ESTIMATION-BASED SOLUTIONS TO INCOMPLETE INFORMATION PURSUIT-EVASION GAMES

A Dissertation

by

TIMOTHY DANIEL WOODBURY

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Chair of Committee,	John E. Hurtado
Committee Members,	Shankar P. Bhattacharyya
	John L. Junkins
	Srinivas Rao Vadali
Head of Department,	Rodney Bowersox

May 2019

Major Subject: Aerospace Engineering

ABSTRACT

Differential games are a useful tool both for modeling conflict between autonomous systems and for synthesizing robust control solutions. The traditional study of games has assumed decision agents possess complete information about one another's strategies and numerical weights. This dissertation relaxes this assumption. Instead, uncertainty in the opponent's strategy is treated as a symptom of the inevitable gap between modeling assumptions and applications. By combining nonlinear estimation approaches with problem domain knowledge, procedures are developed for acting under uncertainty using established methods that are suitable for applications on embedded systems.

The dissertation begins by using nonlinear estimation to account for parametric uncertainty in an opponent's strategy. A solution is proposed for engagements in which both players use this approach simultaneously. This method is demonstrated on a numerical example of an orbital pursuit-evasion game, and the findings motivate additional developments. First, the solutions of the governing Riccati differential equations are approximated, using automatic differentiation to obtain high-degree Taylor series approximations. Second, constrained estimation is introduced to prevent estimator failures in near-singular engagements. Numerical conditions for nonsingularity are approximated using Chebyshev polynomial basis functions, and applied as constraints to a state estimate. Third and finally, multiple model estimation is suggested as a practical solution for time-critical engagements in which the form of the opponent's strategy is uncertain. Deceptive opponent strategies are identified as a candidate approach to use against an adaptive player, and a procedure for designing such strategies is proposed. The new developments are demonstrated in a missile interception pursuit-evasion game in which the evader selects from a set of candidate strategies with unknown weights.

DEDICATION

To Mom and Dad, with love and eternal thanks.

ACKNOWLEDGMENTS

I want to first thank all the people in my life who have been there for me over the years. There are too many to count and too many to list here. Special thanks and love go out to Mom, Dad, Patrick, and Tanja. I would not have made it through the first half of the PhD without the friendship of Steven, Austin, and Jim. While I regret they will not be present for my defense, I am pleased each of them has found a productive life outside of College Station. To all of my other friends and acquaintances in Aerospace Engineering, Hobofo, and the Bryan/College Station community, my deepest thanks and appreciation.

I would like to thank my advisor, Dr. Hurtado, for his patience during the (lengthy) process of this PhD. It has been a challenging experience, at times more and less so than I would have assumed. Dr. Hurtado has kept the lights on by finding funded research, allowing me to remain focused on conducting research. He has been available throughout the dissertation-writing process and has provided feedback on many occasions that has steered my away from unproductive areas of study. He is also an entertaining lecturer, and possesses an encyclopedic knowledge of analytical dynamics; I have enjoyed both courses I took from him.

I thank the members of the PhD advisory committee for their time and assistance. Dr. Srinivas Rao Vadali has been my instructor in no less than four courses in optimal control, spacecraft attitude control, and celestial mechanics. I have found him to be an intelligent, challenging, and fair instructor. One of the few regrets of my graduate study is that I have been unable to take a course from Dr. Shankar Bhattacharyya or Dr. John L. Junkins. I respect them both greatly, and shall have to make do with reading their books instead.

A few other individuals offered helpful suggestions or advice at key intervals in my study. My early correspondence with Dr. Kurt Cavalieri of Sandia National Laboratories illuminated some of the key concepts and pitfalls in his dissertation research, which formed the starting point for the material of Chapter 2. Dr. Manoranjan Majji of Texas A&M University provided feedback on the idea of numerically approximating a neighborhood of solutions to the governing Riccati equations.

This eventually led me to the truncated Taylor Series expansion presented in Chapter 4. Drs. Hans Stigter and Jaap Molenaar of Wageningen University & Research offered helpful correspondence on the subject of their method for nonlinear observability analysis. This method is employed to validate the observability of the primary examples in the dissertation in Appendix A.

CONTRIBUTORS AND FUNDING SOURCES

Contributors

This work was supported by a dissertation committee consisting of Professor John E. Hurtado [advisor], Professor John L. Junkins, and Professor Srinivas Rao Vadali of the Department of Aerospace Engineering and Professor Shankar P. Bhattacharyya of the Department of Electrical Engineering.

All work was completed by the student under the advisorship of Professor Hurtado.

Funding Sources

Graduate study during the PhD was supported from August 2014 through July 2016 by the National Science Foundation Graduate Research Fellowship Program. Sandia National Laboratories provided funding that supported study from August 2016 through May 2019.

TABLE OF CONTENTS

Page

LIST OF FIGURES

FIGURE Page

LIST OF TABLES

TABLE Page

1. INTRODUCTION

1.1 Motivation

This dissertation presents a study of adaptive solutions to differential games in the presence of uncertainty. In presenting this research, it is useful to begin by clearly defining some notation. A *game*, in the mathematical sense, is a general problem in which two or more decision makers, called agents or players, make decisions that are not independent. Usually, the agents attempt to either optimize some objective function or to satisfy some desired boundary conditions. This is a very general description that can apply to many problems that have little relevance to this dissertation. *Dynamic games* refer to a class of problems in which there is some sequential dependence in the behaviors available to agents. This could be as simple as a tree-like branching structure in a discrete-valued game, as in Fig. 1.1. The figure shows a discrete-valued dynamic game in the *extensive form*[2]. In the diagram, Player A chooses either the left or right branch, then Player B makes a choice. Each player receives a reward (R) that is dependent on both players' choices. Dynamic games are contrasted with *static games*. *Differential games* are a subset of dynamic games in which the decision agents are governed by differential or difference equations[5]. In this dissertation, the term "differential games" is preferred because it emphasizes the distinction between aerospace and robotic systems, as compared with other game systems, e.g. economic games.

Pursuit-evasion scenarios refer to any environment in which two or more agents exist and at least one agent has the objective of reducing or increasing the distance between the agents. This very broad definition has been selected because many works in the literature consider either pure pursuit or pure evasion scenarios, in which the behavior of one "team" is considered frozen, and the response of the other player(s) is designed. Pursuit-evasion (PE) can be expressed as a differential game with an objective function relating to the relative range between agents. However, PE scenarios are also extensively studied in the areas of missile defense and aircraft interception, often with little consideration to the classic differential games framework.

The focus of the dissertation is on addressing uncertainty using differential games. Terminology to describe specific types of ignorance that may arise is now defined. Isaacs [5] originally referred to any uncertainty as “incomplete information,” whether it originates from measurement errors, uncertainty in the relative dynamics, or other sources, and many works follow this convention. A more descriptive taxonomy is adopted by Cavalieri [6]. In this work, information in a game is categorized in terms of completeness, perfectness, and certainty. Information is complete if the opponent’s objective function and the opponent inputs (controls) are known. Perfect information implies is no measurement error present. Certain information indicates that the governing differential equations are known without errors. From its inception, the study of differential games has tended to focus on complete, perfect, certain information games. The justifications for this are several. To name a few: such scenarios are more analytically tractable; they avail themselves to the tools of optimal control; and certainty equivalence can be used in many scenarios to neglect some sources of uncertainty.

The terms “zero-sum” (ZS) and “nonzero-sum” (NZS) often arise in the context of differential games. In the ZS game, the sum of all players’ objective functions is always zero. In the two-player ZS game, this implies simply $J_1 = -J_2$. Much attention has been given to ZS games because they model many situations accurately and are often much simpler, in that solution equilibria and proofs

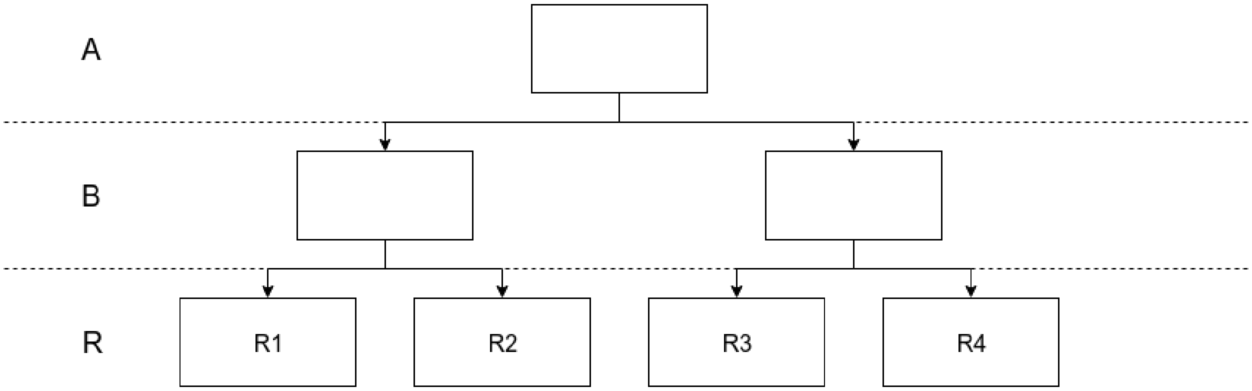

Figure 1.1: Simple discrete-valued dynamic game representation in extensive form

of optimality are more straightforward to show. However, ZS games are not universally applicable, and NZS games are an approach to modelling the uncertainty that occurs in practical problems. Consequently, this dissertation focuses primarily on NZS games.

For the final terminology, different possible game equilibria are defined. While the study of dynamic game equilibria could be the subject of an entire work on its own, consideration here is restricted to the two most common types: Nash equilibria and minimax equilibria. These concepts are explained fully later on, but it is useful to describe them now at a conceptual level. A Nash equilibrium is optimal in the sense that each player obtains a local extremal value, and any deviation by a single player causes that player to suffer a penalty. Effectively it is the noncooperative equilibrium achieved when all players simultaneously play to optimize their objective metrics[7]. In two-player games the Nash equilibrium is often referred to as a saddle-point solution. The term "equilibrium" refers to the fact that no player has any mathematical incentive to deviate from a Nash strategy, and should not be confused with equilibrium concepts in dynamic systems. For NZS games the Nash equilibrium may not exist; much effort has been dedicated to proving existence and uniqueness of Nash equilibria for various games. A minimax equilibrium is found by one player choosing the control strategy that minimizes his maximum objective value[7]. Essentially, this is the approach that ensures the "least bad" worst-case outcome, with respect to a particular objective function. In some specialized cases the minimax and Nash strategies may be equivalent. The minimax equilibrium is generally the less restrictive case and may exist when the Nash equilibrium does not.

Having introduced the basic concepts of differential games, the motivation for their study here can be clearly elucidated. Differential games, as studied in this dissertation, have implications for two broad problems in engineering. The first is conflict modeling. Many military and police engagements involve multiple decision agents. Differential games offer an approach for autonomous vehicles to model and make predictions about the world. This framework is particularly applicable to structured scenarios like missile interception problems, in which opponent objectives can be at least partially anticipated and modeled mathematically a priori. The game framework has implications for civilian systems as well. E.g., consider autonomous cars, in which a vehicle may want to devise a trajectory in the presence of pedestrians with uncertain intentions. A game-theoric solution would attempt to design a trajectory to ensure a maximum distance between the vehicle and any pedestrian for the worst-case behavior of the pedestrian. The second problem of interest is robust control. Games have been used in the past to design control laws that are robust to assumed "worst-case" perturbations by modeling perturbations as actively antagonistic opposing agents. The areas of conflict modeling and robust control are the primary application domains of the work in this dissertation.

The basic subject and applications of this dissertation have been introduced. The remainder of this section continues the introduction to the important topics of the dissertation. The following subsections review foundational works in differential games, introduce the game studied in most of the dissertation, and present the primary objectives of this study. Subsequent sections introduce the mathematical details of linear quadratic differential games, and outline the contents of the dissertation. Now, some significant works in game theory and differential games in particular are reviewed.

1.1.1 Foundational works

Given the very broad definition of game theory just discussed, it should be no surprise that the associated literature is somewhat fractious and uses inconsistent nomenclature. Unifying the literature is beyond the scope of this dissertation, but the author shall endeavor to be self-consistent while providing a reasonable overview of relevant previous works.

It is generally agreed[5, 2] that the formalization of game theory should be attributed to Von Neumann and Morgenstern[8], who defined game theory in an effort to systematically study the actions of economic decision-makers. While game theory has figured prominently in economic literature, such researches are fairly far removed from the focus of the present work, and Von Neumann is rarely cited in engineering circles. On the other hand, the work of Isaacs[5] is almost universally cited in engineering applications of game theory. Isaacs formally defined the idea of *differential games*, and his ideas influenced a great deal of further study. Isaacs was strongly motivated by applications in war games, and the significance of his simple examples is easily grasped by both academics and career military personnel. This book has been particularly influential among the aerospace engineering discipline for its extensions to missile guidance and control. Isaacs' work was contemporaneous with many developments of optimal control theory, and initial applications of Isaacs' ideas were closely tied to then-new ideas in optimal control.

Isaacs made the useful distinction between games of degree and games of kind. In the game of degree, the objective function has some intrinsic significance, like the terminal distance between agents. This approach naturally leads to an approach based on variational optimal control. The game of kind, by contrast, is typically defined by some sets of terminal conditions, each of which correspond to game outcomes such as "winning" and "losing." Much of Isaacs' work focused on generating barrier surfaces that divide a state space into regions that lead to various terminal conditions under optimal play. Isaacs did not consider incomplete information scenarios except in a speculative final chapter.

The work of Başar and Olsder[2] should be mentioned for its widespread influence. These authors attempted to unify the increasingly fractious nature of the game theory literature. They suggested the definitions of *static* and *dynamic* games already described, and also make an intuitive distinction between *cooperative* and *non-cooperative* games. Their work appears commonly cited in the mathematical and computer science disciplines, but is not as frequently cited in the aerospace literature. The book represents a particular line of thought about games, in which the focus is on determining existence, uniqueness, and types of equilibria that exist for a given game. This mindset, while mathematically quite engaging, makes their work challenging to extend to scenarios in which players do not know the opponent objectives exactly.

Next, attention is given to the particular class of differential games that is the primary focus of this work: *linear quadratic differential games* (LQDGs).

1.1.2 Introduction to linear quadratic differential games

LQDGs are a class of games with linear dynamics in which each player attempts to optimize the value of a quadratic function of the state and all players' controls. For a general N-player game, each player's objective function may be written as in Eq. 1.1. The players are subject to governing linear dynamics. The governing dynamics and associated state are the same for each agent, which provides a coupling between each player's outcomes that makes the game meaningful. The objective functions and linear dynamics may be expressed as follows[7]:

$$J_i = \frac{1}{2}\boldsymbol{x}(T_f)^T[S_{f,i}]\boldsymbol{x}(T_f) + \frac{1}{2}\int_{t_0}^{T_f}\left(\boldsymbol{x}^T[Q_i]\boldsymbol{x} + \sum_{j=1}^{N}\boldsymbol{u}_j^T[R_{ij}]\boldsymbol{u}_j\right)\,\mathrm{d}t,\ \ i \in \{1,2,\ldots,N\} \tag{1.1}$$

$$\text{s.t. } \dot{\boldsymbol{x}} = [A]\boldsymbol{x} + \sum_{j=1}^{N}[B_j]\boldsymbol{u}_j$$

It is generally assumed that the weight matrices $[S_{f,i}]$, $[Q_i]$, and $[R_{ij}]$ are symmetric for all i and j, and all problems studied in the dissertation follow this convention. Each player's goal is obtain a "good" value of J_i, subject to the actions of the other players. It is well-known from the optimal control literature that the objective function is nonunique, meaning the weight matrices can be scaled arbitrarily so long as their relative proportions are the same[6]. For the specific case of $N = 2$, the game objective for each player has the form of Eq. 1.2. Note that **the repeated indices i and j do not imply summation**, and this is true for equations in the remainder of the dissertation.

$$J_i = \frac{1}{2}\boldsymbol{x}(T_f)^T[S_{f,i}]\boldsymbol{x}(T_f) + \frac{1}{2}\int_{t_0}^{T_f}\left(\boldsymbol{x}^T[Q_i]\boldsymbol{x} + \boldsymbol{u}_i^T[R_{ii}]\boldsymbol{u}_i + \boldsymbol{u}_j^T[R_{ij}]\boldsymbol{u}_j\right)\,\mathrm{d}t \tag{1.2}$$

The problem state $\boldsymbol{x}$ is normally expressed in terms of relative coordinates between the agents. However, it may be useful to express the state vector in terms of each individual agent's state. In a non-cooperative game, it is important that the problem satisfy "G-reducibility." Essentially, this mandates that no player be able to improve his objective value arbitrarily by manipulating states that are not controllable by his opponent[9]. **In this dissertation, all problems are expressed in terms of relative coordinates, and each agent's control channel satisfies controllability**, in the absence of the opponent control.

Note that Eq. 1.2 has been written in terms of linear time-invariant dynamics. This assumption is maintained throughout the dissertation for convenience. The work presented in later chapters can be extended directly to time-varying systems. LQDGs are of particular interest because their most common solutions can be expressed as linear, time-varying state feedback controllers. This result is derived later in this chapter. In such cases, the optimal state feedback gain is computed as the solution of an ordinary differential equation (ODE). By contrast, solutions to general games require numerical solutions of either initial value problems or boundary value problems in multiple variables. Well-established methods exist for solving such problems numerically, but require substantially greater time and computational power to compute. This dissertation focuses on LQDGs because their solutions can be computed straightforwardly, allowed for rapid exploration of new problems and concepts. At the same time, LQDGs have a history of use in applied systems, and thus are known to possess sufficient complexity to model real-world problems. It is believed that many of the methods used in this dissertation to achieve adaptive strategies in LQDGs can be applied, with minimal modification, to more general differential games. However, such a study is beyond the scope of the present work.

Having introduced its general areas of study, the primary objectives of this dissertation are now discussed in detail.

1.1.3 Objectives of the dissertation

The primary goal of the dissertation is to study decision-making under uncertainty through the lens of LQDGs. NSZ games reflect the inherent mismatch between an agent's assumptions and its reality, and this interpretation has applications to both real-world conflicts and to robust control. While differential games under various kinds of uncertainty have been widely studied, solutions have not tended to focus on real-time systems with significant computational constraints, which commonly arise in engineering problems. Consequently, the focus of this research is on developing solutions that avoid numerical iteration and scale reasonably well with increasing problem dimension. This dissertation seeks to impart the following capabilities to a player in a LQDG:

- Recognize and adapt to an opponent that plays a known strategy *more favorably* than expected

- Recognize an opponent that plays a known strategy *less favorably* than expected
- Adapt to *unknown opponent strategies*, leveraging domain knowledge to make the problem tractable.

The bulk of this dissertation is devoted to implementing methods to achieve these objectives, or to address underlying numerical challenges and reduce computational complexity for real-time performance.

This section has introduced the objectives of the dissertation from a high-level perspective and defined some essential terms. The next section introduces the details of LQDGs and begins the primary technical content.

1.2 Linear quadratic differential games

LQDGs are a central focus of this dissertation, and this section is devoted to an overview of this topic. First, the solutions to LQDGs considered in this dissertation are presented at a conceptual level. Second, relevant literature in the area is reviewed. Third, the solutions to LQDGs considered are presented at the mathematical level. Fourth and finally, the theoretical costs incurred by each player in LQDGs is reviewed.

1.2.1 Solutions of LQDGs

In optimal control theory, equations of a similar form to Eq. 1.2 are solved using either the necessary conditions of optimality or the solution to the Hamilton-Jacobi-Bellman partial differential equation[1]. While obtaining the optimal solution can be challenging, there is no ambiguity about whether a candidate solution is optimal - if it satisfies the second-order sufficiency conditions for optimality, it is optimal[10]. For differential games it is no longer possible to simply find the control history for $\boldsymbol{u}_i$ that optimizes J_i; some additional characteristics of the solution must be stipulated[7][2, p.11]. For instance, an arbitrary solution may be optimal with respect to one

[1] In principle, of course, the two solutions are identical, but the solution methods differ substantially in practice.

opponent control history but suboptimal with respect to another history. To compute a game solution, each player must make some assumption about how the other players will act over the time horizon of interest. The earlier discussion of equilibrium concepts in games alludes to this fact.

Consideration is now given to the solution of LQDGs. In this dissertation, three solution types are considered. In order of approximate complexity, they are minimax, one-sided optimal (OSO), and feedback Nash equilibrium (FNE) strategies. Each strategy is now reviewed.

Minimax strategies attempt to generate a solution with a guaranteed upper bound. The name arises from the fact that player i attempts to minimize J_i while assuming player j simultaneously attempts to maximize J_i. The framework is equivalent to assuming the opponent plays in a zero-sum fashion. The minimax control for player i can be compactly expressed in terms of her objective function J_i:

$$\boldsymbol{u}_i^{(\mathrm{minimax})} = \mathrm{argmin}_{\boldsymbol{u}_i} \max_{\boldsymbol{u}_j} \{J_i(t, \boldsymbol{x}, \boldsymbol{u}_i, \boldsymbol{u}_j)\} \tag{1.3}$$

The minimax strategy is simple to compute because its solution depends only on the particular cost function of player i, which is known to i. However, the solution is conservative; if player j uses any strategy other than maximizing Eq. 1.2, then player i could obtain better performance with a different strategy. The minimax solution is often used to generate robust control by assuming that $\boldsymbol{u}_j$ models a worst-case disturbance to a dynamic system. Naturally there is a connection between H_∞ and minimax control[11]. In Ref. [12], minimax strategies are referred to as “minimax-safe” strategies, and players use them as a “default” behavior while learning an opponent’s game strategy.

One-sided optimal strategies are simply optimal with respect to a particular opponent control solution $\boldsymbol{u}_j$, which may be given as an open-loop or feedback function. An OSO strategy is often a poor solution to a differential game because the opponent may deviate from the assumed strategy. In this dissertation, the OSO strategy is used when the opponent control strategy is known approximately via estimation. For the case of a linear, state-independent opponent feedback, the OSO strategy is simply a particular form of the classic linear quadratic regulator (LQR) solution. In this dissertation, “OSO strategy” refers specifically to the case in which the opponent control strategy is given from a minimax solution. Other solutions that satisfy optimality with respect to particular strategies are possible, such as the LQDG-gain strategy discussed in Chapter 6. It should be noted that an OSO solution is technically an optimal control solution, and not properly a differential game equilibrium.

Feedback Nash equilibria (FNE) in LQDGs are a particular type of Nash equilibrium solution. General Nash equilibria describe scenarios in which no player gains by unilaterally deviating from her equilibrium strategy. In LQDGs, Nash equilibria are commonly characterized as either "open-loop" or feedback Nash equilibria. The terminology is somewhat unfortunate, since both solutions generally lead to a feedback form of all controls. In the open-loop case, the strategy for $\boldsymbol{u}_i$ is derived while modelling $\boldsymbol{u}_j$ as a function of time, not a function of the state. In the FNE case, player i models $\boldsymbol{u}_j$ as a feedback function of the state. The open-loop case is appropriate in economic games or other environments in which players must commit to a sequence of actions before observing the outcome, but is not suitable for interactions between autonomous, physically situated agents. That is why this dissertation considers only the FNE solution.

This section has introduced the basic LQDG structure and the key solution concepts that are used in the subsequent developments. Having presented the basic LQDG form and offered some comparison to the well-known field of optimal control, it is worth reviewing the background literature in the area of LQDGs.

1.2.2 Literature review

LQDGs describe scenarios in which each player's cost function has a quadratic running cost in the state and controls with a possible quadratic terminal state penalty, and the problem dynamics are linear in the state and control. They are directly analogous to the linear quadratic regulator (LQR) in optimal control theory, and, like the LQR, they have proved analytically tractable and in many cases provide feedback solutions[7]. This subsection focuses on relevant literature regarding the existence of finite-valued equilibrium solutions to particular games. For this analysis, perfect, complete, and certain information is commonly assumed. Unless otherwise stated, it should be understood that all of the following references have this quality.

One objective of the proposed dissertation is to consider players that continuously adapt their strategies based on the opponent's performance in LQDGs. For an autonomous agent, it is important to be aware of strategies that are conjugate, that is, strategies that contain points of infinite feedback gain. In surveying the literature, particular attention has been given to existence conditions for a solution to the LQDG, which is closely related to conjugacy. Solutions are generally divided into closed-loop and open-loop solutions depending on whether equilibria are found in feedback form or not, and the literature can be further classified by whether it addresses ZS or NZS games.

Historically, one of the earliest LQDG works is Ref. [13], in which zero-sum games are studied. This appears to be the first application of variational calculus to LQDGs. A two-player nonzero-sum game is considered with terminal state penalty and running cost on the controls only, and a Nash (saddle-point) equilibrium is sought. A feedback matrix solution is presented, although the solution is different (if equilivalent) to the Riccati matrix expression now commonly associated with LQRs and LQDGs. Second-order conditions are considered including the Legendre-Clebsch and conjugacy conditions. A condition for finiteness of the control energy for pursuer and evader is shown to be equivalent to the absence of conjugate points. For the nonzero-sum differential games, Starr and Ho made the first in-depth study, with a particular focus on LQDGs [7]. This paper calls attention to the need for solutions not corresponding to saddle-point solutions in some scenarios, such as minimax strategies. A class called "noninferior strategies" are also introduced. In general, such strategies only exist if agents cooperate to obtain a mutual benefit, and these solutions are not generally applicable to competitive games. It is shown that the Nash equilibrium leads to coupled Riccati differential equations for the players, while the minimax strategy leads to decoupled equations. Both of these works are foundational and do not contain any detailed study of existence conditions.

Refs. [14] and [15] are closely related works that consider the existence of solutions for a particular form of nonzero-sum LQDGs. Ref. [14] considers games in Hilbert space. The game is nonzero-sum but restricted to a case where players penalize only the state and their own control inputs. Solutions are derived in open-loop form and it is shown that closed-loop solutions can be constructed from the open-loop for the LQDG. For the specific form of LQDG, the game is always “playable.” Ref. [15] assumes the same form of games. By deriving the solution explicitly in closed-loop form, it is shown that the game solution depends on the solution to simultaneous coupled Riccati-like equations. However, the results apply only to the form of LQDG in which players have no weight on the opponent’s control and are not immediately generalizable.

Bernhard[9] presents necessary and sufficient conditions for closed-loop saddle-point solutions in two-player, zero-sum LQDGs. The authors conclude that the absence of conjugate points is not necessary for the existence of a feedback solution in the zero-sum game. The main results are presented in terms of a matrix function of time that is a generalization of the conjugacy conditions presented in Ref. [13]. One condition for existence of a finite-valued cost is that the problem must be “G-reducible,” which implies that the evader cannot control parts of the state-space that are not controllable by the pursuer. Unfortunately, these results do not seem to be immediately generalizable to nonzero sum games. As a counterpart to Bernhard, Delfour [16] considers the existence of open-loop saddle-point solutions to zero sum LQDGs in the presence of conjugate points (called “blow-up points” by the author). This work introduces a theorem for necessary and sufficient existence conditions for an open-loop solution to the LQDG. According to Theorem 2.10, sufficient and necessary conditions for the existence of an open loop saddle point are: (1) the objective function is convex in control u and concave in control v; (2) there exists a symmetrical solution to the zero-sum Riccati differential equation. These results are specialized to the zero-sum case.

There is a significant theoretical overlap between H_∞ control synthesis and zero-sum LQDGs in the area of robust control. The infinite horizon LQDG can be thought of as an H_∞ control problem in which the opponent represents a worst-case disturbance for design. Bryson and Carrier [17] note the connection between LQDGs and H_∞ control synthesis. The authors note that existence conditions for a steady-state solution to the LQDG are related to the H_∞ problem of design to the largest tolerable disturbance energy, although such conditions are not studied in this paper. In appendices, the book by Başar and Bernhard[11] discusses some of the theory relating to the largest tolerable disturbance energy. The authors cite the earlier work of Bernhard[9] and there do not appear to be any more recent developments leading to a more compact expression for the maximum disturbance energy.

General existence conditions for the NZS game are challenging and have not received as much attention. Ref. [18] presents some general existence conditions for Nash equilibrium of a two-player, nonzero-sum LQDG. The solutions to the coupled Riccati differential equations and algebraic Riccati equation are shown to depend on the solution of a matrix equation. Invertibility of the solutions to the matrix equation is shown to be a necessary condition for existence of a solution to the Riccati equations. Controllability of the system through pursuer and evader control influence, and observability of the state through the cost function are shown to be equivalent to invertibility of the solutions. In Wang and Cruz[19], existence conditions for nonzero-sum LQDGs are derived under some restrictions. The solutions to the coupled Riccati-like equations are derived for a second-order, two-player system, under the assumption that the controls are separable (each player can only directly affect one state). Further, solutions to the Riccati equation are assumed to be diagonal at all times, leading to decentralized controls in which each player requires knowledge only of her own state. Both infinite and finite horizon solutions are presented and a generalization to systems of order $2n$ are given. However, the diagonal solution of the Riccati equation is quite restrictive and limits the wider application of this work.

This subsection has reviewed relevant historical literature in the area of LQDGs. Next, the equilibrium solution concepts introduced earlier are applied to develop mathematical solutions to LQDGs.

1.2.3 Solution types

This dissertation considers three types of solutions to the LQDG: minimax, OSO, and FNE solutions. This section derives the governing equations for each case. A critical distinction between the three solutions should be noted. The minimax strategy is always guaranteed to exist for quite modest assumptions that are always satisfied by the problems in this dissertation[9]. The OSO and FNE strategies' existence is conditional on the opponent's strategy and weight magnitudes.

Qualitatively, it may be said that if a player and her opponent's control effort are of very different magnitudes, then finite-valued solutions of the governing equations in the OSO and FNE strategies will not exist. This is an important difference between the three strategies and should be kept in mind. The governing equations for each possible strategy are now derived.

1.2.3.1 Minimax strategy

A minimax player j develops his strategy independently of the opponent's weights. The minimax control satisfies the following expression, subject to the linear state dynamics:

$$\boldsymbol{u}_j^{(\text{minimax})} = \text{argmin}_{\boldsymbol{u}_j} \max_{\boldsymbol{u}_i} \left\{ \frac{1}{2}\boldsymbol{x}(T_f)^T[S_{f,j}]\boldsymbol{x}(T_f) + \frac{1}{2}\int_{t_0}^{T_f} \left(\boldsymbol{x}^T[Q_j]\boldsymbol{x} + \boldsymbol{u}_j^T[R_{jj}]\boldsymbol{u}_j + \boldsymbol{u}_i^T[R_{ji}]\boldsymbol{u}_i\right) \,\mathrm{d}t \right\} \tag{1.4}$$

The solution is developed from a variational optimal control perspective, and is identical in most respects to the classic LQR problem solution[10, 7]. First, a Hamiltonian is formed:

$$H = \frac{1}{2}\left(\boldsymbol{x}^T[Q_j]\boldsymbol{x} + \boldsymbol{u}_j^T[R_{jj}]\boldsymbol{u}_j + \boldsymbol{u}_i^T[R_{ji}]\boldsymbol{u}_i\right) + \boldsymbol{\lambda}^T([A]\boldsymbol{x} + [B_i]\boldsymbol{u}_i + [B_j]\boldsymbol{u}_j) \tag{1.5}$$

The stationarity conditions are applied as follows to obtain necessary conditions on the controls:

$$H_{,\boldsymbol{u}_j} = [R_{jj}]\boldsymbol{u}_j + [B_j]^T\boldsymbol{\lambda} = 0 \rightarrow \boldsymbol{u}_j = -[R_{jj}]^{-1}[B_j]^T\boldsymbol{\lambda} \tag{1.6}$$

$$H_{,\boldsymbol{u}_i} = [R_{ji}]\boldsymbol{u}_i + [B_i]^T\boldsymbol{\lambda} = 0 \rightarrow \boldsymbol{u}_i = -[R_{ji}]^{-1}[B_i]^T\boldsymbol{\lambda} \tag{1.7}$$

The costate equation yields the following:

$$\dot{\boldsymbol{\lambda}} = -H_{,\boldsymbol{x}} = -[Q_j]\boldsymbol{x} - [A]^T\boldsymbol{\lambda} \tag{1.8}$$

The problem is subject to the boundary condition $\boldsymbol{\lambda}(T_f) = [S_{f,j}]\boldsymbol{x}(T_f)$. Assume the existence of a solution of the form[2] $\boldsymbol{\lambda}(t) = [S_j(t)]\boldsymbol{x}$. It can be shown that the first-order optimality condi-

[2] To the best of the author's knowledge, this solution form was first applied to LQDGs by Ho, Bryson, and Baron in Ref. [13].

tions are satisfied if the following Riccati difference equation (RDE) has a solution, subject to the boundary condition $[S_j(T_f)] = [S_{f,j}]$:

$$[\dot{S}_j] = -[S_j][A] - [A]^T[S_j] - [Q_j] + [S_j][E][S_j] \tag{1.9}$$

$$[E] \equiv [B_j][R_{jj}]^{-1}[B_j]^T + [B_i][R_{ji}]^{-1}[B_i]^T$$

Eq. 1.9 has the same form as the LQR RDE. Recall that only the first-order (necessary) conditions of optimality have been applied. For a minimax solution, the second-order (sufficient) conditions of optimality for the differential game require $H_{,\boldsymbol{u}_j\boldsymbol{u}_j} > 0$ and $H_{,\boldsymbol{u}_i\boldsymbol{u}_i} < 0$ [20, p. 280]. Since the Hamiltonian is quadratic in the controls, the second order conditions require $[R_{ji}] < 0$ and $[R_{jj}] > 0$. Therefore, it is concluded that if a non-conjugate solution to Eq. 1.9 exists with $[R_{ji}] < 0$ and $[R_{jj}] > 0$, $\boldsymbol{u}_j = -[R_{jj}]^{-1}[B_j]^T[S_j]\boldsymbol{x}$ is the minimax feedback control for player j.

1.2.3.2 One-sided optimal strategy

In the OSO strategy, the OSO player is assumed to be i and the opponent j uses a minimax strategy[3]. An exact OSO strategy requires knowledge of the governing state weights: $[Q_i]$, $[Q_j]$, $[R_{ii}]$, $[R_{ij}]$, $[R_{jj}]$, $[R_{ji}]$, $[S_{f,i}]$, and $[S_{f,j}]$. The opponent j uses a feedback control $\boldsymbol{u}_j = -[R_{jj}]^{-1}[B_j]^T[S_j]\boldsymbol{x}$, where $[S_j]$ is the solution to Eq. 1.9. The closed-loop problem can then be expressed as a standard optimization problem subject to modified dynamics by substituting j's feedback control into Eq. 1.2. Closed-loop values of the state influence matrix and state weight can be defined as follows:

$$[A'] \equiv [A] - [B_j][R_{jj}]^{-1}[B_j]^T[S_j] \tag{1.10}$$

$$[Q'] \equiv [Q_i] + [S_j][B_j][R_{jj}]^{-1}[R_{ij}][R_{jj}]^{-1}[B_j]^T[S_j] \tag{1.11}$$

Performing the substitution into the governing objective function and dynamics yields a stan-

[3] If player j uses any other strategy, i can still obtain a "one-sided optimal" solution with respect to j. In the dissertation, "OSO" refers to the specific strategy that is optimal against a minimax opponent, which arises naturally in the context of the adaptive LQDGs discussed in Chapter 2.

dard LQR problem with time-varying matrices:

$$J_i = \min_{\boldsymbol{u}_i} \left\{ \frac{1}{2}\boldsymbol{x}(T_f)^T[S_{f,i}]\boldsymbol{x}(T_f) + \frac{1}{2}\int_{t_0}^{T_f} \left(\boldsymbol{x}^T[Q']\boldsymbol{x} + \boldsymbol{u}_i^T[R_{ii}]\boldsymbol{u}_i\right) \, \mathrm{d}t \right\} \tag{1.12}$$

$$\text{s.t. } \dot{\boldsymbol{x}} = [A']\boldsymbol{x} + [B_i]\boldsymbol{u}_i$$

Performing a similar derivation as in the minimax case, the LQR RDE is obtained, subject to the boundary condition $[S_i(T_f)] = [S_{f,i}]$:

$$[\dot{S}_i] = -[S_i][A'] - [A']^T[S_i] - [Q'] + [S_i][B_i][R_{ii}]^{-1}[B_i]^T[S_i] \tag{1.13}$$

A finite solution $[S_i]$ to Eq. 1.13 satisfies the first-order conditions for a minimum. However, some caution must be used. Observe Eq. 1.11. For most noncooperative games, $[R_{ij}] \leq 0$ should be used; otherwise agent i is seeking a strategy for which j's control use is small. Therefore the closed-loop state weight $[Q']$ may not satisfy $[Q'] > 0$. It can be shown that a finite-valued solution of the governing RDE satisfies the second-order convexity condition for optimality irrespective of the properties of $[Q']$. However, for a sufficiently large opponent control, a finite-valued solution of the RDE may not exist. In this case, the second-order Jacobi condition of optimality is violated and the RDE no longer provides an optimal control solution[20]. Qualitatively, this condition arises when the relative effort of the evader is much greater than the pursuer's. For the problems studied in Chapters 2–4, problem weights are chosen to ensure finiteness of the governing RDEs. Chapter 5 considers the problem of determining conditions for existence of finite-valued solutions for both the OSO and FNE engagements.

1.2.3.3 Feedback Nash equilibrium

In the FNE strategy, agents seek saddle-point solutions to their objective functions, subject to shared dynamics. The solution form can be derived using variational principles from optimal control with one Hamiltonian function for each player:

$$H_i = \frac{1}{2}\left(\boldsymbol{x}^T[Q_i]\boldsymbol{x} + \boldsymbol{u}_i^T[R_{ii}]\boldsymbol{u}_i + \boldsymbol{u}_j^T[R_{ij}]\boldsymbol{u}_j\right) + \boldsymbol{\lambda}_i^T([A]\boldsymbol{x} + [B_i]\boldsymbol{u}_i + [B_j]\boldsymbol{u}_j) \tag{1.14}$$

$$H_j = \frac{1}{2}\left(\boldsymbol{x}^T[Q_j]\boldsymbol{x} + \boldsymbol{u}_j^T[R_{jj}]\boldsymbol{u}_j + \boldsymbol{u}_i^T[R_{ji}]\boldsymbol{u}_i\right) + \boldsymbol{\lambda}_j^T([A]\boldsymbol{x} + [B_i]\boldsymbol{u}_i + [B_j]\boldsymbol{u}_j) \tag{1.15}$$

The stationarity conditions yield equations for the control values as follow:

$$\boldsymbol{u}_i = -[R_{ii}]^{-1}[B_i]^T\boldsymbol{\lambda}_i \tag{1.16}$$

$$\boldsymbol{u}_j = -[R_{jj}]^{-1}[B_j]^T\boldsymbol{\lambda}_j \tag{1.17}$$

The costate equations are obtained as follow, recalling that the opponent control must be assumed to be a function of the state:

$$\dot{\boldsymbol{\lambda}}_i = -H_{i,\boldsymbol{x}} = -[Q_i]\boldsymbol{x} - [A]^T\boldsymbol{\lambda}_i - \frac{\partial \boldsymbol{u}_j}{\partial \boldsymbol{x}}^T[R_{ij}]\boldsymbol{u}_j - \frac{\partial \boldsymbol{u}_j}{\partial \boldsymbol{x}}^T[B_j]^T\boldsymbol{\lambda}_i \tag{1.18}$$

$$\dot{\boldsymbol{\lambda}}_j = -H_{j,\boldsymbol{x}} = -[Q_j]\boldsymbol{x} - [A]^T\boldsymbol{\lambda}_j - \frac{\partial \boldsymbol{u}_i}{\partial \boldsymbol{x}}^T[R_{ji}]\boldsymbol{u}_i - \frac{\partial \boldsymbol{u}_i}{\partial \boldsymbol{x}}^T[B_i]^T\boldsymbol{\lambda}_j \tag{1.19}$$

As in the LQR case, boundary conditions for the costates are linear in the state: $\boldsymbol{\lambda}_i(T_f) = [S_{f,i}]\boldsymbol{x}$, $\boldsymbol{\lambda}_j(T_f) = [S_{f,j}]\boldsymbol{x}$. As before, assume that the costate solution is a linear function of the state, yielding $\boldsymbol{\lambda}_i = [S_i]\boldsymbol{x}$ and $\boldsymbol{\lambda}_j = [S_j]\boldsymbol{x}$. The closed-loop controls are given by $\boldsymbol{u}_i = -[R_{ii}]^{-1}[B_i]^T[S_i]\boldsymbol{x}$ and $\boldsymbol{u}_j = -[R_{jj}]^{-1}[B_j]^T[S_j]\boldsymbol{x}$. Substituting these expressions into the FNE controls and into the costate equations yields the following:

$$[\dot{S}_i]\boldsymbol{x}+[S_i]\dot{\boldsymbol{x}} = -[Q_i]\boldsymbol{x}-[A]^T[S_i]\boldsymbol{x}-[S_j][B_j][R_{jj}]^{-1}[R_{ij}][R_{jj}]^{-1}[B_j]^T[S_j]\boldsymbol{x}+[S_j][B_j][R_{jj}]^{-1}[B_j][S_i]\boldsymbol{x} \tag{1.20}$$

$$[\dot{S}_j]\boldsymbol{x}+[S_j]\dot{\boldsymbol{x}} = -[Q_j]\boldsymbol{x}-[A]^T[S_j]\boldsymbol{x}-[S_i][B_i][R_{ii}]^{-1}[R_{ji}][R_{ii}]^{-1}[B_i]^T[S_i]\boldsymbol{x}+[S_i][B_i][R_{ii}]^{-1}[B_i][S_j]\boldsymbol{x} \tag{1.21}$$

Substitution of the closed-loop dynamics in for $\dot{\boldsymbol{x}}$ yields coupled RDEs for $[S_i]$ and $[S_j]$:

$$[\dot{S}_i] = -[S_i][A] - [A]^T[S_i] - [Q_i] + [S_i][E_{ii}][S_i] + [S_i][E_{jj}][S_j] + [S_j][E_{jj}][S_i] - [S_j][E_{ij}][S_j] \quad (1.22)$$

$$[\dot{S}_j] = -[S_j][A] - [A]^T[S_j] - [Q_j] + [S_j][E_{jj}][S_j] + [S_j][E_{ii}][S_i] + [S_i][E_{ii}][S_j] - [S_i][E_{ji}][S_i] \quad (1.23)$$

$$[E_{ii}] \equiv [B_i][R_{ii}]^{-1}[B_i]^T \quad (1.24)$$

$$[E_{jj}] \equiv [B_j][R_{jj}]^{-1}[B_j]^T \quad (1.25)$$

$$[E_{ij}] \equiv [B_j][R_{jj}]^{-1}[R_{ij}][R_{jj}]^{-1}[B_j]^T \quad (1.26)$$

$$[E_{ji}] \equiv [B_i][R_{ii}]^{-1}[R_{ji}][R_{ii}]^{-1}[B_i]^T \quad (1.27)$$

Eqs. 1.22 and 1.23 are coupled equations that govern the FNE case. Due to the assumed symmetry of the control penalty matrices, the terms $[E_{ii}]$, $[E_{jj}]$, $[E_{ij}]$, and $[E_{ji}]$ are all symmetric. In this dissertation, Eqs. 1.9, 1.13, 1.22 and 1.23 are all referred to as **generalized RDEs** to emphasize that the solutions are related but distinct in most cases from the classical Riccati difference equations arising in optimal control and estimation. As noted by authors such as Engwerda[21, p. 53], a linear feedback Nash equilibrium exists if and only if the coupled RDEs have symmetric solutions $[S_i]$, $[S_j]$ on the interval of interest.

It is worth comparing the OSO and FNE RDEs of Eqs. 1.13 and 1.22. If the closed-loop matrices in Eq. 1.13 are expanded, the two equations have identical forms. Mathematically, an adaptive player i's gain is governed by the same differential equation for both OSO and FNE cases. The opponent gain equations are distinct; in the OSO case, j's gain is independent of i's weights, but in the FNE case, j's gain is a coupled function of $[S_i]$.

In considering LQDGs under uncertainty, an agent may encounter an opponent who plays a strategy for which no equilibrium solution exists; or, due to uncertainty, an agent may believe the opponent has such a strategy. For example, if an evader in a LQDG is willing to use much more control energy than a pursuer, the generalized RDEs will in general become singular at some point.

It is desirable to have a test for existence of a solution, given a candidate set of weights. Such a test would prevent the agent from attempting to numerically integrate through a singularity, and could be used as a trigger to engage some contingency plan(s). There appear to be no general conditions for guaranteeing the existence of a solution to the coupled FNE RDEs of Eqs 1.22 and 1.23, as noted by multiple authors[19, 22]. From practical experience, the most reliable way to determine if a finite-valued RDE solution exists is to attempt to numerically integrate its solutions from T_f to t_0. A numerical procedure for determining the existence of a singularity in the interval of integration is needed, and a proposed approach is discussed in Chapter 5.

This section has presented the generalized RDEs governing the three types of LQDG solutions that are used. Next, the objective function cost associated with each solution type is discussed.

1.2.4 Optimal game costs

This subsection discusses the objective function costs associated with a given LQDG. First, the cost-to-go associated with each LQDG solution is given. Second, the concepts of zero-sum and nonzero-sum games are reviewed in the context of uncertain games.

1.2.4.1 Minimax cost-to-go

For the minimax case it is necessary to determine the guaranteed cost level for a player, which corresponds to the cost-to-go in the ZS game. The minimax cost-to-go is simply a quadratic function of the state, as in the LQR problem. This can be readily derived. Let J_{c2g} indicate the cost to go from time t to T_f:

$$J_{c2g} = \frac{1}{2}\boldsymbol{x}(T_f)^T[S_{f,j}]\boldsymbol{x}(T_f) + \frac{1}{2}\int_t^{T_f} \left(\boldsymbol{x}^T[Q_j]\boldsymbol{x} + \boldsymbol{u}_j^T[R_{jj}]\boldsymbol{u}_j + \boldsymbol{u}_i^T[R_{ji}]\boldsymbol{u}_i\right)\,\mathrm{d}\tau \tag{1.28}$$

The terminal cost is converted to an integral cost plus a quadratic function of the current state as follows:

$$\boldsymbol{x}(T_f)^T[S_{f,j}]\boldsymbol{x}(T_f) = \boldsymbol{x}(t)^T[F(t)]\boldsymbol{x}(t) + \int_t^{T_f} \frac{d}{d\tau}(\boldsymbol{x}^T[F]\boldsymbol{x})\,\mathrm{d}\tau \tag{1.29}$$

Substituting the ZS game controls $\boldsymbol{u}_j = -[R_{jj}]^{-1}[B_j]^T[S_j]\boldsymbol{x}$ and $\boldsymbol{u}_i = -[R_{ji}]^{-1}[B_i]^T[S_j]\boldsymbol{x}$

and simplifying yields the cost-to-go in terms of $[\dot{F}]$:

$$J_{c2g} = \frac{1}{2}\boldsymbol{x}(t)^T[F(t)]\boldsymbol{x}(t) + \frac{1}{2}\int_t^{T_f} \boldsymbol{x}^T\Big([A]^T[F] + [F][A] + [Q_j] - [S_j][E][F] - \dots \\ [F][E][S_j] + [S_j][E][S_j] + [\dot{F}]\Big)\boldsymbol{x}\,\mathrm{d}\tau \quad (1.30)$$

Letting $[F] = [S_j]$ causes the integrand to become uniformly zero, and the minimax value of the cost-to-go for agent j depends only on $[S_j]$ and the current state:

$$J_{c2g} = \frac{1}{2}\boldsymbol{x}(t)^T[S_j]\boldsymbol{x}(t) \quad (1.31)$$

It is emphasized that Eq. 1.31 gives the value of using the minimax strategy from time t only if the opponent uses the maximizing control strategy. If the opponent uses any other control solution, then the actual cost incurred shall be reduced. However, this fact does not guarantee system stability or instability, which must be evaluated separately. Furthermore, the cost “ceiling” of Eq. 1.31 assumes complete, perfect, and certain information, and does not account for uncertainty arising due to sensing errors and other sources of uncertainty, all of which will degrade that actual cost achieved.

1.2.4.2 One-sided optimal cost-to-go

The OSO case has already been shown to be an LQR problem with time-varying state weights. Therefore the solution to Eq. 1.13, if it exists and is finite-valued, gives the cost-to-go if agent i uses a one-sided optimal strategy against a minimax agent j[10]:

$$J_{c2g} = \frac{1}{2}\boldsymbol{x}(t)^T[S_i]\boldsymbol{x}(t) \quad (1.32)$$

1.2.4.3 Feedback Nash equilibrium cost-to-go

The cost-to-go for agent i playing against j, in which both agents use the FNE strategies, can be derived in a similar fashion to the minimax cost-to-go. The process begins with the FNE cost-to-go from time t:

$$J_{c2g} = \frac{1}{2}\boldsymbol{x}(T_f)^T[S_{f,i}]\boldsymbol{x}(T_f) + \frac{1}{2}\int_t^{T_f} \left(\boldsymbol{x}^T[Q_i]\boldsymbol{x} + \boldsymbol{u}_i^T[R_{ii}]\boldsymbol{u}_i + \boldsymbol{u}_j^T[R_{ij}]\boldsymbol{u}_j\right) \mathrm{d}\tau \tag{1.33}$$

Applying Eq. 1.29 and substituting the closed-loop values of the control yields the following expression:

$$\begin{aligned} J_{c2g} = \frac{1}{2}\boldsymbol{x}(t)^T[F]\boldsymbol{x}(t) + \frac{1}{2}\int_t^{T_f} \bigg(\boldsymbol{x}^T\big([Q_i] + [S_i][B_i][R_{ii}][B_i]^T[S_i] + [S_j][B_j][R_{ij}][B_j]^T[S_j]+ \\ [F](A - [B_i][R_{ii}]^{-1}[B_i]^T[S_i] - [B_j][R_{jj}]^{-1}[B_j]^T[S_j])+ \\ (A - [B_i][R_{ii}]^{-1}[B_i]^T[S_i] - [B_j][R_{jj}]^{-1}[B_j]^T[S_j])^T[F] + [\dot{F}]\big)\boldsymbol{x}\bigg) \mathrm{d}\tau \end{aligned} \tag{1.34}$$

Substitution of $[F] = [S_i]$ again makes the integrand zero and so the optimal cost-to-go is a quadratic function of the state:

$$J_{c2g} = \frac{1}{2}\boldsymbol{x}(t)^T[S_i]\boldsymbol{x}(t) \tag{1.35}$$

It should be noted that all of these results for the costs-to-go can be found in the literature; e.g., Ref. [10] for the OSO cost-to-go and Ref. [21, p. 364] for the FNE cost-to-go. The results are reproduced here to provide context and because the numerical values of the estimated cost-to-go are used later in the dissertation to quantify tradeoffs between candidate strategies in LQDGs.

Having presented the optimal costs-to-go for all three solution strategies, a brief discussion of ZS and NZS games is now given.

1.2.4.4 An interpretation of nonzero-sum games for uncertainty modeling

The significance and interpretation of ZS and NZS games merit some attention. As has been mentioned earlier in this chapter, ZS games have been extensively studied. Indeed, this approach is intuitive for modeling a large range of potential conflicts, even in engineering systems. Within the context of this dissertation, **NZS games are significant because they indicate a mismatch between an agent's assumptions and the reality it experiences.** For instance, in a missile interception engagement, a NZS game models a discrepancy between an agent's assumed model of its opponent and the actual behavior of the opponent. Qualitatively, this discrepancy can be interpreted as the opponent acting less or more "aggressively" than an agent's expectation.

This section has shown that for each LQDG strategy considered, the cost-to-go for an acting player is always a quadratic function of the state whose value depends on the solution to the player's generalized RDE. Furthermore, the importance of NZS games as a way of modeling uncertainty has been emphasized. The final section of this chapter outlines the remainder of the dissertation.

1.3 Outline of the dissertation

The dissertation is organized into eight chapters, including the present one. The remaining chapters are now summarized.

- Chapter 2 presents an estimation-based approach to account for parametric uncertainty in a LQDG. The primary contribution of this chapter is to extend previous research to the case in which both players are adaptive.

- Chapter 3 considers an orbital pursuit-evasion game with both individual simulations and Monte Carlo examples. The results directly motivate the development of Chapters 4 and 5.

- Chapter 4 proposes incremental improvements in the adaptive gameplay. A statistical test is proposed to prevent a player from acting while uncertainty is large. To avoid repeated numerical integration of the generalized RDEs, a high-order Taylor series approximation to the RDE solutions is obtained using automatic differentiation.

- Chapter 5 proposes a procedure to prevent numerical singularities in the generalized RDEs by numerically quantifying boundaries on opponent weights. The boundaries are used in conjunction with inequality-constrained estimation to improve estimation for near-singular engagements. Statistical tests are proposed for determining when an engagement is singular, enabling a player to take remediating action when a singular engagement is detected.

- Chapter 6 addresses engagements in which the form of the opponent's strategy is uncertain.

The opponent is assumed to be a hybrid dynamic system with modes corresponding to different strategies, selected according to domain knowledge. This model enables a new class of strategies to oppose an adaptive agent, and these are referred to as “deceptive” strategies.

- The new concepts of Chapters 4–6 are explored in extensive numerical examples in Chapter 7. A tactical missile interception engagement is considered and framed as a LQDG. Individual simulations and Monte Carlo results are studied to evaluate the performance of various proposed techniques.

- Chapter 8 summarizes the dissertation and identifies promising directions for future work.

The current chapter has motivated the choice to study differential games under uncertainty. LQDGs are identified as a convenient class of games for which feedback solutions are straightforward to obtain, and the basic solutions and generalized RDEs used in the remainder of the dissertation are presented. The next chapter presents an estimation-based strategy for agents in LQDGs that is appropriate for either the one-sided or two-sided adaptive game solutions.

2. ADAPTIVE GAMEPLAY OVERVIEW

This chapter presents a strategy for adaptive gameplay in NZS games by using nonlinear estimation to obtain estimates of opponent weights. The resulting approach is appropriate for the first objective of this dissertation: recognizing and adapting to a “less aggressive” opponent when the form of the opponent’s strategy is known. This chapter extends previous work of Cavalieri [6] to the case in which both agents play adaptively. Doing so necessitates a change in state parameterization and incurs a computational burden that is addressed in Chapter 4.

The chapter begins with a review of related literature in the area of adaptation in differential games and/or pursuit/evasion engagements. Next, the primary sequential estimation concepts used are reviewed. This is followed by the presentation of the main adaptive gameplay approach. The final section considers the problem of state parameterization in adaptive LQDG gameplay, and presents a parameterization distinct from Cavalieri’s.

2.1 Adaptive gameplay literature review

Any approach that uses data to improve or modify game play for one or more sides in a game is considered “adaptive” for the purposes of this discussion. Of specific interest are scenarios in which information is any combination of incomplete, imperfect, or uncertain, and players attempt to improve their play over the duration of one or more executions of the game. This idea has not been widely studied in the differential games literature; consequently, some relevant references from the area of linear-quadratic optimal control are also included in this section, as there is much theoretical overlap between the two problems.

