization*. Berlin, Heidelberg, New York: Springer-Verlag, 1999.
- [117] J. C. Spall, *Introduction to Stochastic Search and Optimization*. New Jersey: Wiley-Interscience, 2003.
- [118] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *IEEE Transactions on Evolutionary Computation*, vol. 1, pp. 67–82, 1997.
- [119] P. M. Pardalos and M. G. C. Resende, *Handbook of Applied Optimization*. New York: Oxford University Press, 2002.
- [120] T. G. Kolda, R. M. Lewis, and V. Torczon, "Optimization by direct search: new perspectives on some classical and modern methods," *SIAM Review*, vol. 45, no. 3, pp. 385–482, 2003.
- [121] R. Fletcher, *Practical Methods of Optimization*. New York, NY: John Wiley and Sons, 1987.
- [122] A. R. Conn, K. Scheinberg, and P. L. Toint, "Recent progress in unconstrained nonlinear optimization without derivatives," *Math. Programming*, vol. 79, pp. 397–414, 1997.

- [123] A. R. Conn, K. Scheinberg, and P. L. Toint, "On the convergence of derivative-free methods for unconstrained optimization," *Approximation Theory and Optimization: Tributes to M. J. D. Powell*, pp. 83–108, 1997.
- [124] M. J. D. Powell, "Uobyqa: Unconstrained optimization by quadratic approximation," Tech. Rep. NA2000/14, DAMTP, University of Cambridge, 2000.
- [125] J. F. Blowey, A. W. Craig, and T. Shardlow, *Frontiers in Numerical Analysis*. Berlin, Heidelberg, New York: Springer-Verlag, 2002.
- [126] D. P. Bertsekas, *Nonlinear Programming*. Belmont, Massachusetts: Athena Scientific, 1995.
- [127] W. M. Hirsch and S. Smale, *Differential Equations, Dynamical Systems and Linear Algebra*. New York, NY: Academic Press, 1974.
- [128] M. W. Hirsch, S. Smale, and R. L. Devaney, *Differential Equations, Dynamical Systems and an Introduction to Chaos*. New York, NY: Academic Press, 2004.
- [129] J. Cortés, "Achieving coordination tasks in finite time via nonsmooth gradient flows," in *Proceedings of Conference on Decision and Control*, 2005.
- [130] V. I. Utkin, *Sliding Modes in Control and Optimization*. Berlin, Heidelberg, New York: Springer-Verlag, 1994.
- [131] H. K. Khalil, *Nonlinear Systems*. Upper Saddle River, NJ: Prentice Hall, 2001.
- [132] C.-T. Chen, *Linear System Theory and Design*. New York Oxford: Oxford University Press, 1999.
- [133] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.
- [134] R. Freund, "Lecture notes of 15.084j / 6.252j nonlinear programming, spring 2004," 2004. MIT OpenCourseWare.
- [135] K. M. Passino, *Biomimicry for Optimization, Control, and Automation*. Berlin, Heidelberg, New York: Springer-Verlag, 2005.
- [136] D. P. Bertsekas and J. N. Tsitsiklis, "Gradient convergence in gradient methods with errors," *SIAM Journal of Optimization*, vol. 10, no. 3, pp. 627–642, 2000.
- [137] K. L. Moore, "Iterative learning control: An expository overview," *Applied and Computational Controls, Signal Processing, and Circuits*, vol. 1, no. 1, pp. 425–488, 1998.