Much of the work in aerospace engineering has focused on classic missile guidance problems. In Ref. [23], the authors compare the effectiveness of missile intercept guidance laws based on LQDGs against linear-quadratic optimal guidance laws. As noted by the authors, the optimal control approach essentially assumes the opponent control input to be zero and calculates an optimal feedback gain. The problem is assumed subject to control constraints that are enforced by clipping the commanded controls. In simulation results, the target is assumed to be maneuvering at its control limit to escape. This maneuver greatly degrades the performance of the optimal guidance law compared to the differential game-based guidance laws. In a closely related work, Anderson[24] again compares the differential game and optimal control guidance laws, this time in the presence of actuator dynamics for the interceptor. Guidance laws are designed assuming particular values of the maximum target acceleration. In general it is found that the differential game formulation makes the guidance laws more robust when the actual target acceleration exceeds the value used for guidance design. By considering the target acceleration as inexactly known, the two papers explore some characteristics of uncertain differential games; however, the guidance solutions developed are taking advantage of the same robustness that relates LQDGs to H_∞ control, rather than adapting to measured values of the opponent's performance. Belapolsky and Ben-Asher [25] conduct an interesting study in which the effect of control saturation effects are not considered during control design but are present in implementation. In the study, a LQDG is contrasted with a "degenerate" problem in which optimal feedback controls are substituted into the linear-quadratic regulator cost function, leading to perturbation controls about the feedback trajectory. The authors find the degenerate control is less susceptible to control saturation, at the cost of being less responsive to opponent maneuvers. In Ref. [26], a LQDG formulation is used to derive a missile interceptor guidance law with a specified terminal intercept angle. Linearized dynamics are used in the LQDG cost function and the guidance strategy is derived under the saddle-point assumption. The guidance law is evaluated in Monte Carlo against a missile with a square wave acceleration profile having random amplitude and phase. Existence conditions for conjugate points in the feedback solution are also considered. The conjugate point is closely related to the "maximum disturbance energy" threshold for the H_∞ problem. The critical value is computed and used to confirm that selected control laws will not experience any points of conjugacy. Fonod and Shima [27] use a multiple-model estimator with adaptive controller to identify the guidance law and gain values used by a target missile and perform interception. The paper takes a "default" perspective of identifying a model from a discrete set of possible gains, but then introduces a "reduced-order" model in which the gains for each class of guidance law are estimated as continuous, unknown parameters. The target's guidance law and gains are assumed to be constant, but the authors note that an Interacting Multiple Model (IMM) estimator could be used if the target's guidance law is expected to change mid-flight. Turetsky and Shima [28] consider a target-attacker game in which the attacker is allowed to switch between allowed linear guidance laws at prescribed times. The paper takes the perspective of the target and derives evasive strategies. A "missile endgame" scenario is considered in which there is insufficient time to identify the attacker's strategies from data. The attacker is assumed to choose from a finite set of possible strategies and switch times. The corresponding hybrid evasion strategies for both attacker and target are derived from a matrix game perspective. In Ref. [29], a pursuit-evasion game is considered in which the pursuer has hybrid dynamics. The hybrid dynamics refer to the closed-loop feedback gains of the pursuer, which means effectively that the pursuer chooses a time to switch between two possible feedback games. By analysis of the continuity of the zero-effort miss variable, the authors show that the game can be converted to one with impulsive dynamics, and address the problem using an appropriate framework.

In relatively recent literature, attention has been given to adaptive, or learning, strategies in games for general robotic systems. In Ref. [30], a two-player zero-sum LQDG is considered in which players do not know the state dynamics (uncertain information). In this paper, dynamic uncertainty is modeled in "norm-bounded, one-block form." Because of the uncertainty in the dynamics, a saddle-point solution is difficult to evaluate. The paper develops suboptimal strategies that guarantee a threshold value of the objective function; thus, the solution bears similarities to the minimax strategy, while explicitly accounting for uncertainty in the dynamics in the control strategy. The dissertation of Satak [12], and derived publications[31, 32], consider the problem of learning the objective function and strategy of an opponent in a two-player game. A general learning algorithm based on the numerical solution to the Hamilton-Jacobi-Isaacs equation is used. One limitation is that knowledge of the opponent control inputs was assumed. Kopf et al. [33] uses the idea of noncooperative LQDGs to determine an automatic control policy for systems jointly controlled by humans and machines that lack explicit cooperation. In this paper, the usual LQDG objective function is used but the weights used by the human player are assumed a priori unknown. Based on observed trajectories, which include the state and control histories of the human, the human's weights are estimated using an Inverse Reinforcement Learning scheme, and then are used to solve a one-sided optimal control problem for the automatic feedback control. In Ref. [34], an actor-critic reinforcement learning approach is combined with a neural network value function approximation to adaptively solve the Hamilton-Jacobi-Isaacs partial differential equation online. This adaptive strategy is derived for cooperative games.

The dissertation of Ref. [6] is a direct precursor to the present work. In that paper, nonlinear estimation is used to obtain adaptive solutions to LQDGs in the presence of uncertain opponent weights and uncertain relative dynamics. Each player is assumed to employ feedback linearization, such that the relative dynamics can be modelled as effectively linear. The latter concept was previously explored in Ref. [35] for spacecraft guidance synthesis, but does not appear to have come into widespread use. Ref. [6] restricts consideration to a single form of the problem in which one agent plays a nonadaptive minimax strategy. No consideration of a scenario in which both players are adaptive is made. Additionally, the form of the opponent's strategy is assumed known with parameteric uncertainty, but no consideration was made of how the form of the opponent control can be obtained in practice. The present dissertation seeks to extend that work to address these areas of open research.

This completes a survey of related work in adaptive gameplay. The next section briefly reviews nonlinear sequential estimation using Kalman filter approaches.

2.2 Nonlinear estimation

The Kalman Filter is a widely-used estimator that is optimal for systems with linear dynamics and measurements with Gaussian white noise process and measurement noise[36]. The filter has been widely studied and its details can be found in numerous references, such as Ref. [3]. For brevity, most of the implementation details are omitted in this dissertation, and only the most basic details are discussed. The Kalman Filter consists of an iterative propagate-update cycle in which prior state estimates are propagated to a new time and updated with new state measurements[1]. The selection of Gaussian process noise is an approximation in many practical problems, and typically necessitates a tuning process on the part of the designer.

Many practical problems fail to satisfy the requirements of linear dynamics with linear measurements. An enormous volume of research has been conducted to develop various extensions and approximations of the Kalman Filter for nonlinear systems. In this dissertation, the Extended Kalman Filter (EKF) and Unscented Kalman Filter (UKF) are used. The EKF is a classic extension of the Kalman Filter for nonlinear problems. Effectively, local linearizations of the state and measurement models are used to approximate the optimal propagation and update steps for the nonlinear system. The UKF, on the other hand, replaces the prior state by a deterministic particle approximation according to the Unscented Transformation (UT). The weighted statistical mean and covariance of the updated particles is used to reset the estimated mean and covariance for the next cycle of the filter. Additional details of the EKF and UKF descriptions are given in Appendix B, for completeness.

In this dissertation, both EKF and UKF are used. The UKF is theoretically accurate to a higher order than the EKF and can have better convergence properties[4]; however, in the numerical examples of this dissertation, the performance between the two filters was not found to be significantly different. Rather, the UKF is primarily of interest because it does not require explicit Jacobians of the process and measurement models to be developed. For the work presented in this dissertation, the EKF is used when analytical Jacobians of the process model are straightforward to obtain. The primary adaptive LQDG strategies are expressed in terms of parameters for which analytical Jacobians cannot be evaluated in general. For the associated estimators and the Taylor series extension in Chapter 4, the UKF is employed.

This section has briefly described nonlinear estimation using Kalman Filters. The EKF and UKF are introduced and used heavily in the remainder of the dissertation. The next section presents the approach for adaptive gameplay via estimation.

[1] In the uncommon case of continuous measurements, then both propagate and update steps occur simultaneously and continuously.

2.3 Estimation-based game approach

Chapter 1 presents the theoretical governing equations and costs-to-go assuming all agents have complete information about the game state and the weights and strategies of opponents. However, for the OSO and FNE strategies, a player's strategy, encapsulated in a time-varying feedback gain, depends on the weights used by her opponent. Since noncooperating agents do not generally communicate a set of agreed-upon weights in advance, it must be assumed that perfect knowledge of the opponent weights is not available a priori in realistic games. This reality has led the minimax controller to be used in practice for its robustness properties and independence of the opponent's weights. However, if the opponent is playing according to the assumed strategy and accurate estimates of the opponent weights are obtained, then the objective function value can, in some cases, be decreased below the minimax cost by implementing an appropriate solution. For now, the question of identifying the opponent's strategy is set aside, and an agent is assumed to know if the OSO or FNE solution is appropriate. Identifying the correct strategy to use is revisited in Chapter 6.

This section presents the basic approach to achieving adaptive gameplay in LQDGs. First, relevant background that forms the basis of the current dissertation is presented. Second, an adaptive gameplay strategy suitable for two agents playing adaptively against one another is proposed to extend previous research. Third, the proposed extension is formalized into governing generalized RDEs. Fourth, a modified state parameterization for estimation is identified to avoid a two-point boundary value problem.

2.3.1 Adaptive gameplay background and motivating work

The fundamental idea behind adaptive gameplay is that an agent should observe the opponent and update its prediction of the opponent's future actions based on observations. Various methods for achieving this have been considered in previous studies. One challenge in playing adaptively against a potentially adaptive opponent is ensuring a tractable representation. This topic is addressed in a pursuit-evasion scenario in Satak and Hurtado[31]:

> Consider reviewing the information required by the pursuer to model the evader's behavior, who is also learning. The pursuer wants to learn the evader's behavior. The pursuer knows that the evader is learning, therefore the evader must have some opinion about the pursuer. However, the evader also knows that the pursuer is learning. Therefore to model the evader's behavior, it is not enough for the pursuer to know the evader's objective and the evader's model of the pursuer. The pursuer also needs to know what the evader thinks about the pursuer's model for the evader. A similar argument is true for the evader as well. If the two players continue with their arguments, the amount of information required for them to perfectly model each other's behavior will have infinite levels of information ...

The point is that an adaptive player must make some assumption that allows her to compute a control solution from a finite amount of information. Additionally, the player should ensure that the required information is observable. In this chapter, the opponent is assumed to choose strategies that lead to either OSO or FNE game solutions. The solution of a LQDG using either an OSO or FNE solution depends on the solution of one or more generalized RDEs that are coupled in nature. Obtaining a solution requires, at a minimum, knowledge of the opponent weights and the value of the opponent Riccati gain at one time. Given these weights, the value of $[S_j]$ at any time is needed to solve the associated RDEs. The relevant RDEs are repeated below. Eq. 2.1 applies for the OSO problem and Eq. 2.2 for the FNE problem.

$$[\dot{S_j}] = -[S_j][A] - [A]^T[S_j] - [Q_j] + [S_j][E][S_j] \tag{2.1}$$

$$[\dot{S_j}] = -[S_j][A] - [A]^T[S_j] - [Q_j] + [S_j][E_{jj}][S_j] + [S_j][E_{ii}][S_i] + [S_i][E_{ii}][S_j] - [S_i][E_{ji}][S_i] \tag{2.2}$$

$$[\dot{S_i}] = -[S_i][A] - [A]^T[S_i] - [Q_i] + [S_i][E_{ii}][S_i] + [S_i][E_{jj}][S_j] + [S_j][E_{jj}][S_i] - [S_j][E_{ij}][S_j] \tag{2.3}$$

In this dissertation, two primary models of the opponent strategy are used. The corresponding adaptive gameplay strategies are referred to as the "adaptive OSO" and "adaptive FNE" solutions, respectively. The adaptive OSO strategy is equivalent to the estimation-based adaptive gameplay of Ref. [6]. In that work, a "one-sided" adaptive game solution is considered. The opponent is assumed to use a minimax LQDG solution, such that the opponent feedback gain is a fixed function of time independent of the state. By estimating the opponent weights, an adaptive player can approach a perfect information OSO strategy by using the estimated value of the opponent weights to compute a new feedback gain.

For simplicity, agent i shall be taken as the pursuer and j as the evader. In Ref. [6], player i estimates the values of $[Q_j]$ and $[R_{ji}]$ as static parameters. The value of $[S_j]$ is estimated as a time-varying function of the opponent weights with governing equation given by Eq. 2.1. The effect of $[S_j]$ is observable through its influence on the opponent feedback gain. The corresponding optimal closed-loop gain $[S_i]$ is then obtained by first integrating Eq. 2.1 forward in time from t to T_f, then integrating Eq. 2.3 backward in time from T_f, where the boundary condition $[S_{f,i}]$ is given.

This completes the review of the relevant work of Ref. [6]. Next, the extension of this concept to the case of two adaptive agents is considered.

2.3.2 Extension to two adaptive agents

In extending previous work to a "two-sided adaptive" game, it is essential that an agent's assumptions lead to a tractable, observable form. Furthermore, each agent's assumptions should lead to a game solution that satisfies the Nash equilibrium property under some conditions. This property, qualitatively speaking, requires that no agent has an incentive to deviate in her strategy. This second point suggests that, in the limit as the players' estimation errors become small, the solution should approach the perfect information FNE strategy. Following the example of the minimax solution, an adaptive FNE strategy is defined assuming the worst-case value of the **opponent knowledge**. Under this model, each agent has a given set of weights for a LQDG and estimates the value of the opponent weights, **while assuming that the opponent knows the true value of the estimating player's weights**. Then, each agent implements the FNE control given by his known weights and estimates. It is clear that, in the worst case of a perfect information opponent, an agent obtains estimates that satisfy a Feedback Nash Equilibrium to within estimation error tolerance.

Recall that Cavalieri estimated the static opponent weights, $[Q_j]$ and $[R_{ji}]$, and the current-time value of $[S_j]$ subject to its governing RDE. In seeking to apply this work to this new, FNE game strategy, two problems are immediately encountered. First, the generalized RDEs are unstable in forward time for most problems of interest. This means that small errors in estimating $[S_j]$ are amplified in the forward integration process. Second, the value of $[S_j]$ is estimated at t, but the value of $[S_i]$ is fixed at T_f. For the OSO problem, this is overcome because Eq. 2.1 is independent of $[S_i]$. However, for the FNE problem, both coupled equations must be satisfied simultaneously, leading to a two-point boundary value problem (2PBVP). While various solutions to the 2PBVP exist, most solutions are iterative and may not be suitable for real-time implementation. Of the two problems identified, the first problem might be overcome, but the second problem is more severe and indicates another approach may be beneficial. This is the topic of Section 2.3.4. The next subsection presents the model used in this dissertation for LQDGs in which both agents play adaptively.

2.3.3 Adaptive FNE approach

A key assertion of this dissertation is that an adaptive strategy in differential games under incomplete information must satisfy one of two possible conditions. Either the strategy must assume the adaptive agent is more sophisticated than the opponent, or the strategy will describe reality with some bias or other error. This is essentially the same argument made in Chapter 2 of Satak[12], from a different perspective. Motivated by this reference, an adaptive FNE strategy has been outlined that assumes an opponent possesses perfect information. Under this assumption, all uncertainty in the opponent's time-varying gain as a function of unknown opponent weights. An adaptive player solves the following generalized RDEs, in which $[\hat{\cdot}]$ denotes a matrix of estimated values. The adaptive player uses the estimated opponent gain $[\hat{S}_j(t)]$ to propagate her state estimates and implements the feedback gain given by $[S_i(t)]$:

$$[\dot{S}_i] = -[S_i][A] - [A]^T[S_i] - [Q_i] + [S_i][E_{ii}][S_i] + [S_i][E_{jj}][\hat{S}_j] + [\hat{S}_j][E_{jj}][S_i] - [\hat{S}_j][E_{ij}][\hat{S}_j]$$

$$[\dot{\hat{S}}_j] = -[\hat{S}_j][A] - [A]^T[\hat{S}_j] - [\hat{Q}_j] + [\hat{S}_j][E_{jj}][\hat{S}_j] + [\hat{S}_j][E_{ii}][S_i] + [S_i][E_{ii}][\hat{S}_j] - [S_i][\hat{E}_{ji}][S_i]$$

Following the example of Ref. [6], the estimated opponent parameters would be the elements of $[Q_j]$, $[R_{jj}]$, $[R_{ji}]$, and $[S_j(t)]$. However, as the next subsection discusses, it is preferable to use an alternate state parameterization.

If the opponent has perfect information, the adaptive FNE player's strategy is optimal in the limit as the estimation error becomes arbitrarily small. When the opponent has imperfect information and plays the adaptive FNE strategy, there is a mismatch between the two players because neither has perfect information but both assume the opponent has perfect information. In general, this mismatch leads to a bias in the estimation of the opponent weights. This mismatch persists if the players increase their sophistication until and unless one player chooses to play suboptimally by truncating his model at some level of complexity. It is expected that for most problems of practical interest, the improvement in the objective function for these more sophisticated strategies will be insignificant, and consideration in the dissertation is restricted to the adaptive OSO and adaptive FNE approaches already outlined. The information mismatch between players and associated suboptimality are noted as avenues for improvement in a future study.

This subsection formally defines the adaptive FNE strategy used in this dissertation. The following subsection presents and justifies a modified state parameterization for estimation purposes.

2.3.4 Parameterizaton for two-sided adaptive gameplay

This section presents an alternate state parameterization for the opponent's Riccati gain in adaptive LQDG solutions. The terminal value of the opponent's gain, $[S_{f,j}]$, is estimated, rather than the current-time value of $[S_j]$. For estimation purposes, the value of the Riccati gain as a time-varying function can be estimated as Cavalieri[6] did, or the value of the Riccati gain at some epoch time can be estimated as a time-invariant parameter. The Riccati gain can effectively be parameterized by either approach for estimation, and it remains to choose among the available options most suitable for real-time adaptive LQDG gameplay. The time varying-value of $[S_j]$ is attractive from the perspective of estimation because it influences the dynamics as a gain on the true state. However, due to their coupling, $[S_j]$ and $[S_i]$ can only be solved as a 2PBVP when agents use adaptive FNE strategies.

To avoid the 2PBVP, $[S_{f,j}]$ is considered as an alternate parameterization. Under this approach, the state estimates are used to propagate $[S_j]$ backward in time simultaneously with $[S_i]$ for both the OSO and FNE problems. This parameterization has two associated challenges. First, the Jacobian of the state dynamics with respect to $[S_{f,j}]$ cannot be evaluated analytically for a general system. Second, the steady-state behavior of the system is independent of its terminal value. If the transient phase of a particular LQDG is relatively short, then $[S_{f,j}]$ may not be very observable. The first challenge can be overcome by using an estimator that does not require analytical Jacobians, such as the UKF. Little can be done about the second challenge, but its effects are mitigated somewhat; if $[S_{f,j}]$ is weakly observable, it is because its effect on the system is small compared to other weights. Thus, **under the proposed adaptive OSO and FNE solutions, the unique elements of the opponent matrices $[Q_j]$, $[R_{jj}]$, $[R_{ji}]$, and $[S_{f,j}]$ are estimated** simultaneously with the relative state variables using a UKF.

This section has presented the adaptive OSO strategy developed previously by Cavalieri[6] and formalized an adaptive FNE strategy based on similar principles. A discussion of the sub-optimality of implementable adaptive strategies has been given, and the proposed adaptive FNE strategy has been justified as being optimal in the case of a complete information opponent. The state parameterization used by Ref. [6] leads to a 2PBVP when applied to the adaptive FNE problem. Instead, the terminal value of the opponent Riccati matrix is estimated for both adaptive OSO and FNE gameplay.

In summary, this chapter introduces the estimation-based adaptive LQDG solution that is the primary focus of the dissertation. Relevant literature on adaptation and learning in differential games and optimal control problems is discussed. Some basics of nonlinear estimation with Kalman filtering are reviewed. The main focus of the chapter is on extending the work of Ref. [6] to games in which both players act adaptively. Each player achieves a tractable strategy by assuming her opponent has perfect information. The estimated state is modified to estimate the terminal value of the opponent’s Riccati gain, rather than the time-varying value, avoiding a 2PBVP. This is a departure from previous work. In the next chapter, a numerical implementation of an orbital pursuit-evasion game is considered to gain insight into the proposed approach and identify some limitations.

3. APPLICATION: ORBITAL PURSUIT-EVASION GAME[1]

The previous chapter introduces the estimation-based adaptive LQDG approach that is the focus of the dissertation. By applying the FNE concept in agents' sought solutions, the one-sided optimal game solution introduced in Ref. [6] is extended to two-sided adaptive engagements in which both players are adaptive. This chapter considers an orbital PE game under incomplete information. The primary motivation is to demonstrate the proposed approach and highlight some practical shortcomings of the implementation, which motivate developments in later chapters.

This chapter is organized as follows. First, a brief overview of the problem and some relevant background literature is given. Next, the governing dynamic equations are reviewed. This section is followed by a discussion of the adaptive gameplay structure and associated problem weights. Finally, numerical results are presented in the form of single simulation summaries and Monte Carlo simulations. To begin this chapter, an overview of the problem is presented.

3.1 Problem overview

PE engagements involving spacecraft have been studied for many decades. The orbital PE game can model various scenarios of interest, including ICBM interception, anti-satellite operations, and noncooperative rendezvous/docking. In this chapter, the relative motion of two bodies is analyzed using the Hill-Clohessy-Wiltshire (HCW) equations of motion. The HCW equations describe, to first order, the relative motion of satellites acted on by two-body gravity about a reference orbit. For a circular reference orbit, the HCW equations yield linear equations of relative motion for which the satellite motion in the orbital plane decouples from the out-of-plane motion[37]. In this chapter, the two spacecraft engage in a PE game, using the HCW equations to model linear relative dynamics. The remainder of the current section summarizes related research in orbital PE games.

[1] Part of this chapter is reprinted with permission from "Adaptive play via estimation in uncertain nonzero-sum orbital pursuit-evasion games" by T. Woodbury and J.E. Hurtado, 2017, AIAA SPACE Forum, AIAA 2017-5247, Copyright 2017 by Timothy Woodbury and John E. Hurtado.

PE games for orbital spacecraft have been studied for many years, almost always in the context of complete, perfect information games. Early research focused on application of the then-new principles of optimal control and barrier surface ideas introduced by Isaacs[5]. In Woodward[38], optimal thrust angle histories are computed for pursuit-evasion games involving spacecraft in Earth orbit. The solution is obtained by linearizing about a circular reference orbit, then solving using the necessary conditions for minimax control strategies. Anderson [39] presents an analytically derived barrier surface in a planar pursuit-evasion game between low-thrust spacecraft[2]. The dynamics are linearized about a circular reference orbit. The work of Menon and Calise [35] uses feedback linearization to enable the application of linear-quadratic pursuit-evasion game theory to nonlinear spacecraft control problems. The dynamics are formed in Cartesian, inertial coordinates. The nonlinear system dynamics are replaced with pseudocontrol terms that linearize the system. The linear-quadratic game is solved in terms of the new controls, and the nonlinear feedback controls that produce the desired pseudocontrols are then computed. A follow-up paper, Ref. [40], enhances the discussion by considering more realistic mission scenarios and nonlinear dynamics.

More recent studied in orbital PE games have focused on applications of numerical techniques developed for optimal control problems. Ref. [41] applies multiobjective optimization to a spacecraft pursuit evasion game. A multiphase direct transcription approach is used, leading to a multiple objective nonlinear programming problem. Solutions are then found by either building an equivalent weighted scalar system or by using a trade-off approach. A planar spacecraft pursuit-evasion game is presented using an inertial reference frame. Pontani and Conway [42] use a pursuit-evasion formulation for the problem of intercepting an evading spacecraft. Saddle-point solutions are obtained for the problem where the pursuer tries to minimize time-to-capture and evader tries to maximize it. The solution is found using a semi-direct collocation with nonlinear programming algorithm. This framework enforces the analytical necessary first-order conditions, combined with an evolutionary algorithm to generate an initial guess. This study considers a complete, perfect information game. Ref. [43] addresses a decentralized sensor management technique

[2] In this work, the "barrier surface" divides the workspace into two regions: one in which an optimal pursuer is guaranteed capture, and one in which an optimal evader is guaranteed to evader capture "in the short term."

to enable orbital pursuit-evasion activities. This work is primarily concerned with a negotiation game for assigning ground-based sensors to track particular targets. Based on the measured values from the sensors, open-loop optimal strategies for the purser and evader are implemented. The EKF, UKF, and linear minimum mean square error filter are employed to track targets. The dissertation of Hafer [44] applies sensitivity methods to the solution of orbital PE games. The contributions include homotopy techniques for obtaining feedback control solutions and altitude-constrained game solutions. Ref. [45] addresses optimal solutions to the orbital pursuit evasion problem. The HCW equations are used to formulate the problem dynamics. A linear-quadratic solution is compared against a nonlinear solution based on the state-dependent Riccati equation. The solution with the nonlinear formulation shows an improvement in the cumulative cost. A follow-up, Ref. [46], enhances this analysis to include the effect of solving the problem on a more coarse time scale, as well as additional numerical simulations.

The summarized works in the area of orbital PE have primarily been interested in determination of optimal or equilibrium strategies for perfect information environments. The current dissertation's focus on adaptive game solutions under uncertainty is distinct in this regard. The next section summarizes the HCW relative motion dynamics.

3.2 Problem dynamics

The Hill-Clohessy-Whiltshire (HCW) equations for relative motion are used to define realistic initial conditions and external dynamics for the pursuer and evader spacecraft. Ref. [37] gives the relative Cartesian coordinates of one body about a second in terms of the mean motion n. For a circular orbit, motion in the orbital plane decouples from the out-of-plane motion, leading to a solution in terms of relative position coordinates x, y and their derivatives. The out-of-plane motion forms independent dynamics that should be assumed to have different weighting from the in-plane components. When out-of-plane motion is present, uncertainty can be addressed by an independent estimator using the same principles used to develop the in-plane LQDG play. For simplicity, this chapter restricts consideration to the in-plane components of motion.

The full relative equations of motion are presented before simplifying to the case of a circular

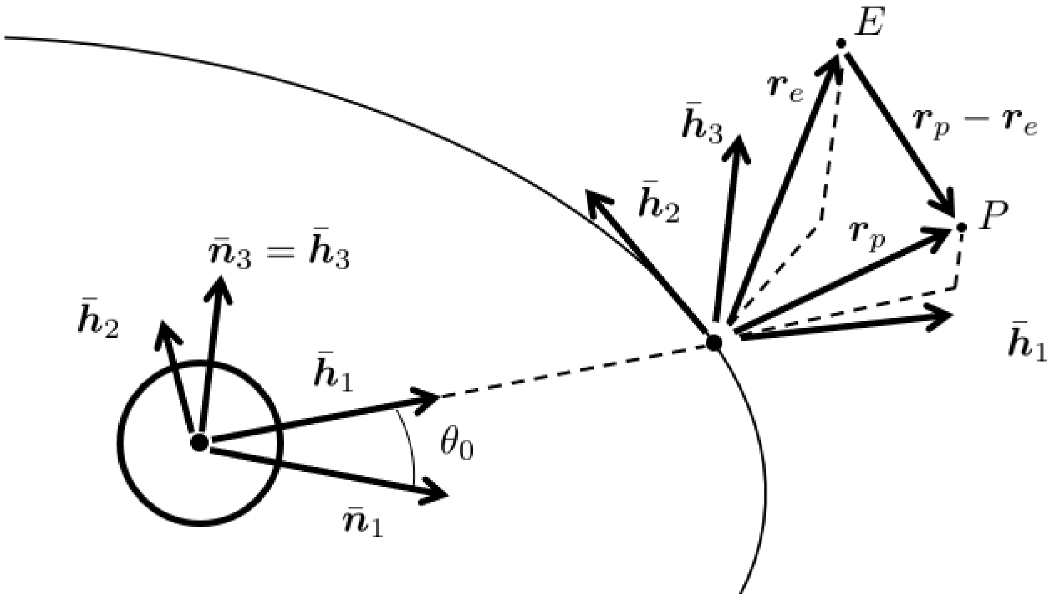

Figure 3.1: Graphic showing the definition of various reference frames and relative position vectors in orbital PE engagement

reference orbit. The motion of the agents is described relative to a reference two-body orbit. The reference orbit is a conic section and the two-body motion can be characterized in terms of the distance from the central body to the orbiting body, $r_0(t)$, and the polar angle $\theta_0(t)$. Note that the subscript “0” is used to denote the reference orbit, and that all quantities thus denoted may be time-varying. The semimajor axis of the reference orbit is denoted by a_0, and it is constant for the two-body problem. The reference frames and some coordinates used are shown in Fig. 3.1. Let an inertial reference frame n^+ be centered at the center of mass of the orbited body. The $\bar{\boldsymbol{n}}_1$ axis points towards periapsis and $\bar{\boldsymbol{n}}_3$ is aligned with the angular momentum vector of the orbiting body about the central body. The n^+ frame position vector of the orbiting body is given from $\boldsymbol{r} = r_0(\cos\theta_0 \bar{\boldsymbol{n}}_1 + \sin\theta_0 \bar{\boldsymbol{n}}_2)$. The governing equations for the orbiting body are given below in terms of the standard gravitational parameter μ:

$$\ddot{r}_0 = r_0\dot{\theta}_0^2 - \frac{\mu}{r_0^2} \tag{3.1}$$

$$\ddot{\theta}_0 = -\frac{2\dot{r}_0\dot{\theta}_0}{r_0} \tag{3.2}$$

The relative motion equations are conveniently described in the rotating Euler-Hill reference frame, also known as the local vertical local horizontal (LVLH) frame, which is denoted h^+. Frame h^+ is aligned with the spacecraft position vector such that the position vector is $\boldsymbol{r} = r_0\bar{\boldsymbol{h}}_1$, and thus h^+ is defined by a simple three-axis rotation through θ_0 from n^+. The motion of a spacecraft in the vicinity of the reference orbit is commonly described in terms of the coordinates x, y, and z, each corresponding to displacements along the axes $\bar{\boldsymbol{h}}_1$, $\bar{\boldsymbol{h}}_2$, and $\bar{\boldsymbol{h}}_3$. By re-writing the governing equation for the relative position of the neighboring spacecraft in terms of the reference orbit's coordinates, relative equations of motion for the displacement coordinates can be obtained. The equations are subject to acceleration terms u_x, u_y, and u_z:

$$\ddot{x} - 2\dot{\theta}_0\dot{y} - \ddot{\theta}_0 y - \dot{\theta}_0^2 x = \frac{\mu}{r_0^2} - \frac{\mu(r_0 + x)}{\left((r_0 + x)^2 + y^2 + z^2\right)^{3/2}} + u_x \tag{3.3}$$

$$\ddot{y} + 2\dot{\theta}_0\dot{x} + \ddot{\theta}_0 x - \dot{\theta}_0^2 y = -\frac{\mu y}{\left((r_0 + x)^2 + y^2 + z^2\right)^{3/2}} + u_y \tag{3.4}$$

$$\ddot{z} = -\frac{\mu z}{\left((r_0 + x)^2 + y^2 + z^2\right)^{3/2}} + u_z \tag{3.5}$$

It is convenient for numerical purposes to rescale the problem dynamics so that all terms are of approximately the same order of magnitude. In the following equations, primed quantities indicate derivatives with respect to the new time variable, τ, and dots indicate derivatives with respect to t. In the subsequent sections of this chapter, the distinction is neglected, and dots are used to indicate derivatives with respect to the new time variable. The constants α, a, b, and c are introduced and used to normalize the states and control inputs as follows. The scale factors used in the numerical implementation in this chapter are summarized in Table 3.1. Define the alternate time variable τ as follows:

$$\tau \equiv \alpha t \tag{3.6}$$

Let the state vector $\boldsymbol{\xi}$ be associated with the x–y components of the displacement, with corresponding control vector $\boldsymbol{u}$ as follows:

$$\boldsymbol{\xi} = \begin{bmatrix} x/a & y/a & x'/b & y'/b \end{bmatrix}^T \tag{3.7}$$

$$\boldsymbol{u} = \begin{bmatrix} u_x/c & u_y/c \end{bmatrix} \tag{3.8}$$

For the particular case of a circular reference orbit, the nonlinear equations approximately simplify to linear, time invariant equations. Taking the first-order Taylor series expansion of the motion about the reference trajectory, and substituting the mean motion $n_0 \equiv \sqrt{\mu/a_0^3}$ yields linear equations in which x and y are decoupled from z and its derivatives. Using this model and the rescaled variables, the orbital plane components of the HCW dynamics have the following state-space form[37]:

$$\boldsymbol{\xi}' = \begin{bmatrix} 0 & 0 & \frac{b}{a} & 0 \\ 0 & 0 & 0 & \frac{b}{a} \\ \frac{3n^2a}{\alpha^2 b} & 0 & 0 & 2n/\alpha \\ 0 & 0 & -2n/\alpha & 0 \end{bmatrix} \boldsymbol{\xi} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \frac{c}{b\alpha^2} & 0 \\ 0 & \frac{c}{b\alpha^2} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \tag{3.9}$$

The HCW equations lead to linear coupled x–y dynamics and undamped out-of-plane motion (in the absence of control). In the remainder of this chapter, the out-of-plane motion components shall be neglected.

Eq. 3.9 describes the relative motion of a satellite about the reference orbit defined by $r_0(t), \theta_0(t)$. That is, the equations describe relative motion about a nonmaneuvering target. The problem can be transformed into a PE engagement in the vicinity of the reference orbit by defining relative state vectors for pursuer and evader, as $\boldsymbol{\xi}_p$ and $\boldsymbol{\xi}_e$. It is obvious that the relative state vector, $\boldsymbol{z} \equiv \boldsymbol{\xi}_p - \boldsymbol{\xi}_e$

$\alpha\,(s^{-1})$	$a\,(m)$	$b\,(ms^{-1})$	$c\,(ms^{-2})$	$n\,(s^{-1})$
10^{-2}	10^{4}	10^{4}	1	3.4366×10^{-4}

Table 3.1: Dynamic scale factors

has the same state influence matrix with control contributions from each player:

$$\boldsymbol{z}' = \begin{bmatrix} 0 & 0 & \frac{b}{a} & 0 \\ 0 & 0 & 0 & \frac{b}{a} \\ \frac{3n^2a}{\alpha^2 b} & 0 & 0 & 2n/\alpha \\ 0 & 0 & -2n/\alpha & 0 \end{bmatrix} \boldsymbol{z} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \frac{c}{b\alpha^2} & 0 \\ 0 & \frac{c}{b\alpha^2} \end{bmatrix} (\boldsymbol{u}_p - \boldsymbol{u}_e) \tag{3.10}$$

This section has presented the governing relative dynamics of a PE game taking place in the neighborhood of a circular reference orbit. The governing equations for motion in the orbital plane are independent of the out-of-plane motion and vice versa. For convenience, the problem has been rescaled so that all states are of approximately the same magnitude. The following section describes the implementation of adaptive game play for the orbital PE game.

3.3 Additional considerations and problem weights

The previous section presents the governing relative equations of motion for the orbital PE game. The current section describes the adaptive gameplay applied to this example problem. Each player has a quadratic objective function and plays either a minimax or an adaptive strategy. Adaptive strategies assume a parameterized form for the opponent's gain as a function of time and use sequential estimation for the uncertain parameters. Observability analysis of the chosen form of the opponent weights is conducted, and results that confirm observability are presented in Appendix A.2.

This section contains three parts. First, the basic game description and the way in which players use adaptation is described. Second, the equations for sequential estimation are presented.

3.3.1 Adaptive game description

Players act according to the relative, linear equations of motion in the vicinity of a circular reference orbit. Each player has a quadratic cost function in the relative states and controls, as in Eq. 1.2. The terminal time is assumed to be known and fixed by some external knowledge. The pursuer is denoted by the subscript p and the evader by e; under this convention, $[R_{pe}]$ is the weight that the pursuer places on the evader's control. It is assumed that players place equal weights on the X and Y components of position, velocity, and acceleration. That is, all position-level states have the same weight, all velocity-level states have the same weight, and all controls have the same weight. It is further assumed that each player uses diagonal weights. Players act under planar motion. Under these assumptions, for planar motion using the dynamics of Eq. 3.10, each players' weights take on the following form, for $i \in \{e, p\}$:

$$[Q_i] = \mathrm{diag}(q_{1,i}, q_{1,i}, q_{2,i}, q_{2,i}) \quad [R_{ie}] = \mathrm{diag}(r_i, r_i) \quad [S_{f,i}] = \mathrm{diag}(s_{1,i}, s_{1,i}, s_{2,i}, s_{2,i}) \tag{3.11}$$

Under this framework there are a maximum of five distinct unknown opponent parameters. Note that without loss of generality, the convention of Ref. [6] has been followed, in which $[R_{pp}]$ and $[R_{ep}]$ are each taken to be identity.

Players are either static (minimax) or adaptive in their solutions. Under the minimax strategy, a player solves Eq. 1.2 according to the minimax principle and obtains a time-varying feedback gain that ensures a minimum objective function value against a worst-case opponent control. Under the adaptive strategy, a player uses either the OSO or FNE approach as appropriate. An adaptive player is always assumed to know the type of strategy used by the opponent. That is, the opponent's control is subject only to parametric uncertainty. This is a limiting assumption, and Chapter 6 explores solutions when this assumption is invalid. At each time including the initial time, adaptive players implement the control that is optimal against the assumed form of the opponent strategy for the current values of the state estimates.

Adaptive players initially assume the opponent's weights are at their zero-sum values with some prescribed uncertainty.[3] This is a critical choice of initial condition and merits some particular attention. The adaptive OSO and FNE strategies have been implemented under the assumption that the estimated weights correctly converge to the true weights. This is in contrast to many other adaptive gameplay techniques which iterate or optimize the control signal until a local extrema of the objective function is achieved. In the estimation-based adaptive game solution, the estimates are assumed to be correct and the optimal control for the estimated opponent weights is implemented. It is possible, particularly during the initial transient when uncertainty is high, that the adaptive agent may implement controls that are suboptimal against the unknown true values of the opponent's weights. The zero-sum condition, it should be recalled, is a saddle-point solution of the game at which both the FNE and minimax control strategies become equivalent[7]. By setting the initial estimates of the opponent weights to the zero-sum values, the resulting control shall have the minimax property. Even if no additional information is obtained from observations, the adaptive control strategy ensures a minimum worst-case outcome. This choice of initial conditions has the effect of biasing the closed-loop control toward the minimax strategy. However, it should be noted that, since the convergence properties of the Kalman filter are notoriously difficult to ensure or predict, the transient behavior of the closed-loop filter is not guaranteed to be optimal.

This completes the description of the adaptive game implementation. The next subsection presents the governing equations for the filter used in adaptive LQDG gameplay.

3.3.2 Filter equations

Each agent is assumed to have full-state measurements of the relative position and velocity terms with Gaussian measurement noise. The governing state dynamics are obtained from Eq. 3.10. Let matrices $[A]$ and $[B]$ denote the state and control influence matrices in that equation. The governing equation for the relative state, $\boldsymbol{z} \equiv \boldsymbol{x}_p - \boldsymbol{x}_e$, has the following form in terms of each player's control:

$$\dot{\boldsymbol{z}} = [A]\boldsymbol{z} + [B](\boldsymbol{u}_p - \boldsymbol{u}_e) \tag{3.12}$$

[3] In practice, the magnitude of the initial uncertainty is chosen heuristically based on the convergence properties of the estimator.

Each agent models the opponent weights as stationary white-noise processes with large initial uncertainty. The governing equations for agent p are now presented; obtaining the similar equations for agent e is left as an exercise for the reader. Let $\hat{\boldsymbol{\theta}}_e$ denote a vector representation of e's five unique governing weights. An adaptive agent p assumes the following closed-loop dynamics:

$$\dot{\boldsymbol{z}} = \Big([A] + [B][R_{ee}]^{-1}[B]^T[S_{ee}(t, \hat{\boldsymbol{\theta}}_e)]\Big)\,\boldsymbol{z} + [B]\boldsymbol{u}_p \tag{3.13}$$

$[S_{ee}(t, \hat{\boldsymbol{\theta}}_e)]$ is governed by either the uncoupled RDE of Eq. 1.9 for a minimax evader or the coupled generalized RDE of Eq. 1.23 for a FNE evader.

For an explicit example, consider planar motion under the HCW dynamics. The estimated state vector consists of the four elements of the relative state and the five unique opponent weights:

$$\hat{\boldsymbol{x}} \equiv \begin{bmatrix} x/a & y/a & x'/b & y'/b & q_{1,e} & q_{2,e} & r_e & s_{1,e} & s_{2,e} \end{bmatrix}^T \tag{3.14}$$

The governing process model in state-space form may be summarized as follows. A disturbing process noise vector $\boldsymbol{w} \in \mathbb{R}^5, \boldsymbol{w} \sim \mathcal{N}(\boldsymbol{0}, [Q])$ is assumed:

$$\dot{\hat{\boldsymbol{x}}} = \begin{bmatrix} \frac{b}{a}\hat{x}_3 \\ \frac{b}{a}\hat{x}_4 \\ \frac{3n^2a}{\alpha^2 b}\hat{x}_1 + 2\frac{n}{\alpha}\hat{x}_4 + \frac{c^2}{\hat{x}_7 b^2\alpha^4}\Big(\hat{x}_1 s_6(\hat{\boldsymbol{\theta}}_e) + \hat{x}_2 s_7(\hat{\boldsymbol{\theta}}_e) + \hat{x}_3 s_3(\hat{\boldsymbol{\theta}}_e) + \hat{x}_4 s_{10}(\hat{\boldsymbol{\theta}}_e)\Big) + \frac{c}{b\alpha^2}u_1 + w_1 \\ -2\frac{n}{\alpha}\hat{x}_3 + \frac{c^2}{\hat{x}_7 b^2\alpha^4}\Big(\hat{x}_1 s_8(\hat{\boldsymbol{\theta}}_e) + \hat{x}_2 s_9(\hat{\boldsymbol{\theta}}_e) + \hat{x}_3 s_{10}(\hat{\boldsymbol{\theta}}_e) + \hat{x}_4 s_4(\hat{\boldsymbol{\theta}}_e)\Big) + \frac{c}{b\alpha^2}u_2 + w_2 \\ w_3 \\ w_4 \\ w_5 \\ 0 \\ 0 \end{bmatrix} \tag{3.15}$$

The elements of the opponent's Riccati matrix solution are denoted as follows and subject to terminal boundary conditions:

$$[S_{ee}(t, \hat{\boldsymbol{\theta}}_e)] = \begin{bmatrix} s_1 & s_5 & s_6 & s_8 \\ s_5 & s_2 & s_7 & s_9 \\ s_6 & s_7 & s_3 & s_{10} \\ s_8 & s_9 & s_{10} & s_4 \end{bmatrix} \tag{3.16}$$

$$s_1(T_f) = s_2(T_f) = \hat{x}_8, s_3(T_f) = s_4(T_f) = \hat{x}_9$$
$$s_5(T_f) = s_6(T_f) = s_7(T_f) = s_8(T_f) = s_9(T_f) = s_{10}(T_f) = 0$$

The relative state terms, comprising the four components of $\boldsymbol{z}$, are measured with uncorrelated Gaussian errors. Using the presented process and measurement model, adaptive agents implement a UKF to obtain iteratively better estimates of the opponent weights as the engagement unfolds. At each time instance, the posterior opponent weight estimates are used to re-solve the governing RDE for the adaptive player by integrating backward from the terminal time to the present time. This process generates an updated state feedback gain that is then implemented. Each agent is treated as implementing her control with a zero-order hold.

This completes the description of the filter equations. Next, the problem weights used in numerical examples are presented.

3.3.3 Problem weights for numerical examples

To conclude this section, the numerical values used in subsequent simulations are now reviewed. Table 3.2 presents the problem settings. The table shows the initial relative state, the simulation horizon, and the sample period $\Delta\tau$ in scaled units. The values P_z, P_Q, P_R, and P_S denote each player's initial covariance associated with the relative state, the opponent state weight, the opponent control weight on the evader state, and the terminal state penalty. The value r_k denotes the state measurement error variance, and all states are measured with independent zero-mean

$\boldsymbol{z}(0)$	τ_f	$\Delta\tau$	P_z	P_Q	P_R	P_S	r_k
$(-5, 10, 0.1, -0.1)$	6.0	0.05	0.1	7.5	1.5	0.01	10^{-4}

Table 3.2: Simulation settings for finite horizon simulations.

$q_{1,p}$	$q_{2,p}$	r_p	$s_{1,p}$	$s_{2,p}$	$q_{1,e}$	$q_{2,e}$	r_e	$s_{1,e}$	$s_{2,e}$
3.16	8.91	-2.00	1.00	1.00	-1.00	-1.00	1.80	-1.2	-0.9

Table 3.3: Player weights for orbital PE game

Gaussian errors. The effect of the chosen initial state covariance is to place a large uncertainty on $[Q_j]$ and $[R_{je}]$, which are found to have good observability properties. These values are strongly correlated with the opponent steady-state gain, which is known to be observable in the absence of significant disturbances. The initial uncertainty in $[S_{f,j}]$ is quite small by comparison. Simulation results indicate that the terminal Riccati value is poorly observable at best, and the estimator does not converge with larger initial errors, for the game weights chosen. These game weights are reviewed next.

Table 3.3 shows the values of the unique elements of the opponent gains used in this problem. Pursuer and evader weights are denoted by p and e respectively. Because each player takes $[R_{ip}] = [\mathbb{I}_{n\times n}]$, the weights are simple to interpret. E.g., $\|q_{k,p}\| > \|q_{k,e}\|$ indicates that P places a greater weight on the running state magnitude than E. Similarly, $\|r_e\| < \|r_p\|$ indicates that E places a smaller weight on her control use than P imagines her to do. E places a larger terminal weight on the relative position than P, but a smaller weight on the relative velocity components. In a nominal simulation with no adaptation, it should be qualitatively expected that P will steadily reduce the relative states for most of the simulation. As the simulation approaches the terminal time, E will gradually expend greater control energy to increase the relative position states, which may be enough to reduce their rate of decrease and/or increase the relative states.

This section has described the application of the adaptive gameplay developed in the previous chapter to the orbital PE game. The complete filter equations for one player in a planar game have been presented. Finally, the numerical values used in simulations have been presented. These numerical simulations are the focus of the next section.

3.4 Results

Numerical simulations are conducted to evaluate the effectiveness of the proposed adaptive OSO and FNE strategies. Results are presented in three subsections. Each section begins with a discussion of the metrics used to evaluate agent effectiveness, before presenting results. First, individual simulations are presented for the finite horizon problem. Second, the performance of an adaptive agent playing against a static agent with increasingly inaccurate information is examined to justify the use of adaptative gameplay used in this chapter. Last, Monte Carlo analysis of the three types of adaptive game is shown.

3.4.1 Individual results - finite horizon

Individual simulations are conducted to gain insight into the behavior and performance of adaptive agents. Adaptive agent performance is evaluated by comparing simulations with adaptive gameplay against a reference static simulation in which both agents use minimax strategies. In many of these simulations, it is found that the position, velocity, and control histories often are qualitatively indistinguishable between static and adaptive gameplay. Consequently, the performance of adaptive agents is characterized by their estimation accuracy and cost-to-go change. Estimation accuracy is evaluated by estimation error plots showing the parameter errors and associated 3σ covariance bounds. The 3σ bounds are computed from the diagonal elements of the covariance matrix and off-diagonal elements are ignored.

The cost-to-go based on actual simulation performance is computed by subtracting the running cost from the total cost incurred over the simulation. The change in the cost-to-go is evaluated as the difference between the cost-to-go under adaptive gameplay and under static gameplay. Negative values of the cost-to-go change indicate the adaptive player performs better than the corresponding minimax player in the static game. The cost-to-go change is plotted as a function of time to determine if there are trends relating to initial or terminal transient behavior. The total cost

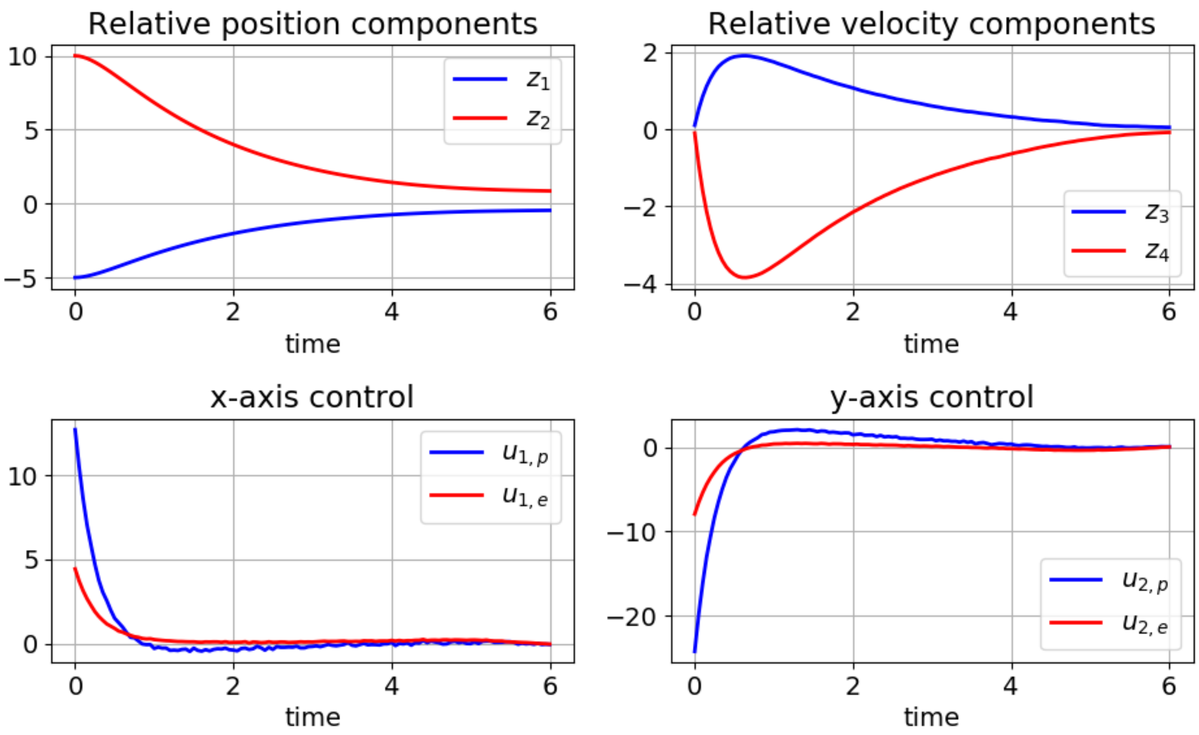

Figure 3.2: Finite horizon gameplay with minimax strategies. In the legend, P and E are used to denote pursuer and evader respectively.

under adaptive gameplay and the reference cost are also reported. The reference cost is denoted as "BL cost," where BL denotes "baseline."

Now, the play of static agents using minimax strategies with imperfect information is considered. In this scenario, the agents use both UKFs to estimate only the relative state while using minimax control gains. The pursuer has greater state weight than the evader, causing her to use more control energy and stabilize the system. The evader has a larger terminal state penalty than the pursuer does, which causes the position history response to level out near the end of the simulation. The resulting system, shown in Fig. 3.2, converges slowly from the initial state. The control magnitudes are large initially but decay quickly. The pursuer expends much more control energy than the evader, particularly at the initial time.

When the pursuer plays adaptively against a static evader, there is little change visible in the state and control histories. The objective function value and estimation errors are shown in Fig.

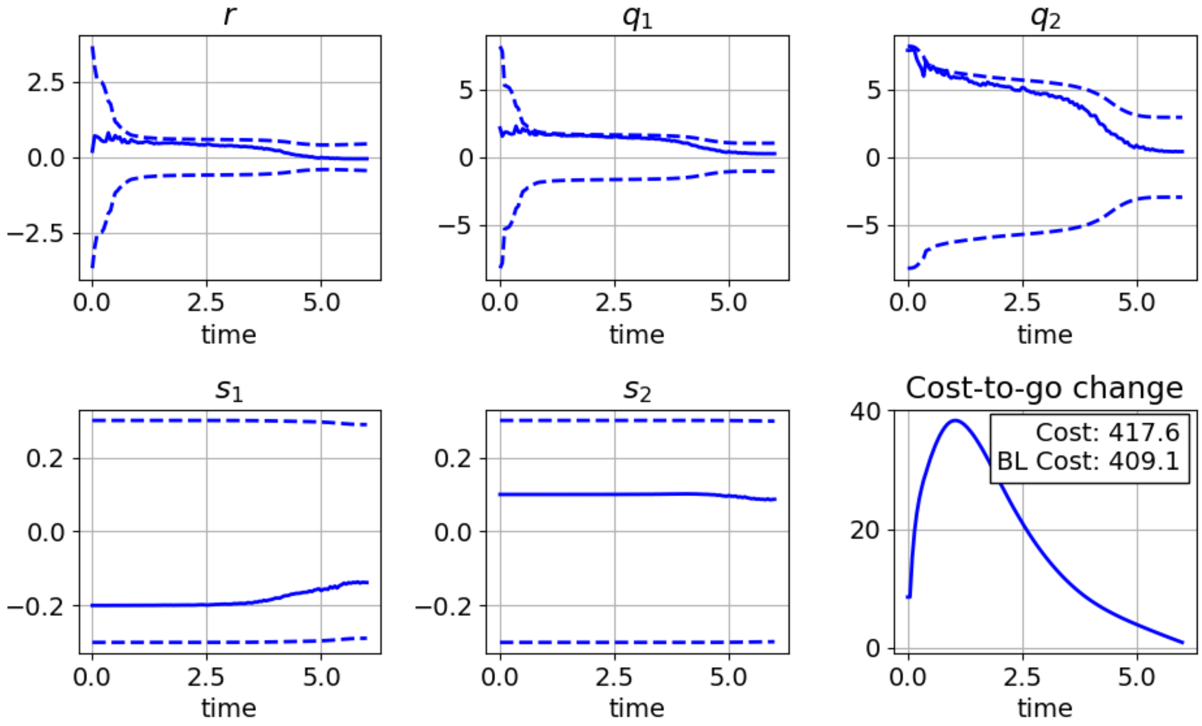

Figure 3.3: Pursuer estimation errors and objective function values for a finite-time simulation

3.3. This figure plots the estimation errors in each parameter and the associated 3σ bounds from the covariance diagonal elements. The relative state errors track the measurements and are omitted for brevity. The parameter estimates trend toward the direction of reduced error as more data are acquired, and are bounded by the typical 3σ errors. However, the convergence of the parameters is relatively slow, and it is clear that there are significant biases in the parameters for most of the simulation. Both the pursuer control weight, r, and the position-level state penalty q_1 show relatively fast convergence of the error. The velocity-level state penalty q_2 shows much slower error convergence. However, all three parameters converge to small errors of order less than one by the end of the simulation. Convergence of the terminal Riccati weights, by contrast, is poor. Even though the error magnitudes are small, the estimator only reduces the error in each element slightly, and only during the final two time units of the simulation, when the opponent's RDE solution departs from its steady value. The cost-to-go change is shown in the last subplot. The true

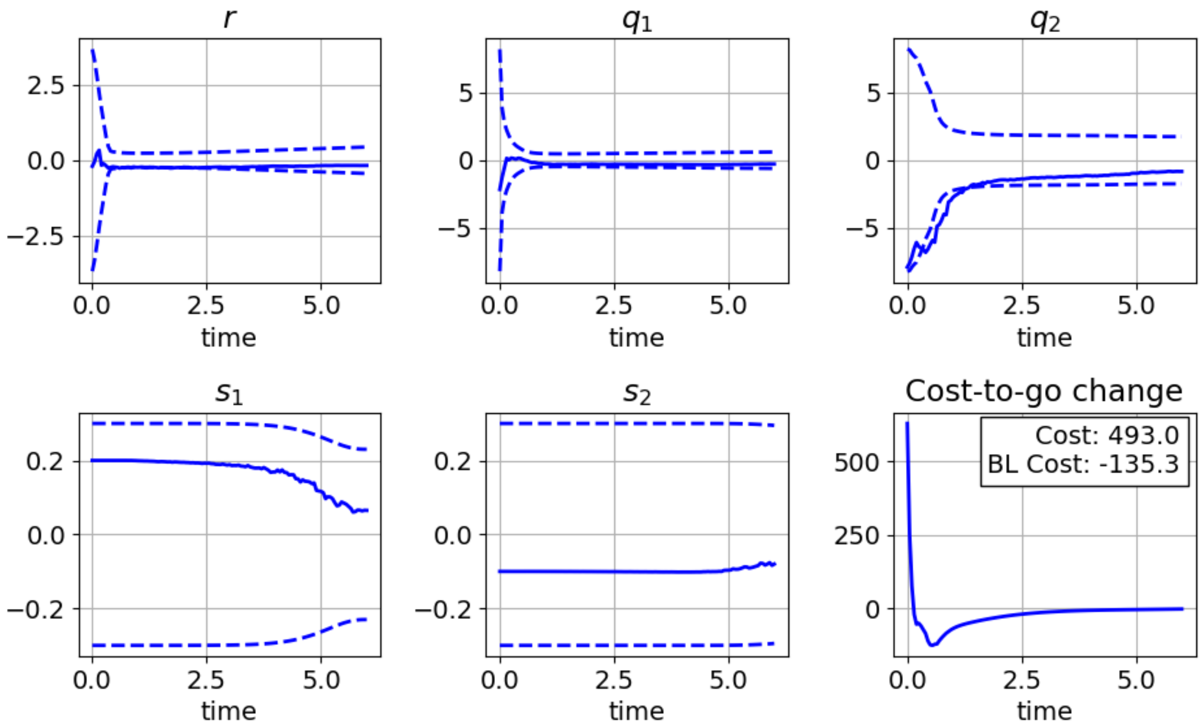

Figure 3.4: Evader estimation errors and objective function values for a finite-time simulation

cost-to-go as a function of time minus the static game baseline value is plotted. When the relative cost-to-go is positive, the adaptive agent has a larger cost-to-go than in the baseline case, and is interpreted as performing worse than the baseline case. Under this interpretation, the adaptive OSO pursuer has a noticeably poor initial response. The cost-to-go under adaptive gameplay, in fact, becomes steadily worse until about $\tau \approx 1$. After that time, the relative cost-to-go decreases and approaches zero as $\tau \rightarrow \tau_f$. The total cost incurred by the adaptive pursuer is about 2% greater than the baseline cost. Overall, the performance shows that the adaptive OSO pursuer is able to obtain accurate estimates of the evader's weights, but with large estimation errors for most of the simulation. While playing according to these biased estimates, the OSO pursuer achieves about a 2% drop in overall performance compares to a minimax baseline reference. That is, due to estimation errors, the performance is worse than playing statically for the initial conditions and weights chosen.

Fig. 3.4 shows the estimation errors and overall performance when the OSO evader plays against a static pursuer. There are two main differences between this plot and Fig. 3.3. First, the estimation errors are reduced much more quickly. Second, the adaptive player incurs a much greater penalty. Regarding the estimation errors, the r and q estimates mainly converge within the first time unit. The covariance bounds and terminal errors are noticeably smaller. The estimates of the terminal state weights are still noticeably difficult to estimate, but the s_1 weight exhibits much better convergence. The finding that the evader has better observability of the pursuer weights is not surprising. The system is nonlinear and the pursuer control inputs are much larger than the evader's. It is reasonable to expect that the pursuer's control gain, which is a direct function of his weights, should be more observable than the evader's control gain.

Turning now to the OSO evader's cost-to-go, it is clear that the evader pays a huge initial penalty. After the initial penalty, the OSO evader outperforms the reference value. This poor initial transient can be explained by the large initial mismatch in the estimated and truth values of the opponent's weights. These errors lead to a large initial control use from the evader, during which a large penalty is accrued. For further consideration, the state histories corresponding to this scenario are also plotted in Fig. 3.5.

The state and control histories with an adaptive evader and static pursuer are noticeably different than the baseline case of Fig. 3.2. In considering Fig. 3.5, the relative positions are no longer monotonic and the evader initially increases the relative distance. Consequently, the terminal position is slightly farther from the origin than in the static case. The evader demonstrates huge initial control inputs. This leads to an increase in the pursuer control as well; however, the pursuer control profile is much smoother. The control history of the evader is a consequence of overconfidence, in which the evader acts according to highly uncertain parameter estimates as if the estimates are accurate. This poor transient performance represents a research challenge, and is addressed in Chapter 4.

Figs. 3.6 and 3.7 show simulation results when both agents simultaneously estimate the other's weights, using the coupled generalized RDEs. In this scenario, the state histories are not noticeably

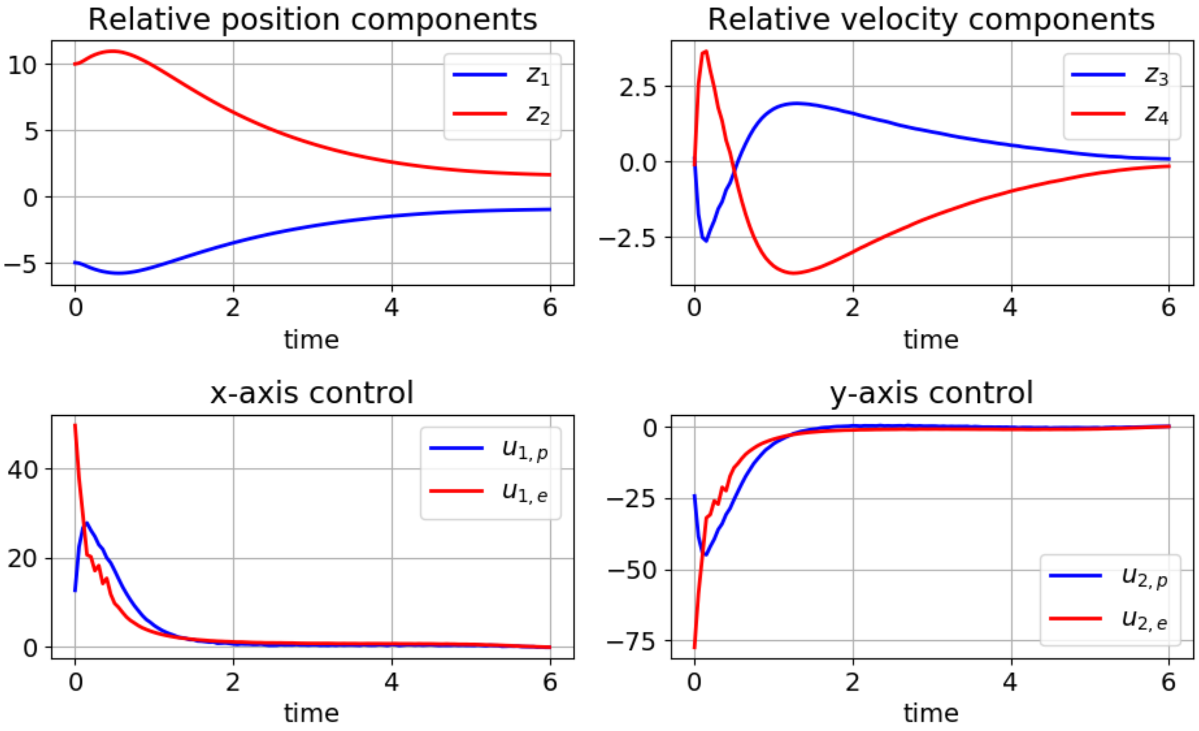

Figure 3.5: Simulation states for the adaptive evader with static pursuer

different from the static case. In examining the estimation errors, it is clear that the estimates are more biased than in the OSO cases. Recalling that the adaptive FNE strategy represents a truncation of a complete model, this bias should be expected. It appears that the pursuer estimates are more biased than the evader estimates. The evader also appears to outperform the pursuer in terms of its objective function. The FNE pursuer's cost is higher than the baseline, but better than the OSO pursuer cost. The maximum amplitude of the relative cost-to-go for the FNE pursuer is also much lower than in the OSO engagement. It can be speculated that the evader's estimated parameters, as in the OSO case, are more observable than the pursuer's. The FNE evader achieves a total cost that is essentially equivalent to the baseline cost. There is a small initial transient during which the FNE evader outperforms the baseline case, lasting until about $\tau = 1.7$. Afterward, the FNE evader cost-to-go is higher than the baseline. The overall interpretation of the cost-to-go changes can be summarized as follows. While playing against an adaptive pursuer, the adaptive FNE evader is

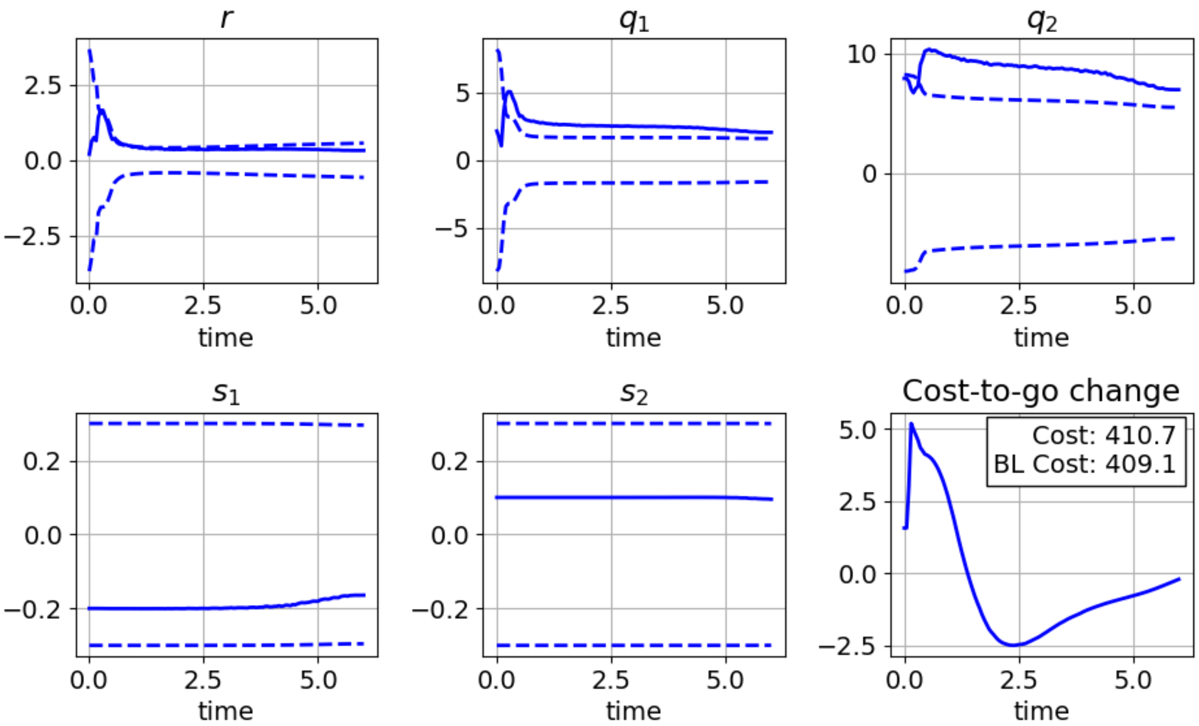

Figure 3.6: Pursuer estimation history for the two-sided finite game

able to recover her reference cost. The adaptive FNE pursuer performs slightly worse than in the baseline case. The adaptive pursuer incurs a large initial penalty, and if penalty could be overcome, performance would be improved.

The difference in performance between the FNE pursuer and evader may be explained partially by the relative observability of the uncertain parameters. It has been shown in the OSO engagements that the evader has better observability of the pursuer's weights than the reverse. This trend appears to continue in the FNE engagement as well. If so, then the evader's state estimation error should be reduced more quickly than the pursuer's, which would lead the evader to adapt more quickly than the pursuer. This is one possible explanation for the relative performance of the two agents.

As a comparison of all four cases, Fig. 3.8 shows the trajectories in the relative reference frame. The initial position is indicated by an open circle and has the same value for all three

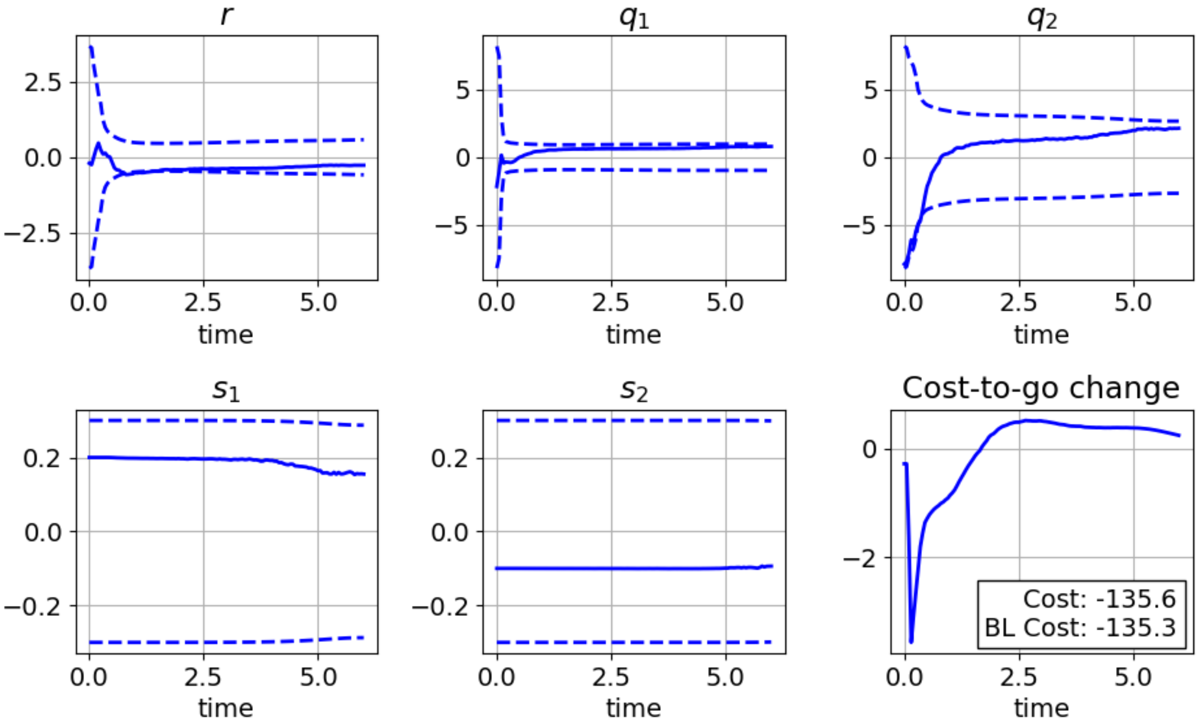

Figure 3.7: Evader estimation history for the two-sided finite game

cases. The terminal position is indicated by an "X" and varies slightly. From examination, the static case, in which no agent is adaptive, is most similar to the two-sided adaptive case. This reinforces the idea that the two-sided game is essentially recovering the static case - neither agent gains a substantial advantage. The one-sided engagements have similar terminal states but different underlying trajectories. The engagement with OSO evader in particular has the most distinctive trajectory, with the initial range between agents initially increasing, and a noticeable amount of "cross-track" motion orthogonal to the nominal capture trajectory.

The individual simulation results presented in this subsection demonstrate two primary trends. One, the asymmetry in the control weights used by each player makes the adaptive evader's estimation performance much better than the adaptive pursuer's in general. Two, there is a large transient effect on performance due to the initial uncertainty. Once that transient effect passes, the adaptive evader typically outperforms his baseline, and the adaptive pursuer's performance improves little

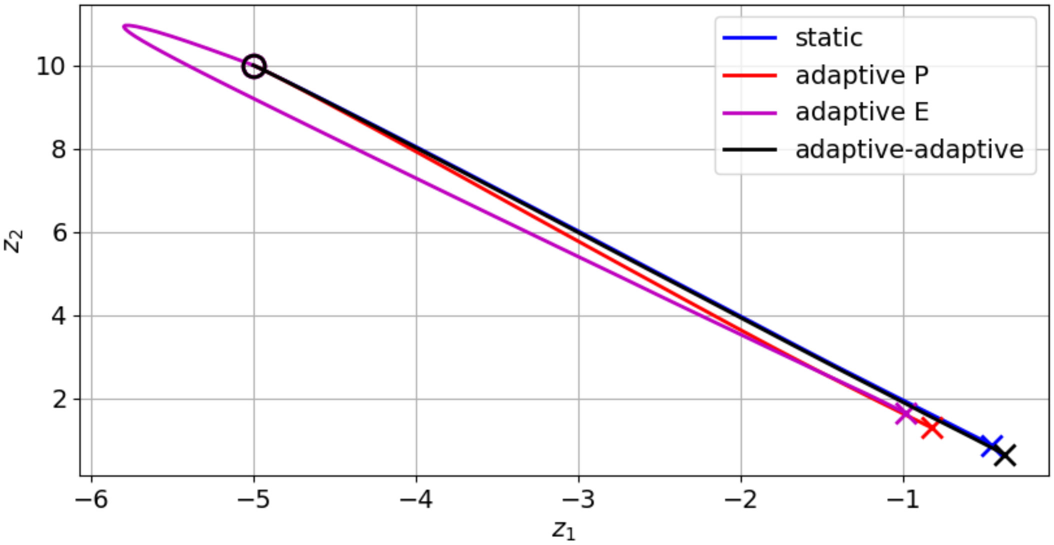

Figure 3.8: Trajectory comparisons for all four finite horizon cases

because his estimation performance is poor.

This subsection has considered the finite horizon LQDG engagement between various adaptive agents. The following subsection performs a small Monte Carlo analysis to analyze the behavior of adaptive agents playing against static agents with progressively poorer information.

3.4.2 Convergence of adaptation

As noted previously, the adaptive FNE approach is mismatched if both players use it simultaneously, because there is an error between the assumptions of the model and the actual approach used by the opponent. In practice, the effect of this mismatch is that each player obtains estimates of the opponent weights that are slowly time-varying, in such a fashion that errors in the estimate of the closed-loop opponent gain remain small. Since the weight estimates are time-varying and no longer unique, it is challenging to analyze the equilibrium properties of these strategies. Indeed, in this dissertation it is assumed that the mismatched adaptive FNE strategies achieve asymptotically a Nash equilibrium. To justify this assumption, in this subsection an adaptive FNE player plays against an opponent who implements a non-adaptive FNE strategy with some error.

The idea is a simple one; under perfect information, a Nash equilibrium exists at particular weight values. When the opponent has perfect information, the adaptive player should approach the FNE performance with no bias. As the opponent's information becomes less accurate, the adaptive player should perform no worse than in the exact FNE engagement. While this behavior does not prove the mismatched strategies are in equilibrium, it does demonstrate that the adaptive strategy is achieving the goal of improving performance in the presence of opponent errors.

To analyze this performance, an opponent information metric is defined. Monte Carlo simulation with fixed weights and initial conditions are conducted as the opponent information degrades from perfect information to minimax information. The distribution of the adaptive player's total cost incurred as a function of the opponent information quality is then considered. To analyze the distributions, a sort of modified box-whisker plot is employed to characterize distributions as a function of an independent variable. A box-whisker plot is a traditional statistical tool for visualizing a distribution of data. To visualize the distribution as a function of time, the traditional box-whisker components are simply plotted as functions of time. The approach is briefly summarized. The experimental mean and median are plotted as measures of the center of the distribution. The upper and lower quartiles[4] are plotted to give some idea of where the data are concentrated. (Half the data fall within these bounds.) Finally, the "whiskers" commonly used in box-whisker plots are shown to give some idea of the spread within the data. The upper whisker corresponds to the last value less than the upper quartile plus 1.5 times the interquartile range (the difference in quartiles). Similarly, the lower whisker corresponds to the last value greater than the lower quartile minus 1.5 times the interquartile range[47]. Qualitatively, the "box" values indicate where the data is concentrated and whether there is any asymmetry in the distribution. The "whisker" values indicate the extrema of the data and are intended to discard outliers. The resulting plots are useful for observing transient effects on the data distribution and convergence to asymptotic limits.

To perform this analysis, let an opponent information metric be defined as a value on $[0, 1]$. The opponent's model of the adaptive player's weights are obtained by linearly interpolating between

[4] Corresponding to the 25th and 75th percentiles, respectively

the true values and the zero-sum values. An information metric of 0 indicates the opponent has perfect information and 1 indicates the opponent uses the zero-sum weights and has no information. To evaluate performance, fixed values of the initial conditions and weights corresponding to Tables 3.2 and 3.3 are used and the resulting objective function value obtained by the adaptive player is evaluated across 100 simulations. Both adaptive pursuer and adaptive evader are considered. Values of the information metric corresponding to $0, 0.2, 0.4, 0.6, 0.8, 1.0$ are used.

Fig. 3.9 shows the distribution of the objective function value for the adaptive players across 100 simulations. Overall, these figures show the distribution of the overall cost as a functions of the opponent information quality. It is clear from the figures that the performance of the adaptive FNE player improves as the opponent's information quality deteriorates. The change in the overall cost is greater than the extent of the whiskers for the perfect information case, which indicates that the extent of the improvement is statistically significant. The change in the objective function is larger for the pursuer than for the evader. Additionally, the adaptive evader appears to experience diminishing returns as the opponent information approaches the ZS case, while the distribution for the adaptive pursuer is relatively linear. Both the change in the objective function and the curvature of the plots are functions of the weights used by both players, and the trends noted are not expected to generalize to all possible weights.

The results of this section are not sufficient to demonstrate global properties of the adaptive FNE strategy. What has been shown is that, for particular values of the opponent weights, an adaptive player consistently improves her objective function value as the opponent's information is eroded from perfect information to no information. This finding indicates that the adaptive strategy is achieving some improvement in reaction to errors in the opponent's knowledge, which is the minimum that should be expected of any useful adaptive strategy. Thus, while this subsection has not demonstrated optimality, it has validated that the adaptive LQDG play meets a basic requirement to be useful. The next subsection presents a Monte Carlo analysis of the orbital PE game.

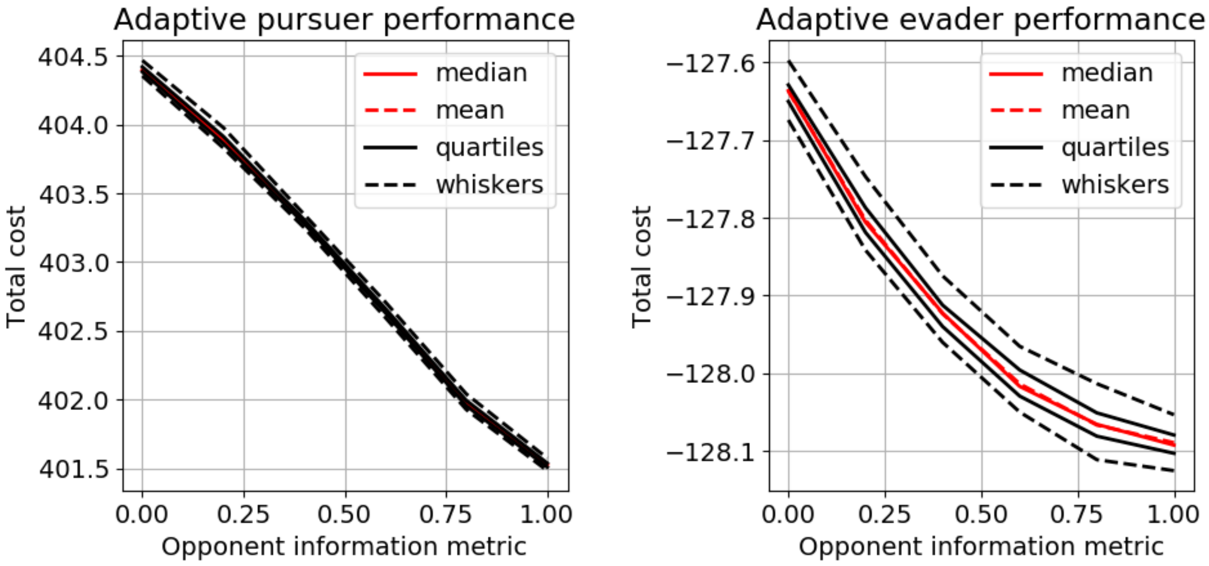

Figure 3.9: Performance of adaptive FNE player against a static FNE player with deteriorating information

3.4.3 Monte Carlo analysis

In Monte Carlo analysis, initial states and/or player weights are sampled from some assumed distributions and the resulting performance of the agents is analyzed. When initial states are sampled, all values are sampled independently. Initial positions are drawn from a uniform distribution on $[-5, 5]$ and initial velocity from a uniform distribution on $[-2, 2]$. When player weights are sampled, it is found that the generalized RDEs become unstable if sampled blindly from a Gaussian density of modest covariance, and so some additional sampling logic must be employed. Player weights are sampled from a Gaussian distribution whose mean is the true value of the opponent weight and whose covariance is one-tenth the value assumed in Table 3.2. If the weights satisfy stable RDEs, then the weights are used. If the weights lead to singular solutions, then the weights are re-sampled until stable values are found. Stability is evaluated by numerically integrating the RDEs backward from the terminal time. If any element of the associated RDEs exceeds a numerical threshold, the engagement is identified as unstable. Note that the value of the sample covariance is reduced from the initial covariance assumed by the adaptive player because larger values were found to sample unstable engagements at an exceedingly high rate.

Monte Carlo simulations are evaluated according to three metrics. First is the number of simulations that terminate prematurely. This can happen for several reasons, such as a large estimate covariance magnitude, indicating divergence of the estimator, or estimated opponent weights that lead to a singular RDE solution and unimplementable control. In general, the number of premature terminations is an indicator of how robust an estimator performs in a given circumstance.

The second metric is called the agent's loss function. An agent's loss function $\mathcal{L}_i$ and normalized loss function $\ell_i(t)$ are defined as functions of time as follows:

$$\mathcal{L}_i(t) = J^*_{c2g,i}(t) - J_{c2g,i}(t) \tag{3.17}$$

$$\ell_i(t) \equiv \frac{J^*_{c2g,i}(t) - J_{c2g,i}(t)}{\|J^*_{c2g,i}(t)\|} = \frac{\mathcal{L}_i(t)}{\|J^*_{c2g,i}(t)\|} \tag{3.18}$$

$J^*_{c2g,i}$ denotes the perfect information cost-to-go for agent i and $J_{c2g,i}$ denotes the cost-to-go based on the actual cost incurred during the simulation. The perfect information cost-to-go depends on both agent's strategies and is defined as follows. For an adaptive OSO agent playing against a minimax agent, the OSO agent's perfect information cost-to-go is computed by using the true values of the minimax agent's weights. The minimax agent's perfect information cost-to-go is found by computing the optimal RDE solution for the minimax agent when the adaptive OSO agent uses his perfect information strategy[5]. For the two-sided FNE case, both agents' optimal costs-to-go are found from the perfect information FNE game. The perfect information costs-to-go are computed from the solutions of the associated RDEs and the true state history from simulation with imperfect information. From the definition of the loss function, its nominal value should generally be negative; if an opponent plays optimally, then the best an agent can hope to do is to approach a zero value of the loss function. In practice, the loss function can be positive when both players play according to imperfect information, and this occurs in many simulations. This loss function is defined using the cost-to-go rather than total accrued cost to make the metric less

[5] Hence, the perfect information "minimax" agent strategy is not actually minimax.

sensitive to transient effects. The loss function is analyzed both in terms of the total cost-to-go, evaluated at the initial time, and also in terms of the time histories of the loss function. For the time histories, the distribution plots introduced in Sec. 3.4.2 are employed to consider the loss function distributions as functions of time.

The third and final metric for evaluating Monte Carlo results is the mean square error (MSE) in estimating the opponent's gain. It is expected that estimation of the opponent weights incurs a bias when both players are adaptive, as discussed previously. As a result, the accuracy in estimating the opponent weights is not a good metric of the performance of the estimator. Instead, it is more useful to consider the accuracy in predicting the opponent gain as the simulation develops. The MSE is computed for each simulation across all times and gain elements as follows:

$$MSE_K = \frac{1}{8N} \sum_{i=1}^{4} \sum_{j=1}^{2} \sum_{k=1}^{N} (\hat{K}_{ij}(t_k) - K_{ij}(t_k))^2 \tag{3.19}$$

Monte Carlo results are presented in three parts. First, constant initial conditions and agent weights are selected and the resulting aggregate performance is analyzed. It is useful to conduct simulations for a constant set of initial conditions to gain insight into the behavior of each player's objective function. The objective function is in general more sensitive to the initial state than to individual parameter uncertainty. In the second part, the initial conditions are allowed to vary according to a uniform distribution, and the resulting performance is analyzed. Finally, the problem weights vary with constant initial conditions.

3.4.3.1 Baseline case with fixed initial states and weights

As a baseline case, 100 simulations are conducted with fixed problem weights and initial conditions. The only source of uncertainty is in the measurement errors and the resulting influence on the estimation and control of each agent. Figs. 3.10–3.11 plot the distribution of the loss function as a function of time for each of the three simulation cases. These figures plot $\mathcal{L}_i$, the unnormalized cost-to-go loss, as a function of time. Fig. 3.10 plots the loss metric for OSO evader and pursuer. The distribution plots are of the same type as Fig. 3.9, in which the median, mean, upper and

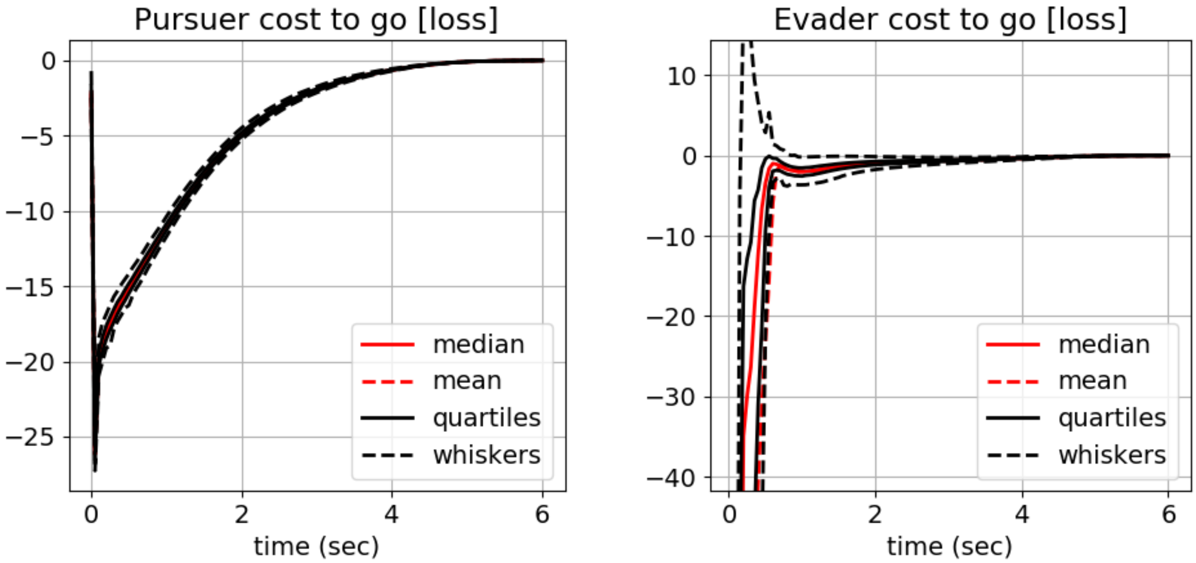

Figure 3.10: Distribution plot of the loss function for OSO evader (left) and OSO pursuer (right)

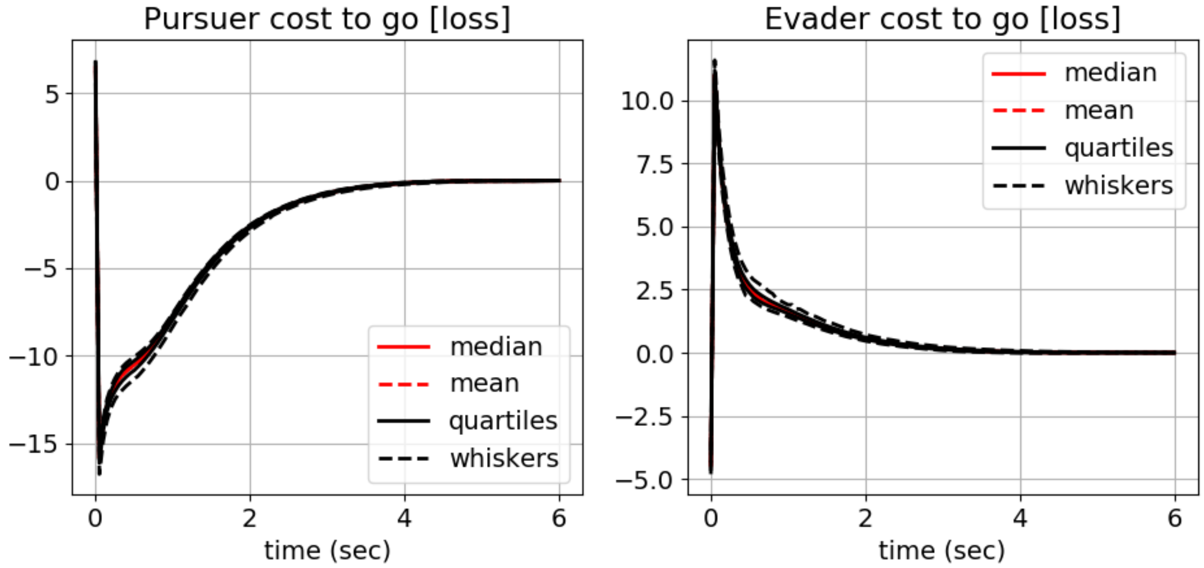

Figure 3.11: Distribution plot of the loss function for opposing adaptive pursuer and evader

lower quartiles, and whiskers are plotted to give a sense of the overall distribution of the data as a function of time. These plots show the same general trends in cost-to-go as were observed in the individual simulation results. The pursuer's initial loss is small but quickly spikes to a large negative value. The cause of the spike is a rapid increase in the adaptive cost-to-go, apparently caused by a transient response when the adaptive gameplay begins. The OSO evader also initially suffers a large loss, corresponding to the poor transient response when the uncertainty is high. The transient lasts approximately three-quarters of a time unit. After that point, the distribution including the whiskers are approximately bounded above by zero, and the entire distribution approaches zero cost-to-go uniformly as $\tau \to \tau_f$. Noticeably, the adaptive evader approaches zero loss more quickly than the adaptive pursuer once the initial transient is overcome. Considering the whiskers, the OSO pursuer has much lower variation in performance than the OSO evader. The performance of the OSO pursuer's loss function appears to be close to deterministic, with only small changes in loss experienced due to measurement sampling effects.

Fig. 3.11 shows the loss metric as a function of time when both players use the adaptive FNE approach. The results suggest the two-sided adaptive FNE game is nearly zero-sum, in that the pursuer and evader loss functions are nearly mirrored about the x-axis. The curves are not identical, however. At the initial time, the pursuer has a positive loss and the evader has a negative value, indicating that the pursuer cost-to-go at $\tau = 0$ is lower than the perfect information baseline value, and the evader cost-to-go is higher. After an initial transient, the trend is reversed, with the pursuer metric indicating worse performance than the baseline and the evader performing better. This result is similar to what was shown previously in Sec. 3.4.1 for the same weights and initial conditions.

Tables 3.4–3.5 report the remaining performance metrics. Table 3.4 reports the distribution of the relative loss function, ℓ_i, evaluated at the initial time. This metric quantifies the overall cost incurred under adaptive play relative to the perfect information baseline. The median and the upper and lower quartiles are reported to give some idea of the spread within the data. The loss metrics are reported for the static agents as well as adaptive ones. The results indicate that the adaptive pursuer against minimax evader is able to nearly recover perfect information play, having a loss metric typically between -0.6% and -0.4%. Because of the adaptive pursuer, the minimax evader incurs a loss of about -6% generally. As noted earlier, the OSO adaptive evader incurs

Agent	Adaptive?	Lower quartile	Median	Upper quartile
Pursuer	✓	-0.0063	-0.0052	-0.0041
Evader		-0.060	-0.059	-0.059
Pursuer		0.54	0.59	0.70
Evader	✓	-2.7	-2.3	-2.1
Pursuer	✓	0.015	0.016	0.016
Evader	✓	-0.035	-0.034	-0.032

Table 3.4: Summary of cost function loss statistics in 100 simulations with sampled weights

a huge penalty due to the initial transient. This transient skews the OSO evader's relative loss function to values of about -200% of the baseline. Because of this highly suboptimal initial play, the minimax pursuer actually attains better performance than he should expect to against a static player. In the two-sided adaptive FNE scenario, the relative loss distributions are very tight. The pursuer typically improves by about 1.6% relative to his baseline, and the evader metric is about 3.4% worse than the baseline. As the distribution plots show, the improvement in cost-to-go is highly dependent on the transient response, and a small change in either player's initial response could easily change the relative loss metric from positive to negative or vice versa. The overall picture can be summarized as follows: the OSO pursuer essentially recovers perfect information gameplay. The OSO evader incurs a large penalty due to suboptimal play when the uncertainty is large. In the two-sided game, the pursuer plays a little better and the evader a little worse than the baseline case.

Table 3.5 shows metrics of the MSE in opponent gain estimation distribution. The MSE metric excludes simulations that terminated prematurely. The mean, median, and quartile values of the MSE across different simulations is then reported in Table 3.5. The results indicate that the OSO pursuer obtains estimates of the minimax evader's gains with a typical MSE that is about one-third of the MSE of an OSO evader estimating a minimax pursuer's gain. It is interesting that the pursuer MSE is smaller than the evader MSE, given the lower estimation errors shown for the evader than for the pursuer. However, the evader gain is generally smaller than the pursuer gain, so it is reasonable that the gain MSE is lower for the pursuer estimating the evader's gain. In the

Agent	Lower quartile	Median	Mean	Upper quartile
Adaptive Pursuer	0.014	0.017	0.017	0.021
Adaptive Evader	0.049	0.051	0.054	0.056
Adaptive Pursuer	0.025	0.041	0.039	0.053
Adaptive Evader	0.31	0.46	0.47	0.66

Table 3.5: Summary of opponent gain estimation mean squared error statistics in 100 simulations with sampled weights

adaptive FNE engagement, both player's MSE metrics deteriorate, but the evader's error becomes much larger than the pursuer's. It is reasonable to conclude that the larger loss metric experienced by the evader in this case can be attributed in part to this larger error in gain estimation.

For the case of fixed initial states and weights, there are no failed simulations in the OSO simulations. There are only four failed simulations in the two-sided FNE simulations. Based on this result, the robustness of the agents to measurement uncertainty alone is acceptable.

Baseline Monte Carlo results have been presented. Next, the effect of sampled weights with constant initial conditions is considered.

3.4.3.2 Sampled weights with constant initial conditions

In the following discussion, the weights used by each player are sampled according to a nominally normal distribution. The distribution is only nominally normal because some sampled values correspond to games for which there is no finite-valued RDE solution, and these values are discarded and re-sampled. No consideration is made of the "distance" from the sampled values to the point of instability. For instance, if chosen weights are "close," in the space of parameters, to unstable weights, then small estimation errors may cause the estimated RDE solution to become singular even when the perfect information solution is nonsingular. Hence, a greater number of failed simulations are to be expected under this new sampling paradigm.

1000 Monte Carlo samples are used to evaluate performance, using the same metrics introduced previously. Fig. 3.12 shows the loss function for the OSO evader and pursuer. The OSO evader loss function shows a huge initial variance at $\tau = 0$. After about half a dozen measurements, the

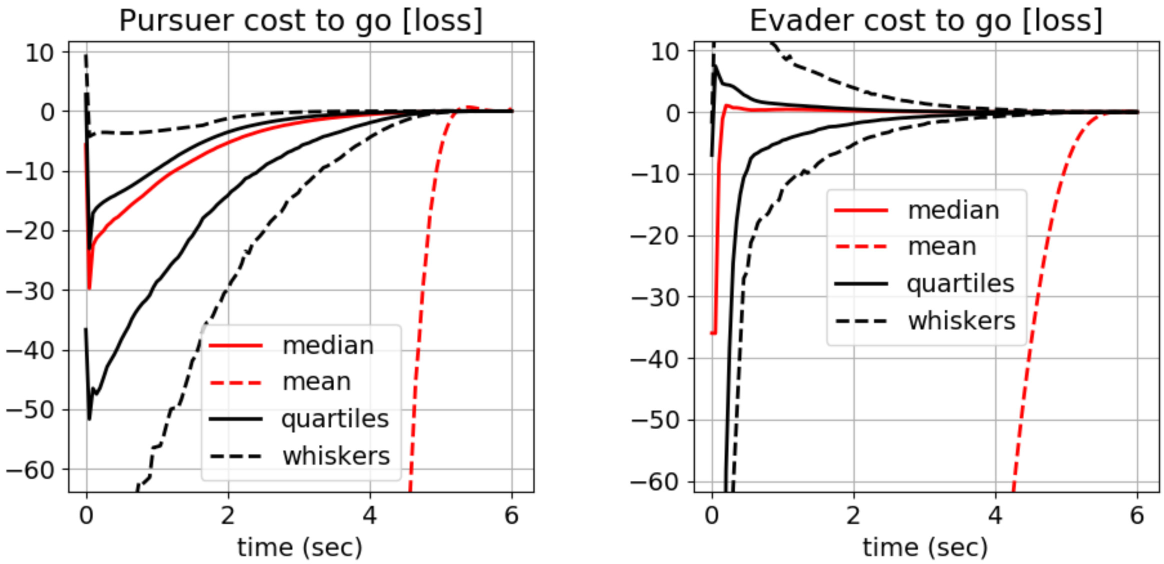

Figure 3.12: Distribution plot of the loss function in 1000 simulations with sampled weights for OSO evader (left) and OSO pursuer (right)

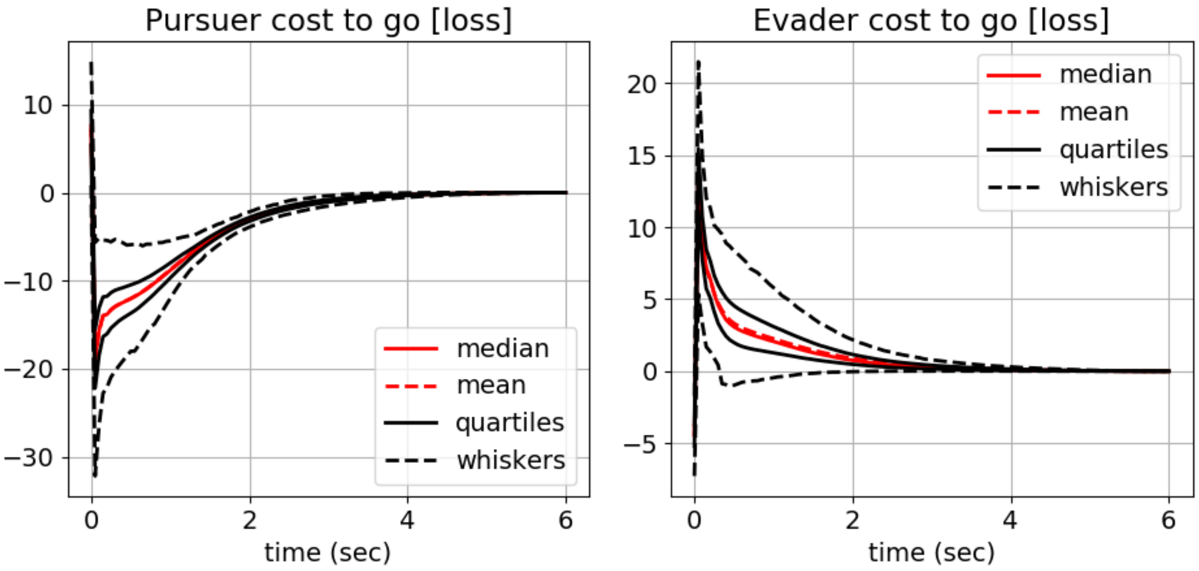

Figure 3.13: Distribution plot of the loss function in 1000 simulations with sampled weights for opposing adaptive pursuer and evader

Agent	Adaptive?	Lower quartile	Median	Upper quartile
Pursuer	✓	-0.092	-0.013	0.0073
Evader		-0.11	-0.041	0.10
Pursuer		0.018	0.066	1.2
Evader	✓	-2.9	-0.33	-0.095
Pursuer	✓	0.014	0.017	0.021
Evader	✓	-0.045	-0.034	-0.027

Table 3.6: Summary of cost function loss statistics in 1000 simulations with sampled weights

median loss value approaches zero, but the upper quartile remains noticeably larger for about one time unit. The upper whisker decays even more slowly, and cannot be said to have converged until about $\tau = 4$. The mean loss metric is very skewed and is not visible on the plot for most of the simulation time. The OSO pursuer also shows a large initial variance in the loss, but the whiskers are tighter and are generally negative-valued. The median value is close to the upper quartile. The distributions converge to near zero by approximately $\tau = 4$. Convergence of the distribution to zero indicates there is little that can be gained by additional adaptation or estimation from this time onward. Overall, the distributions show similar median responses to the baseline case, but huge growth variation in performance due to the sampled weights. Based on the performance in individual simulations, it is likely that this increase in variability is attributable to poor transient responses due to initial errors.

Fig. 3.13 shows the loss function distributions for the two-sided adaptive FNE game. The results are comparable to those of Fig. 3.11 but with much wider uncertainty in each distribution. However, the variability in the two-sided adaptive FNE game is much less than was displayed in the adaptive OSO games with sampled weights. The whiskers converge to values near the quartiles by approximately $\tau = 1$, and the overall loss distribution converges to near zero by about $\tau = 3$ or $\tau = 4$. The plots indicate that, after suffering a penalty due to the initial conditions, the adaptive evader typically enjoys an improvement in the cost-to-go at subsequent times.

Tables 3.6–3.8 summarize the remaining performance metrics. Table 3.6 shows the relative loss metric. It manifests similar general trends to the baseline case, but with much greater variance

Agent	Lower quartile	Median	Mean	Upper quartile
Adaptive Pursuer	0.011	0.018	0.028	0.032
Adaptive Evader	0.069	0.095	0.12	0.13
Adaptive Pursuer	0.011	0.016	0.021	0.025
Adaptive Evader	0.064	0.086	0.095	0.11

Table 3.7: Summary of opponent gain estimation mean squared error statistics in 1000 simulations with sampled weights

between the upper and lower quartiles. The median OSO pursuer loss is about -1.3% but the interquartile range is about 9.9%, roughly two orders of magnitude larger than in the baseline case. The introduction of sampling improves the performance of the OSO evader substantially. Although the lower quartile value is still on the order of several times the baseline value, the median and upper quartile values are on the order of tens of percents. In the adaptive FNE game, the pursuer relative loss metric is typically between 1.4% and 2.1%, and the evader metric between -4.5% and -2.7%. The results in this table generally follow the same trends as the baseline case but with much greater uncertainty. It remains true that OSO pursuers typically are able to attain similar performance to their perfect information counterparts, and initial-condition-based transient errors degrade OSO evader performance. In the adaptive FNE case, pursuers typically are able to improve over their baseline, and evader experience a loss.

Table 3.7 summarizes the MSE gain estimation error experienced across simulations. Pursuers achieve errors on the order of 0.01 in both the OSO and FNE engagements. The evader errors are typically larger while being of the same order of magnitude; the upper quartile is on the order of 0.10. Both players now achieve slightly smaller errors in the FNE case than in the OSO case. This is a reversal of the trend seen in the baseline engagement, although it is not clear if the change in the errors is statistically significant. With sampled player weights and fixed initial conditions, the MSE in gain estimation is almost indistinguishable whether the OSO or FNE strategy is used.

Lastly, Table 3.8 summarizes the number of failed simulations in each case. About 10% of pursuer OSO simulations failed to complete, while the number is much higher in the other cases. About 40% of evader OSO engagements failed, and about 25% of adaptive FNE engagements

Agent	Failed simulations
Adaptive Pursuer	114
Adaptive Evader	395
Adaptive Pursuer and Evader	258

Table 3.8: Summary of failed simulations in 1000 simulations with sampled weights

failed. It is clear that the overall estimator robustness is highly sensitive to the game weights. The performance demonstrated here is likely to be unacceptable in most practical systems. Motivated by this finding, improving the robustness of the adaptive gameplay by quantifying the "stability bounds" associated with the LQDG game weights is a focus of Chapter 5.

This completes the description of results for sampled weights with constant initial conditions. Next, sampled initial conditions with constant agent weights are considered.

3.4.3.3 Sampled initial conditions with constant weights

In this section, initial conditions are chosen such that the initial position components are from $\mathcal{U}(-5, 5)$ and the initial velocity components from $\mathcal{U}(-2, 2)$. All relative state components are independent. Latin Hypercube sampling is used to ensure the sampled values are spread throughout the uniform domain[48, 49][6]. The agent weights are the constant values specified in Table 3.3. As in the previous set of Monte Carlo results, 1000 sampled values are chosen.

Fig. 3.14 shows the OSO evader and pursuer cost-to-go loss distributions. The evader loss metric, as before, manifests very large initial uncertainty that typically leads to a large penalty. By $\tau = 1$, much of the uncertainty has been reduced, but the distribution does not fully converge until about $\tau = 4$. The large uncertainty indicates the OSO evader is quite unreliable; sometimes the performance is much worse than the baseline, and sometimes it is better. The OSO pursuer behavior, on the other hand, shows relatively little variability. The whisker height is much smaller during the initial transient. The median loss values are small, on the order of 1, and the distribution converges steadily as $\tau \to \tau_f$. The OSO pursuer loss is consistently small.

[6] Note that Latin Hypercube designs were *not* used previously because the engagement stability with respect to weights made the samples potentially dependent.

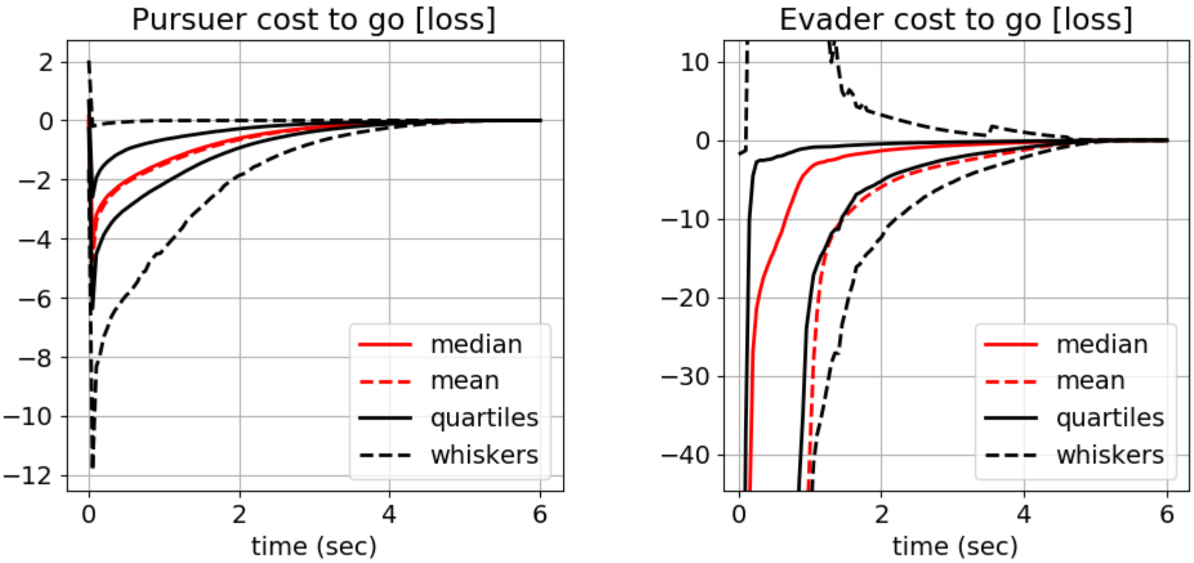

Figure 3.14: Distribution plot of the loss function in 1000 simulations with sampled initial conditions for OSO evader (left) and OSO pursuer (right)

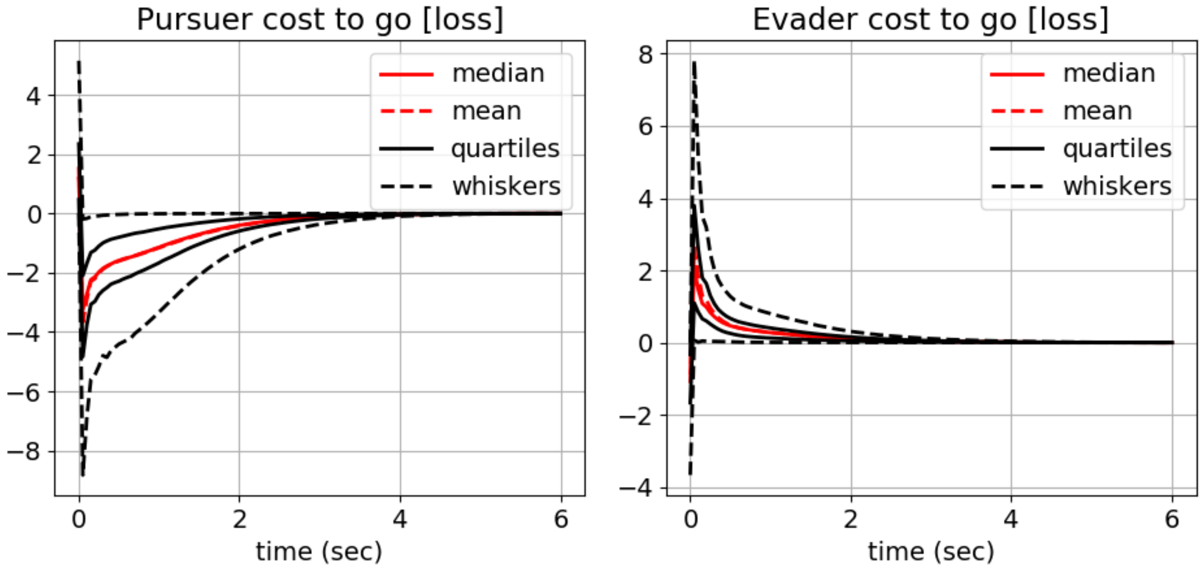

Figure 3.15: Distribution plot of the loss function in 1000 simulations with sampled initial conditions for opposing adaptive pursuer and evader

Agent	Adaptive?	Lower quartile	Median	Upper quartile
Pursuer	✓	-0.0050	0.0036	0.014
Evader		-0.12	-0.085	-0.030
Pursuer		0.83	2.5	14
Evader	✓	-42	-7.5	-2.8
Pursuer	✓	0.013	0.027	0.039
Evader	✓	-0.064	-0.050	-0.029

Table 3.9: Summary of cost function loss statistics in 1000 simulations with sampled initial conditions

Fig. 3.15 shows the loss function distributions in the adaptive FNE game. The evader distribution is initially quite sharply peaked before decaying gradually from a postive value. After the initial timestep, the evader loss whiskers appear to be bounded below by zero. The pursuer transient response is less peaked, and gradually decays from a small negative value to zero. By comparison with the OSO evader distribution, the adaptive FNE evader distribution is much tighter and more consistent. It appears that in the two-sided FNE engagement, there is less overall variability in either agent's cost. This phenomenon was also observed in the case with sampled weights and constant initial states.