- [138] B. Porat and A. Neohorai, "Localizing vapor-emitting sources by moving sensors," *IEEE Transactions on Signal Processing*, vol. 44, pp. 1018–1021, 1996.
- [139] P. Ogren, E. Fiorelli, and N. E. Leonard, "Cooperative control of mobile sensor networks: adaptive gradient climbing in a distributed environment," *IEEE Transactions on Automatic Control*, vol. 29, pp. 1292–1302, 2004.
- [140] E. W. Justh and P. S. Krishnaprasad, "Equilibria and steering laws for planar formations," *Systems and Control Letters*, vol. 52, pp. 25–38, 2004.
- [141] D. J. Klein and K. A. Morgansen, "Controlled collective motion for trajectory tracking," in *Proceedings of the American Control Conference*, 2006.
- [142] J. A. Marshall, M. E. Broucke, and B. A. Francis, "Pursuit formations of unicycles," *Automatica*, vol. 42, pp. 3–12, 2006.
- [143] R. M. A. Lewis, J. Ostrowski and J. Burdick, "Nonholonomic mechanics and locomotion: The snakeboard example," in *International Conference on Robotics and Automation*, 1994.
- [144] F. Bullo and A. D. Lewis, "Kinematic controllability and motion planning for the snakeboard," *IEEE Transactions on Robotics and Automation*, vol. 19, pp. 494–498, 2003.
- [145] T. Balch and R. C. Arkin, "Behavior-based formation control for multirobot teams," *IEEE Transactions on Robotics and Automation*, vol. 14, pp. 926–939, December 1998.
- [146] R. Olfati-Saber and R. M. Murray, "Distributed cooperative control of multiple vehicle formations using structural potential functions," in *Proceedings of the IFAC World Congress*, (Barcelona, Spain), June 2002.
- [147] D. H. Kim, H. O. Wang, G. Ye, and S. Shin, "Decentralized control of autonomous swarm systems using artificial potential functions: Analytical design guidelines," in *Proceedings of the IEEE Conference on Decision and Control*, December 2004.
- [148] R. L. Raffard, C. J. Tomlin, and S. P. Boyd, "Distributed optimization for cooperative agents: Application to formation flight," in *Proceedings of the IEEE Conference on Decision and Control*, December 2004.
- [149] H. Yamaguchi, "A cooperative hunting behavior by mobile-robot troops," *The International Journal of Robotics Research*, vol. 18, pp. 931–940, September 1999.

- [150] J. P. Desai, J. Ostrowski, and V. Kumar, "Controlling formations of multiple mobile robots," in *Proc. 1998 IEEE Int. Conf. Robotics and Automation*, (Leuven, Belgium), pp. 2864–2869, May 1998.
- [151] P. Ögren, E. Fiorelli, and N. E. Leonard, "Cooperative control of mobile sensor networks: Adaptive gradient climbing in a distributed environment," *IEEE Transaction on Automatic Control*, vol. 49, pp. 1292–1302, August 2004.
- [152] J. R. Lawton, B. J. Young, and R. W. Beard, "A decentralized approach to elementary formation maneuvers," in *Proceedings of the IEEE International Conf. on Robotics and Automation*, (San Francisco, CA), pp. 2728–2743, April 2000.
- [153] J. H. Reif and H. Wang, "Social potential fields: A distributed behavioral control for autonomous robots," *Robotics and Autonomous Systems*, vol. 27, pp. 171–194, May 1999.
- [154] E. Rimon and D. E. Koditschek, "Exact robot navigation using artificial potential functions," *IEEE Transaction on Robotics and Automation*, vol. 8, pp. 501–518, October 1992.
- [155] N. E. Leonard and E. Fiorelli, "Virtual leaders, artificial potentials and coordinated control of groups," in *Proceedings of the IEEE Conf. Decision and Control*, (Orlando, FL), pp. 2968–2973, December 2001.
- [156] J. Yao, R. Ordóñez, and V. Gazi, "Swarm tracking using artificial potentials and sliding mode control," in *Submitted to the Conference Decision and Control*, 2006.
- [157] V. Gazi and R. Ordóñez, "Target tracking using artificial potentials and sliding mode control," in *Proceedings of the American Control Conference*, vol. 6, pp. 5588–5593, 2004.
- [158] C. Zhang and R. Ordóñez, "Decentralized adaptive coordination and control of uninhabited autonomous vehicles via surrogate optimization," in *Proc. American Control Conf.*, (Denver, CO), June 2003.
- [159] C. Zhang, R. Ordóñez, and C. Schumacher, "Multi-vehicle cooperative search with uncertain prior information," in *Proc. AIAA Guidance, Navigation and Control Conference*, (Providence, Rhode Island), August 2004.
- [160] A. P. Aguiar, L. Cremean, and J. P. Hespanha, "Position tracking for a nonlinear underactuated hovercraft: Controller design and experimental results," in *Proceedings of the IEEE Conference on Decision and Control*, 2003.