Tables 3.9–3.11 summarize the loss function, gain estimation MSE, and failed simulations with sampled initial states. First, consider Table 3.9. In the game with OSO pursuer, the pursuer loss values are typically near zero. The upper quartile is about 1.4%, indicating better performance than the baseline. The minimax evader in the same simulations experiences loss values typically on the order of one to ten percent. These results suggest a significant degradation in evader cost, but minimal change in pursuer cost due to the adaptive pursuer play. In the OSO evader game, the initial conditions skew the results and lead to poor performance for the evader. In the two-sided FNE game, the pusuer typically achieves small positive values and the evader achieves small negative values. All of these trends are consistent with the other Monte Carlo cases. The uncertainty has increased, as measured by the wider interquartile range for both pursuer and evader. Overall the findings still indicate an advantage for the pursuer in the FNE game, while the evader is struggling to "keep up." However, it is also clear that this outcome is a function of the initial guesses for each

Agent	Lower quartile	Median	Mean	Upper quartile
Adaptive Pursuer	0.011	0.013	0.014	0.015
Adaptive Evader	0.074	0.088	0.096	0.11
Adaptive Pursuer	0.014	0.016	0.019	0.020
Adaptive Evader	0.040	0.051	0.061	0.071

Table 3.10: Summary of opponent gain estimation mean squared error statistics in 1000 simulations with sampled initial conditions

agent, and that the cost-to-go at subsequent times in the simulation usually is better for the evader, once its estimates have begun to converge.

Table 3.10 summarizes the gain estimation MSE statistics. The OSO pursuer MSE values are on the order of 0.01 and have a very small interquartile range, indicating very consistent performance across simulations. The OSO evader has larger errors of the same magnitude and an interquartile range that is about an order of magnitude larger. Essentially, the OSO evader has slightly higher typical errors but significantly larger variance in performance. In the adaptive FNE game, the pursuer MSE median errors increase slightly and the evader median errors descrease noticeably. The change in pursuer MSE is small and may not be statistically significant, but the evader MSE appears to be statistically significant based on the quartile values. It is interesting that the error metric increases for the pursuer in the FNE game and descreases for the evader. This may be related to the initial weight estimates, which are unfavorable to the evader but appear to be less unfavorable in the FNE game.

Lastly, Table 3.11 summarizes the number of failed simulations in each case. Essentially, only the OSO evader engagement has any significant number of failures, with 136 out of 1000. It should be noted that this is significantly less than occured in the case of sampled weights only, but still a large fraction of simulations. This finding indicates that the OSO evader is particularly fragile compared to the OSO pursuer.

The Monte Carlo results demonstrate four clear trends:

1. The OSO adaptive pursuer performance typically approaches that of the perfect information baseline case. While the estimator is robust to initial state changes, the OSO pursuer still

Agent	Failed simulations
Adaptive Pursuer	0
Adaptive Evader	136
Adaptive Pursuer and Evader	6

Table 3.11: Summary of failed simulations in 1000 simulations with sampled initial conditions

fails in a large number of simulations when sampled weights are used.

2. The OSO adaptive evader performance is heavily skewed by a poor initial transient response, which is mitigated slightly when sampled weights are used. The adaptive evader exhibits high uncertainty, long after the initial time, and extremely poor robustness to sampled values of the pursuer's weights.

3. In adaptive FNE games, the pursuer consistently improves over his baseline by a small amount, while the evader suffers overall relative losses of about -5% typically. This performance metric is highly dependent on the initial conditions. It appears that after overcoming an initial transient, the adaptive evader's performance typically improves more than the adaptive pursuer's does.

4. Despite estimator fragility with respect to the agent weights, the performance and MSE error metrics are seen to be more sensitive to the initial state than to the values of the opponent weights. It seems like that this happens because a large number of simulations with sampled weights terminated prematurely, which has biased the metrics of the remaining simulations towards better values.

Given the performance demonstrated thus far, it seems reasonable to expect that sampled initial states and agent weights simultaneously would produce a large number of prematurely terminated simulations, and are unlikely to produce new trends in terms of agents' performance. Hence, for the sake of brevity, the discussion of Monte Carlo results is limited to the results presented thus far. The next section summarizes the chapter and presents conclusions.

3.5 Conclusions

This chapter has studied the use of sequential estimation to enable adaptation in agents playing a nonzero-sum incomplete, imperfect information PE game. An agent plays against an opponent whose behavior is known to be either static or adaptive, and the appropriate Riccati equations are used to predict an opponent's future actions according to the best estimate of the opponent's objective function weights at each time. The linear relative dynamics of the HCW equations allow the use of the well-developed LQ PE game in the problem formulation.

There are four primary conclusions from the numerical examples. (1) Estimating the opponent's terminal state penalty, $[\hat{S}_f]$, poses an unusual challenge. $[\hat{S}_f]$ is not very observable in the sequential estimation framework, because the opponent's Riccati matrix is in a steady state for most of the simulation. However, the opponent weights overall can be estimated with lower uncertainty in the finite horizon game, because the presence of the terminal weight causes the opponent's behavior to deviate from the estimated models more significantly and improves observability. (2) The evader in this scenario generally gets a more precise estimate of the pursuer's weights than the pursuer does of the evader's. This is because the pursuer's greater control use makes his actions more observable. Likely because of this improved estimation accuracy, the evader typically achieves better cost-to-go metrics *after the effects of initial estimation errors have been overcome*. (3) To limit the effect of initial errors when the uncertainty is high, an agent should employ a "safe" strategy until his estimates converge, or take some other action to obtain smooth control corrections. Since this step is not employed in the chapter, most simulations show an initial transient in which a cost penalty is accrued. The tradeoff, naturally, is that waiting too long to begin using the estimates will reduce the potential benefits for an agent.

The results shown in this chapter suggest several avenues for further consideration. The numerical implementation of adaptive LQDG gameplay suffers from quite slow evaluation times, due to the need to repeatedly re-solve the generalized RDEs via numerical integration. This problem was present in the earlier work of Ref. [6], but is more prevalent in the adaptive FNE case, which requires each player to numerically integrate coupled RDEs in a UKF. Additionally, a rigorous framework for deciding when to begin using the estimates of the opponent's weights is needed to prevent the poor transient response present in many simulations. Efficiently approximating RDE solutions and safely switching to the adaptive strategy are both covered in Chapter 4. The Monte Carlo results with sampled weights and fixed initial conditions clearly demonstrate a fragility of the estimation process with respect to the weights. It is believed that many of these failed simulations can be attributed to nearly-unstable engagements in which the estimated weights stray into the unstable region, causing a numerically singular control to be computed and leading to the failure of the simulation. A method for more robustly estimating the opponent weights in the presence of engagement stability considerations is presented in Chapter 5. Another challenge not addressed in this chapter is how an adaptive agent might distinguish between a static and adaptive opponent. This area is discussed more in Chapter 6, which proposes an approach based on multiple-model estimation.

4. PARAMETERIZATION, PROBABILISTIC LOOP CLOSURE AND RICCATI SOLUTION APPROXIMATION

The previous chapter considers a detailed example of an orbital pursuit-evasion game in which agents use adaptive gameplay to try to improve their overall performance. While the estimation underlying adaptation shows reasonably good performance, the simulations demonstrated several shortcomings that are not addressed in the work of Ref. [6]. The present chapter focuses on improvements to the estimation-based adaptive gameplay, while using the same basic adaptive gameplay approach. In particular, three topics are addressed. First, the LQDG estimation problem parameterization is modified to produce identical equations from both agents' perspectives. This change is made primarily for convenience in the numerical implementation of adaptive gameplay. Second, a rigorous method for switching from the minimax strategy to adaptive strategy based on the estimation uncertainty is introduced and used in all future results. Finally, a method for approximating neighboring solutions of the generalized RDEs using the Taylor series is introduced and demonstrated. To begin, the problem re-parameterization is introduced.

4.1 Re-parameterization

In the work of Ref. [6] and the orbital PE game of Chapter 3, the pursuer is designated as player i and $[R_{ii}] = [R_{ji}] = [\mathbb{I}_{n\times n}]$ is used. There is no loss of generality, and this approach is reasonable for the minimax and OSO games. However, a different choice of parameterization leads to more compact expressions of the FNE governing equations. Choosing instead $[R_{ii}] = [R_{jj}] = [\mathbb{I}_{n\times n}]$ eliminates matrix inverse terms in the generalized RDEs. Furthermore, it produces a useful symmetry between agents i and j. After simplification, the governing RDE for $[S_i]$ is given as follows:

$$[\dot{S}_i] = -[S_i][A]-[A]^T[S_i]-[Q_i]+[S_i][E_{ii}][S_i]+[S_i][E_{jj}][S_j]+[S_j][E_{jj}][S_i]-[S_j][E_{ij}][S_j] \quad (4.1)$$

$$[E_{ii}] \equiv [B_i][B_i]^T \quad (4.2)$$

$$[E_{jj}] \equiv [B_j][B_j]^T \tag{4.3}$$

$$[E_{ij}] \equiv [B_j][R_{ij}][B_j]^T \tag{4.4}$$

The governing FNE equation for $[S_j]$ is simply given by interchanging the indices i and j in the previous expressions. This assumption also simplifies the closed-loop dynamics such that i's closed-loop equation only depends explicitly on $[S_j(t)]$ and not on $[R_{jj}]$. In principle, any linear system can be transformed to the suggested form simply by a linear transformation on the variables. Furthermore, any objective function for i in which $[R_{ii}]$ is equal to a constant times identity can be transformed into this form without a change of variables by simply changing the relative scale of the problem weights. This change is significant for numerical implementation because it allows the same code to be used for both agents while retaining a compact problem representation.

The choice of $[R_{ii}] = [R_{jj}] = [\mathbb{I}_{n\times n}]$ has the undesirable effect of slightly changing the interpretation of the problem weights for both agents. For simplicity of the discussion, let i be the observing agent, whether in an OSO or FNE solution. If i is a pursuing agent, then he must have $[Q_i] \geq 0$ and $[S_{f,i}] \geq 0$ for the problem to be physically meaningful. Similarly, in a competitive game where j and i are opposing one another, $[R_{ij}] < 0$ must be satisfied. On the other hand, if i is an evader, then he must choose $[Q_i] \leq 0$ and $[S_{f,i}] \leq 0$, but the requirement $[R_{ij}] < 0$ remains. That is, under the altered problem representation, **two of the problem weights have a sign dependence on the role of the acting agent**. This should be kept in mind while reviewing the subsequent results.

This section has summarized a re-parameterization of the LQDG system that is more convenient for the FNE game. This system parameterization is used in all following work. The following section presents a straightforward procedure to prevent adaptive agents from acting according to highly uncertain estimates.

4.2 Cost-based loop closure

As noted in both Refs. [6] and [12], as well as Chapter 3, it is not generally appropriate to immediately use the learned values of an opponent's weight, because errors in the initial estimates may lead to an overall reduction in the game outcome. In both of those dissertations, an arbitrary observation window was specified by the control designer. During this period, the adaptive player used a minimax strategy. His estimates were assumed to converge when the observation window expired, and the estimates were used to generate subsequent values of the control. This approach is generally acceptable in practice, but has two disadvantages. First, it requires tuning for specific scenarios. Second, even with tuning, the convergence of the nonlinear estimates is dependent on the state trajectory, so an agent in practice could wait longer than necessary, or not long enough. Since nonlinear Kalman Filtering is used to obtain estimates, a straightforward statistical approximation can be used to determine when the uncertainty is sufficiently low to begin using estimates to generate control values.

At the beginning of a scenario, an agent has access only to his own weight values. Unless good a priori estimates of the opponent weights are known at initialization, the agent should implement a strategy that is independent of the opponent's settings, such as the minimax strategy. The minimax strategy ensures a "cost ceiling;" as discussed earlier, the cost-to-go with minimax weights always has an upper bound given as a quadratic function of the current state. That upper bound is independent of the opponent strategy. The adaptive agent wishes to observe the opponent, develop good estimates of the opponent weights, and exploit those estimates to achieve better performance than would otherwise be possible. This behavior can be achieved by stipulating that the adaptive agent should not switch to a candidate strategy unless that strategy is expected to result in a lower cost than the minimax cost ceiling.

The estimated values are assumed to be approximately Gaussian. The cost-to-go (C2G) under the OSO or FNE strategies is simply a nonlinear function of the estimates. By using the estimate means and covariances, the cost-to-go and its covariance can be approximated. Let the full state vector for estimation be $\hat{\boldsymbol{z}} \equiv \begin{bmatrix} \hat{\boldsymbol{x}}^T & \hat{\boldsymbol{\theta}}^T \end{bmatrix}^T$. The estimated cost-to-go for i if he acts using his state estimates is a simple function of his Riccati matrix solution:

$$\hat{J}_{C2G} = \frac{1}{2}\hat{\boldsymbol{x}}^T[S_i(\hat{\boldsymbol{\theta}})]\hat{\boldsymbol{x}} \tag{4.5}$$

The mean and covariance of $\hat{J}_{C2G}$ can be approximated to first order under the assumption that the state estimate $\hat{\boldsymbol{z}}$ is Gaussian. The Unscented Transform (UT) is employed to approximate the mean and covariance of the C2G[4, p. 155]. If it assumed that i correctly knows the strategy of his opponent, then the estimation errors are the only source of uncertainty in Eq. 4.5. It remains only to determine an appropriate threshold on $\hat{J}_{C2G}$ for which it is acceptable to begin using the estimates for closed-loop control.

A simple approach is to require a high probability that the true cost-to-go is less than the minimax ceiling. This can be achieved by requiring $Pr(J_{C2G} < J_{minimax}) > P_{\alpha}$, where P_{α} is some threshold near zero. $P_{\alpha} = 0.99$ is typically used in the dissertation. Both the cost-to-go and the minimax cost depend on the current state, and hence are random variables. Consequently, the UT is used to evaluate the mean and covariance of $J_{\delta} \equiv J_{C2G} - J_{minimax}$. If the higher moments of J_{δ} are assumed negligible, then $Pr(J_{\delta} < 0)$ is simply given by the cumulative distribution function (CDF) of the approximately Gaussian random scalar J_{δ}. The CDF, denoted by $F_{J_{\delta}}(0)$, is straightforward to evaluate, given the mean and covariance approximated by the UT.

To highlight the use of this cost-based initialization, return to the example of the orbital pursuit-evasion game. In Fig. 3.4, the evasive player incurs a huge initial penalty by acting according to the estimates of the pursuer's weights when the uncertainty is large. The same scenario is repeated in Fig. 4.1 using the cost-based initialization. In this plot, the evader is denoted as agent j, and has the objective of minimizing the cost function J_j. The evader transitions from using the minimax strategy to the one-sided optimal strategy after less than five measurements typically. The true adaptive gameplay cost (computed assuming perfect information) is shown on the left plot and is evaluated against the cost incurred for the same scenario with no adaptation. The adaptive cost smoothly departs from the static cost and achieves a lower overall value. The left plot shows the estimated and true values of the change in cost to go obtained by switching from the minimax strategy to the OSO strategy. The right plot shows the estimated and true change in the cost-to-go from switching to the adaptive OSO strategy. After about four measurements, the uncertainty is low enough that the 3σ bounds on the estimated cost-to-go change are negative, and the closed-loop

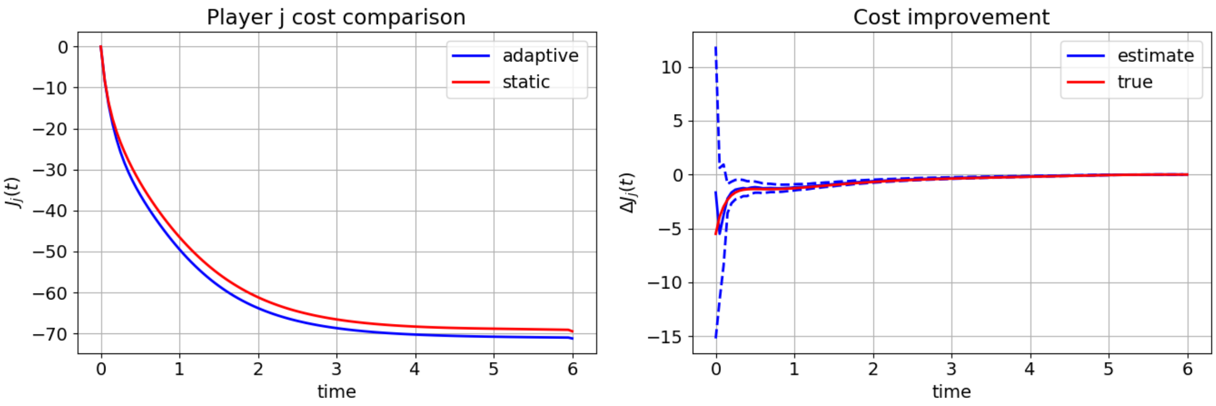

Figure 4.1: Cost improvements in evasive agent for orbital pursuit-evasion game

control is used. This figure shows that the cost-based initialization eliminates the transient penalty in a fashion that is automatically adaptive to the data quality, enabling a reduction in overall cost.

This completes the description of the procedure to prevent adaptive agents from acting under high levels of uncertainty. This method is labeled “cost-based loop closure” to reflect the fact that estimates are prevented from being used in the closed-loop control until uncertainty is sufficiently low. It is important to note that the default control strategy is still a feedback structure. This procedure, unless otherwise noted, is used in all future results addressing adaptive game-playing agents.

The first two sections of this chapter have presented relatively straightforward addenda to the existing work of Chapter 2. In the remaining section, a Taylor series expansion of the generalized RDE solutions is used to obviate the need to repeatedly solve the RDEs by numerical integration. This enhancement is more complicated and is the subject of the remainder of the current chapter.

4.3 Truncated Taylor series Riccati solution approximation

This section presents an approach for approximating neighboring solutions of the generalized RDEs using truncated Taylor series approximations. Automatic differentiation (AD) is leveraged to numerically evaluate the Taylor series coefficients. AD is a software approach to numerically evaluating derivatives that has primarily been employed in problems involving large-scale optimization, such as engineering design optimization and training of recurrent artificial neural networks[50]. Some applications in the areas of guidance, navigation, and control of aerospace engineering vehicles have been published, such as Ref. [51], which has influenced this section. In this section, AD is employed to numerically expand RDE solutions about a nominal set of opponent weights.

This section begins by introducing AD and its application to RDE solution approximation. Subsequently, a scalar numerical problem is considered to evaluate the AD approach.

4.3.1 Automatic differentiation

Many functions of practical interest can be approximated by a truncated Taylor series expansion about a nominal solution point. By using the Taylor series approximation to the RDEs, an adaptive player can numerically solve for her optimal gain as a polynomial function of the opponent's weights. These solutions can be computed offline, prior to the engagement, and are valid in some neighborhood of the expansion point. This approach nicely complements the use of the nonlinear Kalman filter for estimation, which in general only converges for initial estimates “near” the true values and cannot tolerate arbitrarily large initial errors. By using the Taylor series approximation, an agent effectively eliminates repeated numerical integration of the RDEs, at the cost of a small loss in solution accuracy, and a requirement to store polynomial coefficients in memory.

The generalized RDEs have no closed-form solutions. Any Taylor series coefficients must be computed numerically. Governing equations for the coefficients as functions of time can be obtained by expanding partial derivatives of the governing ODEs. This can be done by hand, leading to tedious tensor equations that are time-consuming to numerically integrate. Alternately, computer symbolic algebra can be used to obtain the governing ODEs, but this often leads to an explosion of terms for higher-order derivatives. AD offers an alternative path to evaluating derivatives up to arbitrary order. All values of the higher-order derivatives are computed and stored a priori, and the accuracy of the online update is limited only by the resources available to store the derivatives in memory and perform array operations to evaluate the RDE approximation. Once the AD software is configured, all the end user must do is write code to numerically integrate the governing RDEs for nominal values of the opponent weights, and the software automatically computes the required derivatives in the background.

AD is employed to evaluate arbitrary-order Taylor series expansions of the RDE solution. The full class of AD methods is broad, but all share the characteristic of evaluating the numerical value of a floating-point operation(s) and one or more of its derivatives. AD is commonly implemented using operator overloading, allowing a user to write code in a standard form while derivatives are seamlessly evaluated in the background when code executes. AD is employed as an alternative to computer symbolic manipulation, which can become impractical for determining governing ODEs for high-order sensitivity derivatives. AD is often applied to problems in numerical optimization to obtain Jacobian information for gradient-based solutions[52]. A great number of AD software packages exist. Many are intended specifically for optimization problems, and only directly allow the user to evaluate the Jacobian and Hessian matrices. To obtain arbitrary-order Taylor series approximations, an AD tool that computes higher-order derivatives is needed.

The class of AD methods optimized for univariate Taylor polynomials (UTPs) offers a partial solution to the higher-order derivatives. Univariate Taylor series expansions of an operation can be evaluated very efficiently using AD. Software exists in various languages for computing UTPs. This dissertation has used the open-source `AlgoPy` Python software, whose development is discussed in Ref. [53]. However, UTP methods generally only directly compute unmixed partial derivatives in the expansion of an operation. To obtain the mixed partials needed for the Taylor series expansion, directional derivatives of the underlying function are expanded to high orders and a linear combination of directional derivatives can be used to obtain the mixed partials[52]. Efficient evaluation of these higher-order partials from UTPs has been a topic of study[54]. A brief overview is now given. Consider the Taylor series expansion of a vector function $f(\boldsymbol{x})$ about the point $\boldsymbol{a} + \boldsymbol{v}\tau$. By taking the expansion as $\tau \to 0$, the Taylor expansion reveals the directional derivatives of $f(\boldsymbol{x})$ along $\boldsymbol{v}$:

$$f(\boldsymbol{a} + \boldsymbol{v}\tau) = f(\boldsymbol{a}) + \frac{\partial f}{\partial \boldsymbol{x}}^T \boldsymbol{v}\bigg|_{\boldsymbol{x}=\boldsymbol{a}} \tau + \frac{1}{2}\boldsymbol{v}^T \frac{\partial^2 f}{\partial \boldsymbol{x}^2}\boldsymbol{v}\bigg|_{\boldsymbol{x}=\boldsymbol{a}} \tau^2 + \dots \tag{4.6}$$

For example, assume a second-order expansion of f is desired, and evaluate the directional derivatives associated with the following directions: $\boldsymbol{v}_1 = (2,0)$, $\boldsymbol{v}_2 = (0,2)$, $\boldsymbol{v}_3 = (1,1)$. This produces the following three Taylor series expansions:

$$f(\boldsymbol{a}+\boldsymbol{v}_1\tau) \approx f(\boldsymbol{a}) + 2\left.\frac{\partial f}{\partial x_1}\right|_{\boldsymbol{x}=\boldsymbol{a}} \tau + 2\left.\frac{\partial^2 f}{\partial x_1^2}\right|_{\boldsymbol{x}=\boldsymbol{a}} \tau^2 \tag{4.7}$$

$$f(\boldsymbol{a}+\boldsymbol{v}_2\tau) \approx f(\boldsymbol{a}) + 2\left.\frac{\partial f}{\partial x_2}\right|_{\boldsymbol{x}=\boldsymbol{a}} \tau + 2\left.\frac{\partial^2 f}{\partial x_2^2}\right|_{\boldsymbol{x}=\boldsymbol{a}} \tau^2 \tag{4.8}$$

$$f(\boldsymbol{a}+\boldsymbol{v}_3\tau) \approx f(\boldsymbol{a}) + \left.\left(\frac{\partial f}{\partial x_1} + \frac{\partial f}{\partial x_2}\right)\right|_{\boldsymbol{x}=\boldsymbol{a}} \tau + \frac{1}{2}\left.\left(\frac{\partial^2 f}{\partial x_1^2} + \frac{\partial^2 f}{\partial x_2^2} + 2\frac{\partial^2 f}{\partial x_1 x_2}\right)\right|_{\boldsymbol{x}=\boldsymbol{a}} \tau^2 \tag{4.9}$$

The terms $\left.\frac{\partial f}{\partial \boldsymbol{x}}^T \boldsymbol{v}_i\right|_{\boldsymbol{x}=\boldsymbol{a}}$ and $\left.\boldsymbol{v}_i^T \frac{\partial^2 f}{\partial \boldsymbol{x}^2} \boldsymbol{v}_i\right|_{\boldsymbol{x}=\boldsymbol{a}}$ in Eq. 4.6 are computed directly by the UTP AD software package. Using these computed terms, the desired mixed partials in Eqs. 4.7–4.9 can be found by rewriting the equations in matrix form. By requiring equality hold between Eqs. 4.6 and 4.7–4.9 for all values of τ, it is clear that the first and second order derivatives must satisfy the following matrix expressions:

$$\left.\begin{bmatrix} \frac{\partial f}{\partial \boldsymbol{x}}^T \boldsymbol{v}_1 \\ \frac{\partial f}{\partial \boldsymbol{x}}^T \boldsymbol{v}_2 \\ \frac{\partial f}{\partial \boldsymbol{x}}^T \boldsymbol{v}_3 \end{bmatrix}\right|_{\boldsymbol{x}=\boldsymbol{a}} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \\ 1 & 1 \end{bmatrix} \left.\begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \end{bmatrix}\right|_{\boldsymbol{x}=\boldsymbol{a}} \tag{4.10}$$

$$\left.\begin{bmatrix} \boldsymbol{v}_1^T \frac{\partial f}{\partial \boldsymbol{x}} \boldsymbol{v}_1 \\ \boldsymbol{v}_2^T \frac{\partial f}{\partial \boldsymbol{x}} \boldsymbol{v}_2 \\ \boldsymbol{v}_3^T \frac{\partial f}{\partial \boldsymbol{x}} \boldsymbol{v}_3 \end{bmatrix}\right|_{\boldsymbol{x}=\boldsymbol{a}} = \begin{bmatrix} 4 & 0 & 0 \\ 0 & 4 & 0 \\ 1 & 1 & 2 \end{bmatrix} \left.\begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} \\ \frac{\partial^2 f}{\partial x_2^2} \\ \frac{\partial^2 f}{\partial x_1 x_2} \end{bmatrix}\right|_{\boldsymbol{x}=\boldsymbol{a}} \tag{4.11}$$

Eqs. 4.10 and 4.11 are linear equations in the unknown derivatives $\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \frac{\partial^2 f}{\partial x_1^2}, \frac{\partial^2 f}{\partial x_2^2}, \frac{\partial^2 f}{\partial x_1 x_2}$. Eq. 4.10 is clearly overconstrained, and a unique solution can be found by either using a full-rank submatrix of the coefficient matrix or a pseudoinverse. Eq. 4.11 can be solved by direct inversion of the coefficient matrix. This procedure allows the higher order mixed partial derivatives of f to be extracted from the UTP expansions computed by an AD program.

A general procedure for extracting arbitrary order Taylor series partial derivatives of a function is now presented. The procedure is presented without proof, but is straightforward to confirm using properties of polynomials. Let the desired maximum polynomial degree be d. Let the eth degree directional derivative along $\boldsymbol{v}_i$ of f be denoted as $D^e_{\boldsymbol{v}_i} f(\boldsymbol{x})$. For each $e \leq d$, the directional derivative can be related to the mixed partials by a polynomial expansion as follows:

$$D^e_{\boldsymbol{v}_i} f(\boldsymbol{x}) = \left(\sum_{k=1}^{n} v_i(k) \frac{\partial}{\partial x_k} \right)^e f \tag{4.12}$$

In Eq. 4.12, the multinomial acts as an operator on the scalar function f with argument $\boldsymbol{x} \in \mathbb{R}^n$. $v_i(j)$ is the jth element of the vector $\boldsymbol{v}_i$. This operator determines the coefficient matrix; it can be verified that it produces the coefficient matrices of Eqs. 4.10–4.11. All that remains is to specify the directions $\boldsymbol{v}_i$ along which derivatives are taken. In principle, any set of vectors that produces a full-rank coefficient matrix at degree d can be used. A convenient choice is the $\boldsymbol{v}_i$ having 1-norm equal to d and whose elements are in the set of positive integers, $\mathbb{Z}^+$[54].

Eq. 4.12 and the chosen set of $\boldsymbol{v}_i$ are sufficient to define the coefficient matrices for all polynomial degrees $0 < e \leq d$. The AD program is used to compute the numerical values of the directional derivatives, and the corresponding values of the mixed partials are extracted by inversion of the coefficient matrices. This computes a vector of the mixed partials. To evaluate the truncated Taylor series approximation to $f(\boldsymbol{x})$ at a new value of $\boldsymbol{x}$, array operations are used to compute appropriate monomial functions of $\boldsymbol{x}$, and the dot product with the vector of mixed partials evaluates the contribution of the eth degree in the polynomial expansion.

In this dissertation, the Taylor series expansion of the numerical solution to an ODE is desired. AD methods are commonly implemented using operator overloading, which makes them convenient for many applications, but incompatible with most existing numerical integration functions. To support the use of AD Taylor expansions of ODE solutions, a fourth order Runge-Kutta-Fehlberg integrator (RKF45) method is implemented using the adaptive stepsize of Ref. [55]. Taylor expansions are to be computed offline a priori, and the efficiency of the numerical integration is not a significant priority.

This subsection has discussed how sensitivity derivatives are used to approximate updated solutions to the RDE in this dissertation. First-order derivatives have been obtained by analytically deriving the governing ODEs for the OSA game problem. To enable higher-order derivatives to be computed, automatic differentiation has been used to directly evaluate Taylor series expansions numerically. The class of AD methods based on univariate Taylor polynomials is used, necessitating some postprocessing to extract the mixed partials of the underlying functions. The use of AD software also requires the re-implementation of a numerical integration function for compatibility with AD data types, for which RKF45 is used. This software enables arbitrary order sensitivity derivatives of an ODE to be numerically determined. To develop an understanding of the practical use of the AD Taylor series expansion, the next section considers a numerical example.

4.3.2 Performance comparison – scalar problem

To validate the truncated Taylor series approximation to the generalized RDE solution, the performance of the truncated series is compared with the performance of the numerical solution from integration. Consider a simple problem with scalar dynamics. The problem dynamics and objective functions for the adaptive pursuer i and minimax evader j are given as follow:

$$\dot{x} = ax + u_i + u_j \tag{4.13}$$

$$J_i = \frac{1}{2} s_{f,i} x(T_f)^2 + \frac{1}{2} \int_0^{T_f} \left(q_i x^2 + u_i^2 + r_{ij} u_j^2 \right) \,\mathrm{d}t \tag{4.14}$$

$$J_j = \frac{1}{2} s_{f,j} x(T_f)^2 + \frac{1}{2} \int_0^{T_f} \left(q_j x^2 + u_j^2 + r_{ji} u_i^2 \right) \,\mathrm{d}t \tag{4.15}$$

Consider the following numerical values: $a = 0.1, q_i = 4, r_{ij} = -1.05, s_{f,i} = 10, T_f = 3$. Under the OSO strategy, pursuer i estimates the values of q_j, r_{ij}, and $s_{f,j}$. For simplicity in this example, i uses no control, which allows the performance of different estimators to be compared on the same measurement sequence.

The numerical values considered are collected in Table 4.1. The performance of the "full" estimator, using the UKF and repeated integration of the RDEs, is compared against the Taylor

q_i	q_j	r_{ij}	r_{ji}	$s_{f,i}$	$s_{f,j}$	T_f	Δt	$x(0)$
4.0	-3.0	-1.05	-.92	2.0	-2.5	2.0	0.02	-0.05

Table 4.1: Table of values used for scalar study of Taylor approximation

series-based RDE approximation for increasing polynomial degree. The feasible region $-4 \leq q_j \leq -3$, $-0.952 \leq r_{ji} \leq -0.88$, $-3.0 \leq s_{f,j} \leq -1.0$ is defined for the opponent weights. For this comparison, all estimators use the same process noise model, which places uncorrelated Gaussian noise on each of the estimated states. A process noise of variance 10^{-8} is placed on the $\dot{x}$ dynamics, and of variance 10^{-6} on each of the equations for the opponent weights. All measurements have a variance of 10^{-4}, and the same measurement sequence is used for all estimators. Initial covariances of $P_{xx} = 10^{-4}, P_{rr} = 10^{-4}, P_{qq} = 10^{-1}, P_{s_f s_f} = 1$ are chosen judiciously to envelope the initial estimate errors while avoiding instability of the RDEs.

Fig. 4.2 shows the performance of estimators as the polynomial degree is increased. The upper four charts indicate the estimated and true states. The bottom four charts show the estimation errors and associated 3σ bounds computed from covariance diagonals. Taylor series approximations up to degree three are shown. While it is clear that the filter is not well-tuned, the convergence of the Taylor series to the full problem solution for increasing d is clear. As the Taylor series degree increases, the approximation-based estimators more closely track the full UKF estimator. The largest difference comes in the increase from degree one to degree two. Degrees higher than two are considered, but not plotted as there are diminishing returns. The parameter estimates eventually converge with negligible bias, although the convergence is relatively slow. This is explained by the large initial errors in r_{ji} and q_j. The linear (degree one) Taylor approximation actually achieves smaller errors for most of the simulation, but reaches a biased estimate as time increases.

This section has discussed a sensitivity-based approach to approximating the RDE solution. First, AD is introduced and employed to obtain higher-order terms in a Taylor series approximation of the RDEs. These approximations eliminate the need for repeated integration of the RDEs, at the cost of reduced accuracy far from the point of Taylor series expansion in parameter space. Finally,

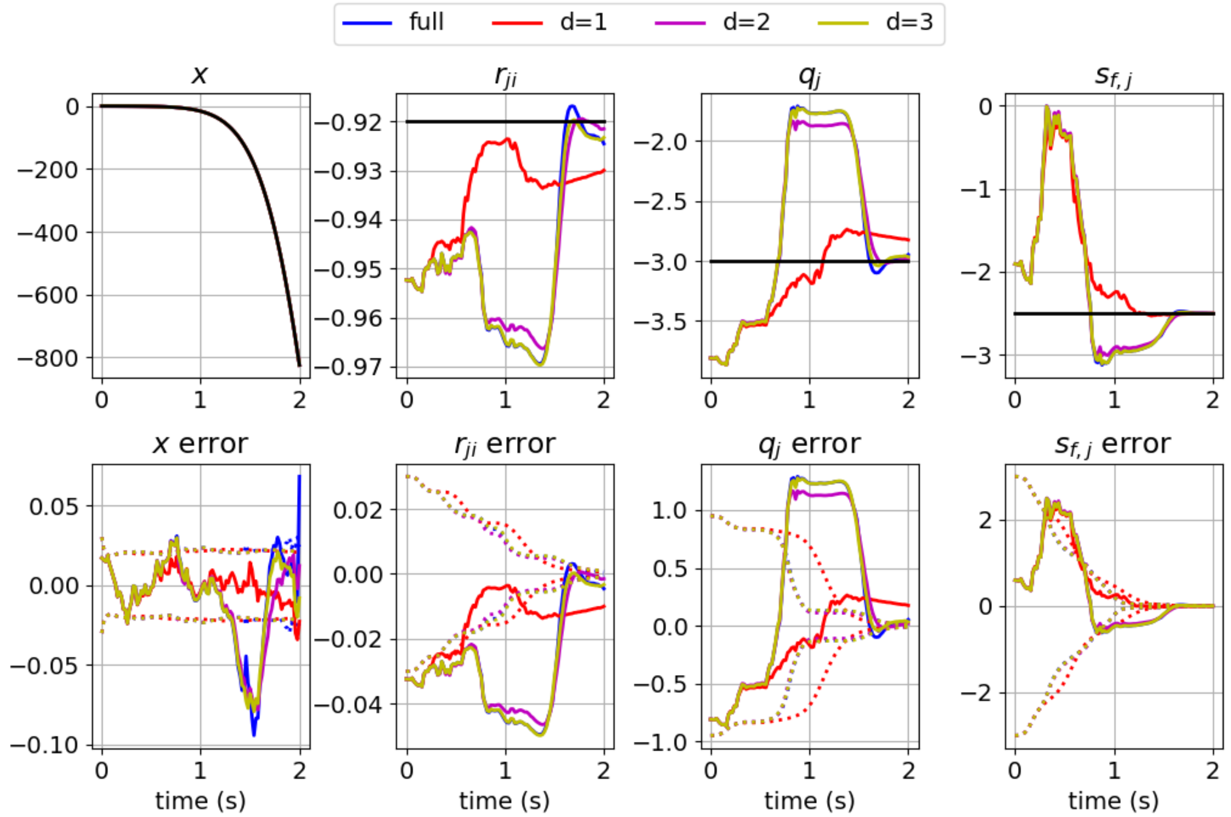

Figure 4.2: Comparison of the estimates and errors between the full LQDG estimator and the Taylor series-based filter. Upper figures show the truth values in black.

to demonstrate the merits of the truncated Taylor series approximation, a scalar open-loop example has been considered. This example shows the estimation performance converging to that of the full RDE solution as the Taylor series degree increases.

This chapter introduces a number of background improvements to address some shortcomings of the adaptive LQDG play introduced in Chapter 2. A slightly modified system parameterization is used to allow for a consistent notation between pursuer and evader, enabling more efficient code re-use in implementations. A straightforward cost-based loop-closure is introduced to allow an agent to intelligently switch from minimax strategy to adaptive strategy only when the uncertainty is sufficiently low. Finally, AD is used to obtain Taylor series approximations to the numerical RDE solution as functions of time and of the opponent weights. This innovation enables the adaptive player to efficiently use the unscented transform without the need to repeatedly solve the generalized RDEs, with performance in practice that approaches the original RDE solution accuracy. A scalar problem has been introduced and used to justify the basic performance of the Taylor series approximation. Chapter 7 presents numerical results demonstrating the cost-based loop closure and Taylor series expansion.

This chapter has focused on incremental improvements that make the adaptive LQDG play significantly more practical. In subsequent chapters, the new research directions suggested by the results of Chapter 3 are considered. In the next chapter, constrained estimation is introduced to reduce the rate of numerical failures in simulations for which the opponent parameters are near the limits of engagement stability.

5. CONSTRAINED ESTIMATION FOR NEAR-SINGULAR ENGAGEMENTS

Previous chapters have presented and refined an approach for adaptive gameplay in LQDGs. This approach permits an agent to adapt to NZS opponent weights and improve his performance by optimizing his control with respect to the true opponent weight values. However, not every configuration of player and opponent weights has a nonsingular solution. In a PE game, this could occur, e.g., if an evader is willing to use much more control energy to avoid capture than a pursuer is to achieve capture.

Previous discussions in Chapters 2 and 3 have noted the existence of such "singular engagements." These engagements are characterized by nonfinite values of the generalized Riccati gains during the game's time horizon. Even when the true game weights are selected to ensure finiteness of the Riccati matrices, the adaptive LQDG solution may fail if the **estimated** weights converge to an incorrect value for which no finite solution exists. This is especially likely at the start of an engagement when the uncertainty is high.

Thus, there are two motivations for studying the problem of singular engagements. First, an agent in a nonsingular engagement may still encounter numerical singularities in her RDE solutions due to estimation errors. Second, an agent may find herself in an engagement for which a solution does not exist because of the value of the opponent's weights. In the latter context, the agent can take some application-specific remediating action. For example, an agent can change her weights so that a finite-valued solution exists, or change her overall objective (e.g., switching into an information-gathering mode rather than a capture mode).

This chapter studies the problem of singular engagements for which the governing RDEs become unbounded. An approach based on constrained estimation is proposed both to prevent singularities due to errors in estimation, and to detect engagements for which finite-valued solutions are not possible. The chapter restricts its focus to implementing estimation algorithms and detecting when an incomplete information engagement has no finite solution. The question of what action should be taken when encountering opponent weights that lead to a singularity is taken to be application-specific and is left for future consideration.

The most significant technical challenge in this chapter is in obtaining explicit inequality constraints to apply to the estimated state. It has already been stated that there are no known, general analytical conditions for which nonsingular solutions to the generalized RDEs exist[19]. On the other hand, it is relatively inexpensive to use numerical integration to evaluate whether a candidate set of weights has a finite-valued RDE solution. Consequently, most of the present chapter focuses on developing a procedure for numerically approximating the inequality constraint in the space of opponent parameters.

The remainder of this chapter presents the approach for detecting engagement stability in detail. First, a scalar example problem is studied. This problem admits analytical conditions for engagement stability, and is used in subsequent sections. Next, an approach to approximating the set of stable opponent weights using Chebyshev polynomial functions is introduced. Third, constrained estimation based on probability density function (PDF) truncation is presented to address the nonlinear inequality constrained problem. Fourth, the procedure for detecting a singular engagement is presented. Finally, numerical results, including Monte Carlo analyses, are generated to demonstrate the utility of the approach.

5.1 Motivation

This section presents problem with scalar dynamics having a set of analytical stability bounds in the space of evader weights. Consider the scalar OSO problem considered in Sec. 4.3.2. A set of numerical values for i's weights and box constraints for j's weights are shown in Table 5.1. Note that the box region has been expanded to include weight values for which stability is not satisfied. The stability of the RDE solutions in this region can be characterized by the existence of real steady state solutions of the correct sign. The existence of such solutions places constraints on both agents. For the minimax evader j, the requirements are simply $q_j \leq 0$ and $0 > r_{ji} > -1$. For the adaptive OSO pursuer i, the existence of a real positive definite s_i^* requires the magnitude

Variables		Box region	
T_f	3.0	r_{ji}	
Δt	0.05	-.97	-.88
a	0.1	q_j	
r_{ij}	-1.05	-6	-2
q_i	4	$s_{f,j}$	
$s_{f,i}$	10	-3	-1

Table 5.1: Table of values used for scalar OSO study of constrained estimation

of j's steady-state Riccati solution, s_j, be bounded as follows:

$$\left| s_j - \frac{a}{1 + r_{ij}} \right| \leq \frac{\sqrt{-r_{ij}a^2 - q_i(1 + r_{ij})}}{1 + r_{ij}} \tag{5.1}$$

By requiring the true solution for s_j^* be greater than or equal to the lower bounds of Eq. 5.1, an inequality constraint on the magnitude of q_j is obtained. Satisfaction of this condition ensures the existence of a positive definite steady-state pursuer gain.

$$q_j \geq \bar{s}_j^2(1 + {r_{ji}}^{-1}) - 2a\bar{s}_j \tag{5.2}$$

$$\bar{s}_j \equiv \frac{a + \sqrt{-r_{ij}a^2 - q_i(1 + r_{ij})}}{1 + r_{ij}}$$

To validate these results, the solutions of the OSO RDEs are evaluated numerically over a grid of values of q_j and r_{ji}. If the numerical solution at any time exceeds an absolute value of 10^6, it is classified as unstable. The results and the lines corresponding to the inequality constraints are shown in Fig. 5.1. The numerical solution matches the theoretical limits, indicating that the steady-state analysis adequately characterizes stability in the region of interest.

This section presents background motivation for studying the constraints in LQDGs. This section introduced a problem with scalar dynamics and validated analytical bounds for engagement stability. The analytical bounds are used later in this chapter to validate the proposed numerical procedure for predicting engagement stability. The next section presents the approach to approxi-

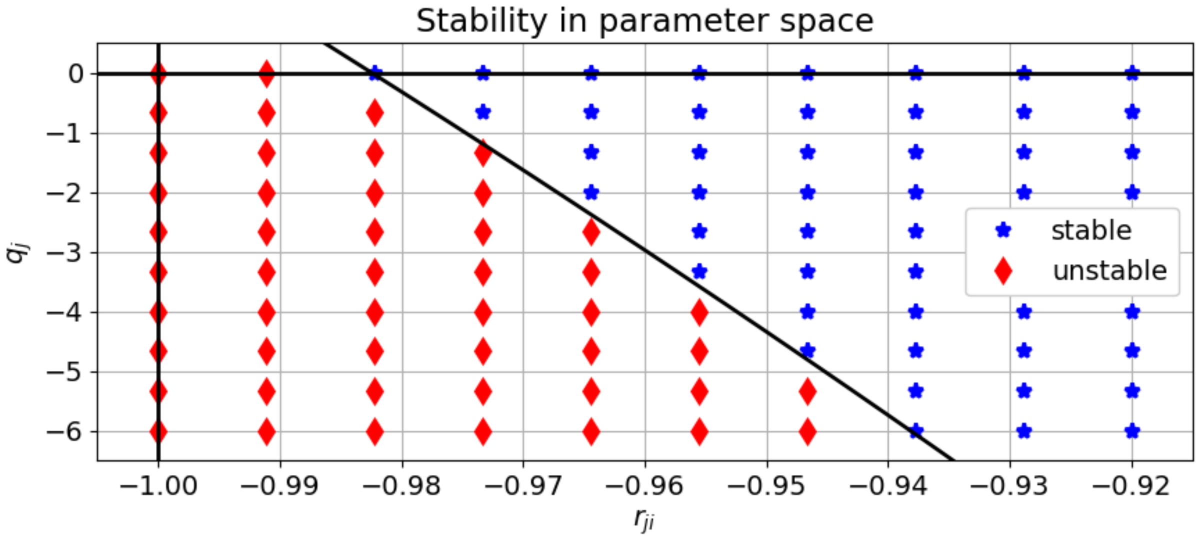

Figure 5.1: Numerically evaluated stability in scalar problem parameter space, with solid line showing theoretical stability boundaries

mating the engagment stability condition numerically.

5.2 Numerical quantification of instability conditions

The general LQDG problem lacks analytical conditions for which the generalized RDE solution guarantees optimality and/or Nash equilibrium conditions. Singular solutions to the generalized RDEs are a particular problem because they lead to excessively unbounded control commands. Constrained estimation is suggested as a method for preventing an adaptive LQDG player from commanding singular control values. To implement constrained estimation, a functional form of the constraints on the opponent's weights is needed. This section investigates numerical quantification of these constraints.

It is relatively inexpensive to test a given set of weights for singularity by brute-force numerical integration with a numerical tolerance on the maximum absolute value of the RDE solution elements. This section proposes a simple iterative procedure for identifying points on the the "stability manifold" separating the regions in the opponent's parameter space that lead to stable and unstable engagements, respectively. Subsequently, Chebyshev polynomial basis functions are used to approximate the stability manifold over a region of interest for numerical evaluation.

5.2.1 Approximation of the stability interface

This objective of this section is to numerically approximate constraints on the opponent's weights that guarantee nonsingular solutions of the generalized RDEs. The boundary between regions in which solutions are and are not singular is referred to as the "stability manifold," for convenience; in general, the stability manifold may consist of non-intersecting closed sets. Numerically computing points on the stability manifold generally requires iteration from a starting point. It is felt that performing this computation online is impractical. Instead, the approach used is to approximate the stability manifold with basis functions and use the approximate constraint for estimation. Fitting basis functions is the topic of the next subsection. Before that can be investigated, a procedure is needed for obtaining points on the stability manifold in offline computations. That is the subject of the current subsection.

Identifying the stability manifold represents an unusual sort of root-finding problem. On one side of the manifold, the generalized RDE solutions are well-defined and exhibit no special numerical challenges. On the other side, one or more elements of the solutions become singular before the terminal time. This characteristic prevents the direct use of most standard root-finding algorithms, which are generally not well-equipped to find solutions in the presence of singular solutions. Motivated by Fig. 5.1, a reasonable approach is to assume the user has some crude knowledge of the parameter values of interest and perform a bracketing search for a point at which the Riccati matrix norm becomes singular to within a tolerance at T_f. For example, in Fig. 5.1, a lower bound of -0.92 can be defined for r_{ji}, and an upper bound of -0.99. Given these bounds, the approximate solution for the value of r_{ji} on the manifold can be found as a function of q_j by searching on the interval between the upper and lower limits on r_{ji}.

The bisection method is a simple but reliable method for finding a root of a function on an interval. It is applied to the problem of numerically approximating the stability manifold. Let the opponent have N independent weights. One weight is specified as the iteration weight and upper and lower bounds on it are specified by an engineer. The upper and lower bounds must satisfy the condition that one endpoint lies in the stable region and the other lies in the unstable region. The bisection algorithm begins by selecting a point halfway in between the stable and unstable endpoints. The RDE solution is evaluated at the midpoint. If the solution is singular, the unstable endpoint is set equal to the midpoint, and the function starts again on the new interval. Otherwise, the stable endpoint is updated, and iteration continues on the new interval. Iteration continues until a numerical tolerance on the difference between the stable and unstable endpoints is met. The bisection method is a well-known algorithm and its application to this problem is straightforward.

To evaluate the stability of the numerical RDE solutions, event detection is used in conjunction with numerical integration. Event detection is a common feature of modern ordinary differential equation (ODE) solvers including MATLAB's `ode45`[56] and SciPy's `solve_ivp`[57] functions. Event detection allows the numerical solution to be terminated conditionally if the norm of the Riccati matrix exceeds a numerical threshold. If an event is detected, then the problem weights are identified as unstable.

This subsection outlines a simple procedure to numerically approximate a point on the stability manifold using the bisection method for search along a fixed interval. In the next subsection, this procedure is used to fit a basis function approximation to a set of points lying on the stability manifold.

5.2.2 Constraint approximation with Chebyshev polynomial basis functions

To effectively adapt to uncertainty in the opponent parameters while ensuring parameter estimates remain inside the stable region of the parameter space, the stability interface must be treated as an explicit constraint on the parameter estimation problem. Since the stability interface cannot, in general, be analytically quantified, it must be approximated numerically using the method of the previous subsection. The current subsection reviews the motivation for numerically approximating the stability manifold. This is followed by a discussion of possible solution methods. Chebyshev polynomial basis function approximation is chosen because it is relatively scalable and straightforward to implement. Lastly, an alternative constraint approximation choice that may be more generalizable is reviewed.

The decision to numerically approximate the stability interface is justified in light of the constrained estimation problem. Constrained estimation forces a state estimate to satisfy one or more possibly nonlinear constraints:

$$\boldsymbol{g}(\hat{\boldsymbol{x}}, t) \leq \mathbf{0} \tag{5.3}$$

A usable characterization of the stability interface should have two properties. First, it should be able to evaluate, in a binary fashion, whether a given point $\hat{\boldsymbol{x}}$ in parameter space satisfies Eq. 5.3. Second, it should contain an explicit functional representation or approximation to the left-hand side of Eq. 5.3 that can be used to solve for a point that satisfies equality of the equation. These two requirements enable a constrained estimation algorithm to determine if a constraint is violated and update the state estimate to a value that satisfies the constraint. Any approach to quantifying the stability interface that satisfies these properties should be compatible with most constrained estimation procedures.

In considering this problem, two classes of algorithms are natural choices: algorithms for direct function approximation of the stability interface, and algorithms for classifying points as belong to stable or unstable parts of the opponent's parameter space. The latter approach is potentially attractive, particularly if the stability manifold has high curvature, or is not one-to-one over the region of interest. In the dissertation, the function approximation approach is used, primarily motivated by simplicity of implementation.

In numerically approximating the stability interface, there are two main considerations. First, the numerical approximation should have at minimum reasonably smooth first derivatives; this allows the approximation to be compatible with a wide range of possible estimation strategies. Second, the approach should be somewhat scalable as the dimensionality of the problem increases. Ultimately, many different approaches may work, depending on the problem dimension and the nature of the stability interface. The approach presented in this section is thought to be suitable for a range of problems of small to medium dimensions, and exhibits good tradeoffs between accurate function approximation and algorithm complexity.

Chebyshev polynomials[1] are a well-known choice of polynomial basis functions. Recent work has highlighted the suitability of Chebyshev polynomials for least-squares-based function approximation in high dimensions[58]. The procedure for efficiently fitting Chebyshev polynomials is summarized in Appendix C. In Section 5.5, Chebyshev polynomials are used to evaluate the stability limits of the simple example problem (cf. Fig. 5.5).

The advantages and disadvantages of the application of Chebyshev approximation to the constrained estimation problem are now discussed. The two primary advantages both stem from the orthogonality properties when applied to least-squares function approximation. First, the Chebyshev coefficients can be computed using pure matrix algebra operations, which can readily be evaluated. Second, the method scales reasonably well as the dimension of the independent variable space increases. These qualities make it a useful choice that is straightforward to implement and achieves low approximation error in many cases. There are several disadvantages to the Chebyshev approximation approach. First, although the method scales well in regard to increasing dimensions, the procedure is still memory intensive for high-degree approximations in high dimensions, which may limit the achievable accuracy. Second, some error is incurred by using this approach, since the stability interface is only being approximated. The magnitude and acceptability of the error must be evaluated on a case-by-case basis. Third, the solution procedure of Ref. [58] does not offer a rigorous approach for deciding how many basis functions should be used, and this question must generally be addressed by either manual iteration or some separate statistical analysis. Fourth and finally, Chebyshev approximation cannot be applied if the stability interface cannot be represented by a one-to-one function.

This section has discussed numerical approximation of the stability manifold and selected a Chebyshev basis function approach. The following section discusses constrained estimation for the LQDG problem.

[1] Specifically, this work refers to the Chebyshev polynomials *of the first kind*.

5.3 Constrained estimation

Estimation in the presence of equality and inequality constraints is not frequently considered in aerospace engineering applications. However, an extensive background literature exists, and has been applied to problems such as chemical process modeling[59], turbofan engine health[60], and ground vehicle tracking[61]. For realtime applications, the most popular approaches seem to be derived from either the Kalman filter or moving horizon estimation approaches. This section begins with a brief overview of the categories of possible solution methods. Second, a PDF truncation approach is selected, based on the LQDG problem environment, and an elegant sample implementation is outlined.

5.3.1 Constrained estimation overview

A useful nomenclature for constrained estimation approaches is identified by Straka et al [61]. This work suggests three categories of constrained estimation approaches exist: (1) description-modifying approaches, in which an equality constraint is either eliminated from the system parameterization, or is treated as a deterministic "pseudo-measurement" with zero noise; (2) optimization approaches, such as moving horizon estimation or its one-step analog, recursive nonlinear dynamics data reconciliation[62], which solve for state values that minimize a residual error subject to the constraints; (3) estimate-constraining approaches, which commonly augment the standard nonlinear Kalman estimation procedure with some additional correction to account for the presence of constraints. It should be noted that description-modifying approaches are usually amenable only to equality constraints, and are not suited for the LQDG problem. Optimization-based approaches are attractive but tend to be more computationally intensive. It may not be possible to implement these methods on all systems. Consequently, it is felt that estimate-constraining approaches are the most general class of estimation solutions that are appropriate for the LQDG problem, and this shall be the focus of the following discussion.

Simon [63] provides a useful overview of the primary methods used across all three categories of the constrained estimation literature. From this survey paper, the equality constrained problem appears to have received the greatest attention in the existing literature. Classical methods implement modifications to the EKF and are well-suited to handle linear equality and inequality constraints. The primary approaches are as follow: psuedo-measurements, in which a constraint is treated as a zero-noise measurement and appended to the measurement vector; projection, in which an updated state is projected onto the constraint surface to minimize some distance metric in state space; and PDF truncation, in which the updated state PDF from the Kalman update is truncated at a constraint surface and re-normalized, producing a shifted estimate and smaller covariance. Of these, only projection and PDF truncation are well-suited to handle inequality constraints. When constraints are nonlinear, a diverse array of methods have been considered, such as the second-order EKF, Moving Horizon Estimation, and various extensions to particle filters.

In the LQDG application, it is desirable to retain relatively modest computational demands to ensure the proposed solution is suitable for a broad range of candidate systems. Thus, exploiting one of the modifications to the EKF is felt to be a reasonable starting point. Of these approaches, both pseudo-measurements and projection require an exact root of the constraint to be enforced. This process is trivial for linear constraints, but in general requires numerical iteration with nonlinear constraints. Since the methods operate directly on the mean of the updated state, accuracy of the solved root may be critical to ensure the updated state lies outside the inadmissible region. By contrast, PDF truncation operates on the covariance of the posterior state, and has the effect of shifting the conditional mean via the truncation process. If the truncated covariance magnitude is larger than numerical errors in the resolved root, then the numerical tolerance of root-finding can be relaxed, and first-order accurate methods may be adequate.

5.3.2 PDF truncation discussion

The procedure for constraint application using PDF truncation is presented in Appendix B. Essentially, it consists of sequentially applying each of the inequality constraints to modify the posterior estimated state and covariance. The procedure does have some unusual operational characteristics that should be reviewed[60]. First, the truncated state estimate is biased, even for fully linear systems with linear constraints. This is a drawback to PDF truncation that other constrained estimation approaches overcome. It should be noted that other error metrics, such as root mean square error, may still be better for the PDF truncation filter than for the unconstrained estimate. Second, the truncated state $\tilde{\boldsymbol{x}}_k^+$ is nonunique if the constraints are not independent. This is not expected to be a significant problem for the LQDG, in which only one constraint is expected in most problems. Third and finally, the PDF truncation process is not properly recursive. Consider, for example, a constant scalar state with no process noise bounded below by 0:

$$x_{k+1} = x_k \tag{5.4}$$

$$x_k \geq 0 \tag{5.5}$$

Let the measurement model $y_k = x_k$ have infinite measurement covariance, such that $[P_{k+1}] = [P_k]$ for the unconstrained system. If the PDF truncation algorithm is applied blindly to the posterior state, and the truncated state is fed back into the filter as the prior state for the next iteration, then the resulting state estimate will increase monotonically while the covariance decreases monotonically. This is obviously unacceptable. Two possible remedies are identified, and are referred to as the "recursive" and "nonrecursive" implementations of PDF truncation. Essentially, the recursive implementation only applies the truncation algorithm using constraints that are violated after the measurement update. The nonrecursive implementation never modifies the filter state, and only uses the PDF truncation to modify the state before it is used in external calculations (e.g. for closed-loop control). This chapter studies both recursive and nonrecursive implementations as an exercise. In the LQDG estimation problem, the recursive method is the natural choice because it prevents the UKF propagation step from attempting to solve for potentially singular opponent gains. However, when the Taylor series approximation to the RDE solution introduced in Chapter 4 is used, the nonrecursive method can be used.

Appendix B also summarizes two approaches to applying the PDF truncation algorithm for nonlinear equality constraints of the type that arise in the LQDG. The method of Simon [64] uses the Jacobian of the constraint about its pre-constrained value to obtain approximately linear constraints. The approach of Tully [65] operates directly in the "constraint space," and uses a first-order accurate pseudo-measurement to transform the truncated PDF back into the estimated state space. The two algorithms differ only in their extension to nonlinear systems. Both are first-order accurate and it is not clear how they compare. In this chapter, both approaches are implemented and compared in the later numerical example in this chapter. This section has discussed constrained estimation procedure and chosen a PDF truncation procedure for the LQDG problem. The problem of detecting singular engagements is now considered.

5.4 Singular engagement detection

The method of estimating opponent weights in a LQDG depends on observability of the opponent's game weights through their influence on the measured relative dynamics. The opponent weights define her feedback gain, and the time-varying nature of the RDE gain solution allows observability of the underlying game weights. As an alternative to estimating the LQDG weights, the opponent feedback gain can be estimated directly by modeling it as a stationary white-noise process. This approach has many disadvantages if the opponent is actually using a LQDG-derived strategy. It will not track the time-varying gain as well, and it cannot be used to extrapolate the future value of opponent gains. However, this "nonparametric" gain estimator can be reliably used even in situations where the adaptive player's RDE solution is singular. By comparing the nonparametric estimator and the constrained LQDG estimator, the presence of unstable opponent gains can be detected. In this section, a procedure for using the nonparametric estimator to classify engagement stability is presented.

The proposed approach is summarized in the flowchart of Fig. 5.2. Two nonlinear Kalman Filters are implemented and executed in parallel. First is a constrained adaptive LQDG estimator of either the OSO or FNE type, as appropriate for the scenario. Second is a nonparametric estimator in which the opponent's influence is modelled as a stationary white-noise process with some appropriately chosen process noise magnitude[2]. The constrained LQDG estimator is referred to as

[2] Additional sophistication can be introduced by considering various adaptive process noise filters to adapt to scenarios with greater uncertainty.

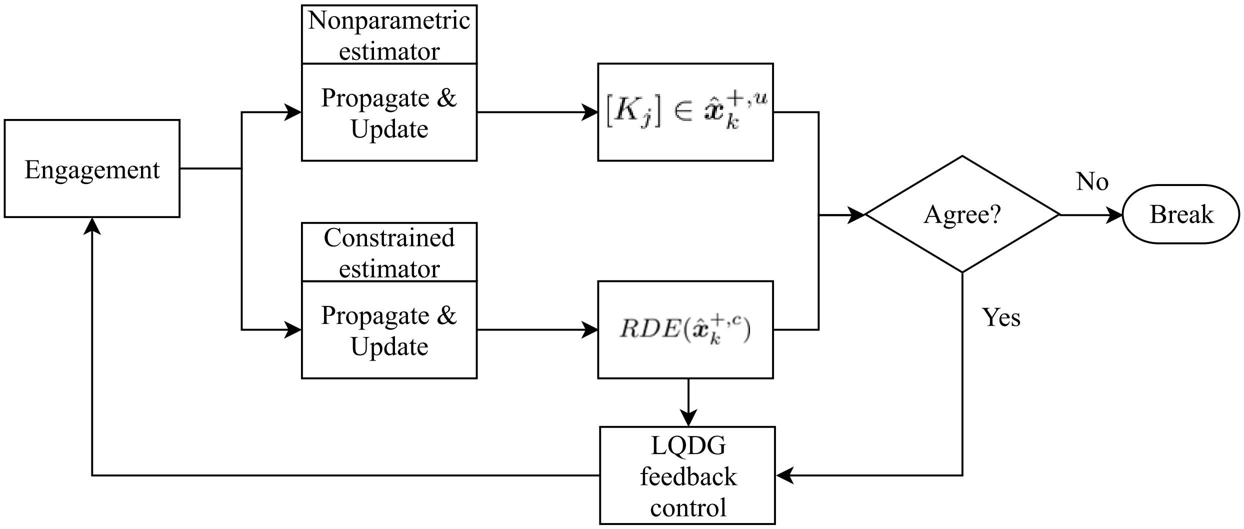

Figure 5.2: Flowchart of parallel filter logic

"filter 1" and the nonparametric estimator as "filter 2." The two estimators have the same measurement model. When the opponent gains are in the stable region of the parameter space, then the gain computed from filter 1 should agree with that of filter 2. When this occurs, the values of filter 1 should be used to develop an appropriate strategy to counter the opponent. When the opponent gains are in the unstable region, then the gain computed from filter 1 will disagree with that of filter 2. In this case, the objective is no longer to develop an appropriate counter-strategy. Instead, the goal is to identify with some appropriate statistical confidence when the two filters disagree. Subsequently, the estimating player can take appropriate, application-specific action. For instance, if capture is not possible, an adaptive pursuer might switch into an information-gathering mode, rather than a capture mode. In other scenarios, a player may be able to change her gains to achieve a stable RDE solution, but this will not always be feasible.

Design and implementation of the filters is relatively straightforward. All that is needed to implement the procedure of Fig. 5.2 is a method to determine if the two filters agree. With the adaptive LQDG scenario, two methods can be used. First, some LQDG problems may implicitly bound the closed-loop opponent gain. For instance, the opponent gain elements may be bounded above or below for all solutions of stable engagements. This can occur in the OSO engagement, since the adaptive player's Riccati gain can become singular while the minimax player's gain remains finite. However, in the FNE engagement, the Riccati solutions are coupled and both will become singular simultaneously. So, this approach is only usable for OSO problems. A different approach is to directly evaluate the filter 2 estimates in terms of their statistical consistency with respect to the estimates from filter 1. Inconsistency indicates an unstable engagement. This approach is more general, and has a good theoretical basis. However, it is likely to be sensitive to the statistical consistency of the two estimators, and may require more manual labor in tuning the filters. To conclude this section, methods based on each approach are presented in technical detail.

Three approaches for testing if the parallel filter estimates agree are considered. Two (Methods 0 and 1) are based on implicit limits on the opponent gain elements. The remaining approach (Method 2) is based on statistical tests for consistency between the parallel estimates. Before presenting these methods, some notation is first defined. Let filter 2 produce a direct estimate of the opponent gain in terms of a vector of random variables having the assuming distribution $\mathcal{N}(\hat{\boldsymbol{x}}_2, [P_2])$. Filter 1 produces a constrained estimate of LQDG parameters, and then computes the opponent gain as a function of the parameters. Denote by $\hat{\boldsymbol{\theta}}$ the parameter estimates from filter 1 and $\hat{\boldsymbol{x}}_1(\hat{\boldsymbol{\theta}})$ the estimated value of the opponent gain elements under filter 1. Under the LQDG constraints, $\boldsymbol{\theta}$ is assumed to be constrained to lie in some admissible region. If the volume defined by all admissible values of $\hat{\boldsymbol{x}}_1(\boldsymbol{\theta})$ is also finite, then Methods 0 or 1 can be used to detect instability. Otherwise, Method 2 should be used. To begin, Methods 0 and 1 are described.

5.4.1 Implicit opponent gain bounds

Methods 0 and 1 operate by exploiting the stability bounds that exist in the set of stable LQDG game weights and feedback gains. Basically, Method 0 compares the closed-loop opponent gain from filter 2 against the set of admissible gains for which a nonsingular RDE solution exists. If the probability that any gain element lies outside the admissible gain region is too large, the engagement is classified as unstable. Method 1 operates in the space of opponent LQDG weights $\boldsymbol{\theta}$ by attempting to find a value of $\boldsymbol{\theta}$ that produces the gain estimate $\hat{\boldsymbol{x}}_2$. If the probability of the solution $\boldsymbol{\theta}^*$ under the estimate of filter 1 is too small, the engagement is classified as unstable.

Method 0 is the simplest approach considered. The approach overestimates the probability that a state estimate $\hat{\boldsymbol{x}}_2$ lies in the stable region of the opponent's gain space, but it is less expensive to evaluate online than Method 1. Under Method 0, each element of $\hat{\boldsymbol{x}}$ is assumed to be independent, and the extremal values of each element are computed based on the admissible values of $\boldsymbol{\theta}$. The admissible values of $\boldsymbol{\theta}$ are characterized by exploiting the Chebyshev constraint approximation. Under the Chebyshev constraint model, a boundary associated with θ_N as a function of $\theta_1, \ldots, \theta_{N-1}$ has already been defined using a grid of points in parameter space. The extrema of the jth element of $\hat{\boldsymbol{x}}$ can be characterized as follows: for each Chebyshev CGL node $\theta_1(k), \ldots, \theta_{N-1}(k)$, find θ_N that satisfies the constraints and maximizes (minimizes) $\hat{x}(j)$. The global extrema of each element over the CGL nodes defines upper and lower bounds on $\hat{x}(j)$ associated with stable engagements. The elementwise extrema $l_b(j) \leq \hat{x}(j) \leq u_b(j)$ are determined offline. Online, the estimate $\hat{\boldsymbol{x}}_2$ is found to support the hypothesis of a stable engagement if there is an admissible $\hat{\boldsymbol{x}}$ that satisfies the constraints and has a high likelihood under a Gaussian assumption for the estimate. An α-level test is implemented as follows. Let $\hat{\boldsymbol{x}}^* = \operatorname{argmin}_{\boldsymbol{x}} (\boldsymbol{x} - \hat{\boldsymbol{x}}_2)^T [P_2]^{-1} (\boldsymbol{x} - \hat{\boldsymbol{x}}_2)$, subject to $l_b(j) \leq \hat{x}(j) \leq u_b(j)$, $j = 1, \ldots, n$. If the following inequality is satisfied, then $\hat{\boldsymbol{x}}_2$ supports the hypothesis of a stable engagement with a false-positive rate of α:

$$(\boldsymbol{x}^* - \hat{\boldsymbol{x}}_2)^T [P_2]^{-1} (\boldsymbol{x}^* - \hat{\boldsymbol{x}}_2) < F_{\chi^2}^{-1}(1 - \alpha, n) \tag{5.6}$$

$F_{\chi^2}^{-1}(a, b)$ is the inverse cumulative distribution function of a chi-square random variable with probability level a and b degrees of freedom. Because this approach assumes the elements of $\hat{\boldsymbol{x}}^*$ are independent of one another, it overestimates the volume of the set of admissible opponent gains. However, Method 0 solves a problem that is guaranteed to be convex, unlike Method 1; hence, in general, Method 0 is numerically advantageous to Method 1, which may be significant in some applications. The test is demonstrated graphically in Fig. 5.3 for estimating an opponent gain with two unique elements. The figure shows the inadmissible region in the space of variable $\boldsymbol{x}$ as shaded. The upper and lower bounds on the elements of $\boldsymbol{x}$ are drawn from the admissible

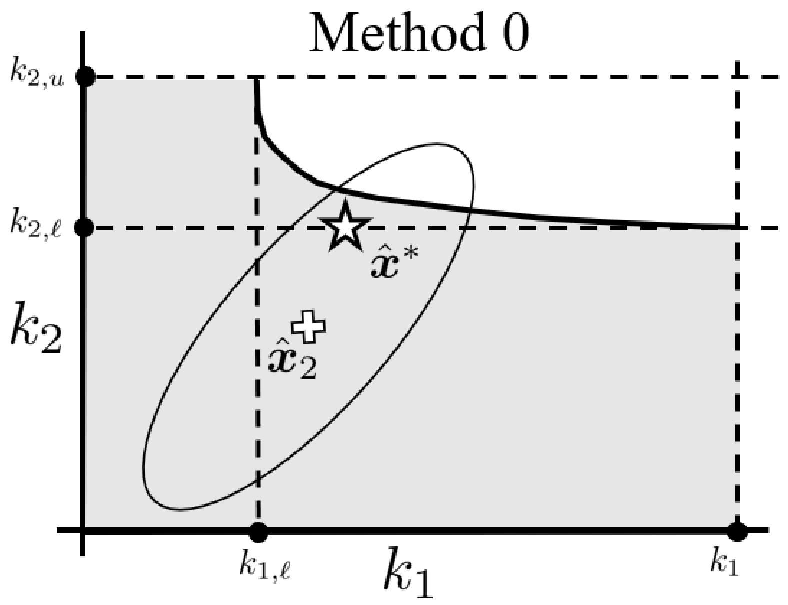

Figure 5.3: Graphical explanation of Method 0 statistical test

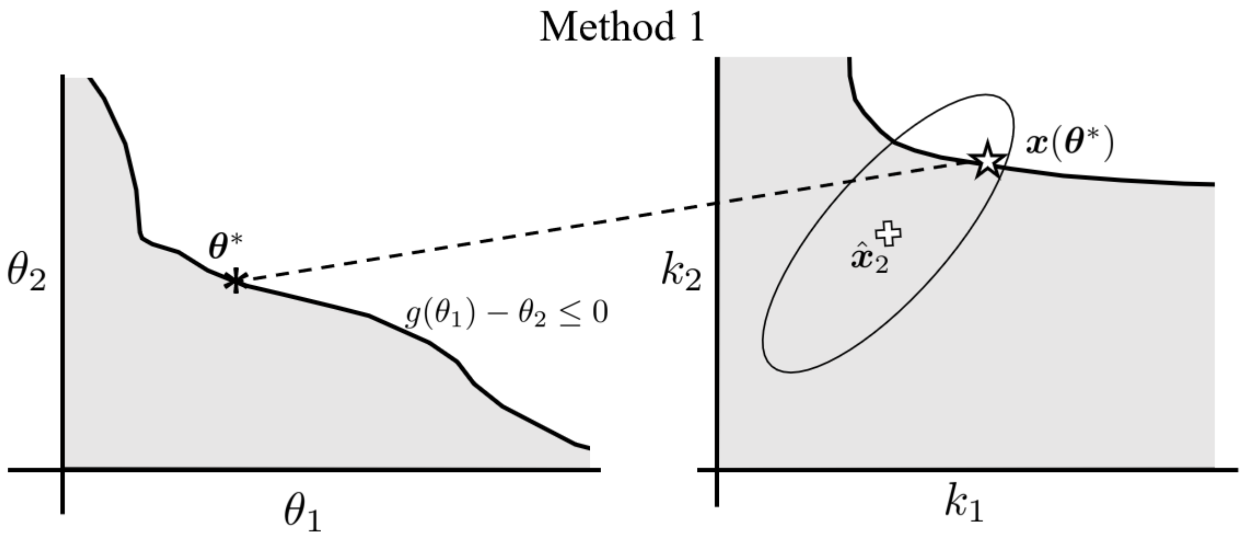

Figure 5.4: Graphical explanation of Method 1 statistical test

region, and the maximum likelihood value of $\hat{\boldsymbol{x}}^*$ that satisfies the constraints is shown. The test evaluates if the computed value of $\hat{\boldsymbol{x}}^*$ is sufficiently likely, given the estimated distribution of $\hat{\boldsymbol{x}}_2$.

Method 1 solves the same problem as Method 0 by optimizing over the parameter space, $\boldsymbol{\theta}$, using the existing bounds used for constrained estimation. Because $\hat{\boldsymbol{x}}_2$ is theoretically independent of $\boldsymbol{\theta}$, a test procedure can be implemented by finding the value of $\boldsymbol{\theta}$ that maximizes the probability that $x(\boldsymbol{\theta})$ comes from the same distribution as $\hat{\boldsymbol{x}}_2$. That is, let $\boldsymbol{\theta}^*$ be the maximum-likelihood estimate of $\boldsymbol{\theta}$ constrained by the theoretical requirements of the stable LQDG:

$$\boldsymbol{\theta}^* = \mathrm{argmax}_{\boldsymbol{\theta}} \left\{ p(\boldsymbol{x}(\boldsymbol{\theta}) \big| \hat{\boldsymbol{x}}_2, [P_2]) \right\} \tag{5.7}$$

$$\mathrm{s.t.} f(\boldsymbol{\theta}^*) \leq 0$$

Let $\boldsymbol{x}^*$ denote $\boldsymbol{x}(\boldsymbol{\theta}^*)$. Under the null hypothesis H_0, $\boldsymbol{x}^*$ is an admissible value drawn from the same distribution as $\hat{\boldsymbol{x}}_2$ and the engagement is stable. The alternate hypothesis, H_1, states that $\boldsymbol{x}^*$ comes from a different distribution. Under this model, $(\boldsymbol{x}^* - \hat{\boldsymbol{x}}_2)^T [P_2]^{-1} (\boldsymbol{x}^* - \hat{\boldsymbol{x}}_2)$ is a chi-square random variable with n degrees of freedom, with n being the dimension of $\hat{\boldsymbol{x}}$. A decision rule with false error probability α is defined as follows:

$$(\boldsymbol{x}^* - \hat{\boldsymbol{x}}_2)^T [P_2]^{-1} (\boldsymbol{x}^* - \hat{\boldsymbol{x}}_2) \underset{H_0}{\overset{H_1}{\gtrless}} F_{\chi^2}^{-1}(1-\alpha, n) \tag{5.8}$$

This equation defines a decision rule for determining if the distribution of $\hat{\boldsymbol{x}}_2$ satisfies the theoretical requirements for stability of the LQDG. That is, if the left-hand side of the equation is less than the right-hand side, H_0 is detected as true; otherwise, H_1 is take to be true.

As presented, Method 1 is applied to only the current-time estimate of $\hat{\boldsymbol{x}}_2$. In practice, better performance may be obtained by replacing the left-hand side of Eq. 5.8 by a sum over the last N estimated states and covariances, in which case the degrees of freedom of the right hand side are modified to Nn. This modification may reduce false positives during transients, for example, when $\hat{\boldsymbol{x}}_2$ is changing rapidly. Method 1 is demonstrated graphically in Fig. 5.4. This figure shows the admissible regions in both $\boldsymbol{\theta}$-space and $\boldsymbol{x}$-space as unshaded. The constrained value of $\boldsymbol{\theta}^*$ that maximizes the likelihood of $\boldsymbol{x}(\boldsymbol{\theta}^*)$ given $\hat{\boldsymbol{x}}_2$ is shown on the left plot. It it mapped into $\boldsymbol{x}(\boldsymbol{\theta}^*)$ on the right plot, and the resulting value is found to support the hypothesis of a nonsingular engagement if the likelihood of $\boldsymbol{x}(\boldsymbol{\theta}^*)$ is sufficiently high.

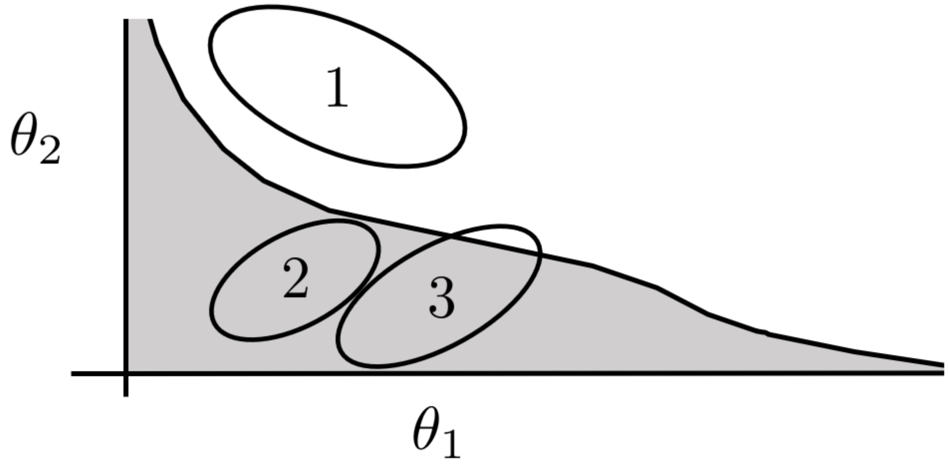

Figure 5.5: Illustration of one limitation of the method 1 test showing the inadmissible region and various candidate PDFs as ellipses

The advantage of Method 1 is that it is simple to understand and to tune. Furthermore, it is not highly sensitive to statistical consistency of the estimator for $\hat{\boldsymbol{x}}_2$, and is independent of the estimator for $\hat{\boldsymbol{x}}_1$. While the computation of $\boldsymbol{\theta}^*$ can require iterative numerical methods, in practice any $\boldsymbol{\theta}$ that satisfies Eq. 5.8 is sufficient to demonstrate stability of the estimated state. One shortcoming of this decision rule is that no consideration is made of the overlap between the admissible region and the PDF of $\boldsymbol{x}_1$. This is demonstrated graphically in Fig. 5.5. The figure shows the inadmissible region as shaded and candidate PDFs of the estimate for θ_1, θ_2 as 99% covariance ellipses. PDFs 1 and 3 both satisfy the Method 1 test with a 1% false-positive rate, even though it can be argued qualitatively that PDFs 2 and 3 are more similar to one another. It is felt that, in practice, careful tuning of the acceptance region of the decision rule is required to obtain an acceptable rate of false positives and negatives[3].

This completes the description of Methods 0 and 1. Method 2 is now presented.

5.4.2 Filter consistency test

Method 2 is proposed as an alternative for scenarios in which the admissible region for $\hat{\boldsymbol{x}}_2$ is unbounded in all directions. This may occur in FNE games where $[S_i]$ and $[S_j]$ tend to $\pm\infty$ at

[3] *Widening* the acceptance region is achieved by *reducing* the value of α for the false-positive rate.

similar rates. In this case, differences in the estimated densities of $\hat{\boldsymbol{x}}_1$ and $\hat{\boldsymbol{x}}_2$ are taken as indicators of engagement instability. If the assumptions underlying the constrained estimate $\hat{\boldsymbol{x}}_1$ are valid, then the two estimates should achieve the same values as long as the estimators are well-tuned. If the two estimates are seen to disagree significantly, this is interpreted as indicating the assumptions of the constrained estimate are validated and the underlying engagement is unstable.

Method 2 directly compares the estimate $\hat{\boldsymbol{x}}_1$ to the density defined by $\mathcal{N}(\hat{\boldsymbol{x}}_2, [P_2])$. In developing a test for "agreement" between the two estimators, some practical considerations must be made. At first glance, both estimators approximate the underlying variables as multivariate normal distributions, and one could implement a test for statistical consistency of, e.g., $\hat{\boldsymbol{x}}_1 - \hat{\boldsymbol{x}}_2$, which has a known mean and covariance. In practice, performance of the estimators is sensitive to process noise tuning, and it is not possible to ensure statistical consistency without a sophisticated process noise model that is adaptive to the particular nonlinear dynamics of the system. It is felt that such an approach would be time-consuming to develop and would eventually require ad hoc modifications to achieve acceptable error rates. Instead, a test is developed by comparing the estimate $\hat{\boldsymbol{x}}_1$ to the density described by $\mathcal{N}(\hat{\boldsymbol{x}}_2, [P_2])$. The nonparametric filter approximates the time-varying dynamics of the opponent gains as a white-noise process, which is inherently a crude approximation. It is expected this filter will have a larger covariance than Filter 1 under normal operation. Furthermore, Filter 2 is expected to lag the true gain, which manifests as a gain estimation bias. Because of this bias, $\hat{\boldsymbol{x}}_2$ is unlikely to look like a random variable drawn from $\mathcal{N}(\hat{\boldsymbol{x}}_1, [P_1])$, but it is plausible that $\hat{\boldsymbol{x}}_1$ could look like a random variable drawn from $\mathcal{N}(\hat{\boldsymbol{x}}_2, [P_2])$ due to the larger value of $[P_2]$. This test is implemented as follows by comparing sequential estimates from the two filters:

$$\sum_{k=1}^{N} (\hat{\boldsymbol{x}}_{1,k} - \hat{\boldsymbol{x}}_{2,k})^T [P_{k,2}]^{-1} (\hat{\boldsymbol{x}}_{1,k} - \hat{\boldsymbol{x}}_{2,k}) \underset{H_0}{\overset{H_1}{\gtrless}} F_{\chi^2}^{-1}(1-\alpha, Nn) \tag{5.9}$$

Eq. 5.9 explicitly assumes a sum over multiple times t_k. However, the estimate of $\hat{\boldsymbol{x}}_2$ is expected to be biased, and the difference $\hat{\boldsymbol{x}}_{1,k} - \hat{\boldsymbol{x}}_{2,k}$ is likely to have a nonzero average value. The value of N should not be chosen to be too large, or the bias error will lead to an excess number of false positives. In practice, it is suggested that a minimum of K consecutive test statistics that reject the null hypothesis should be required to detect an unstable engagement. That is, if Eq. 5.9 is evaluated after each filter update over a sliding window for some short N, and the left-hand side exceeds the right-hand side K consecutive times, then the hypothesis of a stable engagement should be rejected. Requiring K consecutive rejections of the null hypothesis improves the robustness of the test to outliers and reduces false positives during transients.

This section has presented three statistical tests for evaluating if parallel estimators support the hypothesis of a stable engagement. Methods 0 and 1 are based on comparing the estimated value of $\hat{\boldsymbol{x}}_1$ against the set of opponent gains for which stable engagements are possible. Because these methods depend only on outputs from one estimator, they are felt to be less sensitive to tuning and statistical conditioning than Method 2, and are preferable in general. However, Methods 0 and 1 can only be applied to the OSO engagement. Method 1 better accounts for interdependence among elements of the opponent gain matrix, but may form a nonconvex problem that is more difficult to solve online. Method 0 should be used only when the demands of realtime applications prohibit Method 1 from being used. Method 2 directly compares the state estimate from filter 2 against the distribution described by filter 1. The test is sensitive to errors in both filters and is likely to be more difficult to tune so that acceptable rates of false positives and false negatives are obtained. However, Method 2 can be implemented even in the FNE game where no limits on the opponent gain exist.

This section has discussed the problem of detecting when an opponent's gains lead to an unstable engagement. The next section applies the parallel estimation algorithm to the scalar problem introduced earlier in the chapter.

5.5 Numerical example

In this section, the full process of stability-constrained LQDG estimation is demonstrated through a numerical example. The LQDG with scalar dynamics used in Sec. 5.1 is considered. The pursuer always uses the adaptive OSO strategy against a minimax evader. To begin, the Chebyshev function approximation to the stability bound in parameter space is computed and presented. Subsequently, the performance of constrained and unconstrained LQDG estimators for nearly unstable

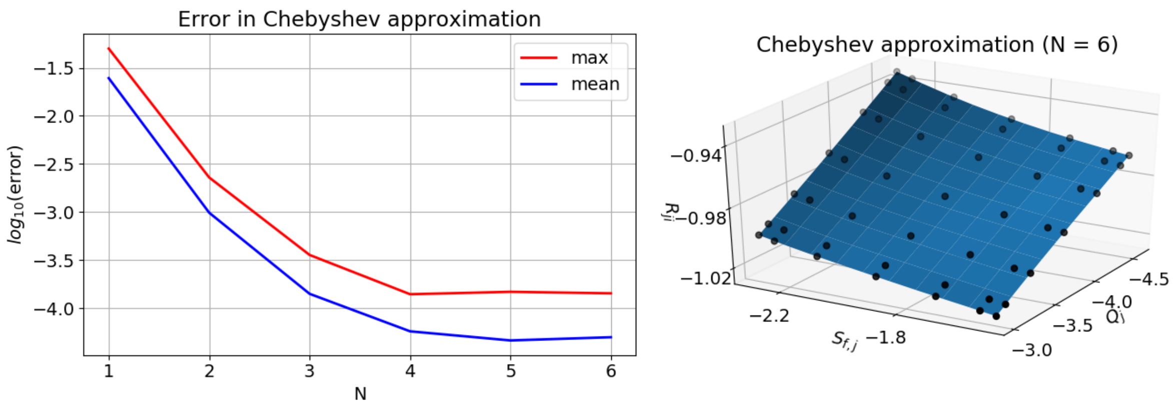

Figure 5.6: Convergence of Chebyshev interface approximation for increasing number of basis functions

values of the evader parameters are compared in terms of estimation errors. Monte Carlo analysis is also used to compare a large number of metrics between the unconstrained estimation and the various constrained estimation implementations considered in this chapter. Finally, the parallel estimation approach is used to classify engagements as stable or unstable. The effectiveness of the three methods for evaluating engagement stability is compared in Monte Carlo simulations.

5.5.1 Stability constraint approximation

To begin, the Chebyshev function approximation to the stability interface is computed. r_{ji} is treated as a dependent function of q_j and $s_{f,j}$ along the interface. Fig. 5.6 shows the convergence of the Chebyshev approximation for increasing numbers of basis functions. On the left, the maximum error and the average error between the approximation and truth values at gridded points are plotted. The grid points are the CGL nodes associated with polynomial degree $N+1$; this choice is intended to evaluate the goodness-of-fit in between CGL nodes at degree N. On the right, the Chebyshev function approximation to the stability interface and the exact function values for the CGL nodes at $N = M = 6$ are plotted. The results show sharply diminishing returns for increasing N above 4. The Chebyshev approximation of the stability interface on the right appears to be qualitatively reasonable. Based on the results, $N = 4$ is an appropriate polynomial degree for approximation, and is used in the subsequent simulations.

On reviewing Fig. 5.6, it is clear that the numerical approximation to the stability manifold is dependent on $s_{f,j}$, even though it is asserted in Sec. 5.1 that the analytical stability constraints depended on r_{ji} and q_j alone. This is an effect of the definition of numerical stability, which is evaluated over the finite time $t \in [0, T_f]$. If a system becomes singular in the neighborhood of $t = 0$, the terminal value of $s_{f,j}$ influences how quickly the Riccati matrix grows to exceed the defined numerical limits for nonsingularity. By increasing T_f arbitrarily, the stability manifold approximation can be made to approach the conditions for existence of a steady-state RDE solution. This is demonstrated in Fig. 5.7, which plots the analytical bounds of Fig. 5.1 with numerical approximations for increasing T_f at constant values of $s_{f,j}$. The dashed lines indicate the extrema of q_j used for Chebyshev basis function fitting, and the blue and red regions denote the stable and unstable regions, respectively, identified in earlier testing. Fourth-degree polynomials are used. For short time horizons, the instability at small magnitudes of q_j is poorly approximated by the basis function fit. This is because, for small q_j, the rate of growth in the Riccati matrix is slow, and the numerical tolerance is not exceeded on short time horizons. As T_f increases, the stability manifold approximations approach the analytical bound, and the bound can be approximated to any desired precision by increasing T_f further.

A convergence study has been conducted to determine the appropriate order of Chebyshev polynomials for this problem. Further, it has been shown that the analytical problem stability bounds of Section 5.1 are recovered by the numerical approximation in the limit as the time horizon becomes large. Next, the obtained Chebyshev constraint approximation is used to apply constrained estimation to the adaptive OSO LQDG strategy.

5.5.2 Constrained estimation effectiveness

This subsection compares the estimation error and failure rate of constrained and unconstrained estimators for nominally stable values of the opponent parameters. To highlight the differences in the two cases, values of q_j and $s_{f,j}$ are sampled from an initial covariance, and the corresponding value of r_{ji} lying *on* the Chebyshev constraint surface is chosen for the nonadaptive evader's weights. This results in a large number of nearly unstable engagements in which unconstrained

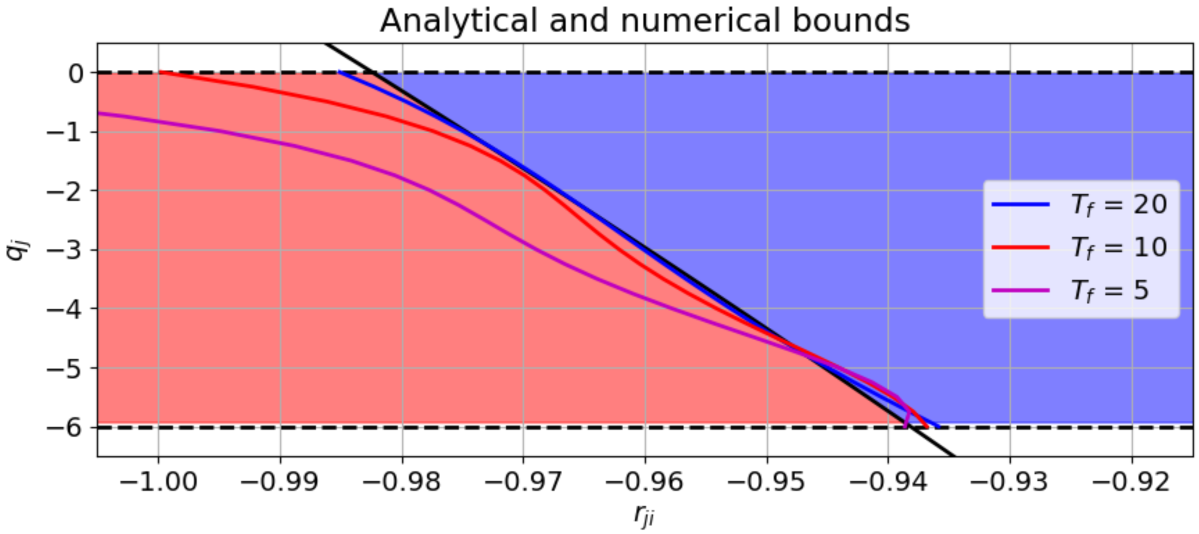

Figure 5.7: Comparison of the analytical stability manifold and its approximation for increasing values of T_f

estimation fails at a relatively high rate. To demonstrate the utility of the constrained estimation approach, the performance of the constrained estimator is compared against the unconstrained UKF. First, individual simulation runs are highlighted for discussion. Second, a small Monte Carlo analysis is conducted to evaluate trends in the performance of the constrained and unconstrained estimation algorithms.

Figs. 5.8 and 5.9 show estimation performance for simulations using constrained and unconstrained estimation, respectively. A fixed set of opponent weights lying on the stability interface is used: $q_j = -3.43, r_{ji} = -0.972$, and $s_{f,j} = -1.90$. With constrained estimation, the estimator performance is qualitatively good. The pursuer uses the adaptive OSO strategy and the evader uses the minimax strategy. The Simon constraint algorithm is applied, using the nonrecursive constraint strategy with the approximate stability manifold identified in the previous subsection[4]. The state estimation errors and covariances typically converge to small values by the end of the time horizon. The errors in the weight estimates exhibit a bias for most of the simulation, but this is typical of the performance achieved in earlier LQDG examples. Comparing this case against Fig. 5.9, it is

[4] Qualitatively similar histories are achieved using the recursive constraint strategy.

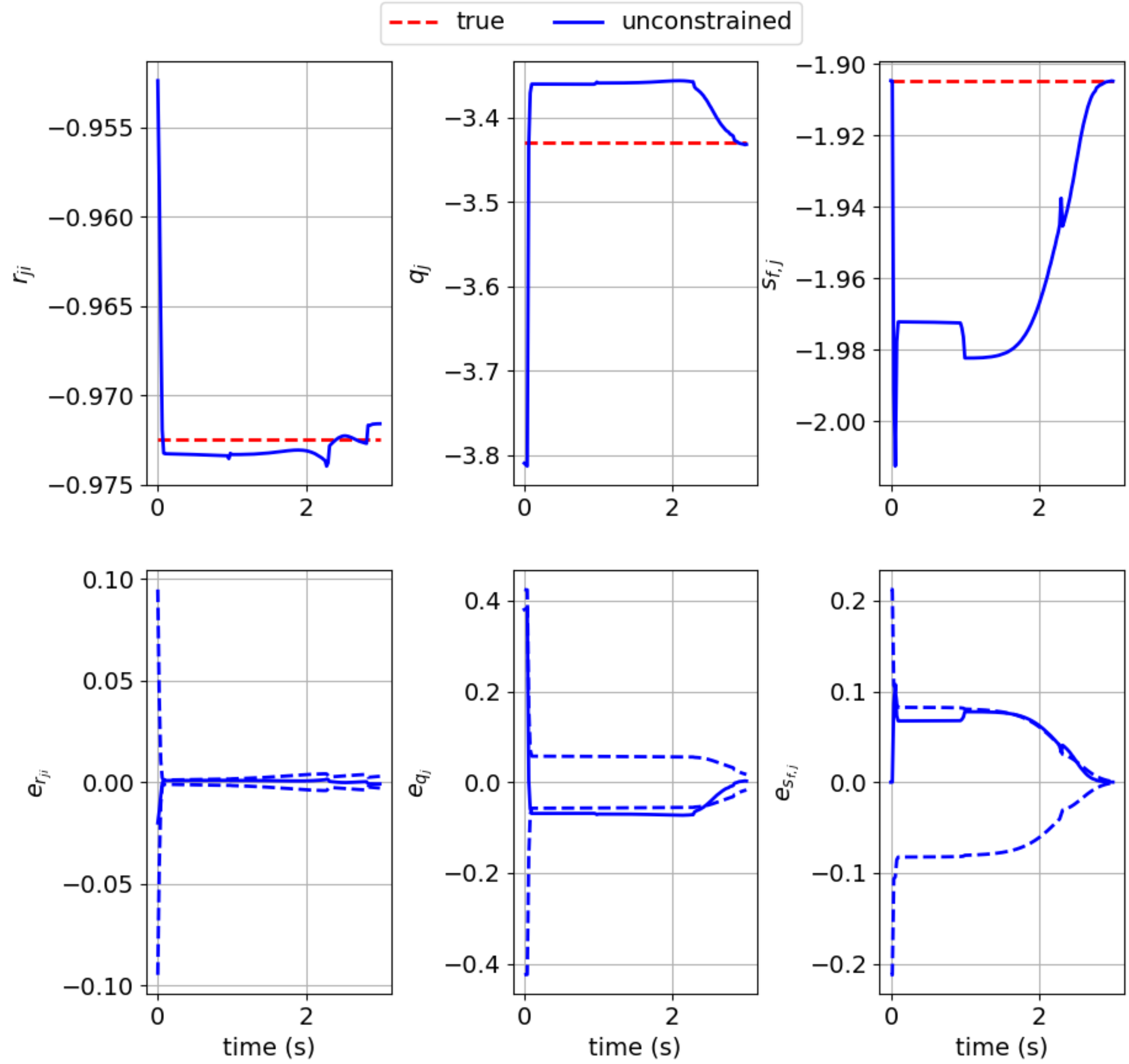

Figure 5.8: Individual simulation estimation performance using constrained estimation. Top: true states and estimates. Bottom: estimation errors (solid line) and 3σ bounds (dashed lines).

clear that the unconstrained estimator performs poorly in the same scenario. The unconstrained estimator fails after the second measurement update and there is no subsequent measurement history. The constrained and unconstrained estimates have the same state and covariance history up until the time the unconstrained estimate fails. The constrained estimate history continues after experiencing a large step change in the state $s_{f,j}$. That step change corresponds to the constraint violation being detected, triggering the PDF truncation and producing a sharp drop in the covari-

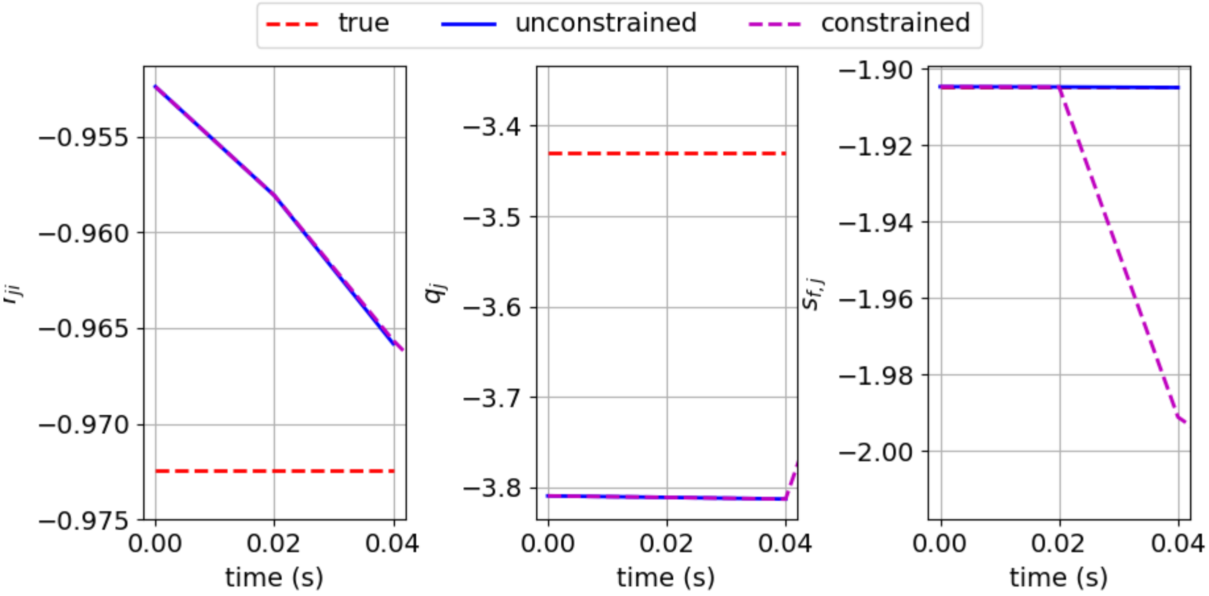

Figure 5.9: Individual simulation estimation error histories using unconstrained estimation

ance. To further quantify the difference in performance, Monte Carlo simulation results are now presented.

For Monte Carlo simulations, the initial state and opponent weight values are sampled according to a Latin Hypercube approach. The initial state is drawn from $\mathcal{U}(-25, 25)$. The opponent weights are sampled on the Chebyshev stability interface, such that $q_j \sim \mathcal{N}(3.81, 2 \times 10^{-2})$ and $s_{f,j} \sim \mathcal{N}(-1.90, 5 \times 10^{-3})$, and r_{ji} is the corresponding value on the stability interface. This sampling scheme is used to ensure the unconstrained estimator fails in a large number of cases. The sampled states and weights are then applied to each estimator. Each estimator experiences the same conditions, up to the measurement noise, which is chosen to be small. For each estimator, the following performance metrics are considered. A simulation is considered to have failed when the commanded control exceeds a predefined threshold of 10^{10}. The number of failures, N_{fail}, and the statisical mean μ_{T_f} and standard deviation S_{T_f} associated with the failure time is tabulated for each estimator. The mean square error (MSE) in the estimates of evader weights are also tabulated.

Table 5.2 compares the performance of the various constrained estimation approaches against the unconstrained UKF in Monte Carlo simulation. The constrained estimation strategies use the

Case	Unconstrained	Tully Recursive	Simon Recursive	Tully Nonrecursive	Simon Nonrecursive
N_{fail}	403	177	173	167	171
μ_{T_f}	1.71	1.59	1.52	1.62	1.59
S_{T_f}	1.44	0.757	0.79	0.711	0.752
$MSE(r_{ji})$	2.88×10^{-6}	3.82×10^{-6}	3.76×10^{-6}	3.71×10^{-6}	3.02×10^{-6}
$MSE(q_j)$	0.000124	0.00101	0.00148	0.00119	0.00159
$MSE(s_{f,j})$	4.56×10^{-8}	2.07×10^{-6}	1.05×10^{-5}	2.49×10^{-6}	4.86×10^{-6}

Table 5.2: Summary of Monte Carlo simulations of unconstrained and constrained estimation for scalar test cases sampled from the stability interface. 1000 simulations are executed for each case.

constraint algorithms of either Simon[64] or Tully[65], and use either the recursive or nonrecursive formulations of PDF truncation, as discussed in Sec. 5.3. In the table, $\mu_{(\cdot)}$ and $S_{(\cdot)}$ denote the sample mean and standard deviation of variables. All four combinations of constrained estimation implementations are compared. Their error metrics display small differences between the different methods. Among the mean square error metrics, the constrained estimators uniformly have an error of about an order of magnitude larger in estimating q_j. The errors for all estimators in estimating r_{ji} are of the same order of magnitude. The unconstrained estimator achieves a consistently lower MSE metric in r_{ji}. The unconstrained estimator achieves a MSE on the order of 10^{-6} for $s_{f,j}$. This is two orders of magnitude lower than most of the constrained estimation algorithms, and three orders of magnitude lower than the Simon algorithm using the recursive implementation.

The main findings of Table 5.2 are now summarized. The simulations with constrained estimation finish at about twice the rate as the unconstrained UKF. The average simulation time at failure is similar for all five cases, although the standard deviation is much smaller for the constrained estimators, which suggests a more consistent failure modality in those simulations. The error metrics are generally one or two orders of magnitude larger when constrained estimation is used instead of unconstrained estimation. Overall, the Monte Carlo results validate that the constrained estimation approach succeeds in significantly reducing the number of simulation failures. The constrained estimators show order-of-magnitude or greater increases in the MSE metrics for estimating q_j and $s_{f,j}$. This is consistent with the expectation that the PDF truncation algorithm leads to biased state estimates.

This subsection has compared the failure rates of constrained and unconstrained estimation algorithms in a LQDG. For the final subsection, the parallel estimation algorithm is considered and the effectiveness of the various suggested statistical tests is evaluated.

5.5.3 Instability detection tests

The numerical results so far have shown the performance of the numerical stability interface approximation and the constrained estimation algorithm. Now, the effectiveness of the procedure for detecting engagement instability is evaluated. For this procedure, the opponent weights are drawn from a Gaussian distribution that is chosen so that about one-third of samples fall in the unstable region: $P_{q_j,q_j} = 2 \times 10^{-2}, P_{r_{ji},r_{ji}} = 10^{-3}, P_{s_{f,j},s_{f,j}} = 5 \times 10^{-3}$. Latin Hypercube sampling is employed to ensure the sampled values approximate the distribution. Each of the three methods proposed in this chapter for evaluating agreement between the parallel estimators is implemented. The effectiveness of the statistical test is evaluated in terms of its rates of false positives and false negatives. As reported, false positive rate denotes the fraction of stable engagements that are incorrectly classified as unstable, and false negative rate denotes the fraction of unstable engagements that are classified as stable. The number of simulations that failed for other reasons, such as singular covariance or excessively large control, are also compared. The latter can occur when an unstable engagement is not detected before the control becomes large.

Each of the three statistical tests is individually tuned on a small number of simulations to achieve an acceptable compromise between false positives and false negatives. For Method 0, the most recent three estimates are used to compute the test statistic, and one exceedence of the threshold classifies an engagement as unstable. Method 1 uses the last three estimates three estimates with three consecutive exceedences to detect instability. Method 2 uses the last two estimates and requires five consecutive exceedences to detect instability.

Table 5.3 reports results for 1000 simulations using each method. It is clear that the greatest challenge is achieving an acceptable rate of false negatives. As seen in the table, Method 0 has the best performance in terms of error rates, with no false positives and about a 26% false negative

Method	False positive rate	False negative rate	Unstable engagements	Stable engagements	Other failed simulations
Method 0	.000	.264	352	648	52
Method 1	.00779	.355	358	642	19
Method 2	.0324	.575	351	649	47

Table 5.3: Comparison of detection performance with 1000 Monte Carlo simulations

rate. Method 1 has a small number of false positives (less than 1%), and about 36% false negatives. Method 2 achieves poor performance, having approximately a 3.2% false positive rate and 58% false negative rate. Method 2, in particular, was found to be very challenging to tune. In terms of other sources of simulation failure, Method 1 has just under half as many failures as the other two cases. Failures occur when a large control or state value is attained, and are prevented when singular engagements are correctly identified before a large control is commanded. The total number of failed simulations is too small for statistical significance, and it is concluded that failures occur at a rate of about 2%–5% in all methods.

This chapter leverages constrained estimation to overcome potentially nonsingular solutions to the generalized RDEs governing the LQDG solutions introduced in Chapter 2. The effective limits on opponent weights for nonsingular solutions are found approximately by numerical integration and are referred to as the "stability manifold." Chebyshev polynomials are used to approximate the stability manifold for online operations. Using the approximate stability manifold representation, constrained estimation is shown to significantly reduce the number of simulation failures due to singular RDE solutions for true opponent weights lying on the stability manifold. By comparing constrained estimates against a nonparametric unconstrained estimate of the opponent gain, approaches for detecting unstable engagements are presented and tested. In tests, the proposed approaches are shown to achieve low rates of false positives, but somewhat high rates of false negatives, in which unstable engagements are not classified as such.

Not studied in this chapter is the question of what an adaptive player should do when he recognizes an engagement for which he has no finite-valued solution. In some applications, if a player has sufficient control authority, he may be able to change his game weights to obtain a stable strategy. This may not be practical for control-limited players. Nonetheless, the concept of updating the game weights to adapt to the larger-than-expected opponent control is an appealing option, and has implications for the $H\infty$ control problem, as well. This idea is suggested as an area for future research, but is not explored further in this dissertation. Instead, the next chapter finally returns to an important fundamental question: how can a player recognize when an opponent is using a particular strategy, such as the minimax strategy?

6. MULTIPLE MODEL ESTIMATION FOR UNCERTAIN OPPONENT STRATEGIC FORM

This dissertation so far as considered the problem of how a player in a differential game should respond to an opponent who uses a *known* strategy with some inexactly known weights. In Chapter 2, solutions are suggested based on appropriate assumptions about the sophistication of the opponent. Chapters 4 and 5 suggest incremental improvements that address some limitations of the initial implementation of those ideas. Qualitatively speaking, the suggested methods improve a player's response when the opponent's strategy is *more favorable* than expected and offer a means to recognize when the opponent's strategy is *less favorable*.

Thus far, nothing has been said about how a player can know the form of his opponent's strategy sufficiently well that the adaptive gameplay can be used. The adaptive OSO and FNE strategies support *parametric uncertainty* in the opponent strategy, but they assume the *form* of the strategy itself is known. It is useful to differentiate between these two sources of uncertainty. For the remainder of the dissertation, "parametric uncertainty" refers to uncertainty in the exact numerical values of a particular representation of an opponent's strategy, and "uncertainty of form" to uncertainty in the true representation of the opponent's strategy. In general, an opponent is free to use any objective function or method to derive his control strategy, and the opponent may not have the same objective as the adaptive player. This potential challenge has of course been recognized in the literature on games. Various learning-based or adaptive methods have been proposed. For instance, Ref. [34] uses an actor/critic learning structure to obtain iteratively better solutions to the Hamilton-Jacobi-Isaacs (HJI) PDE that governs differential games. Satak [12] uses a similar approach to obtain basis function approximations to the HJI equation.

These generic learning-based methods achieve the objective of identifying another player's contribution to an agent's performance, but they are not an appropriate solution in all domains. Learning-based methods require data to achieve precise convergence, and may not be appropriate in scenarios with short engagement times. In such a situation, an agent may be better served by exploiting domain knowledge to suggest a set of probable opponent strategies subject to parametric uncertainty. If any of the models match the observed opponent behavior closely, then the adaptive agent can still enjoy the benefits of estimation-based adaptation to learn the opponent's effective weights. In an effort to address uncertainty in the form of the opponent's strategy and account for the possibility of switching-based opponent deception, this chapter applies Interacting Multiple Model (IMM) estimation to LQDGs. IMM estimation is an approach developed in the field of target tracking for estimating the state of a target with hybrid dynamics – that is, the dynamic equations are described by one of a set of candidate models.