VITA

Education

- B.S. in Automatic Control September 1995 – July 1999
Nanjing University of Science and Technology Nanjing, China
- M.S. in Electrical Engineering August 2001 – August 2003
University of Dayton Dayton, OH
- M.S. in Applied Mathematics August 2003 – May 2005
University of Dayton Dayton, OH
- Ph.D in Electrical Engineering August 2003 – May 2006
University of Dayton Dayton, OH

Journal Publications

- Chunlei Zhang, Antranik Siranosian and Miroslav Krstić, "Extremum Seeking for Moderately Unstable Systems and for Autonomous Target Tracking Without Position Measurements," submitted to Automatica.
- Chunlei Zhang, Raúl Ordóñez and Corey Schumacher, "Coordination and Control of Groups of Mobile Autonomous Agents Using Surrogate Optimization," submitted to Journal of Intelligent and Robotic Systems.
- Chunlei Zhang and Raúl Ordóñez, "Numerical Optimization-based Extremum Seeking Control with Application to ABS Design," IEEE Transaction on Automatic Control, accepted for publication.
- Chunlei Zhang, Qin Sheng and Raúl Ordóñez, "Notes on the Convergence and Applications of Surrogate Optimization," Discrete and Continuous Dynamic Systems, supplement volume 2005.

Peer Reviewed Conference Publications

- Chunlei Zhang and Raúl Ordóñez, "Extremum Seeking Control based on Numerical Optimization and State Regulation – Part I: Theory and Framework," submitted to Conference on Decision and Control 2006.

R702032640

The HF Group

Indiana Plant

T 052546 F 7 00



6/8/2006

- Chunlei Zhang and Raúl Ordóñez, "Extremum Seeking Control based on Numerical Optimization and State Regulation – Part II: Robust and Adaptive Control Design," submitted to Conference on Decision and Control 2006.
- Chunlei Zhang and Raúl Ordóñez, "Non-gradient Extremum Seeking Control of Feedback Linearizable Systems with Application to ABS Design," submitted to Conference on Decision and Control 2006.
- Chunlei Zhang, Daniel Arnold, Nima Ghods, Antranik Siranosian and Miroslav Krstić, "Source Seeking With Nonholonomic Unicycle Without Position Measurement – Part I: Tuning of Forward Velocity," submitted to Conference on Decision and Control 2006.
- Chunlei Zhang, Antranik Siranosian and Miroslav Krstić, "Extremum Seeking for Moderately Unstable Systems and for Autonomous Target Tracking Without Position Measurements," in Proc. American Control Conference, June 2006.
- Chunlei Zhang and Raúl Ordóñez, "Numerical Optimization-based Extremum Seeking Control of LTI Systems," in Proc. Conference of Decision and Control, December 2005.
- Shreecharan Kanchanavally, Chunlei Zhang, Raúl Ordóñez and Jeff Layne, "Mobile Target Tracking with Communication Delays," in Proc. Conference of Decision and Control, December 2004.
- Chunlei Zhang, Raúl Ordóñez and Corey Schumacher, "Multi-vehicle Cooperative Search with Uncertain Prior Information," in Proc. AIAA Guidance, Navigation and Control Conference, August 2004.
- Chunlei Zhang and Raúl Ordóñez, "Decentralized Adaptive Coordination and Control of Uninhabited Autonomous Vehicles via Surrogate Optimization," in Proc. American Control Conference, June 2003.