The use of adaptive strategies in differential games suggests new avenues for *opposing* game strategies. There is inherently a sort of leader-follower mismatch between a static player who chooses to play a game according to a fixed strategy and an adaptive player who observes the opponent and tries to optimize her performance with respect to the observed behavior. Adaptive control strategies in general can be thought of as attempts to make a measured system response look like a desired system response. If a player in a game knows her opponent is using an adaptive strategy, then the player can potentially exploit the adaptation. For example, a player could initially play according to one strategy and wait for the opponent to learn that initial strategy. The player would then execute an abrupt change to some true strategy, gaining an advantage during the transient when the opponent is learning. There is some evidence in the literature to support the use of these strategies; Turetsky and Shima found that an attacking missile benefited from switches in its feedback gain during an engagement, even if its target acted optimally[28]. This concept can be viewed as related to the concept of Stackelberg (leader-follower) equilibria in dynamic games [2, p. 131]. This chapter only explores this idea at a very simple level. Nonetheless, the possibility of such switching-based deceptive strategies is something for which an adaptive player wants to account.

The remainder of this chapter is organized as follows. First, IMM estimation is applied to LQDGs to distinguish among candidate opponent strategies and implement an appropriate control. Then, two approaches to obtaining control solutions from the IMM estimates are discussed. In the next section, the LQDG solution against a constant-gain opponent is introduced as an alternative strategy that an opponent might employ in an incomplete information game. Subsequently, the idea of deceptive switching-based strategies is investigated by using a surrogate, reduced-order model to approximate the learning rate of an IMM player. Finally, the proposed deceptive strategies are evaluated against an opponent using an IMM. To begin, IMM estimation is introduced.

6.1 Interacting Multiple Model estimation background

In the area of target tracking, multiple-model (MM) estimation has proven to be a popular approach for performing estimation of the state of a maneuvering target[66]. Broadly speaking, it is assumed that the dynamic system obeys one of a discrete number of candidate dynamic models; in this sense, the dynamic system is assumed to be a hybrid dynamic system. Furthermore, the system is assumed to be linear with Gaussian error sources, or to be approximated as such[1, p. 441]. The multiple-model approach uses a Bayesian framework to obtain estimates of the probability that the system is in a particular mode by propagating a bank of filters and updating the mode probabilities based on their associated measurement likelihoods. In this dissertation, each mode is assumed to correspond to a particular opponent strategy; e.g., mode 1 might be a minimax strategy and mode 2 an adaptive FNE strategy. Given the system state and mode probabilities, game-theoric reasoning can be applied to evaluate and implement candidate strategies. This is the focus of the later sections in this chapter. In the present section, the Interacting Multiple Model (IMM) estimator, which is used for MM estimation, is introduced.

MM estimation seeks to obtain state, covariance, and mode probability estimates for a system with hybrid dynamics. Numerous algorithms and permutations of multiple model estimation exist, and a full review is beyond the scope of the present section. Most approaches share a common structure; a bank of estimators is constructed, often according to Kalman filtering, and the estimate state and covariance is some weighted sum of the estimates of each filter, with weights proportional to some measurement likelihood function. Approaches can be classified as either static or dynamic, in the sense that a static MM estimator assumes the target's mode is constant, and dynamic MM estimation allows for mode switching[1]. The most common static MM estimator is also known as the multiple model adaptive estimator (MMAE), and has been used in many pursuit-evasion applications, such as in Refs. [67, 68, 27]. The assumption of a constant target mode is generally valid for scenarios in which engagement times are very short. When the target mode is allowed to change, it is modeled as a Markov process. Applied rigorously, this assumption requires an exponentially increasing number of filters as more observations become available. To make the problem tractable, suboptimal filters are implemented that combine the histories of models in previous observations. The most common approaches in the literature are the first- or second-order generalized pseudo-Bayesian (GPB) models and the IMM. For r candidate modes, the first-order GPB and IMM algorithms require r parallel filters, while the second-order GPB requires r^2 such filters[66, 1]. The IMM estimator has proven to achieve a good compromise between complexity and performance in applications[66]. It was first introduced by Blom in Ref. [69] as an approximation to the performance of the minimum mean square error estimator for hybrid dynamic systems with mode switching. The IMM has seen wide applications for tracking of maneuvering targets in air traffic control[70], spacecraft[71], and human-robot interaction[72].

For brevity, the IMM algorithm is omitted from this section and summarized in Appendix B. Table B.3 summarizes the complete algorithm. Essentially, the IMM algorithm approximates the Markov process associated with mode changes by performing a likelihood-weighted mixing step at the beginning of each iteration. Then each filter in the ensemble is propagated and updated in the usual fashion for Kalman Filter-based estimators. Mode probabilities are updated according to the measurement likelihood of each filter's estimate. An additional output mixing step is performed if the state is to be used for feedback control. The state mixing step applies only to state elements that are estimated by all models. For example, consider a target tracking application in which two modes respectively assume constant velocity and constant acceleration, and estimate the associated states. The position and velocity level states are mixed, but not the acceleration level states. This dissertation follows the example of Ref. [73] to obtain unbiased estimates by not mixing the unlike states.

This completes the introduction of the IMM estimation algorithm. The next section addresses the process of obtaining a control solution in a differential game from an IMM estimate.

6.2 Game solution under mode probability framework

The IMM estimation algorithm produces three basic outputs: a state estimate, an error covariance estimate, and a mode probability estimate. This section considers two approaches to generating a control solution from these outputs. The first is a standard practice in multiple model filters and uses a weighted sum over the control that would be used for each mode. The second approach is based on a matrix game analysis of the cost-to-go for the adaptive player. The latter approach incurs a computational penalty, but it allows a player to explicitly compare tradeoffs between various possible outcomes weighted by their modal probability. Each approach is described in the following subsections.

6.2.1 Modal probability-mixed control

This control approach is a common practice in MM-based systems, and in the dissertation it is referred to as "IMM control" (IMM-C) for convenience. The IMM state output is a weighted sum of the estimated states from each modal filter (see Eq. B.24). The state is viewed as a linear combination of the states in the various sub-filters. Common practice in control of multiple-model systems has been to assume a similar form for the system control[74, 75]:

$$\boldsymbol{u}_k = \sum_{j=1}^{r} \alpha_j \boldsymbol{u}_k{}^{(j)}, \;\; \sum_{j=1}^{r} \alpha_j = 1 \tag{6.1}$$

The controls $\boldsymbol{u}_k{}^{(j)}$ are designed for the dynamic system described by each sub-filter individually. Approaches are typically either minimum mean squared error (MMSE) or maximum a posteriori (MAP) [27, 76]. In the MMSE case, the weights α_j used are equal to the mode probabilites $\mu_k^{(j)}$. In the MAP case, α_ℓ is chosen to be 1, where $\ell = \mathrm{argmax}(\mu_k^{(j)})$, with all other weights equal to zero. In this dissertation the MMSE approach is used. $\boldsymbol{u}_k{}^{(j)}$ is obtained by computing the game-based control for each individual subfilter.

The IMM-C approach is simple to implement, and the feedback controls associated with each mode are generally computed already as part of the adaptive LQDG solution. The method does not explicitly account for tradeoffs between the various opponent strategies, which limits its ability to identify or anticipate possible opponent switches. Additionally, the use of modal probability weights can degrade performance in practice. Modal probabilities are likelihood-weighted and are prone to small "jumps" or "spikes" when two modes have similar likelihoods for a particular measurement. If the two modes have similar control values, there is no cost, but if the modes have very different controls, then the IMM player may display step changes in his control that degrade performance. Having introduced the IMM-C procedure, the matrix game-based control is now discussed.

6.2.2 Matrix game-based control

As an alternative to the IMM-C approach, the problem of selecting a controller for player i can be reformulated as a matrix game. A brief review of matrix games is now presented. In a traditional, static matrix game in two dimensions, an $n \times m$ matrix with known values is given. Player 1 selects a matrix row and Player 2 selects a matrix column[1]. The players receive values equal to the entry in the matrix at the chosen row and column. In the common, zero-sum formulation, one player is minimizing and one player is maximizing, leading to an equilibrium in either pure or mixed strategies[2, p. 18–36]. In pure strategies, a player always picks a certain column or row; in mixed strategies, a player chooses randomly according to certain probabilities to maximize his expected return. For a matrix game given by matrix $[C]$, the value of the game can compactly be computed as $\boldsymbol{u}^T[C]\boldsymbol{\mu}$, where $\boldsymbol{u}$ and $\boldsymbol{\mu}$ are the mixed strategy probabilities used by Players 1 and 2, respectively. This basic knowledge of matrix games is sufficient for the current discussion.

As an example of a static matrix game, consider Table 6.1. P_1 wants to minimize the game value and P_2 wants to maximize it. P_1 has a "security strategy," akin to a minimax strategy, by picking row 2. This choice ensures a minimum value of the game for the worst-case opponent choice. The value of the security strategy is 1. P_2's security strategy is to choose column 2. The security strategies are not in equilibrium, however. E.g., if P_1 knows P_2 will play a security strat-

[1] Depending on whether the players choose simultaneously or sequentially, the game nature and equilibrium may be altered significantly.

	P_2	
P_1	3	0
	-1	1

Table 6.1: Example 2-dimensional matrix game adapted from Ref. [2, p. 29]

egy, then P_1 should choose row 1 to obtain the game value of 0. This game admits an equilibrium in mixed strategies, corresponding to $\boldsymbol{u} = [\frac{2}{5}, \frac{3}{5}]^T$ and $\boldsymbol{\mu} = [\frac{1}{5}, \frac{4}{5}]^T$. The equilibrium value of the game is $\frac{3}{5}$.

In the context of the uncertain differential game that is the focus of the present chapter, each column of a matrix game can be seen as corresponding to a possible strategy of player j, with each strategy being a mode used in an IMM estimator. Each row of the matrix then corresponds to the possible responses of player i. The ℓ, m entry in the matrix is then populated by computing the predicted cost-to-go for i's response ℓ if j has chosen strategy m. In this way, the control problem can be viewed as a matrix game having continuous-valued, time-varying entries.

Player i has knowledge of j's possible game strategies and an estimate of the probability that j is using each strategy. She is free to design any number of possible strategies to oppose player j. A reasonable approach for i is to choose as her candidate strategies the optimal strategy to counter each of j's individual strategies. This leads to an $r \times r$ matrix game for player i. The diagonal entries of the matrix are populated by the cost-to-go if i has correctly inferred j's strategy. The off-diagonal entries are populated by the costs-to-go if the strategies are mismatched. For a LQDG, these mismatched costs-to-go are straightforward to evaluate by numerically integrating a Riccati-like equation, as is now discussed.

For the particular case of a LQDG objective function for i, the general cost-to-go can be obtained as a function of a Riccati-like variable. Consider the general two-player LQDG objective function given by Eq. 1.2 with $[R_{ii}] = [\mathbb{I}_{n \times n}]$. Let the control laws of both agents be given by a generic feedback function that can be written as the product of a gain and the state: $\boldsymbol{u}_i = [K_i(t)]\boldsymbol{x}$ and $u_j = [K_j(t)]\boldsymbol{x}$. To evaluate the cost-to-go under this model, the terminal cost can be replaced by an integral cost plus a function of the current state, leading to the cost-to-go equation given below:

$$J_{c2g} = \frac{1}{2}\boldsymbol{x}(t)^T[F(t)]\boldsymbol{x}(t) + \frac{1}{2}\int_t^{T_f} \boldsymbol{x}^T\Bigg([\dot{F}] + [Q_i] + [K_i]^T[K_i] + [K_j][R_{ij}][K_j] + \\ ([A] - [B_i][K_i] - [B_j][K_j])^T[F] + [F]([A] - [B_i][K_i] - [B_j][K_j])\Bigg)\boldsymbol{x}\,\mathrm{d}\tau \quad (6.2)$$

The cost-to-go is computed by requiring the integral of Eq. 6.2 go to zero, leaving only the running cost expressed as a quadratic function of the state weighted by the current value of $[F(t)]$. The cost-to-go matrix is the solution to the following Riccati-like equation and boundary condition:

$$[\dot{F}] = -([A]-[B_i][K_i]-[B_j][K_j])^T[F]-[F]([A]-[B_i][K_i]-[B_j][K_j])-[Q_i]-[K_i]^T[K_i]-[K_j]^T[R_{ij}][K_j] \quad (6.3)$$

$$[F(T_f)] = [S_{f,i}]$$

In the case where $[K_i]$ is a time-varying gain independent of the state given from a generalized RDE solution and $[K_j]$ is likewise given by the linear feedback assumed in $[K_i]$, then the solution for $[F(t)]$ collapses to the Riccati gain $[S_i]$. Otherwise, Eq. 6.3 is a more general equation that can be used to evaluate the cost-to-go associated with player j using a different strategy than the one used to compute $[S_i(t)]$.

For a standard matrix game, in which each entry is assumed deterministically known, the formulation based on mode probabilities leads to a trivial solution. j's mode probabilities are assumed known. Consider the matrix game with value $\boldsymbol{u}^T[C]\boldsymbol{\mu}$, where $\boldsymbol{u}$ is a vector indicating i's probability of choosing each strategy, and $\boldsymbol{\mu}$ is the vector of mode probabilities from the IMM. Note that $\boldsymbol{u}$ is elementwise constrained to be positive-valued and the sum of its elements must be one. If i is minimizing, as is the convention in this dissertation, then it is clear that the optimal strategy for i to choose corresponds to $\mathrm{argmin}([C]\boldsymbol{\mu})$. That is, i's optimal response to the mixed strategy of j denoted by $\boldsymbol{\mu}$ is a pure strategy, with no mixing or probabilistic elements. However, i's response may not be the control associated with the most probable mode if one or more relatively probable modes with large penalties are present. One shortcoming of this approach is that it fails to account for different amounts of uncertainty between two candidate strategies; for example, if strategy 1 has a smaller mean but very large uncertainty compared to strategy 2, it might be preferable for i to select strategy 2. This limitation suggests an area for future research, but is not considered in this dissertation.

The control approaches reviewed in this section have been based on the idea that i computes control solutions that are optimal for each of j's possible individual strategies. In the IMM-C approach this leads to a weighted sum over the individual control options. In the matrix game approach, the control strategy is chosen to optimize the expected cost. The effect of the matrix game is to explicitly consider the cost incurred by using a mismatched control strategy, which is not considered in the IMM-C approach.

The matrix game-based control approach as outlined does not consider the possibility that the opponent may change his strategy. However, this framework can be extended to identify possible opponent switching strategies and switch times. For example, an opponent typically benefits very little from switching at the end of the engagement because there is little time for the new strategy to take effect. Using this matrix game framework in an offline fashion, the IMM player can identify opponent switching times and strategies that are expected to benefit the opponent. These strategies can then be appended to the set of candidate opponent strategies, potentially improving robustness to opponent switches. Such an approach is beyond the scope of the current work and is left as an avenue for future study.

This section has introduced two approaches to implementing closed-loop control based on IMM estimates of an opponent's mode in a LQDG. The IMM-C approach is based on a typical strategy for control of systems observed by a MM estimator. A matrix game-based control is introduced as an alternative that allows explicit consideration of tradeoffs. This latter option comes at the cost of propagating additional solutions for the assumed cost-to-go at each time, which is related to the generalized RDEs. In the next section, the LQDG solution under a constant opponent gain assumption is introduced for use in later examples.

6.3 Constant gain opponent solution

To evaluate the utility of IMM estimation for the solution of LQDGs under uncertainty, it is useful to introduce competing opponent strategies that may appear. Perhaps the simplest closed-loop opponent strategy is a constant gain model. Under this approach, the opponent uses a constant gain instead of the time-varying solutions of a minimax or FNE strategy. It is convenient to refer to the control strategy used by a linear-quadratic player whose opponent uses a constant gain as the "LQDG-gain" strategy, in the same way the adaptive OSO and FNE strategies are labelled.

Under the constant gain model, the opponent control is given as $-[K_j]\boldsymbol{x}$ at all times. By substituting the opponent's closed-loop dynamics into the linear quadratic performance index of Eq. 1.2, the optimal control solution for the linear-quadratic player can readily be obtained in terms of a generalized RDE:

$$[\dot{S}_i] = -([A]-[B_j][K_j])^T[S_i] - [S_i]([A]-[B_j][K_j]) - [Q_i] - [K_j]^T[R_{ij}][K_j] + [S_i]^T[B_i][B_i]^T[S_i] \tag{6.4}$$

This equation is subject to the usual boundary condition, $[S_i(T_f)] = [S_{f,i}]$, and is optimal if it satisfies the second-order optimality conditions. Obtaining the solution to the LQDG against a constant gain opponent is a simple matter of modifying the approach used to obtain the adaptive OSO solution.

In a LQDG with uncertainty in the opponent gain, a state estimator must be implemented for the relative state as well as the independent elements of the $m_j \times n$ gain $[K_j]$. A straightforward approach is to model the elements of $[K_j]$ as zero-mean white-noise processes. This is used as the standard approach for estimating the gain of a constant-gain opponent. This filter works well and is relatively straightforward to tune. However, it tends to react poorly to discontinuities, as are induced when the opponent changes strategies during the engagement. To obtain robustness to step changes in the opponent gain, it is useful to introduce a variant of this filter.

As opposed to modeling the opponent gain as a zero-mean white-noise process, assume the opponent gain has a nonzero mean. Then the gain is modeled as a time-varying process whose derivative is constant, and the gain and its derivative are estimated. This approach doubles the number of gain parameters that must be estimated and observability should be checked on a problem-by-problem basis. It is only an approximation in the case of step changes in the opponent gain. In practice, this filter works in conjunction with the constant gain filter to achieve robustness to step changes in the opponent gain through the standard mixing process of the IMM estimator. This estimation model is used in this dissertation to track the opponent gain through step changes. Consequently, it is referred to as the "switching gain" model of opponent behavior in later figure legends, as a contrast with the simple "gain" model in which gain derivatives are zero-mean.

The extension of the LQDG solution to the case of a constant-gain opponent is straightforward. Based on numerical experiments conducted later in this chapter and the next, there are two noteworthy qualities of the LQDG-gain problem that are not intuitively obvious. First is that the LQDG-gain and OSO strategies may be quite similar in practice. Consider a player with a linear-quadratic objective function whose minimax gain changes relatively little over an engagement. If the opponent uses a constant gain that is near the initial-time value of the minimax gain, then the player's LQDG-gain solution is likely to be qualitatively similar to the OSO solution. The adaptive player's two candidate solutions are already constrained to have the same terminal gain, and having a similar opponent gain over the simulation generally leads to feedback gain histories. The second point is that the LQDG-gain strategy may become singular for opponent gain values that are smaller than the steady-state RDE solution. In many problems, such as the missile interception problem of Chapter 7, the game objectives dictate weight values that constrain the terminal gain a player can use. This can lead to a decreasing player gain as the engagement proceeds from $t = 0$ to $t = T_f$. When that happens against a constant-gain opponent, sufficiently large opponent gains drive the generalized RDEs to singular values. This limitation must be kept in mind when designing weights for a LQDG-gain player.

This section has introduced the LQDG-gain strategy. The governing equation is straightforward to derive in the same way the adaptive OSO RDE is obtained in Chapter 1. Two estimators are developed to account for parametric uncertainty in the opponent gain. First is a simple, constant-gain model. The second estimator treats the gain as time-varying with a constant rate, and is labeled a “switching gain” model in later examples. Using these two estimators in conjunction with an IMM achieves robustness to step changes in the opponent gain, which the numerical example at the end of this chapter demonstrates. The next section investigates the design of switching strategies to play against an adaptive player and suggests a simplified model for strategy synthesis purposes.

6.4 Deceptive strategies to oppose adaptive agents

The primary utility of the IMM estimator employed in this chapter is its capacity to track targets whose dynamic modes change. In the context of differential games, it is felt that opponents who play in this fashion merit some attention. An agent playing adaptively typically exhibits a lag with respect to any step change in the opponent’s behavior. This lag corresponds to a transient interval during which the adaptive player’s strategy converges to a strategy that properly counters the opponent. In existing literature on learning and adaptation in differential games, the past actions of an opponent are assumed to predict her future actions, and the possibility of a “dishonest” opponent is not considered. Any transient due to convergence of a player’s estimates occurs only at the beginning of the game.

However, the assumption of a fixed opponent strategy fails to account for a possible weakness of an adaptation-based game strategy. If an opponent knows a player is adapting, then potentially an opponent can mislead the player by switching from one strategy to another. In the presence of infinitely fast adaptation, this danger can be ignored because a player can always immediately switch to a new strategy satisfying optimality or equilibrium properties. In practical systems with finite convergence times, this sort of deception could have significant consequences.

This section introduces the concept of deceptive strategies in differential games. First, a mathematical description of the problem and a numerical example to justify the concept are presented. Then, a more extensive example for a problem with scalar dynamics is presented.

6.4.1 Problem description

In the current problem, agent j is assumed to engage in a differential game with an adaptive agent i. To improve his performance in the game, agent j uses a strategy of playing initially in some suboptimal fashion, followed by switching to some "more optimal" strategy[2]. When i's rate of adaptation is finite, j can improve her overall performance by a judicious choice of her initial game play and switching time. This subsection proposes a simplified model of i used by j to design such deceptive game strategies.

For simplicity, the game is assumed to have only one switch in j's control; more switches can be used, but such a study is beyond the scope of this chapter. Thus, the game is divided into two stages; Stage 1 is the period before the switch, and Stage 2 the remainder of the engagement. In Stage 1, player j plays according to a chosen objective function or control structure. Then in Stage 2, j plays according to her "true" objective function, subject to an assumed transient in player i's gain. This transient corresponds to the finite rate of adaptation or learning present in i's gameplay. The choice of j's Stage 1 behavior should be selected to maximize the advantage j enjoys from the induced transient in i's gains when the switch occurs.

To design such a deceptive strategy, j makes several simplifying assumptions. First, i is assumed to adapt "correctly," in the sense that i's initial and final control responses are optimal with respect to whatever the gain used by j is in each stage. Second, j is assumed to implement the switch instantaneously, gaining a substantial "first mover" advantage. Third, i is assumed to have no advance warning or knowledge of the switch and plays according to a constant objective function. Fourth, each player is assumed to have feedback solutions in every stage of the game.

The following mathematical framework is proposed. In Stage 1, j plays according to some chosen feedback strategy, denoted $\boldsymbol{u}_j{}^{(1)}(\boldsymbol{x}, t)$. In Stage 2, j plays according to his true objective

[2] The post-switch strategy may satisfy various properties relating to robustness, game equilibrium, or optimality.

function. Thus j's control may be summarized as a piecewise continuous function:

$$\boldsymbol{u}_j = \begin{cases} \boldsymbol{u}_j{}^{(1)}(\boldsymbol{x}, t) & t \leq t_{sw} \\ \boldsymbol{u}_j{}^{(2)}(\boldsymbol{x}, t) & t > t_{sw} \end{cases} \tag{6.5}$$

$\boldsymbol{u}_j{}^{(2)}(\boldsymbol{x}, t)$ should satisfy some property, such as optimality, in which case it can be written as follows:

$$\boldsymbol{u}_j{}^{(2)}(\boldsymbol{x}, t) = \min_{\boldsymbol{u}_j} \left\{ J_j \Big|_{\hat{\boldsymbol{u}}_i{}^{(2)*}} \right\}$$

$\hat{\boldsymbol{u}}_i{}^{(2)^*}$ denotes j's model of i's control in stage 2. Player j's goal is to choose $\boldsymbol{u}_j{}^{(1)}$ and t_{sw} to optimize the overall value of J_j over the entire simulation, subject to the model of i's control and the governing dynamics. The model of i's control may take any number of forms of various complexity. For example, the adaptive LQDG strategies proposed in this dissertation are non-deterministic, because their control values are a function of measurements with random noise. The IMM-based control strategies are even more difficult to analyze.

To model the opponent strategy, the following simplified model is proposed. Let $\hat{J}_i$ denote player j's assumed knowledge of player i's objective function. If i is an adaptive player, then it is reasonable to expect his control in each stage to approach the optimal control with respect to j's strategy in each stage. Let $\hat{\boldsymbol{u}}_i{}^{(1)}(\boldsymbol{x}, t) = \min_{\boldsymbol{u}_i} \hat{J}_i \Big|_{\boldsymbol{u}_j{}^{(1)}}$, and let $\hat{\boldsymbol{u}}_i{}^{(2)}(\boldsymbol{x}, t) = \min_{\boldsymbol{u}_i} \hat{J}_i \Big|_{\hat{\boldsymbol{u}}_j{}^{(2)}}$. Note an important simplifying assumption: $\hat{\boldsymbol{u}}_i{}^{(2)}$ is evaluated with respect to $\hat{\boldsymbol{u}}_j{}^{(2)}$, which is not equal to the true control $\boldsymbol{u}_j{}^{(2)}$. Rather, $\hat{\boldsymbol{u}}_j{}^{(2)}$ is chosen either to be in Nash equilibrium with $\hat{\boldsymbol{u}}_i{}^{(2)}$ or to satisfy some other reasonable condition, such as having the minimax property. Effectively, $\hat{\boldsymbol{u}}_i{}^{(1)}$ and $\hat{\boldsymbol{u}}_i{}^{(2)}$ are the controls that i would play **if he had an instantaneous, perfect information response to** j. For control synthesis, j then uses the following model for i:

$$\hat{\boldsymbol{u}}_i = \begin{cases} \hat{\boldsymbol{u}}_i{}^{(1)}(\boldsymbol{x}, t) & t \leq t_{sw} \\ \mu(\boldsymbol{x}, t)\hat{\boldsymbol{u}}_i{}^{(1)}(\boldsymbol{x}, t) + (1 - \mu(\boldsymbol{x}, t))\hat{\boldsymbol{u}}_i{}^{(2)}(\boldsymbol{x}, t) & t > t_{sw} \end{cases} \tag{6.6}$$

That is, player j models i as using a convex combination of her "perfect" responses in Stage 2. The coefficients $\mu(\boldsymbol{x}, t)$ should satisfy $\mu\Big|_{t=t_{sw}} = 1$ and $\mu\Big|_{t=T_f} \to 0$, but are left here as generic terms. In this chapter and the next, μ is modeled as a first-order decay term having magnitude $\mu = e^{-\alpha(t-t_{sw})}$.

The control models suggested are chosen to decouple the design of j's strategy from her model of i's strategy. As proposed, the control design iteration for j follows a simple structure. For some objective function J_j and some proposed $\hat{\boldsymbol{u}}_j{}^{(1)}$, j first derives $\hat{\boldsymbol{u}}_i{}^{(1)}$, $\hat{\boldsymbol{u}}_i{}^{(2)}$, and $\hat{\boldsymbol{u}}_j{}^{(2)}$ according to optimal control or game-theoric principles. Then $\boldsymbol{u}_j{}^{(2)}$ is obtained as a function of $\hat{\boldsymbol{u}}_i$. If j chooses her weights and switch time according to an optimal control perspective, then the pre-switch control and t_{sw} are chosen according to some optimality criteria. Such a study is beyond the scope of the present chapter; in this chapter and the next, switching-based strategies are tuned manually. The goal of tuning is to achieve a predicted objective function value that exceeds the expected value if there was no switch in j's strategy.

6.4.2 Proof of concept example

As a proof of concept, consider a simple scalar system having linear dynamics and quadratic objective functions for both players. The players i and j implement adaptive and switching strategies, respectively, according to the following objective functions:

$$J_i = \frac{1}{2} s_{f,i} x_f^2 + \frac{1}{2} \int_0^T \left(q_i x^2 + u_i^2 + r_{ij} u_j^2 \right) \, \mathrm{d}\tau \tag{6.7}$$

$$J_j = \frac{1}{2} s_{f,j} x_f^2 + \frac{1}{2} \int_0^T \left(q_j x^2 + u_j^2 + r_{ji} u_i^2 \right) \, \mathrm{d}\tau \tag{6.8}$$

$$\dot{x} = ax + u_i + u_j$$

For the purposes of synthesizing a strategy, j is assumed to have perfect information about i's objective function. In Stage 1, j uses a constant linear feedback $k_j^{(1)}$. Thus, i's optimal feedback gain is governed by the following OSO RDE:

$$\dot{k}_i^{(1)} = -2{k_i}^{(1)}(a - {k_j}^{(2)}) - q_i - r_{ij}{k_j}^{(2)^2} - {k_i}^{(1)^2} \tag{6.9}$$

$${k_i}^{(1)}(T_f) = s_{f,i}$$

In the same way, j must compute the theoretically optimal feedback gains for Stage 2, ${\hat{k}_i}^{(2)}$ and ${\hat{k}_j}^{(2)}$, having terminal values $s_{f,i}$ and $s_{f,j}$:

$$\dot{\hat{k}}_i^{(2)} = -2{\hat{k}_i}^{(2)}(a - {\hat{k}_j}^{(2)}) - q_i - r_{ij}{\hat{k}_j}^{(2)^2} - {\hat{k}_i}^{(2)^2} \tag{6.10}$$

$$\dot{\hat{k}}_j^{(2)} = -2{\hat{k}_j}^{(2)}(a - {\hat{k}_i}^{(2)}) - q_j - r_{ji}{\hat{k}_i}^{(2)^2} - {\hat{k}_j}^{(2)^2} \tag{6.11}$$

The true feedback gain used by i after the switch is given from ${k_i}^{(2)} = \mu(t){k_i}^{(1)} + (1-\mu(t)){\hat{k}_i}^{(2)}$. Using this model, j computes an optimal feedback gain according to the following closed-loop RDE, again subject to the boundary condition ${k_j}^{(2)}(T_f) = s_{f,j}$:

$$\dot{k}_j^{(2)} = -2{k_j}^{(2)}(a - {k_i}^{(2)}) - q_j - r_{ji}{k_i}^{(2)^2} - {k_j}^{(2)^2} \tag{6.12}$$

j then implements the feedback control $u_j = -k_j(t)x$, in which the gain is a piecewise function as follows:

$$k_j(t) = \begin{cases} {k_j}^{(1)}(t) & t < t_{sw} \\ {k_j}^{(2)}(t) & t \geq t_{sw} \end{cases} \tag{6.13}$$

For j to design this controller effectively, the overall cost-to-go including the effect of the initial suboptimal control should be computed. Using Eq. 6.3, the cost-to-go is straightforward to compute by integrating another RDE subject to the control gains $k_j(t)$ and its analog, $k_i(t)$.

A numerical example of the resulting gameplay is computed using the values tabulated in Table 6.2. The resulting feedback gains and cost-to-go are shown in Fig. 6.1. The left figure shows the gains computed by player i, as well as the gain subject to the assumed adaptation dynamics. The

a	T_f	t_{sw}	${k_j}^{(1)}$	q_i	r_{ij}	$s_{f,i}$	q_j	r_{ji}	$s_{f,j}$	α_{sw}
0.1	5.0	0.4	-1.0	3.95	-1.04	1.95	-3.92	-.952	-1.90	-5.0

Table 6.2: Values used for scalar proof-of-concept switching-based gameplay

center figure shows the gains used by j during gameplay. The right figure shows the value of the cost-to-go variable for player j under the switching-based strategy as well as for a pure FNE strategy. All values are computed independently of the state. The cost-to-go plot shows that, for the assumed model of player i, the designed strategy achieves a lower theoretical cost from the initial time than the pure FNE game strategy. The cost is summarized as the solution to the general RDE of Eq. 6.3, such that the actual cost incurred will vary depending on the initial state. The theoretical cost-to-go variables are -11.1 for the FNE case and -16.5 for the switching-based strategy.

In reviewing Fig. 6.1, it is apparent that benefiting from the switching-based strategy is dependent on j's ability to rapidly implement changes in his control. On the other hand, the proposed strategy is developed assuming i asymptotically achieves perfect knowledge of j's control law, while in reality i is always subject to sensing uncertainty, estimation bias, and other limiting factors. Thus it seems plausible that a strategy such as this could be used to obtain a meaningful benefit in practical scenarios. Of course, the switching-based strategy has been developed assuming player i uses an adaptive strategy. If i instead used a minimax feedback law, then i will be nonreactive with respect to j's switches. In fact, j will incur a penalty for using the suboptimal initial law with no benefit. This is a risk that must be considered in the design of switching-based strategies.

This subsection has introduced the concept of a deception-based switching strategy. A heuristically tuned example has been presented to justify this choice of strategy. The following section presents numerical examples for this chapter, including an example of the switching-based control strategy.

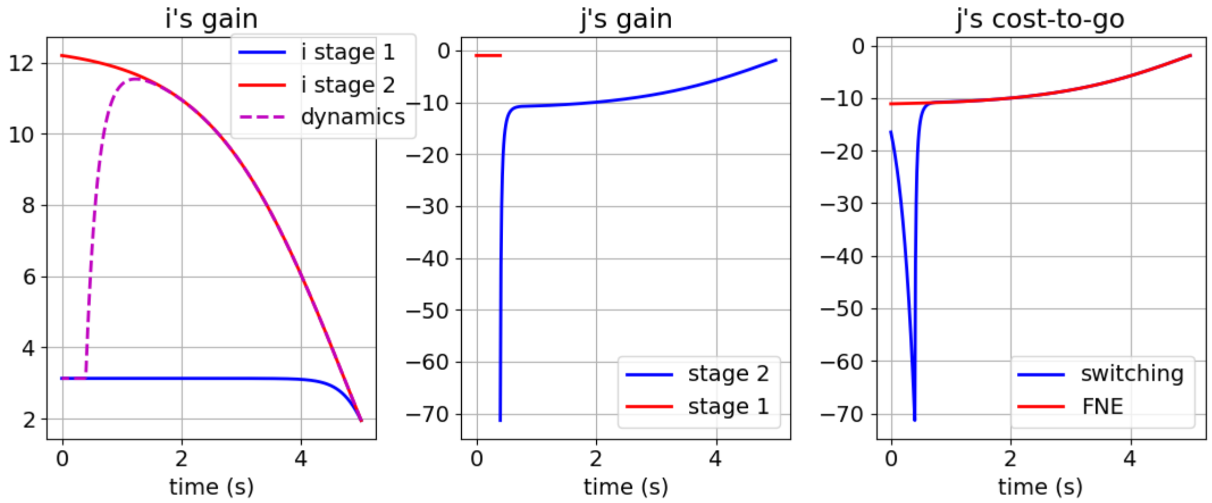

Figure 6.1: Modeled player gains and cost-to-go for the deception-based switching example

6.5 Numerical example

A detailed numerical study is conducted for the problem with scalar dynamics previously used in Sections 4.3.2 and 6.4.2. A new set of problem weights is chosen to highlight qualities relevant to the IMM control problem and switching-based gameplay strategy. These weights are summarized in Table 6.3. For estimation purposes, each agent is assumed to know that $q_i = q_j = 0$, leaving adaptive agents with only two unknown LQDG weights. R_k denotes the measurement variance and Δt the sample period. $P_{q,q}$ and P_{s_f,s_f} denote the initial covariance used for adaptive OSO and FNE estimation.

This section presents results from individual simulations. First, the basic IMM estimation performance is evaluated against simple minimax and constant gain evader strategies. Second, a detailed comparison is made of the IMM-C and matrix game-based control performance against an evader with a static gain. Third, the proposed switching-based nonadaptive game strategy is implemented and demonstrated against an adaptive opponent using IMM estimation.

a	T_f	Δt	q_i	r_{ij}	$s_{f,i}$	q_j	r_{ji}	$s_{f,j}$	R_k	$P_{q,q}$	P_{s_f,s_f}	$x(0)$
0.1	1.0	0.01	0.0	-1.05	7.0	0.0	$-.92$	-6.67	10^{-3}	2.0×10^{-3}	.054	$\sqrt{2}$

Table 6.3: Values used for numerical simulations

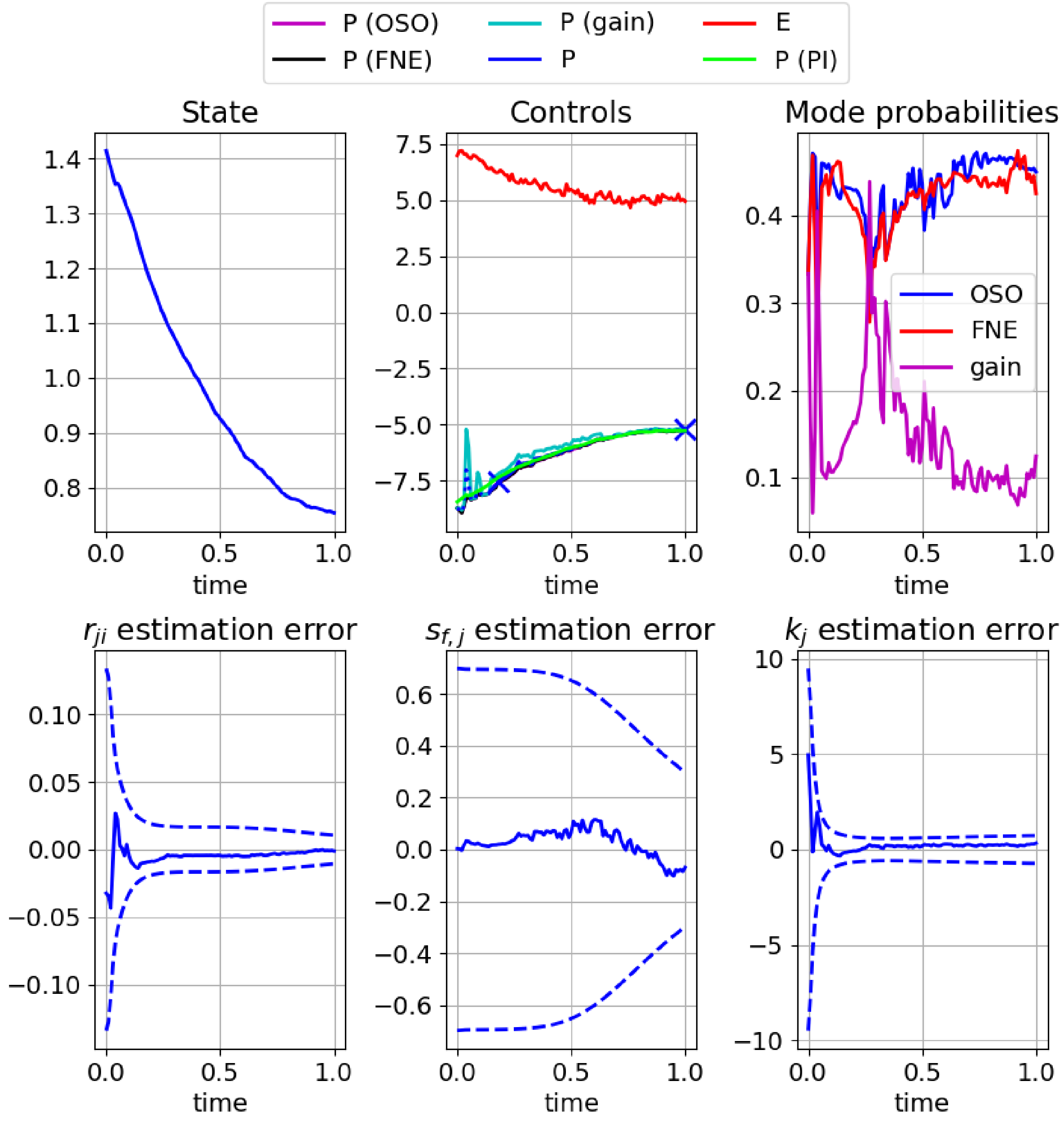

Figure 6.2: Simulation results comparing IMM pursuer against minimax evader

6.5.1 IMM estimation results

This subsection evaluates the basic performance of an adaptive pursuer using IMM estimation against different nonadaptive evader strategies. The IMM-C control strategy is used. Fig. 6.2 summarizes a simulation in which adaptive IMM pursuer plays against a minimax evader. The figure shows the state, control, and mode probability histories, as well as selected estimation states. The plotted estimates of r_{ji} and $s_{f,j}$ are from the OSO estimator; the FNE estimator is qualitatively similar. The control plot shows the true controls used by each player. The pursuer controls are annotated as follows; whenever the agent switches from the minimax strategy and begins using state estimates in the closed-loop control, an "X" is placed, and a dashed line is used to indicate periods in which minimax control is used. The perfect-information (PI) feedback control, which would be used if the evader strategy and weights were known exactly, is plotted for reference. The controls computed by the individual filters (OSO, FNE, and LQDG-gain) are also plotted for reference. The minimax strategy is used until about $t = 0.2$, when the estimator confidence is high enough to begin using estimates in the control. All three modes lead to similar control profiles, which deviate very little from the PI control. The mode probabilities clearly identify the OSO and FNE modes as the most likely, although the IMM is unable to differentiate much between the two options. The estimation errors in the LQDG weights are small and remain so throughout the simulation. The gain estimate has a larger initial error that converges by about $t = 0.1$.

Fig. 6.3 displays simulation results when a constant gain evader plays against the adaptive IMM pursuer. The gain used is $[K_j] = [-2.8]$, which is much smaller than the zero-sum terminal Riccati value of the LQDG. Consequently, the relative state achieves a much smaller value under the constant gain evader. In the control profile, there is a noticeable difference between the controls from the LQDG estimators and the LQDG-gain estimator. The actual control used tracks the PI reference well and converges to a small error by about $t = 0.2$. The mode probabilities clearly indicate that the LQDG-gain mode is the most likely mode for almost all of the simulation. The r_{ji} and $s_{f,j}$ estimates have no truth values. The estimates converge to values far from the initial values, indicating the large error between the LQDG model and the constant gain opponent behavior. The

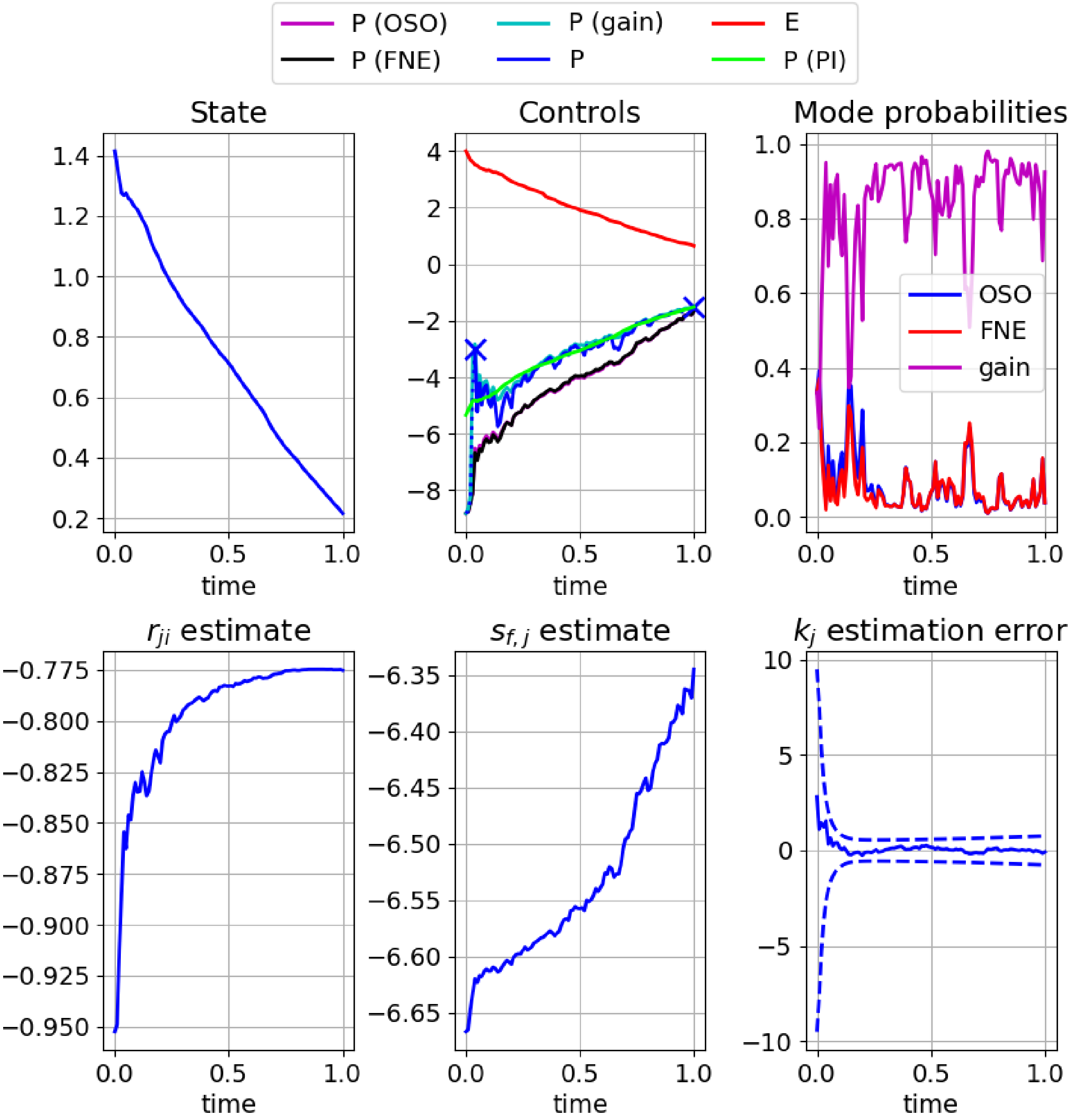

Figure 6.3: Simulation results comparing IMM pursuer against constant gain evader

gain estimate converges to a small error by about $t = 0.15$.

The results shown here indicate that the IMM estimator is able to correctly distinguish between LQDG opponent strategies and fixed-gain opponent strategies when the opponent strategy is unchanging. The IMM estimator is not able to clearly differentiate between the minimax and adaptive

FNE evader strategies for this problem. The adaptive pursuer in this section is able to recover the performance previously demonstrated in adaptive LQDG play. The next subsection compares the IMM-C and matrix game-based control strategies.

6.5.2 Comparison of IMM-C and matrix game-based control

An IMM-based adaptive pursuer with linear quadratic objective function is engaged with an evader who uses a constant gain of $[K_j] = [-2.8]$. The pursuer implements two estimators in the IMM ensemble: an adaptive OSO estimator and a constant-gain estimator [3]. Each mode is considered equally likely initially, and the prior mode transition probability of transitioning from either mode to the other is 1%. This scenario is tested with both IMM-C and matrix game-based control to compare the performance of the two approaches.

Fig. 6.4 summarizes the control, mode probabilities, and cost-to-go incurred when the IMM-C approach is used against the constant-gain evader. The control histories are annoted as in the previous subsection, with "X" used to indicate a transition from minimax control to estimate-in-the-loop control and dotted lines denoting control under the minimax strategy. Under the IMM-C approach, the total cost incurred is 4.0. The reference cost, denoting the expected cost under PI gameplay, is 3.8. The true control quickly approaches the LQDG-gain control, causing it to lie close to the PI control history. The LQDG-gain mode is consistently the most probable. The cost-to-go history shows a large initial slope that diminishes after a few time steps when the estimates enter the closed loop. The reduction in slope of the cost-to-go indicates the cost incurred from that point on has lessened compared to what it might have done.

Fig. 6.5 shows the same performance metrics when the matrix game-based control is used. In addition to plotting the incurred cost-to-go, the pursuer's expectation of the costs-to-go for each candidate strategy are computed and plotted. The results are qualitatively similar to the results with IMM-C control. The total cost incurred is 4.2, a slight increase compared to the IMM-C control. It is apparent why from looking at the plot of control values. During the first several time steps

[3] To highlight the tradeoffs between the OSO and LQDG-gain approaches, the pursuer does not implement the adaptive FNE estimator.

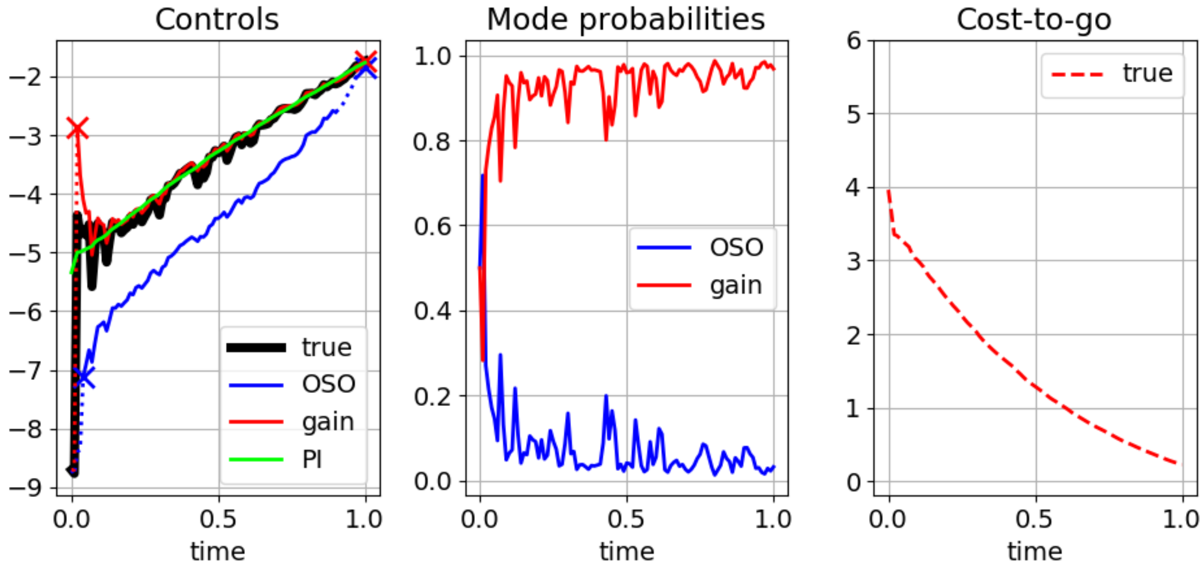

Figure 6.4: Summary results for the scalar example using IMM-C

of the simulation, the matrix game analysis chooses to use the OSO control strategy rather than the LQDG-gain control. This has the effect of using more control energy than is necessary under PI, increasing the overall cost somewhat. Examining the cost-to-go history, it is apparent that the agent believes there is a large potential cost to choosing the gain strategy during the same interval, leading him to use the OSO strategy.

To further investigate the performance of the matrix game-based control, consider Table 6.4. This table summarizes the matrix game computation at certain times of interest during the simulation. Each matrix represents the pursuer's estimate of the cost under a particular pairing of strategies. Columns 1 and 2 correspond to the evader using the minimax and constant gain strategies, respectively. Rows 1 and 2 correspond to the pursuer choosing the OSO and LQDG-gain strategies. The cost matrices are multiplied by the vector of modal probability estimates, and the product vector is the expected value of the game for each of the pursuer's candidate strategies. At $t = 0.01$, after two measurements have been made, each pursuer filter is still using the minimax control, and so each strategy incurs the same cost for a given evader strategy. At $t = 0.02$, the LQDG-gain estimator is confident enough to begin using estimates in the closed-loop control. If

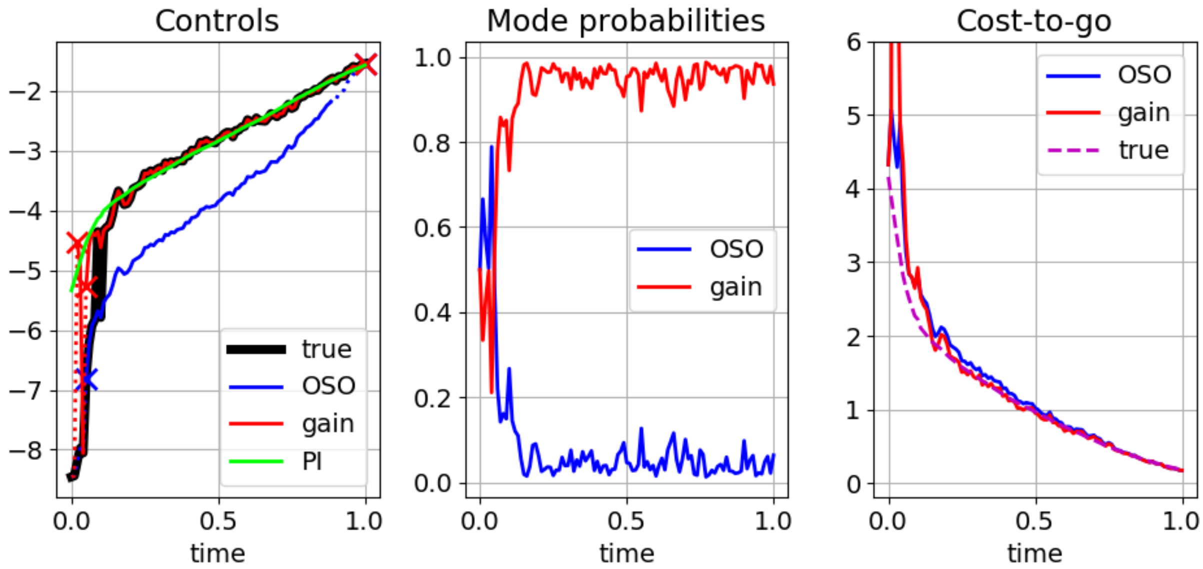

Figure 6.5: Summary results for the scalar example with matrix game-based control

$t = 0.01$ $\begin{bmatrix} 5.7 & 3.83 \\ 5.7 & 3.83 \end{bmatrix} \times \begin{bmatrix} 0.666 \\ 0.334 \end{bmatrix} = \begin{bmatrix} 5.07 \\ 5.07 \end{bmatrix}$	$t = 0.02$ $\begin{bmatrix} 5.24 & 3.84 \\ 34.2 & 2.97 \end{bmatrix} \times \begin{bmatrix} 0.569 \\ 0.431 \end{bmatrix} = \begin{bmatrix} 4.64 \\ 20.7 \end{bmatrix}$
$t = 0.07$ $\begin{bmatrix} 3.34 & 2.76 \\ 5.17 & 2.46 \end{bmatrix} \times \begin{bmatrix} 0.142 \\ 0.858 \end{bmatrix} = \begin{bmatrix} 2.84 \\ 2.85 \end{bmatrix}$	$t = 0.11$ $\begin{bmatrix} 2.93 & 2.5 \\ 3.83 & 2.3 \end{bmatrix} \times \begin{bmatrix} 0.145 \\ 0.855 \end{bmatrix} = \begin{bmatrix} 2.57 \\ 2.52 \end{bmatrix}$

Table 6.4: Snapshots of matrix game computations during simulation

this control is used, the pursuer believes it will incur a large penalty if the evader actually uses a minimax strategy. So, despite the reduced cost-to-go if the evader uses the constant gain strategy, the large potential penalty makes the pursuer choose the OSO strategy. This trend continues for several measurements. At $t = 0.07$, the modal probability associated with the constant gain evader has grown significantly. Still, the cost of being incorrect is too large, and the OSO strategy is used. This continues until $t = 0.11$, the first time at which the expected cost under the LQDG-gain strategy is lower than under the OSO strategy.

The results shown here demonstrate a case in which the matrix game approach leads to a controller that is conservative. It is conservative because it will not switch to the more likely, less aggressive control profile until the possible negative outcomes have become tolerably small. This conservativeness comes at a small cost in performance. There is also a greater demand computational burden required to evaluate the cost-to-go for each combination of strategies. Despite this additional challenge, the matrix game-based strategy seems promising if the number of opponent modes to be evaluated can be limited.

6.5.3 Deceptive switching strategy

For the last set of results in this chapter, the evader uses a nonadaptive strategy based on switching from a constant gain controller to a FNE controller. For simplicity, the evader is allowed perfect information about the pursuer's weights and does not perform estimation. She switches to the perfect information FNE strategy after the switch. The pursuer uses an IMM strategy with adaptive FNE and LQDG-gain modes with the IMM-C approach. For simplicity, the pursuer does not use the cost-based loop closure approach of Section 4.2, which significantly alters the behavior of the pursuer when measurement uncertainty is high. A switch time of $t_{sw} = 0.08$ is used with a pre-switch gain of $[K_j] = [6.0]$. In designing a strategy the evader assumes a first-order time of $\alpha = 30.0$ governing the pursuer's adaptation rate. All other values are as listed in Table 6.3.

Fig. 6.6 displays simulation histories for the state, controls, and evader cost-to-go. The figures compare the predicted performance using the simplified model to the actual performance against a pursuer using IMM estimation. By implementing the switch, the evader is able to increase the relative state to a value greater than its initial value, and achieves a smaller overall cost than is predicted according to a PI FNE solution. The pursuer control and evader cost-to-go plots show the simulated histories as well as the predicted values according to the evader's simplified pursuer model used for strategy design. The plotted values are the predicted closed-loop control and cost-to-go evaluated against the recorded state history from simulation. The simplified model qualitatively captures the characteristics of the full simulation well. There is an initial transient effect in which the prediction error is large. This transient occurs because the pursuer begins with no a priori knowledge of the evader's gain and must converge to an initial response to the

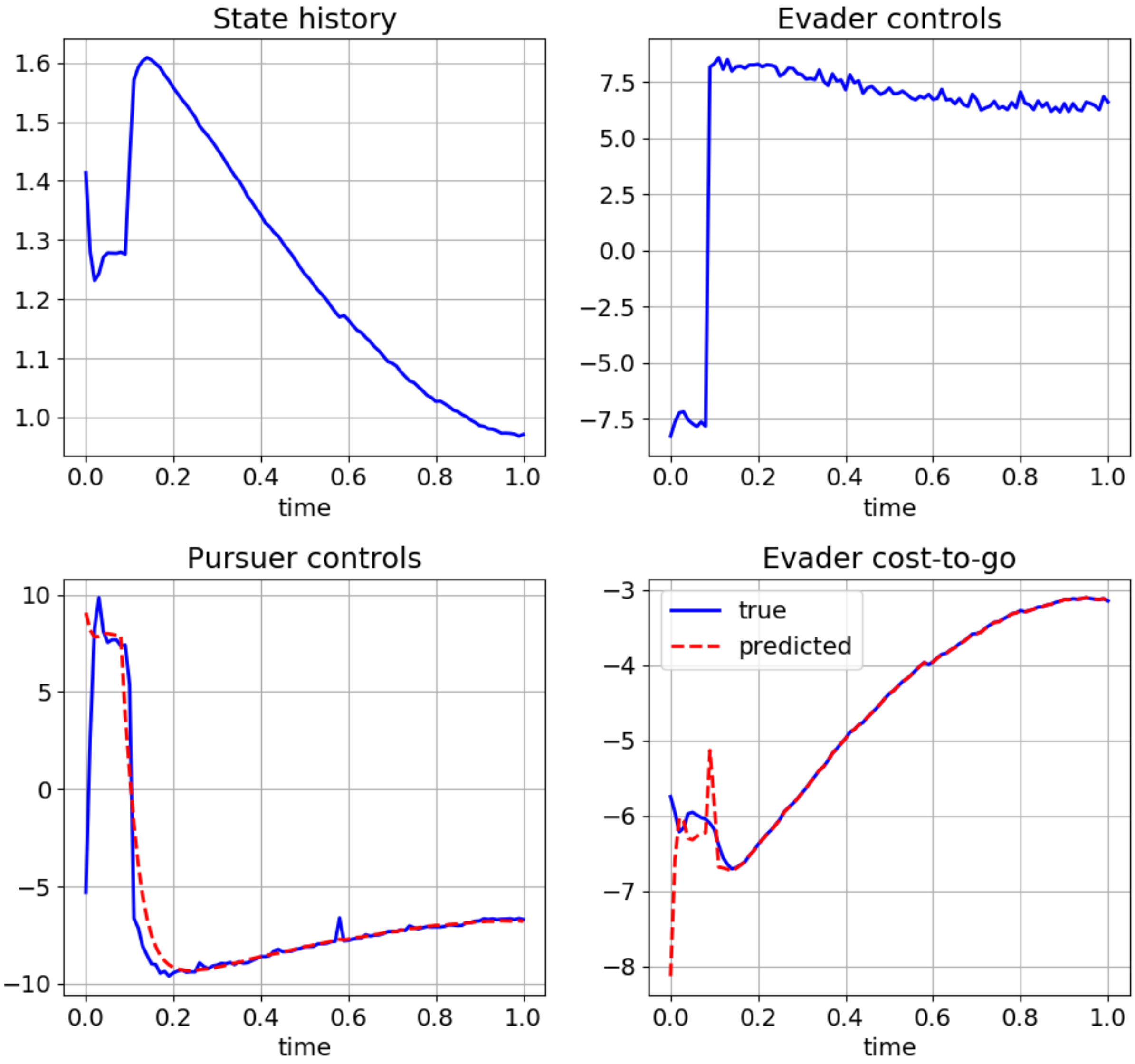

Figure 6.6: Comparison of pursuer controls using IMM estimation and cost-to-go against predicted values

manifested gain strategy. After approximately two time steps, the pursuer "catches up" to the evader's strategy, and from then on the control and cost-to-go predictions and histories match well. There is a slight mismatch in the pursuer's "rise time," after the switch. The initial response is slower than the prediction, but later it converges more quickly. This fact suggests a higher-order model may be needed to fully capture the convergence behavior of the IMM.

Note in Fig. 6.6 that the initial transient reduces the evader's performance significantly. The

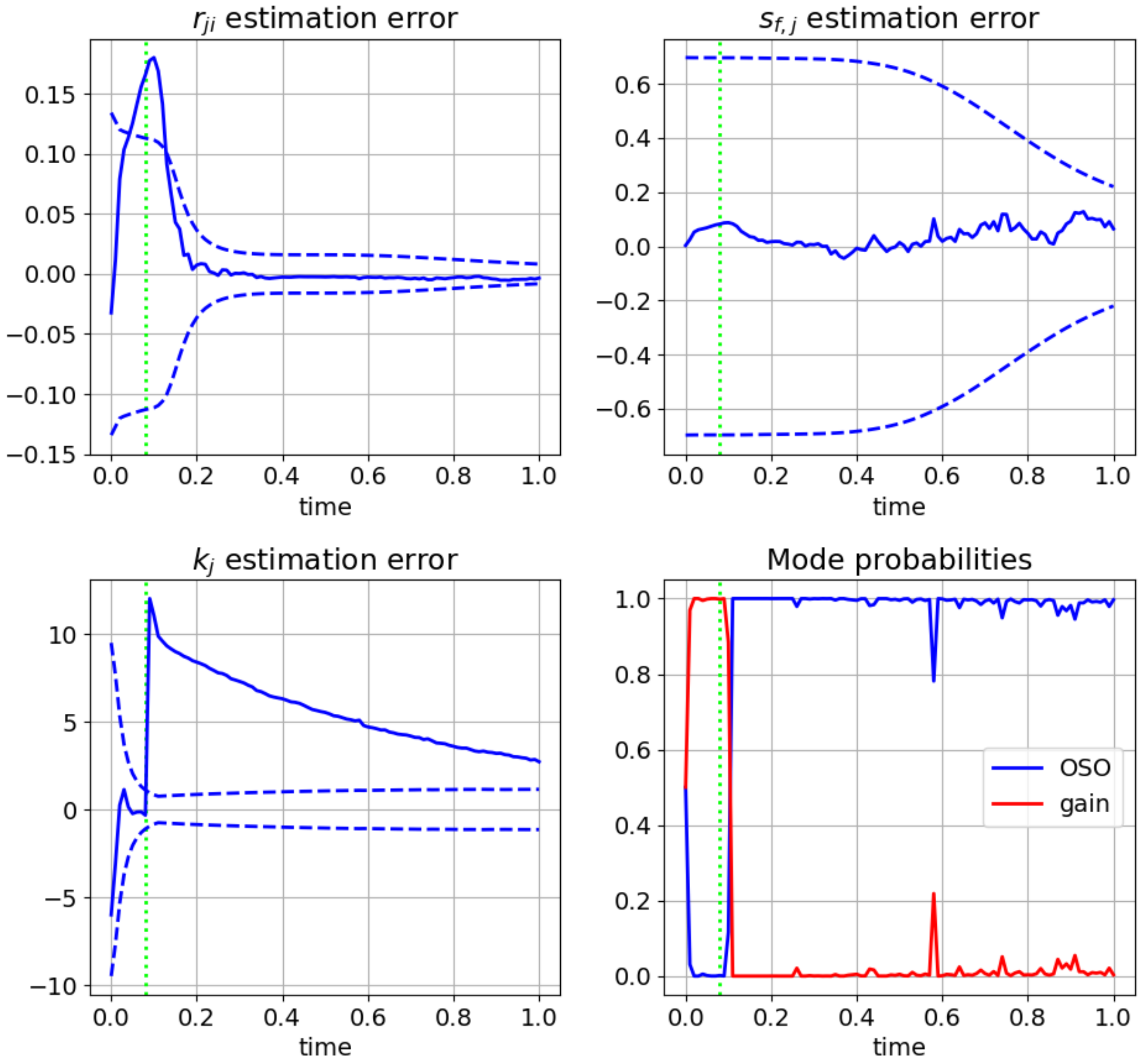

Figure 6.7: Estimation errors during engagement between IMM pursuer and strategy switching evader

full predicted cost-to-go is -8.1, but the actual cost incurred by the evader is -5.7. From the plot of cost-to-go, most of that loss occurs during the first two timesteps. The total predicted cost under the non-switching FNE strategy with perfect information for both players is -5.1, so the switching strategy is still an improvement of about 10% over the non-switching strategy.

Fig. 6.7 shows the pursuer IMM estimation performance during the engagement with the switching evader. The switch time is indicated by a vertical line in each plot. Before the switch, the estimate of r_{ji} begins to diverge, but it quickly converges to the true value in about 0.1 time units. $s_{f,j}$ is essentially unobservable at the beginning of the simulation, so its estimates are not affected by the switch. The estimates of k_j converge quickly to small errors before the switch, then suffer a step change after the switch. The errors reduce slightly as the simulation runs on, and it may be possible to tune the filter to become robust to step changes in the gain. The mode probabilities show that the IMM correctly identifies the constant-gain strategy used before the switch. Within two or three measurements after the switch, the modal probabilities converge to the new correct values. The estimation error plots demonstrate that the IMM estimation behaves essentially as expected for the switching engagement.

When the pursuer uses the matrix game-based control, the evader receives no advantage from switching. Fig. 6.8 shows the pursuer control history and the reference control from each individual estimator. The pursuer uses the FNE strategy the entire time, and the evader does not benefit from the switch. This happens because the "downside" for the pursuer is extremely large if he chooses the LQDG-gain control value and the evader is actually of the FNE type. Note that the evader switching parameters are tuned for the IMM-C pursuer, and it may be possible to re-tune the values to achieve better performance against the matrix game-based pursuer. Nonetheless, it is remarkable that the pursuer is able to overcome the evader's switching strategy by making no change other than explicitly considering the cost of making a bad choice of control.

This section has presented numerical results for a problem with scalar dynamics. First, basic simulations are conducted to demonstrate the IMM estimation is working as intended. The IMM pursuer is able to clearly differentiate between constant gain evaders and minimax evaders, although the minimax and FNE evader strategies are likely not distinguishable. Next, a detailed comparison of the IMM-C and matrix game-based pursuer control approaches is made for the same problem. The results indicate that both control strategies approach the perfect information control. The matrix game-based control is conservative, in that it uses more control energy than is strictly necessary and achieves a lower performance metric than the IMM-C approach. This is a feature of the control approach. The last set of results evaluate a nonadaptive evader strategy based on

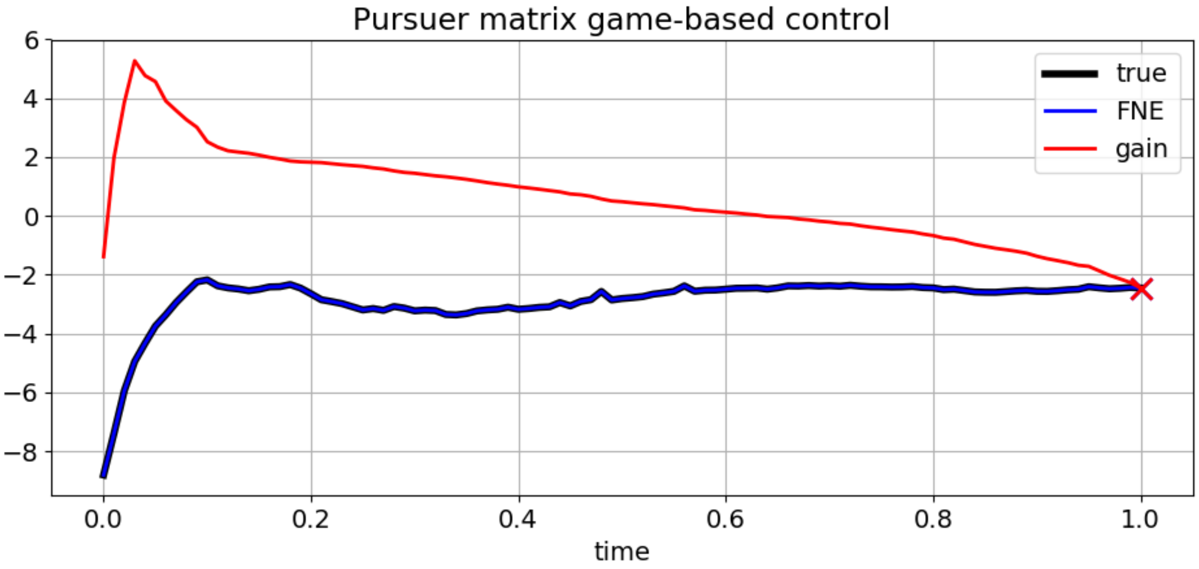

Figure 6.8: Pursuer control against switching evader strategy using matrix game-based control

switching from a constant gain to a FNE strategy. When the pursuer uses the IMM-C approach, the evader performs better under the switching strategy than expected under a simple FNE strategy. The pursuer control and incurred cost histories match the simplified model the evader uses for strategy design. Notably, when the pursuer uses the matrix game-based control, she the evader's switch is not effective because the pursuer continues to use the most conservative controller for the entire simulation. **The matrix game-based control demonstrates an implicit robustness to opponent switches**.

This chapter has presented an approach to pursuit-evasion games in which the opponent strategy is subject to both parametric uncertainty and uncertainty of form. A solution based on MM estimation is considered, using domain knowledge to synthesize candidate opponent responses. The IMM estimation framework leads naturally to two control strategies: the IMM-C approach based on modal probabilities, and a matrix game-based approach. A new strategy based on deception has been proposed for the opponents of adaptive players. Under this approach, a player initially manifests a "false" strategy before switching to one that is optimal. This induces a transient response in the adaptive player, and allows the switching player to improve the value of her objective function. Numerical examples have demonstrated the basic qualities of each of the new elements introduced in this chapter. The IMM estimator correctly differentiates a constant-gain opponent from the adaptive LQDG strategies introduced earlier in this dissertation. The evader is able to use the switching-based strategy to outperform her reference non-switching objective value against an IMM-C pursuer. In examples, the matrix game-based approach demonstrates robustness to opponent switching. This chapter represents the last of new developments in this dissertation. The next chapter considers a detailed missile interception game to explore elements of the new material introduced in Chapters 4–6.

7. APPLICATION: MISSILE INTERCEPTION GAME

This chapter considers an application of the various new concepts introduced in previous chapters to a missile interception pursuit-evasion game. This is a classic problem with multiple agents that is commonly studied under the assumption of linear relative dynamics. A LQDG is set up to demonstrate various techniques, including the cost-based loop closure of Chapter 4, the constrained estimation of Chapter 5, and the IMM estimation and switching strategies of Chapter 6. The chapter has three main parts. First, an overview of the problem is given, emcompassing relevant literature, the problem dynamics, and details of weight selection for realistic games. Second, specific details of the implementation are discussed, such as the selection of polynomial degree for Taylor series approximation. Finally, numerical results highlighting different features of the adaptive gameplay approaches are presented.

7.1 Problem overview

Missile interception problems have been long studied in aerospace engineering. These problems commonly refer to scenarios in which a missile is meant to reduce its range to a target vehicle, which could be another missile or a manned vehicle. The general set of interception problems encompasses disparate scales, ranging from short-range tactical missile engagements to intercontinental ballistic missile interception. In this section, a very generic scenario is considered with scale units to deliberate divorce the results from any specific vehicles. The interception problem is frequently considered without reference to games, and a great deal of existing literature is only loosely related to the topic of this chapter. A small selection of relevant previous works is given, with an emphasis on recent developments. Subsequently, this section presents the general problem dynamics. This is followed by a short discussion of the problem scaling, and then by a discussion of practical missile guidance laws and their translation into a differential game environment.

7.1.1 Related research

The problem of missile interception has been studied for decades. Many of the problems in Isaacs[5, pp. 237–244, 361–366] can be considered abstractions of interception problems under perfect information. A series of later papers[23, 24] derives missile guidance laws under both optimal control and game theory perspectives. A game with realistic control constraints is considered and the robustness to errors in the target acceleration profile is considered. The target acceleration profile is considered a priori unknown, and is not estimated from the available data. Shinar, Guelman, and Green [77] consider a planar PE game. The pursuer is modified with a nonconstant speed based on drag. From the solution to a game of kind, a pursuit feedback law is derived and compared against proportional navigation. The barrier surfaces that define capture regions are also defined. The dissertation of Ref. [6] considers a LQDG involving an intercontinential ballistic missile interception. Planar motion is considered, and feedback linearization is employed to assume a linear form of the relative closed-loop dynamics. Two related papers by Garcia, Casbeer, and Pachter consider a three-agent game in which a cooperating team consisting of a target and a defender missile try to avoid capture of the target by an attacker missile. In Ref. [78], a zero-sum game based on terminal miss distance is considered where the defender is faster than the attacker. All agents are able to turn instantaneously. An analysis based on Apollonius circles is used to derive instantaneously optimal heading commands for the defender team and the attacker. Perfect mutual information with regard to all players' speeds and dynamic constraints is assumed. A sequel, Ref. [79], relaxes the turning rate assumption to consider finite turning rates, and derives control strategies for particular guidance laws chosen by the attacker.

Among recent authors, Shima has been particularly prolific in the area of missile interception problem, producing many practical results. A few are highlighted here. Ref. [80] considers guidance strategies for a missile trying to intercept a slow-moving target that launches an defender missile. Typical guidance laws for the missiles are considered: proportional navigation (PN), pure pursuit (PP), and line-of-sight guidance. Matrix game-theoric analysis is used to compute mixed strategies dictating the guidance laws used by the missile and the defender. Fonod and Shima [27] use multiple model adaptive estimation (MMAE) onboard an evader. The pursuer has a fixed set of guidance strategies with uncertain weights, but executes no switches in strategy. In a related paper, the pursuer is assumed to execute one or two switches in its gain during an engagement[28]. A matrix game analysis is conducted to determine a max-min evader strategy that ensures a minimum miss distance. These papers generally consider missile interception *engagements* without explicitly considering games and specific objective functions. It is obvious, however, that the authors are exploring ideas closely related to those investigated in this dissertation.

This completes a brief review of some related missile interception research. Now, the governing problem dynamics are presented.

7.1.2 Problem dynamics

Missile guidance research is commonly executed for analysis of a planar engagement. To obtain general results in this dissertation, the 3D interception problem is first derived, and later specialized for planar dynamics. Each agent is treated as a point mass moving at a constant velocity. This is a customary assumption for homing missiles during the terminal phase, or endgame, of an engagement[28]. There are two agents present. In this subsection, the conventions of missile interception research are followed, and the agents are referred to as the attacker (pursuer) and the target (evader). A graphical representation of the problem is shown in Fig. 7.1. A reference frame is attached to each point mass at its mass center, with a^+ denoting the attacker's reference frame and t^+ the target's. The 1-axis of each reference frame is aligned with the associated body's velocity vector. These reference frames are established to allow expression of body-axis control accelerations, which much be orthogonal to the velocity vector. An inertial reference frame, n^+, is known. The relative reference frame r^+ is defined such that r^+ is centered at the target's center of mass and $\bar{\boldsymbol{r}}_1$ is aligned with the position vector of A relative to T. That is, the 1-axis of the r^+ frame is aligned with the line-of-sight vector from the target to the attacker. It is customary to develop the velocity and acceleration-level kinematic equations in the r^+ frame, while frames a^+ and t^+ are used only to relate applied forces to body-axis accelerations.

Using the defined coordinates, the relative reference frame kinematics are now developed. r^+

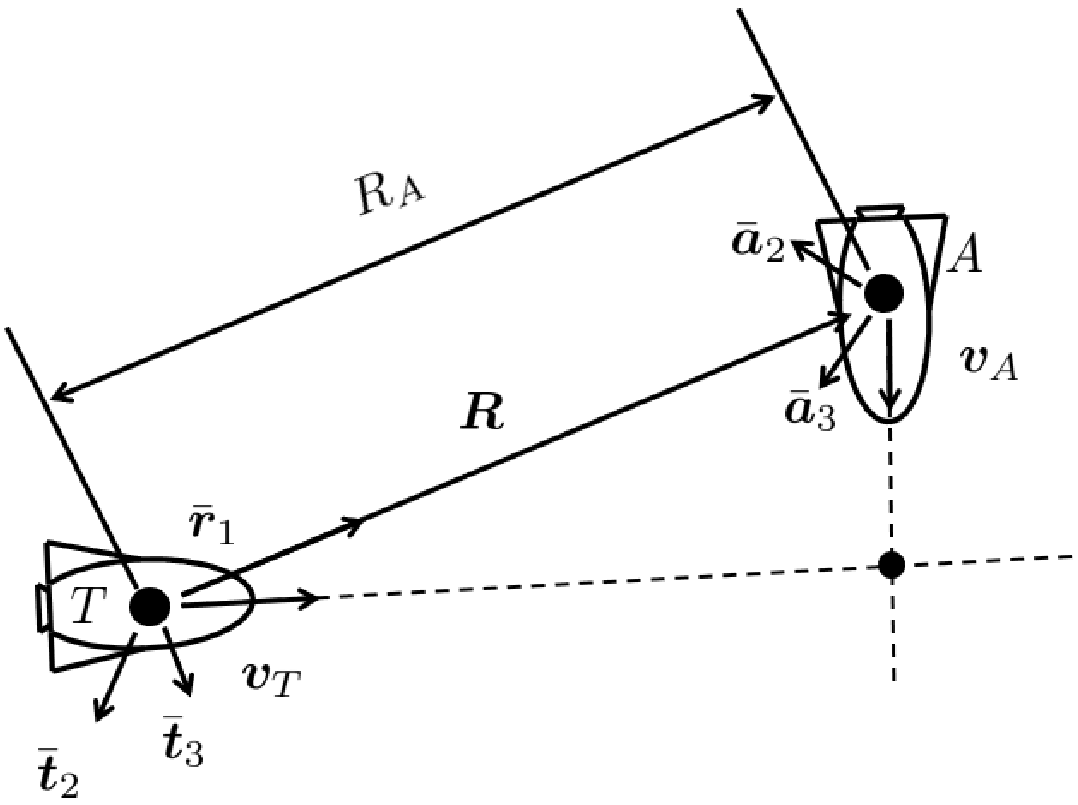

Figure 7.1: Missile interception problem geometry

is defined relative to the inertial reference frame by a simple 3 rotation through Γ_A followed by a 2-axis rotation through ξ_A. Using these angles, the r^+ frame relative position vector $\boldsymbol{R}$ and its velocity and acceleration $\boldsymbol{V}$ and $\boldsymbol{A}$ can be developed as follows:

$$\boldsymbol{R} \equiv \boldsymbol{r}_A - \boldsymbol{r}_T \tag{7.1}$$

$$[\boldsymbol{R}]_r = \begin{bmatrix} R_A \\ 0 \\ 0 \end{bmatrix} \tag{7.2}$$

$$[\boldsymbol{V}]_r = \begin{bmatrix} \dot{R}_A \\ \dot{\Gamma}_A R_A \cos \xi_A \\ -R_A \dot{\xi}_A \end{bmatrix} \tag{7.3}$$

$$[\boldsymbol{A}]_r = \begin{bmatrix} \ddot{R}_a - R_A(\dot{\xi}_A^2 + \dot{\Gamma}_A^2 \cos^2 \xi_A) \\ \ddot{\Gamma}_A R_A \cos\xi_A + 2\dot{\Gamma}_A(\dot{R}_A \cos\xi_A - R_A\dot{\xi}_A \sin\xi_A) \\ -R_A\ddot{\xi}_A - 2\dot{R}_A\dot{\xi}_A - \dot{\Gamma}_A^2 R_A \cos\xi_A \sin\xi_A \end{bmatrix} \tag{7.4}$$

It is also useful to expand the accelerations of each agent in its own reference frame, since that is the frame in which the aerodynamic forces are commonly known. Let each agent's reference frame be defined relative to the inertial reference frame by a three-axis rotation through γ_A (resp. γ_T) followed by a two-axis rotation through θ_T (resp. θ_T). The velocities satisfy $\boldsymbol{v}_A = v_A\bar{\boldsymbol{a}}_1$ and $\boldsymbol{v}_T = v_T\bar{\boldsymbol{t}}_1$. The accelerations in the respective reference frames have the following values:

$$[\boldsymbol{a}_A]_a = v_A \begin{bmatrix} 0 & \dot{\gamma}_A \cos\theta_A & -\dot{\theta}_A \end{bmatrix}^T \tag{7.5}$$

$$[\boldsymbol{a}_T]_t = v_T \begin{bmatrix} 0 & \dot{\gamma}_T \cos\theta_T & -\dot{\theta}_T \end{bmatrix}^T \tag{7.6}$$

A reasonable model for the guidance and control of each agent is to assume the existence of an inner-loop controller that enables each agent to achieve desired body-axis accelerations by manipulation of its actuators to generate prescribed aerodynamic forces. Control laws are typically developed in terms of either body-axis accelerations or the flight path angles, which are related by multiplicative factors.

An interception problem with linear relative dynamics is obtained by analyzing the motion of the agents relative to some nominal, kinematically feasible relative position history whose terminal state satisfies $\boldsymbol{R}(T_f) = \boldsymbol{0}$ with a positive terminal closing velocity. The closing velocity is obtained from the projection of the relative velocity onto the relative position vector as follows: $V_c = (\boldsymbol{v}_T - \boldsymbol{v}_A) \cdot \bar{\boldsymbol{r}}_1$. Any number of candidate reference histories of $\boldsymbol{R}(t)$ can be used, accounting for various factors such as inverse-square gravity, vehicle drag effects, and so on. It is common practice in research to consider a flat-Earth (constant gravity) model with the reference trajectory satisfying straight-line motion for each agent[81, 27, 28]. It is straightforward to show that under this simple collision triangle model, the kinematic coordinates take on the following histories:

$$\bar{R}_A(t) = R_A(0)\left(1 - \frac{t}{T_f}\right) \tag{7.7}$$

$$\bar{\Gamma}_A(t) = \Gamma_A(0), \bar{\xi}_A(t) = \xi_A(0)$$

Furthermore, the positive closing velocity condition is trivially satisfied if $R_A(0) > 0$, which can always be enforced by proper selection of the initial conditions. Under these reference values, the velocity and acceleration kinematics take the following values:

$$\bar{\boldsymbol{V}} = -\frac{R_A(0)}{T_f}\bar{\boldsymbol{r}}_1 \tag{7.8}$$

$$\bar{\boldsymbol{A}} = \boldsymbol{0} \tag{7.9}$$

Furthermore, the angular velocity of the line-of-sight reference frame becomes zero. The line-of-sight reference frame along the reference trajectory is denoted $\bar{r}^+$. It is convenient to expand the perturbed relative equations of motion in this reference frame, which is an inertial reference frame. Let the perturbed relative position be given in terms of the reference value plus a perturbation term:

$$\boldsymbol{R} = \bar{\boldsymbol{R}} + \delta\boldsymbol{R} \tag{7.10}$$

Further, let the elements of the perturbed displacement be written in the $\bar{r}^+$ frame:

$$[\delta\boldsymbol{R}]_{\bar{r}} = \begin{bmatrix} x & y & z \end{bmatrix}^T \tag{7.11}$$

$\bar{r}^+$ has zero angular velocity, so the inertial derivatives of $[\delta\boldsymbol{R}]_{\bar{r}}$ contain only the derivatives of x, y, and z. Each agent's reference trajectory satisfies rectilinear motion, so the acceleration of each agent along the reference trajectory is zero. Let $[C_{b/a}]$ denote the generic direction cosine matrix specifying the transformation from frame a^+ to b^+. The relative acceleration can be expanded as follows:

$$[\boldsymbol{A}]_{\bar{r}} = [\delta\ddot{\boldsymbol{R}}]_{\bar{r}} = \begin{bmatrix}\ddot{x} & \ddot{y} & \ddot{z}\end{bmatrix}^T = [C_{\bar{r}/a}][\boldsymbol{a}_A]_a - [C_{\bar{r}/t}][\boldsymbol{a}_T]_t \tag{7.12}$$

Expanding all terms in a Taylor series and truncating at first order leads to the following governing equation for the second-order position displacements:

$$\begin{bmatrix}\ddot{x}\\ \ddot{y}\\ \ddot{z}\end{bmatrix} = [C_{\bar{r}/\bar{a}}]\begin{bmatrix}0\\ a_{\gamma_A}\\ a_{\theta_A}\end{bmatrix} - [C_{\bar{r}/\bar{t}}]\begin{bmatrix}0\\ a_{\gamma_T}\\ a_{\theta_T}\end{bmatrix} \tag{7.13}$$

It is critical to note a common practice in missile interception. The governing linearized equations for the relative displacement represent three second-order equations with only two controls available for each agent. The goal of the pursuer must be to minimize the perturbed displacement, and at first glance it appears that she must regulate three uncoupled states using only two controls, a task which can be challenging. However, as Ref. [82] notes, it is only necessary for the pursuer to ensure that the "cross-track" displacements y and z remain small to achieve interception. So long as the perturbed velocity along the line-of-sight $\dot{x}$ is small compared to the closing velocity, nonzero values of x do not change the range between agents at interception. Rather, displacements in x change the time of interception by a small amount, approximately given by x/V_c. Consequently, from the LQDG perspective, it is only necessary to consider y and z as states. A compact state-space representation of the relevant linear dynamics is given as follows:

$$\begin{bmatrix}\dot{y}\\ \dot{z}\\ \ddot{y}\\ \ddot{z}\end{bmatrix} = \begin{bmatrix}[0_{2\times2}] & [\mathbb{I}_{2\times2}]\\ [0_{2\times2}] & [0_{2\times2}]\end{bmatrix}\begin{bmatrix}y\\ z\\ \dot{y}\\ \dot{z}\end{bmatrix} + \begin{bmatrix}[0_{2\times2}]\\ [\bar{D}][C_{\bar{r}/\bar{a}}][\bar{D}]^T\end{bmatrix}\begin{bmatrix}a_{\gamma_A}\\ a_{\theta_A}\end{bmatrix} + \begin{bmatrix}[0_{2\times2}]\\ -[\bar{D}][C_{\bar{r}/\bar{t}}][\bar{D}]^T\end{bmatrix}\begin{bmatrix}a_{\gamma_T}\\ a_{\theta_T}\end{bmatrix} \tag{7.14}$$

$$[\bar{D}] \equiv \begin{bmatrix}[0_{2\times1}] & [\mathbb{I}_{2\times2}]\end{bmatrix}$$

This completes the derivation of the 3D perturbed equations of motion for missile interception. The linearized dynamics are uncoupled, and it is convenient for analysis purposes to use planar analysis in many problems. Letting the z components of motion become zero yields planar dynamics in one second-order coordinate:

$$\begin{bmatrix} \dot{y} \\ \ddot{y} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} y \\ \dot{y} \end{bmatrix} + \begin{bmatrix} 0 \\ \cos(\bar{\Gamma}_A - \bar{\gamma}_A) \end{bmatrix} a_{\gamma_A} + \begin{bmatrix} 0 \\ -\cos(\bar{\Gamma}_A - \bar{\gamma}_T) \end{bmatrix} a_{\gamma_T} \tag{7.15}$$

This subsection has presented the governing equations for the missile interception differential game to be considered in this chapter. Point mass vehicle models are used and the governing kinematics and dynamics are obtained up the acceleration level. The dynamics are linearized by assuming rectilinear reference motion. This is an appropriate assumption for a variety of tactical missile engagements. To ensure all problem variables are of approximately the same order, the introduced dynamics are now rescaled.

7.1.3 Problem scaling

Let a scaled time be defined as $\tau \equiv \alpha^{-1}t$, and let a scaled position vector be defined as $\boldsymbol{r} = c^{-1}\boldsymbol{R}$. Under this model, scaled velocity and acceleration vectors are defined as $\boldsymbol{r}' = \boldsymbol{v}$, $\boldsymbol{v}' = \boldsymbol{a}$, using the following definitions:

$$x' \equiv \frac{dx}{d\tau} \tag{7.16}$$

$$\boldsymbol{r}' = \boldsymbol{v} = \frac{\alpha}{c}\boldsymbol{V} \tag{7.17}$$

$$\boldsymbol{v}' = \boldsymbol{a} = \frac{\alpha^2}{c}\boldsymbol{A} \tag{7.18}$$

In this chapter, c is taken to be 100 m, and α is treated as being between 1 and 4 seconds[1]. In numerical examples, rescaled coordinates are chosen to be on the order of 1, as is the simulation time. Thus, the full-scale problem has a duration on the order of seconds, and takes place at relative ranges of about 100 m, relative speeds on the order of 100 m/s, and accelerations on the order of

[1] Different values may be chosen to alter the engagement duration and dynamic scale as desired.

about 10 g's. The remainder of this chapter uses rescaled problem units in presenting the problem and numerical results. To complete the current section, a brief discussion of missile guidance laws is now given.

7.1.4 Missile interception laws

The LQR framework may be used to derive a form of the common proportional navigation (PN) guidance law. PN is a simple guidance law in which an attacker's acceleration is proportional to the relative line-of-sight rate to the target. Consider the planar dynamics of Eq. 7.15. Assume the target's acceleration is zero, and construct a LQR problem by letting the state penalty go to zero with a nonzero state weight on the position only:

$$J_A = \frac{1}{2} b y^2(T_f) + \frac{1}{2} \int_0^{T_f} a_{\gamma_A}^2 \, \mathrm{d}\tau \tag{7.19}$$

Letting $b \to \infty$ yields a linear feedback control law with time-varying weights[81]. Beginning from this LQR problem, it is clear that if a penalty on the target's acceleration, $\frac{1}{2} R_{AT} a_{\gamma_T}^2$, is appended to the integral cost, a *robust* PN guidance law is obtained by deriving the resulting minimax control. It follows, then, that the missile interception problem can be framed as a LQDG by giving one or both players linear-quadratic objective functions. A realistic LQDG framework considers a near-zero sum engagement with no running state cost and a large terminal penalty only on the relative state y. The numerical examples of this chapter endeavor to use weights chosen according to this fashion. However, this model is relaxed in some examples to obtain results that are of greater academic interest.

This section has reviewed some related background research in missile interception problems relating to differential games and/or acting under uncertainty. The problem dynamics have been derived, linearized, and presented for planar and nonplanar interception. For numerical examples, re-scaled dynamics are introduced to make all terms of order 1. Lastly, a nominal LQDG corresponding to a robust PN guidance law engagement discussed for use in numerical examples. In the next section, details of the numerical implementation of the adaptive gameplay methods of this dissertation are discussed.

7.2 Implementation details

Going back to Chapter 4, novel techniques for improving the adaptive gameplay of Chapter 2 and for handling additional sources of uncertainty have been developed and demonstrated on simple problems. These techniques are now applied to the missile interception problem. It would be excessively tedious to burden the reader with all the details of the numerical implementation. This section summarizes the most relevant details in six subsections:

1. Nominal engagement model

2. General estimation equations for planar and nonplanar problems

3. Selection of Taylor series degree

4. Selection of Chebyshev polynomial degree for constraint approximation

5. Design of evader switching-based strategies

6. Robustness of LQDG estimation to step discontinuities in opponent dynamics

In addition to the six topics presented in this section, observability analysis of the planar LQDG interception problem is conducted. Results in Appendix A.3 demonstrate the problem is observable for the engagement weights and dynamics discussed in this section. To begin, the nominal engagement model used in most simulations is presented.

7.2.1 Nominal engagement model

The numerical results in this chapter consider half a dozen different engagements with various player control and estimation strategies, problem weights, initial conditions, and so on. Most of these conditions represent modifications of a "baseline" engagement that is outlined in this subsection for convenient. First, consider the reference motion about which engagements are linearized. All engagements in this chapter are nominally head-to-head, meaning the vehicle velocities are aligned with the relative position vector. The pursuer has a speed of 5.2 and the evader of 4.8, with a usual initial range of 10.0, and a final time of 1.0. For simplicity, the n^+ frame is aligned with the $\bar{r}^+$ frame, and the evader's initial position is taken as the origin of the inertial reference frame. These conditions correspond to coordinate values of $\bar{\Gamma}_A = 0, \bar{\xi}_A = 0, \bar{\gamma}_A = \pi, \bar{\gamma}_T = \bar{\theta}_T = \bar{\theta}_A = 0$.

Now, the nominal simulation weights are presented. The problem weights are frequently varied between numerical examples, but the following values are used if other values are not specified. In planar engagements, the pursuer weights and state take on the following form:

$$[Q_p] = \text{diag}(1.0, 0.0) \quad [R_{pe}] = \text{diag}(-1.05) \quad [S_{f,p}] = \text{diag}(10.0, 0.0) \quad \boldsymbol{x} = \begin{bmatrix} y & \dot{y} \end{bmatrix}^T \tag{7.20}$$

The effect of these weights is to modify the "robust PN" control of Section 7.1.4 with a small running penalty on $y(t)$. The evader may use several strategies based on either LQDG or constant gain approaches. When the evader uses a LQDG strategy, the pursuer normally assumes a nominal ZS engagement.

Numerical simulations are conducted by propagating the full nonlinear dynamics, while players use the linearized model to obtain control solutions. This linearization typically produces no significant problems for state values of less than about 0.1 in either position or velocity[2]. Full state measurements are assumed with uncorrelated covariances of magnitude 10^{-6}. This completes the overview of numerical simulations. Next, details of the estimation models are discussed.

7.2.2 Estimation equations

For the numerical examples in this chapter, the adaptive pursuer's perspective is taken, and his response to various evader strategies is considered. The evader employs strategies introduced earlier: the minimax strategy, the adaptive FNE strategy, the constant gain approach, and two switching strategies discussed later in this section. The pursuer is considered to have either parametric uncertainty, in which case the pursuer uses the appropriate estimator for the evader's known strategy, or, the pursuer has strategic and parametric uncertainty. In the latter case, the pursuer uses IMM estimation. This subsection details the estimators used by the pursuer and evader.

[2] Recall that in physical units, this corresponds to about $10\ \mathrm{m}$ or $\mathrm{m/s}$.

First, consider the filter equations for an OSO or FNE player. The planar missile guidance problem discussed in Section 7.1.4 has only two nonzero problem weights: a terminal penalty on the relative state y, and a running cost associated with the opponent's control use. The guidance law model is relaxed slightly and the evader is allowed to have a nonzero, though small, running penalty on the state y. The assumption that the running and terminal penalties on $\dot{y}$ are zero is enforced. Thus, in the planar engagement, the pursuer considers the evader to use the following problem weights in conjunction with the linear dynamics of Eq. 7.15:

$$[Q_e] = \text{diag}(q_e, 0) \qquad [R_{ep}] = [r_{ep}] \qquad [S_{f,e}] = \text{diag}(s_{f,e}, 0) \tag{7.21}$$

When the evader also plays adaptively, she considers the pursuer to use the same weight structure. In planar LQDG engagements with only parametric uncertainty, each agent has a maximum of five states to estimate. For estimation purposes, all time-invariant parameters are modelled as zero-mean Gaussian processes, as in previous chapters.

In considering nonplanar engagements, it is assumed that all players place equal penalties on the y and z components of relative motion and there are no penalties on the velocity-level terms. Consequently, extending the engagement to nonplanar motion does not increase the number of time-invariant parameters to estimate. In a nonplanar engagement, a player has seven states comprised of the four relative positions and velocities and the three weights.

A number of engagements consider the evader to use a constant gain model, as discussed in Section 6.3. When the evader does so, he is assumed to have zero velocity gain. This restriction is made not for reasons of observability, but for reasons of filter stability. When the constant gain evader plays against a LQDG pursuer, the pursuer's RDE solution remains nonsingular for a large range of position-level gains and a small range of velocity-level gains. Rather than tuning the relative uncertainty in each gain element or setting up a constrained estimate for the LQDG-gain problem, the velocity gain has been assumed zero for simplicity. A pursuer using the LQDG-gain model estimates two states and one parameter in planar engagements.

An explicit state-space representation of all possible filters is omitted for brevity. The next section considers the Taylor series approximation to the RDE solutions.

7.2.3 Taylor series fitting and convergence

The Taylor series approximation to the generalized RDE solution introduced in Chapter 4 is employed to reduce the simulation execution time. Selecting an appropriate polynomial degree for the truncated approximation requires a tradeoff between memory requirements to store the Taylor series coefficients and truncation error. The truncated series approximation error also implicitly limits the initial covariance in the opponent parameters that can be used. This subsection selects an appropriate error metric and evaluates the approximation error and effective initial covariance to select an appropriate polynomial degree for the interception problem.

Consider an adaptive player i playing against player j. AD is used to obtain a truncated Taylor series approximation to $[S_i]$ and $[S_j]$. The implemented control solution depends on both values; for player i, errors in $[S_j]$ induce estimation bias, while errors in $[S_i]$ induce errors in the implemented control law. A reasonable error metric is simply the maximum of the Frobenius norm of the error in $[S_i]$ and $[S_j]$ due to truncation. A scalar metric is obtained by taking the maximum error norm from either matrix across all integration times over the engagement horizon. This yields a simple error metric that can be directly compared as the polynomial degree is changed. It is useful to consider two-dimensional heatmaps of the error metric as a function of the opponent parameters, allowing a visual assessment of the error sensitivity to the parameters.

A simple approach is taken to determine an initial estimation covariance over which the truncated series approximation is valid. Truncation errors are computed as a function of one variable at a time while other parameters are at their ZS values. All parameters are assumed to have uncorrelated initial errors, and the maximum change in any parameter before which the error exceeds a specified threshold is then taken as a 3σ covariance bound. The elements of the covariance matrix are extracted and the 3σ contours are added to the heatmaps to show the error reduction as the polynomial degree increases.

As a representative example, consider the standard reference trajectories. The series approx-

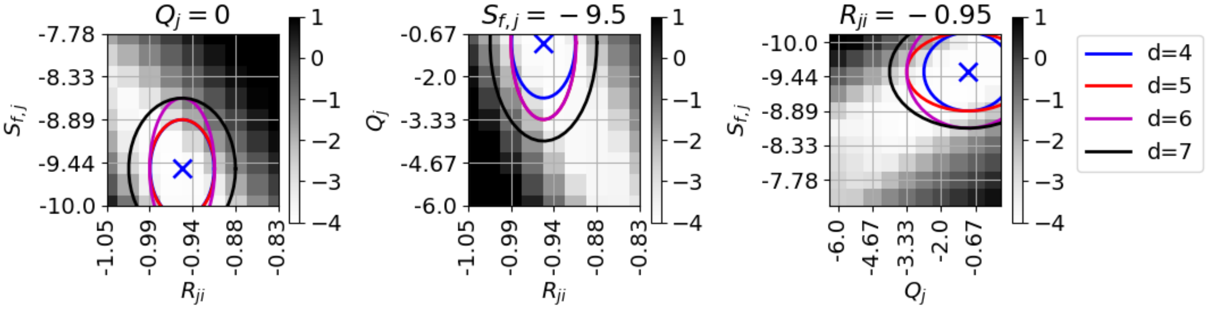

Figure 7.2: Base ten logarithm of truncation error for polynomial expansion in head-to-head interception problem

imation to the RDE solution for the attacker is expanded about the NZS values corresponding to the nominal weights of Eq. 7.20. An error metric tolerance of 10^{-2} is selected. Fig. 7.2 shows the error metric heatmaps and projections of the initial covariance 3σ ellipsoid for increasing polynomial degree. The plotted errors are for the FNE solution approximation. For the OSO case, the error contours and ellipsoid are qualitatively similar, and generally have smaller errors. It is clear from inspection that the truncation errors are highly anisotropic. For a degree 8 approximation, the error ellipse is unchanged compared to degree 7, and the truncated series error outside the ellipse begins to increase. Consequently, a series approximation of degree 7 is chosen for this problem.

The procedure outlined in this subsection is used to determine an appropriate polynomial degree whenever the truncated Taylor series is used. Most scenarios use a polynomial degree of 7, as does the example presented. For the constrained estimation problem, selection of an appropriate basis function order to approximate constraints presents a similar numerical challenge. The next section suggests a sampling-based convergence analysis for the basis function constraint approximation.

7.2.4 Constrained estimation

Chapter 5 introduces inequality-constrained estimation to address numerical singularities in the LQDG control solution that can occur in engagements that are nonsingular under perfect information. An approximate functional form of the stability constraint is obtained by fitting a Chebyshev polynomial surface to a grid of points in the space of the opponent's parameters. It is desireable to use only as many basis functions as are necessary to achieve a small error. Using additional terms may induce undesirable phenomena such the Runge effect. A sampling-based method is used in this section to evaluate convergence.

The Chebyshev polynomial order can, of course, be selected using a similar analysis to that of the previous subsection. However, in fitting the constraint approximation, the problem weights and initial covariance must already be given. A sampling-based method is chosen because it eliminates the need to approximate the true point of instability at the evaluation points, reducing the computation required. Consider a sample of random points in the opponent's parameter space drawn according to the estimator's initial covariance. If enough samples are taken, then some points fall outside the stable region for a given degree of Chebyshev approximation. The Chebyshev polynomial degree should be increased until the number of points outside the stable region no longer changes significantly. Multiple, independent samples are drawn to ensure statistical significance of the results. The primary drawback of this approach is that there is no explicit quantification of the error in the Chebyshev approximation.

Consider the planar interception problem from the perspective of the pursuer. The standard reference trajectories are assumed with the nominal values of the pursuer weights. Chebyshev polynomials are evaluated over a box in the evader's parameter space with edges as shown in Table 7.1. These lengths are chosen to be 1.5 times the assumed initial 3σ estimation covariance. This choice ensures that, in practice, the true value of the opponent parameters falls within the interpolation region of the Chebyshev polynomials - that is, the estimator will fail to converge before constraint surface error starts to grow. Note that values of $R_{ji} \in [-1.3, -0.6]$ are used for the OSO problem and $R_{ji} \in [-1.1, -0.2]$ for the OSO problem. To test convergence of the Chebyshev approximation, 10,000 points are sampled from independent uniform distributions on the grids used to compute the Chebyshev fit. To ensure statistical significance, two independent samples of size 10,000 are generated using Latin Hypercube sampling. The fraction of points lying inside the stable region is compared for each case.

	R_{ji}	$Q_j(1,1)$	$S_{f,j}(1,1)$
Maximum	-0.6/-0.2	-3.21	-0.412
Minimum	-1.3/-1.1	-4.41	-3.40

Table 7.1: Evader weight extrema used to evaluate pursuer Chebyshev constraint approximation

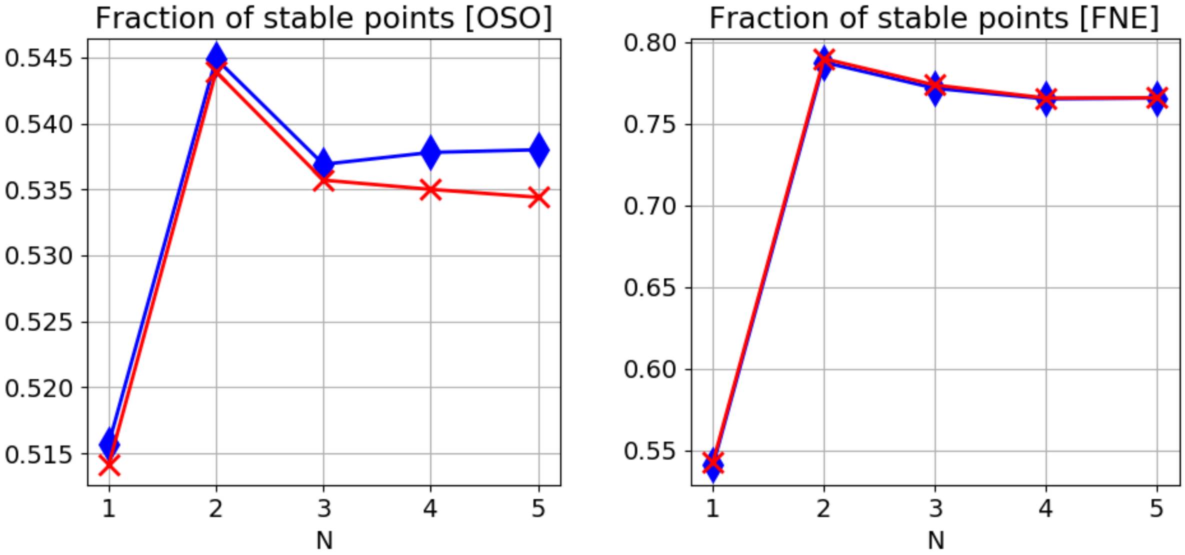

Figure 7.3: Chebyshev convergence results for interception problem, pursuer perspective

Fig. 7.3 shows the convergence behavior for increasing polynomial degree N from the pursuer's perspective in both the OSO and FNE problems. The plotted curves correspond to the two samples used for evaluation. The results agree to two significant figures. For each strategy, the change in the fraction of stable points between $N = 3$ and $N = 4$ is in the third significant figure, indicating the change is not statistically significant for the chosen sample size. From these results, it appears fourth-degree Chebyshev polynomials are adequate to characterize the opponent parameter space of interest. $N = 4$ is chosen for the results shown subsequently.

This analysis is repeated for specific engagements using constrained estimation, and $N = 4$ is used in the results that follow. The next subsection considers the design of nonadaptive, switching-based strategies for the evader.

7.2.5 Deception-based switching strategy

The concept of deception-based strategy switching is introduced in Chapter 6. If a player believes her opponent in a game is playing in a fashion that is reactive to the player's own actions, new approaches to the game become rational. Conceptually, a deceptive player should initially use some non-aggressive strategy long enough for the opponent to become convinced this is the player's true strategy. At a chosen switching time, the deceptive player begins using a more aggressive strategy, taking advantage of the lag induced between the onset of the change and the time the opponent recognizes the change. This subsection considers two such strategies for the evader. In the first strategy, the evader switches between constant gains at two different values. In the second strategy, the evader manifests a constant-gain strategy before switching to a true, LQDG strategy.

7.2.5.1 Gain level switching

In a gain level switching strategy, the evader uses an initially small gain before switching to a larger one. Conceptually, the idea is to "lure" the pursuer into a false sense of security, and take advantage of the transient that occurs while the pursuer adapts to the post-switch relative dynamics. There is no differential game associated with the evader's constant-gain strategy, so the terminal value of y is used as a relevant metric of performance for the evader. For the planar missile interception problem, the following piecewise-defined gain is used for the evader j:

$$K_j(t) = \begin{bmatrix} k_j(t) & 0 \end{bmatrix}, \;\; k_j(t) = \begin{cases} k_1, t < t_{sw} \\ \\ k_2, t \geq t_{sw} \end{cases} \tag{7.22}$$

For simplicity, it has been assumed that the evader places no weight on the velocity state.

There are three parameters that E must choose in this strategy; k_1, k_2, and t_{sw}. To design such a strategy, E assumes a model for P in which P responds optimally to any switch with a delay having first-order dynamics, as suggested and used in Chapter 6. For any Stage 1 and Stage 2 controllers for E, P's response is modelled as follows:

$$\boldsymbol{u}_i = \boldsymbol{u}_2^*(t) + e^{\alpha(t-t_{sw})}(\boldsymbol{u}_1^* - \boldsymbol{u}_2^*) \tag{7.23}$$

$\alpha < 0$ defines the rise time associated with P's response.

For the missile interception problem, E designs a strategy under perfect knowledge of P's true weights, measurement capabilities, and so on. The system is modelled as linear with no measurement error, enabling rapid iteration over different gain values. It is found that the switch must occur very early in the simulation for the switching strategy to outperform a fixed gain strategy. Additionally, the change in gains must be quite large to induce a long transient in the adaptive player's response. In general, this necessitates the evader choosing an initial gain that is **stabilizing** and works against his true long-term interests.

Consider the nominal engagement configuration and weights. Evader parameters of $t_{sw} = 0.1$, $k_1 = -6.0$, and $k_2 = 5.0$ are selected. Based on the observed performance of the pursuer in simulations, a value of $\alpha = -20.0$ is chosen, which leads to a 90% rise time of about 0.12 seconds in the pursuer's response. Fig. 7.4 shows the predicted response of the switching controller for an initial state of $y(0) = -0.1$, $\dot{y}(0) = -0.2$. For comparison, the predicted performance using a fixed gain k_2 for the entire simulation is also plotted. By using an initially stabilizing controller, the evader effectively "tricks" the pursuer into using a destabilizing control input during Stage 1. After the gain switch, the pursuer feedback remains destabilizing initially. The evader exploits this transient effect to increase the magnitude of the relative displacement y.

7.2.5.2 *LQDG with gain switching*

When E's true objective function is of the LQDG type, then the model of Section 6.4 is applied to the planar interception problem. This model is modified to also include P's transient from using an initial minimax gain, according to the same first-order adaptation model. This is an attempt to account for a source of error that was observed in the numerical example of Chapter

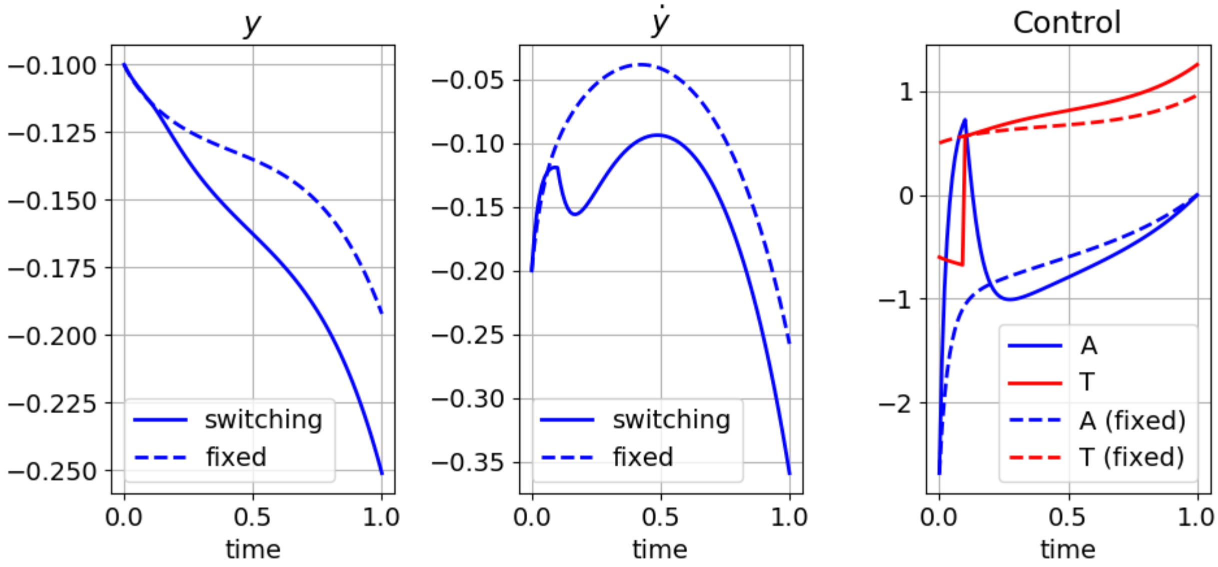

Figure 7.4: Linear states and controls for simplified model for gain switching strategy design

6. In Stage 1, the evader uses a constant gain before switching to the LQDG minimax gain[3] in Stage 2. During Stage 2 the evader exploits knowledge of P's gain and the adaptation dynamics to implement a strategy that is effectively optimal with respect to E's objective function. The simplified model assuming first-order adaptation dynamics is again used to select evader weights that improve nominal performance. Figure 7.5 shows the results for the same problem and initial conditions considered in that example. Results are shown for $\alpha = -20$, $t_{sw} = 0.10$, and a Stage 1 gain value of $[K_1] = [6.5, 0]$. The pursuer's nominal weights are used, but the evader has a NZS objective function corresponding to the following weights:

$$[Q_j] = \text{diag}(-.952, 0) \quad [R_{ji}] = [-.92] \quad [S_{f,j}] = \text{diag}(-9.0, 0) \tag{7.24}$$

Figure 7.5 shows the predicted state, control, and objective function histories. Again, these histories are generated against the simplified model of the adaptive player, and not evaluated against the full IMM in this section. The switching-based strategy leads to an increase in the terminal state

[3] The minimax strategy is used instead of the adaptive FNE strategy for simplicity in tuning the numerical examples later in the chapter.

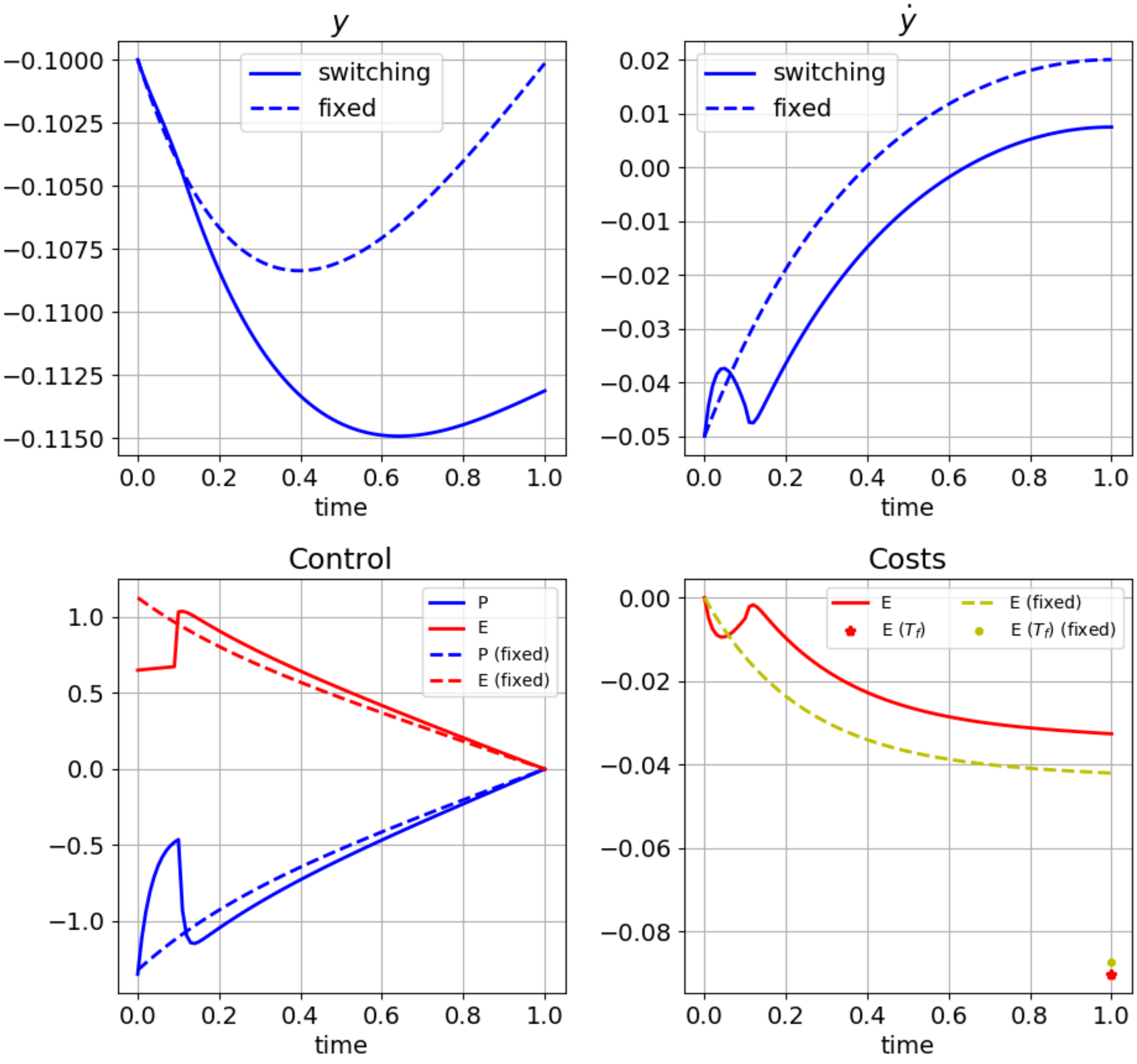

Figure 7.5: Linear states and controls for simplified model for design of the evader's LQDG strategy with gain switching

norm at the cost of greater control energy used. For reference, predicted histories with a fixed, FNE evader strategy are also plotted. The plot of the running cost shows that the switching evader has a larger penalty up to the terminal time, but has a more favorable terminal state, leading to better overall performance. The total cost incurred for the switching strategy is approximately $-.090$. When both players use an optimal, perfect information FNE strategies for the entire problem, the cost incurred by E is $-.087$. The switching strategy represents a predicted improvement of about

3.5% in the LQDG objective function.

This subsection has designed evader switching strategies, which are evaluated in the numerical results of this chapter. To improve robustness to evader switching, the next section describes a new model for the pursuer's IMM ensemble.

7.2.6 New considerations for multiple-model estimation

In this chapter, the pursuer plays adaptively against an evader whose strategy is not known. By using an ensemble of models with nonlinear estimators, the pursuer can respond to evader strategies subject to parameterized uncertainty. The estimation equations for the OSO and FNE strategies are the same as those presented in Chapter 2. Other evader strategies are possible, and IMM estimation is used to account for these potential approaches. It is ultimately not the diversity of possible opponent strategies that presents a major challenge for the adaptive pursuer. Rather, it is the possible presence of step changes in the opponent dynamics, as might occur if the evader uses a deceptive strategy. This subsection considers the estimation challenge associated with step changes in the opponent dynamics.

For any opponent strategy that is does not vary during the full engagement, an estimator can be implemented to account for parametric uncertainty and the appropriate optimal counter-strategy can be developed. However, when the opponent switches strategies mid-engagement, the switch manifests as a step change in the relative dynamics. Step changes are challenging for Kalman filter-based estimation because they appear as highly non-Gaussian errors in the dynamics. A deceptive strategy can cause an estimator to fail in two ways: 1) If the pre-switch dynamics exhibit a large enough error relative to the estimated state, the estimator can diverge; 2) A large step change in the relative dynamics can cause the filter to incorrectly adjust the estimated states in response, potentially leading to divergence of the estimator. It is possible to detect divergence and re-initialize the filter when this occurs. However, the structure of the IMM filter enables an elegant solution that appears to have good practical performance.

Essentially, for each candidate LQDG-based filter in the IMM ensemble, a duplicate filter is created and added to the ensemble. This duplicate has the same initial conditions and covariance, but does not perform measurement updates[4]. The duplicate model maintains a large state uncertainty. Consequently, the duplicate has a low likelihood associated with opponent behaviors over a broad region. If the opponent's initial response is of an LQDG type, then the original model converges to the associated weights with a small covariance and is classified as the most probable mode by the IMM algorithm. If the opponent's initial response is very different from the LQDG response, then the original model will have a smaller likelihood than the duplicate. This will cause the duplicate states and covariance to be mixed with the original model's, preventing the original model estimates from diverging. This effectively maintains the initial estimate and covariance, unless the opponent switches to a LQDG-type strategy. At that point, the original model's covariance starts to shrink and its response dominates the duplicate model's in the state mixing process.

To demonstrate, consider the evader switching strategy demonstrated in Section 7.2.5.2. The two agents are on a nominal, head-to-head intercept trajectory. The switch time is 0.1 and the prior strategy is a constant gain. The post-switch strategy is a ZS FNE strategy. The pursuer uses a simplified IMM containing three modes: a "constant-velocity" opponent gain model of the type introduced in Section 6.3, a LQDG FNE model, and a "high-variance" LQDG FNE duplicate model. The simulation performance for initial states of $y(0) = -0.1, \dot{y}(0) = -0.2$ is shown in Figs. 7.6 and 7.7.

Fig. 7.6 shows the control histories and mode probabilities from the pursuer IMM. The control history shows the pursuer responding initially to the constant gain behavior and quickly switching to the optimal control when the evader switches to an FNE control strategy. The pursuer identifies the FNE mode as more likely on the second measurement after the evader's switch. The mode probabilities show that the duplicate FNE filter, denoted "FNEd", is never considered as the most likely mode. Despite this, Fig. 7.7 demonstrates that the duplicate filter is having the intended effect of allowing the adaptive FNE estimator to re-initialize after the switch occurs. This figure plots the individual filter state estimates, which include the effect of IMM state mixing. During the initial transient, the original LQDG state estimate diverges to a large error as it attempts to

[4] Alternately, the duplicate can be instantiated with an artifically large measurement variance, achieving the same effect.

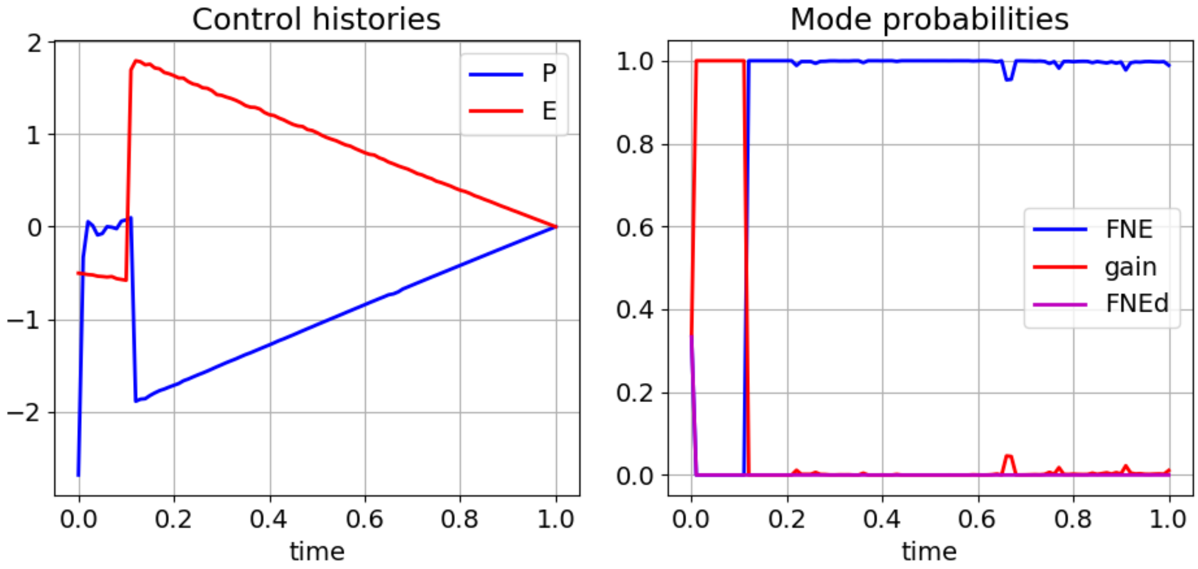

Figure 7.6: Simulation results for a switching evader (E) playing a true LQDG strategy against an adaptive pursuer (P).

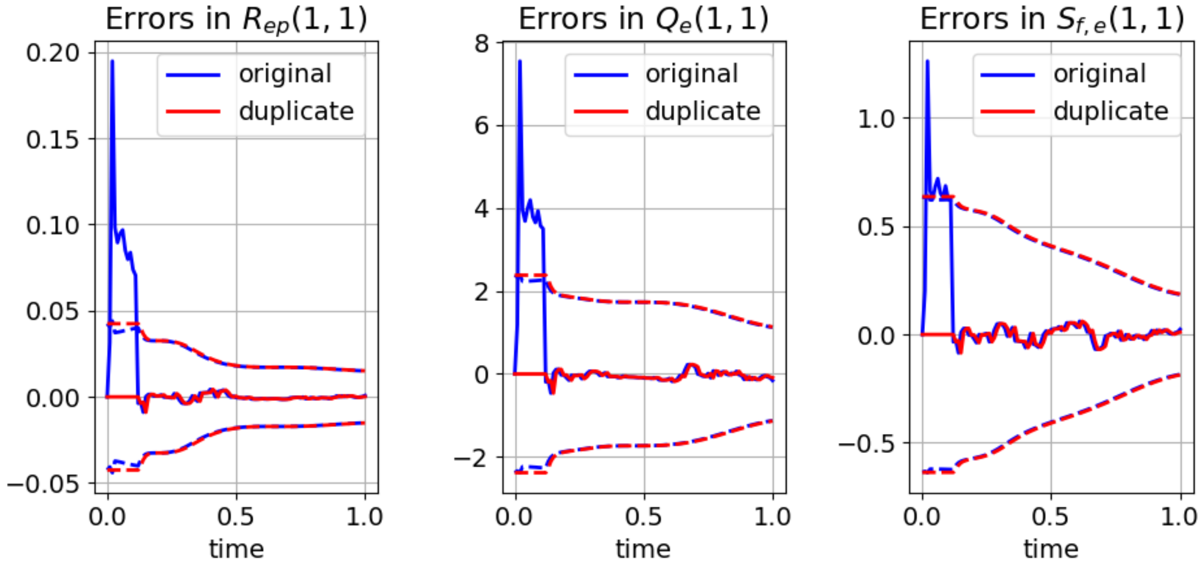

Figure 7.7: State estimation histories for the original and duplicate pursuer LQDG filters playing against a switching evader. Duplicate filter assumes artifically high measurement variance.

model the constant gain opponent. The duplicate state estimate remains at its initial value, which happens to be the truth value. After the evader's switch, the original filter's state estimates are mixed back into the duplicate filter's. Afterwards, the original filter is able to track the true states, and the duplicate filter's states are mixed with those, leading the duplicate state's estimates and covariance to converge. The results indicate the duplicate filter works as intended and achieves smooth re-initialization of the original filter when the switch occurs. The duplicate filter concept is employed in Section 7.3.4 for a pursuer using IMM estimation against a deceptive evader.

This section has described relevant details of the numerical implementation of the missile interception differential game. The nominal engagement trajectories and weights have been summarized. The necessary filter equations have been described. Analysis has been conducted to determine appropriate Taylor series polynomial degrees and Chebyshev polynomial degrees for RDE solution approximation and constrained estimation, respectively. Deceptive nonadaptive strategies have been designed for the evader for gain level-switching and gain switching into LQDG. To improve robustness of LQDG estimators to step changes in the evader dynamics, simple "duplicate" filters with high variance are appended to the set of IMM ensemble models. Having presented these details, the numerical results are now reviewed.

7.3 Results

Numerical simulations are conducted to demonstrate the various advances and techniques outlined in previous chapters. Results include a mixture of individual simulation summaries, to highlight particular aspects of the performance, and Monte Carlo results, intended to quantify the overall performance in a stochastic environment. Before presenting the results, the procedures associated with the Monte Carlo analysis and some miscellaneous details of the simulations are reviewed.

Unless noted otherwise, Monte Carlo samples use simple random sampling in which the pursuer's weights are constant and the evader's weights are sampled according to the pursuer's initial covariance. Initial states are sampled on independent uniform distributions $U[-.1, .1]$. The method of repeated sampling employed in Chapter 3 is used to ensure that all samples yield nonsingular solutions under perfect information.

The number of Monte Carlo simulations to conduct is determined by performing a limited number of convergence studies. Essentially, for each distinct set of player weights or trajectories considered, some set of simulations is conducted twice and compared to validate convergence and the number of significant figures. These results are not discussed in this chapter because they involve comparing tedious charts with very similar values. The convergence study results are included in Appendix D. Typically, 2500 simulations are used when both weights and initial states are sampled, and most performance metrics converge to at least one significant figure for that sample size.

Monte Carlo results are typically quantified using the number of failed simulations, the relative loss function ℓ in the overall cost introduced in Chapter 3, and the estimation errors. Estimation errors are quantified in terms of MSE, error covariance, median absolute error, and 25% and 75% percentile values of absolute error[5]. The MSE is a common metric of estimation performance, but it is sensitive to outliers. By comparing the MSE and median absolute error, it is straightforward to tell if outliers have a significant effect on the performance.

A few addenda should also be mentioned for completeness. In simulations where the pursuer uses an IMM, the IMM-C approach is used to generate control values. When estimation error plots are made, 3σ bounds are computed using the diagonal elements only of the covariance matrix. This truncates some uncertainty and should be kept in mind. When adaptive player costs are addressed, the achieved cost is compared against a reference value. Unless stated otherwise, the reference value is the perfect information (PI) predicted cost-to-go with *continuous* dynamics, assuming each player has exact knowledge of the opponent weights. Each player's controls are implemented with a zero-order hold (ZOH), which can enable the adaptive agent to outperform the reference value in some cases. Based on the numerical results, the cost metrics should typically be considered only to two significant figures, due to the "rounding" effect of the ZOH.

Each of the following subsections is intended to emphasize a particular method, using nu-

[5] Respectively, the lower and upper quartiles.

merical weights chosen to highlight different aspects of the differential game. Five topics are considered. The first topic considers the performance of an adaptive player who knows the strategy used by her opponent. These engagements highlight the improvements introduced in Chapter 4. Second, the effectiveness of constrained estimation in reducing control failures in engagements that approach a stability boundary is examined. Third is the ability of the IMM-based mode estimation to distinguish minimax and adaptive FNE opponent strategies. Fourth is the performance of a pursuer using an IMM against an evader who executes a strategic switch. Fifth is a general study of the performance of a pursuer using the IMM against an evader using selected strategies. All subsections except the last include both individual simulation and Monte Carlo results.

7.3.1 Adaptive LQDG results

The results in this subsection introduce the interception problem and highlight the cost-based loop closure in adaptive agents who are subject only to parametric uncertainty in the opponent strategy. The problem is of the same type studied in Chapter 3. Individual simulations of a planar engagement are discussed first, and use fixed initial conditions and agent weights. The initial state is $y(0) = -0.1, \dot{y}(0) = 0$. The pursuer uses the nominal weights of Section 7.2.1; the evader uses the following weights:

$$[R_{ep}] = [-.97] \qquad [Q_e] = \text{diag}(-2.8, 0.0) \qquad [S_{f,e}] = \text{diag}(-10.25, 0.0) \tag{7.25}$$

The initial parameter covariances used for the opponent r, q, and s_f elements are $5 \times 10^{-4}, 1$, and 9×10^{-2}, respectively, for both pursuer and evader. The initial variance in the relative state estimates is set to ten times the measurement variance. Each adaptive agent models uncorrelated process noise on the state channels $y, \dot{y}, r_{ij}, q_i, s_{f,i}$, with process noise covariance of $\text{diag}(10^{-4}, 10^{-4}, 10^{-3}, 10^{-2}, 1.0)$. Later in this subsection, the estimation and objective function performance are quantified in terms of Monte Carlo results.

First, an engagement in which both players use minimax strategies is considered. Fig. 7.8 shows a summary of the resulting simulation states. The inertial trajectories, perturbed relative

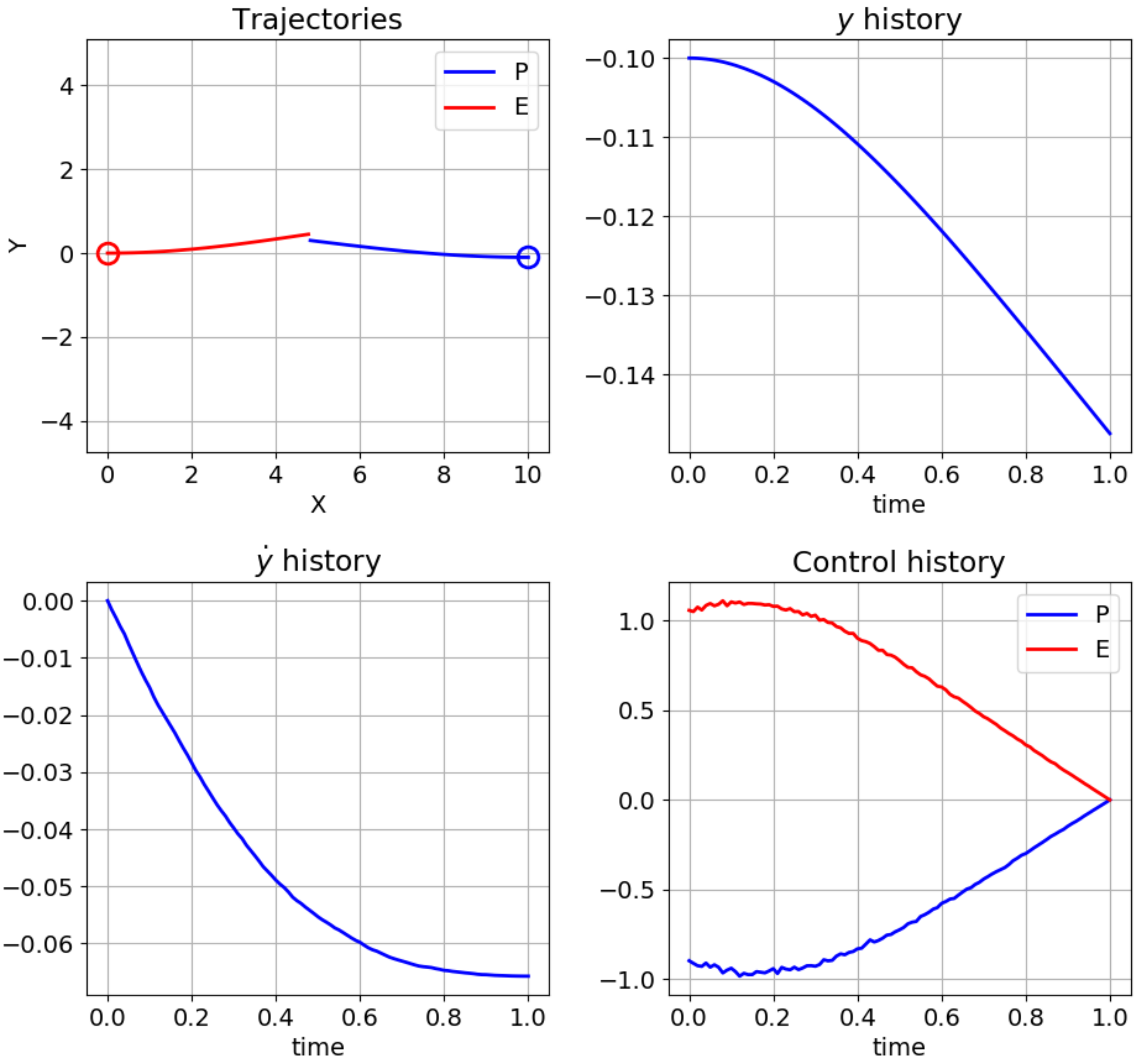

Figure 7.8: Summary simulation states for a planar missile interception engagement with non-adaptive, NZS players

states, and control histories are shown. The initial positions in the inertial space are delineated with open circles. Based on the results shown, the evader appears to use slightly more control energy than the pursuer, and is able to increase the relative displacement slightly over the course of the simulation. However, the increase is relatively mild, and likely within the linear range of the relative dynamics. For this scenario and these initial conditions, the pursuer achieves an objective function value of 0.040, and the evader achieves a value of -0.064. These values are of interest in

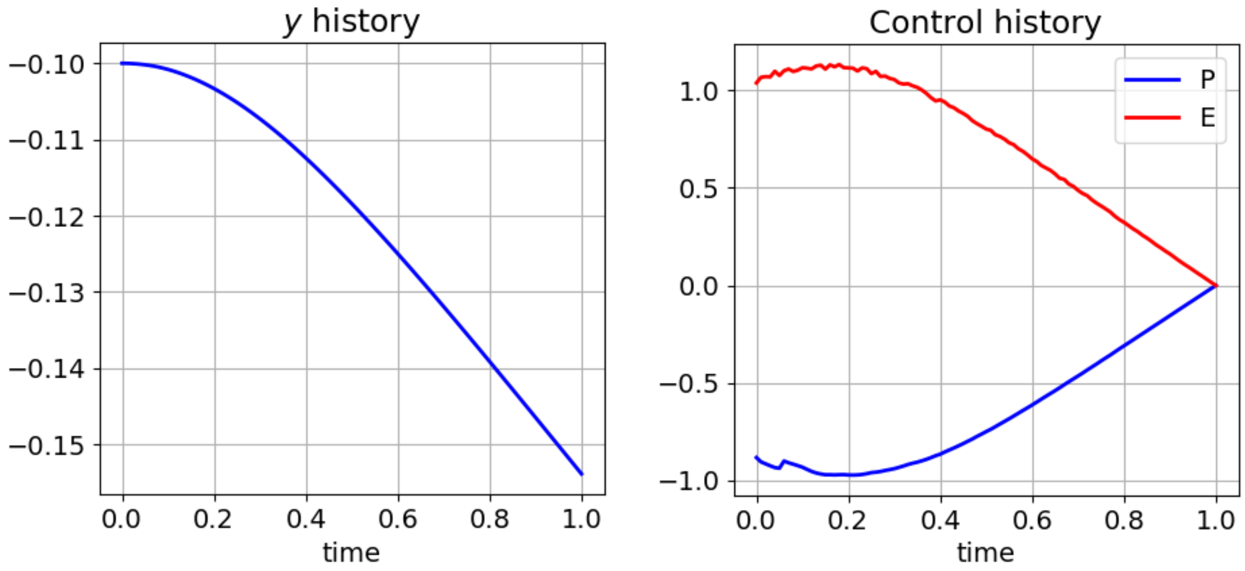

Figure 7.9: Summary simulation states for a planar missile interception engagement with OSO adaptive pursuer

comparing the performance with adaptive strategies.

Fig. 7.9 shows the y state and control histories for the same initial conditions when the pursuer plays according to the adaptive OSO strategy against the minimax evader. (The trajectory and $\dot{y}$ plots show minimal change.) The most noticeable change is a small jump in the pursuer control history occurring around $t = 0.06$. This jump occurs when the pursuer's confidence in his estimate of the cost-to-go exceeds the threshold value and the pursuer begins using his estimates to compute feedback controls. The result is a slight decrease in the pursuer's control energy, which manifests as a small increase in the terminal relative state compared to the non-adaptive engagement. Essentially, the pursuer determines that the evader is using less control energy than initially expected, which allows the pursuer to increase his performance metric by reducing his control use, as well.

The discussion of the adaptive OSO pursuer performance is continued with Fig. 7.10, which summarizes the pursuer's estimation and cost performance. The overall cost incurred matches the cost of the minimax engagement to two significant figures, and indicates that effectively no net change in performance is experienced due to adaptation. The plots show the errors in the estimation of the evader parameters, the cost-to-go history, and the estimate of the closed-loop cost-to-go

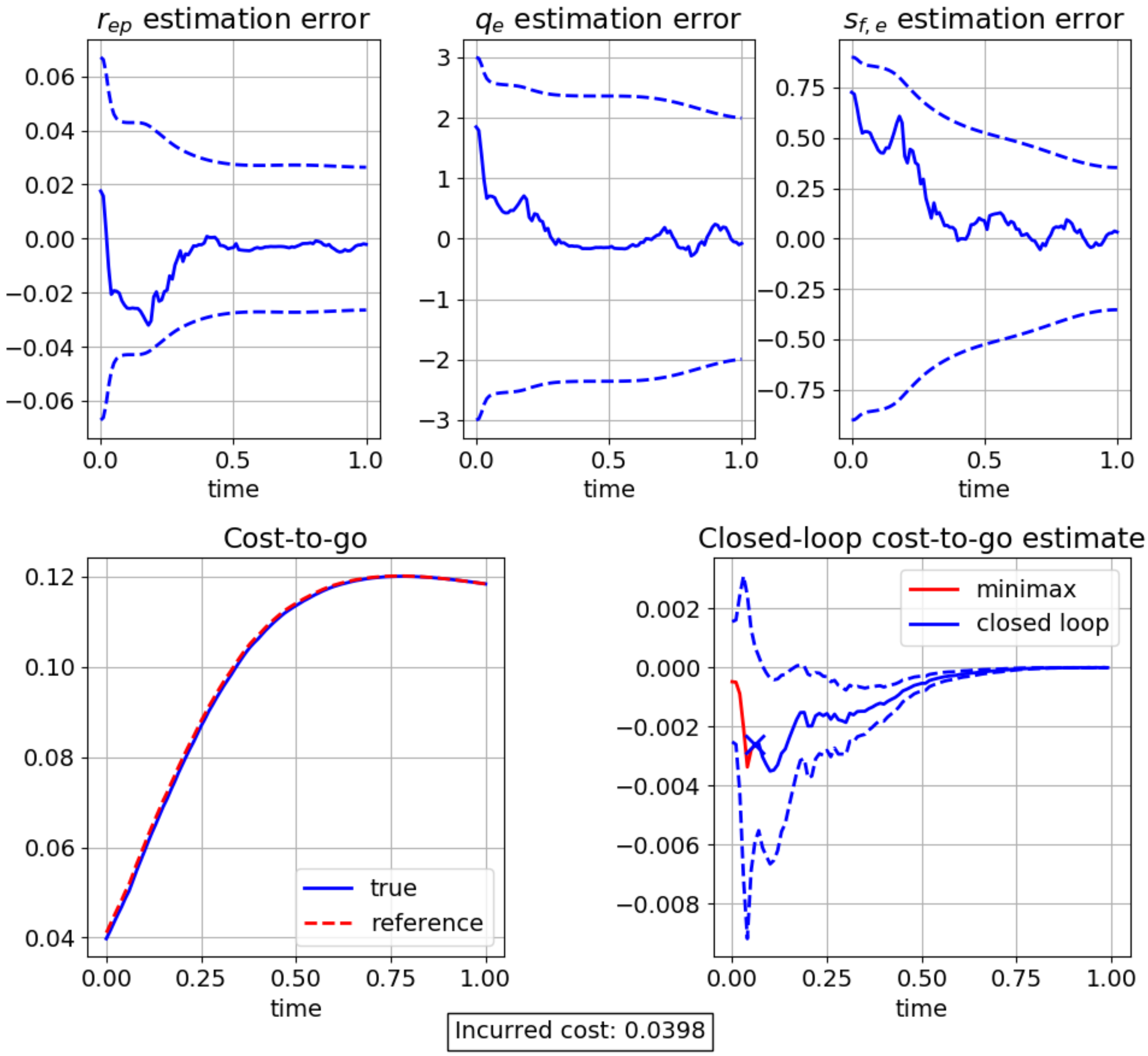

Figure 7.10: Estimation performance for a planar missile interception engagement with OSO adaptive pursuer

change. The relative state estimation errors are qualitatively bounded by the 3σ values and are not plotted for brevity. The parameter estimation errors converge to small values and fall within the 3σ covariances. It appears the estimator is slightly underconfident in this engagement, meaning the process noise could be tuned to smaller values. The cost-to-go lies essentially on top of the reference cost-to-go value. Recall that the reference cost-to-go is computed under the assumption of perfect information. This finding indicates that the initial errors in the opponent parameters has

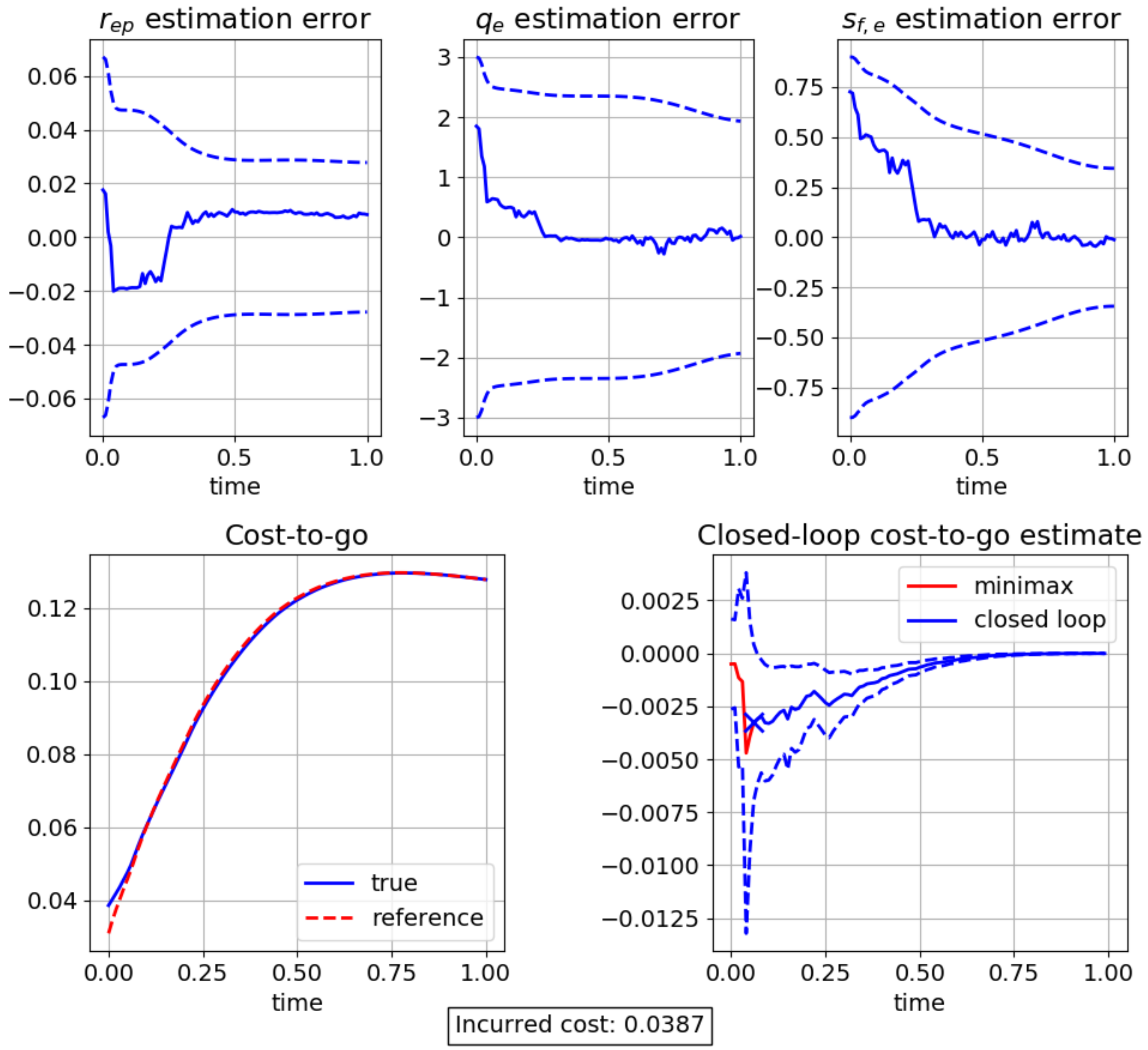

Figure 7.11: Pursuer estimation performance for a planar missile interception engagement with FNE players

a minimal effect on the overall performance for these initial conditions. The bottom right figure displays the pursuer's estimate of the change in cost-to-go achieved by switching from a minimax control to an adaptive OSO control strategy. The 3σ covariance in the cost-to-go is computed according to the unscented transformation and plotted for reference. The "X" indicates the first time at which the estimates are used to generate the control. This time is roughly coincident with the 3σ covariance falling below the y-axis. This time clearly coincides with the small step change

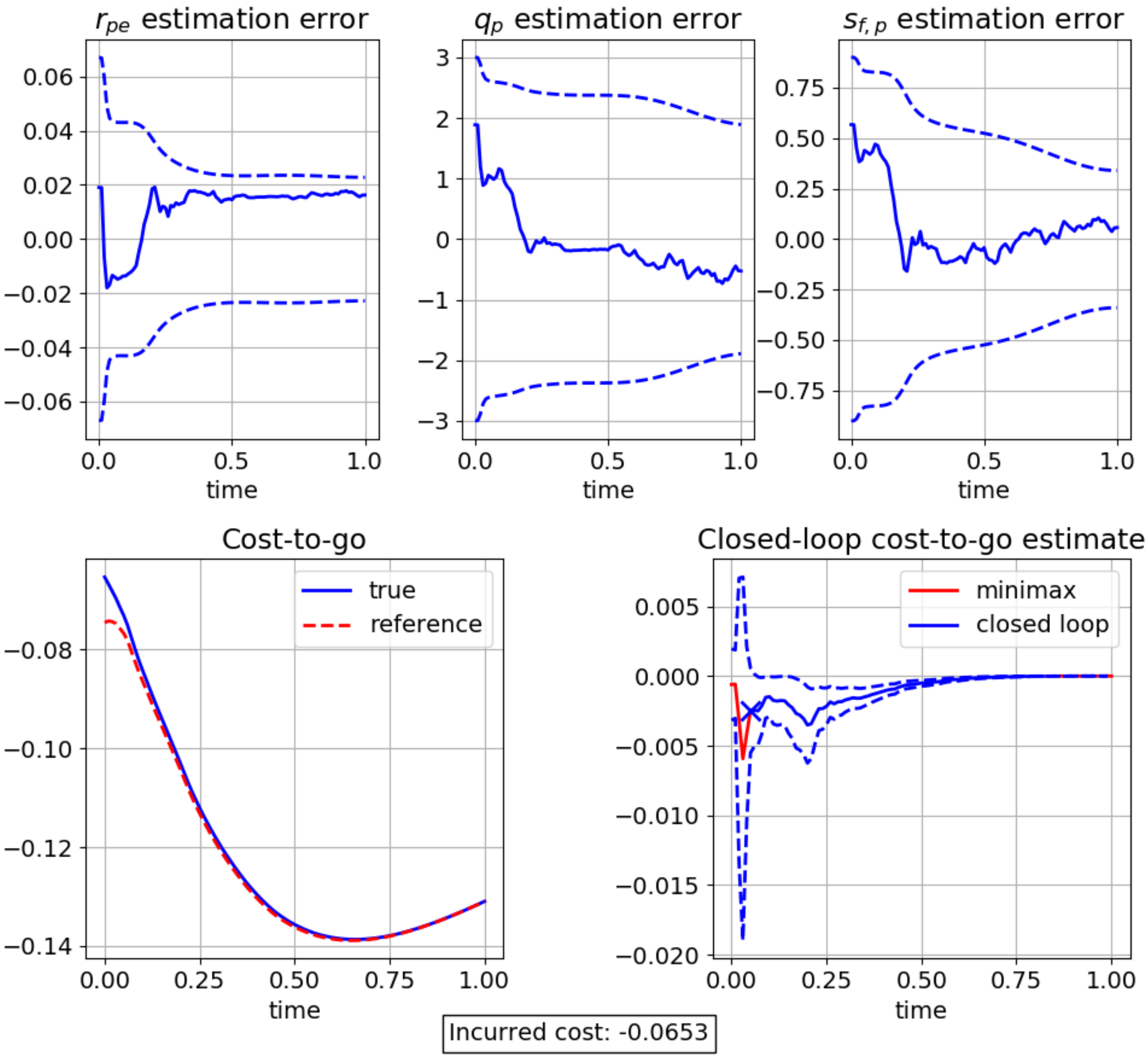

Figure 7.12: Evader estimation performance for a planar missile interception engagement with FNE players

in the opponent control noted in Fig. 7.9.

Figs. 7.11–7.12 compare the pursuer and evader estimation histories when both players use the adaptive FNE strategy. The state and control histories are similar to those shown for previous engagements, and are omitted. Both agents' estimation performance shows reasonable error reduction. As in the OSO case, the estimators are underconfident. Both players show a terminal bias particularly in the estimation of the opponent control penalty term; however, it must be recalled that this is an expected characteristic of the adaptive FNE solution. The pursuer achieves an overall cost of 0.039, and the evader achieves an overall cost of -.065. That is, under adaptive FNE gameplay, the pursuer is able to reduce his cost by the same amount the evader's cost increases. This reflects the near zero-sum nature of the engagement. Additionally, it should be noted that each agent's performance is worse than his PI reference value. The pursuer and evader cost-to-go curves converge to the reference curve by the time each agent begins using the parameter estimates for control, which indicates that the adaptive game strategies are effectively achieving the reference performance with an adaptation delay. This adaptation delay can be reduced by decreasing the relative state measurement covariance, which leads to a corresponding improvement in each player's performance.

The individual simulation results presented in this subsection highlight the performance of adaptive agents using the new developments of Chapter 4. It has been shown that the strategy of adaptively switching from minimax to LQDG strategies based on estimation error eliminates the large initial transients that previously appeared in the PE games of Chapter 3. Now, the performance of adaptive agents in Monte Carlo simulations is considered.

Monte Carlo simulations are conducted to evaluate the pursuer's performance when the evader uses each of the minimax, adaptive FNE, and constant gain control strategies. The pursuer knows the evader's strategy with parametric uncertainty. Table 7.2 summarizes the estimation error metrics for each simulation case. In each set of simulations, there are no failed simulations due to premature termination or excessively large control inputs. The error metrics indicate the error between the estimated value of the opponent gain and the true value. In simulations, gain values are usually on the order of one; results indicate the typical absolute error metrics, quantified by the lower and upper absolute error quartiles, are between 10^{-3} and 10^{-1}. For engagements with minimax or adaptive FNE evaders, the typical MSE is on the order of 10^{-2}. This magnitude is consistent with the absolute error metrics, and suggests that outlier effects are not significantly affecting the error metrics. The pursuer error metrics between the OSO and FNE cases match to one significant figure and indicate there is a minimal change in estimation errors between the two

	MSE	Median Abs Err	Lower Quartile	Upper Quartile
OSO Pursuer	0.0346	0.0242	0.00592	0.0803
FNE Pursuer	0.0342	0.0258	0.00641	0.0808
FNE Evader	0.0263	0.0263	0.00643	0.0845
LQDG-gain Pursuer	0.736	0.0302	0.0119	0.0928

Table 7.2: Summary of gain estimation error metrics for adaptive agents in Monte Carlo simulations (2500 simulations for each evader strategy)

cases. The FNE evader achieves a smaller MSE metric but slightly larger absolute error metrics compared to the pursuer estimation error. The change in MSE is in the first significant figure, but the change in median and quartile values is only in the second significant figure. It is unclear how to interpret this MSE result. Since the other error metrics show little change, it may be that the change in the MSE is due to outlier effects and not due to significant changes in the underlying distribution. In the engagement with constant-gain evader, the pursuer's error metrics are slightly larger than against LQDG evaders. Noticeably, the MSE is an order of magnitude larger. The lower quartile values increase by almost a factor of two compared to OSO and FNE engagements, while the median and upper quartile errors increase by only about 15%. This larger error metric, particularly the MSE, can be partially attributed to the initialization of the LQDG-gain filter. In these engagements the pursuer has a large initial covariance and a large initial error, since the gain estimate is initialized to zero. There is a significant transient effect that must be overcome at the start of the simulation. This contributes to the larger MSE error metric.

Table 7.3 summarizes the relative loss metrics in each set of simulations. It must be recalled that these are relative loss metrics that can be interpreted as percentages; due to the truncation error from using a ZOH for control, they are only accurate to about 1%. In this light, it is clear that when the evader uses the minimax or adaptive FNE strategies, the performance of the adaptive pursuer is not significantly different than the reference performance. When the evader uses the constant gain strategy, there does appear to be a significant loss incurred by the pursuer, relative to the reference value. Typical loss function values are between about -4% and -38% for the pursuer in these scenarios. At first consideration, this appears to a large degradation compared to the OSO

	Median Loss	Lower Quartile	Upper Quartile
OSO Pursuer	0.0038	0.000155	0.00914
FNE Pursuer	-0.000638	-0.0029	0.00146
FNE Evader	4.07×10^{-5}	-0.00283	0.00277
LQDG-gain Pursuer	-0.11	-0.376	-0.0396

Table 7.3: Summary of relative loss function metrics for adaptive agents in Monte Carlo simulations (2500 simulations for each evader strategy)

and FNE results. However, it is reasonable to believe this is primarily a transient effect, due to the larger initial uncertainty. Consider the adaptive FNE cost-to-go history of Fig. 7.11. In this scenario, the pursuer achieves a relative loss metric of about -26%; however, it is qualitatively clear that the pursuer's cost-to-go history converges to the reference value within several measurements, as it is meant to. Overall, in the OSO and FNE engagements, adaptive players effectively recover their reference values because the initial errors are zero mean. In the engagement with constant evader gain, the adaptive pursuer overcomes a nonzero mean initial error and achieves typical loss metrics of between -4.0% and -38%.

Overall, the adaptive LQDG results demonstrate that the new improvements of Chapter 4 are having the intended effect in eliminating harmful initial transients. The typical estimation errors in Monte Carlo show a reduction in the outlier effects. The relative loss achieves very small values in adaptive LQDG solutions and slightly more negative loss values against a constant gain evader, due to the larger initial estimation error. In the next subsection, the use of constrained estimation to reduce simulation failures in near-unstable engagements is considered.

7.3.2 Constrained estimation performance

This subsection presents results using constrained estimation to avoid numerical singularities in the generalized RDEs. First, new problem parameters are chosen to exacerbate the occurrence of singularities. Second, individual simulations are presented for both constrained and unconstrained estimation. Third and finally, Monte Carlo results are presented comparing constrained and unconstrained estimation for sampled values of the evader's weights.

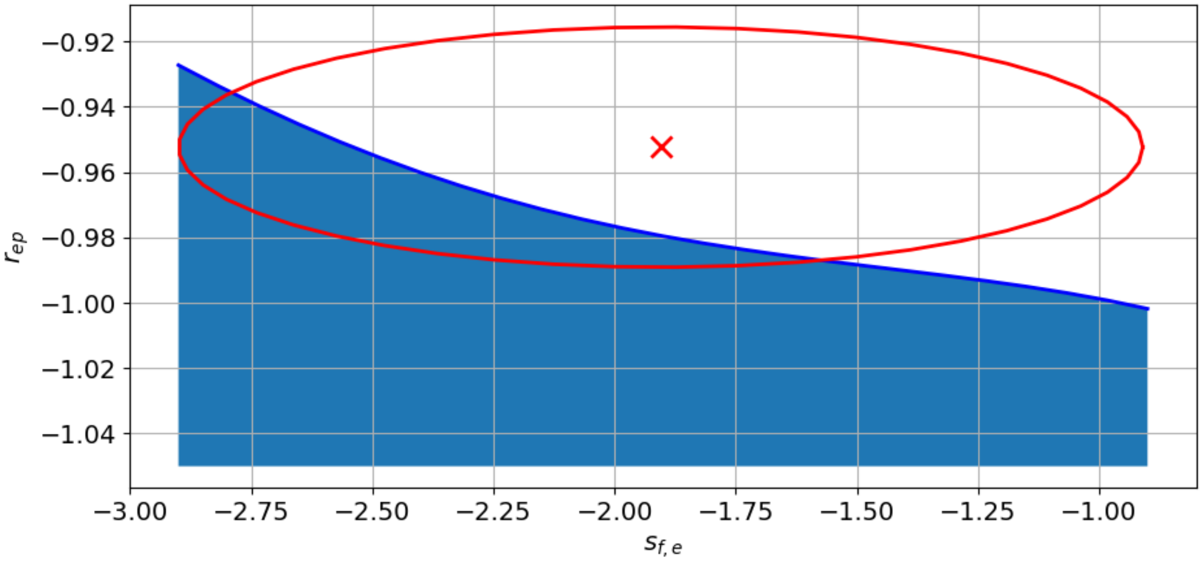

Figure 7.13: Initial covariance 3σ ellipse and Chebyshev stability boundary for constrained estimation engagement with $q_e = -5.71$

To study constrained engagement, a nonplanar scenario with equal problem weights on y and z direction states is used. The effects of constrained estimation are only noticeable if the constraints are active; otherwise the constrained and unconstrained estimators are identical. For Kalman filter-based constrained estimation, the engagement stability boundary must be close enough to the nominal opponent weights that the filter does not diverge if the true weights are near the stability boundary. To study the effects of constrained estimation, the problem parameters are modified to allow for the existence of singular engagements that are within the filter's effective radius of convergence. The time horizon and running state costs are both increased. The player speeds are not changed from the nominal values, and the initial range is increased to match the time horizon. The large running cost increases the sensitivity of the engagement stability to the unknown evader weight $[R_{ep}]$ and allows the engagement to become unstable for relatively small changes in $[S_{f,e}]$. The simulation is nominally zero-sum. The modified time parameters and pursuer weights are displayed in Table 7.4.

For the chosen problem settings, the pursuer has an initial uncertainty in the opponent weights

T_f	Δt	q_p	r_{pe}	$s_{f,p}$
2.0	0.01	6.0	-1.05	2.0

Table 7.4: Summary of problem weights for constrained estimation numerical examples

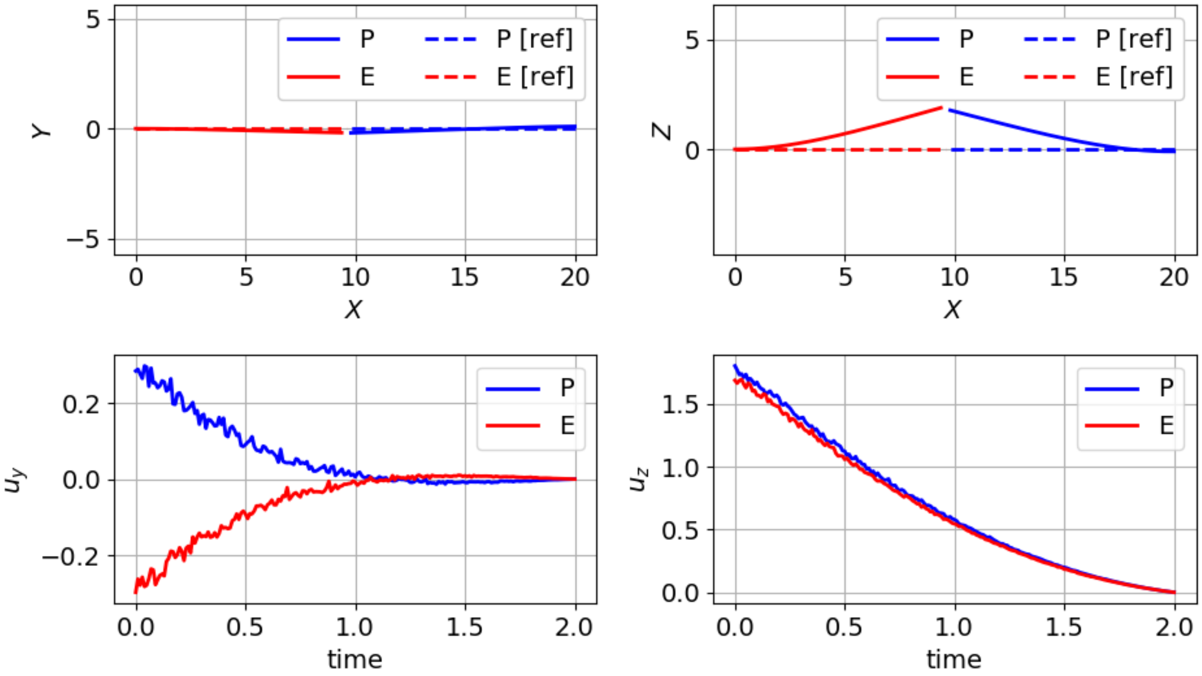

Figure 7.14: Nominal simulation performance in constrained estimation scenario with minimax pursuer and evader

described by the covariance $[P_{p,\theta\theta}] = \text{diag}(1.0 \times 10^{-4}, 0.018, 0.11)$. The covariance applies to the uncertainty in $r_{ep}, q_e,$ and $s_{f,e}$ respectively. Given the problem settings, a fourth-order Chebyshev polynomial is fit as a function of $r_{ep}, q_e, s_{f,e}$ to approximate the engagement stability boundary for a minimax evader. The projection of the stability boundary and the initial 3σ covariance ellipse at $q_e = -5.71$ are plotted for reference in Fig. 7.13. The shaded region in the plot corresponds to values of the parameters for which the problem is unstable. The engagement is stable for values near the zero-sum point, which is denoted by an "X," but becomes unstable as r_{ep} and $s_{f,e}$ increase. The chosen conditions are such that the majority of sampled initial evader weights yield stable engagements, but unstable engagements are reasonably likely.

$y(0)$	$\dot{y}(0)$	$z(0)$	$\dot{z}(0)$	q_e	r_{ep}	$s_{f,e}$
0.1	-0.05	-0.1	-0.05	-5.71	$-.977$	-2.0

Table 7.5: Summary of problem settings for individual simulations using constrained estimation

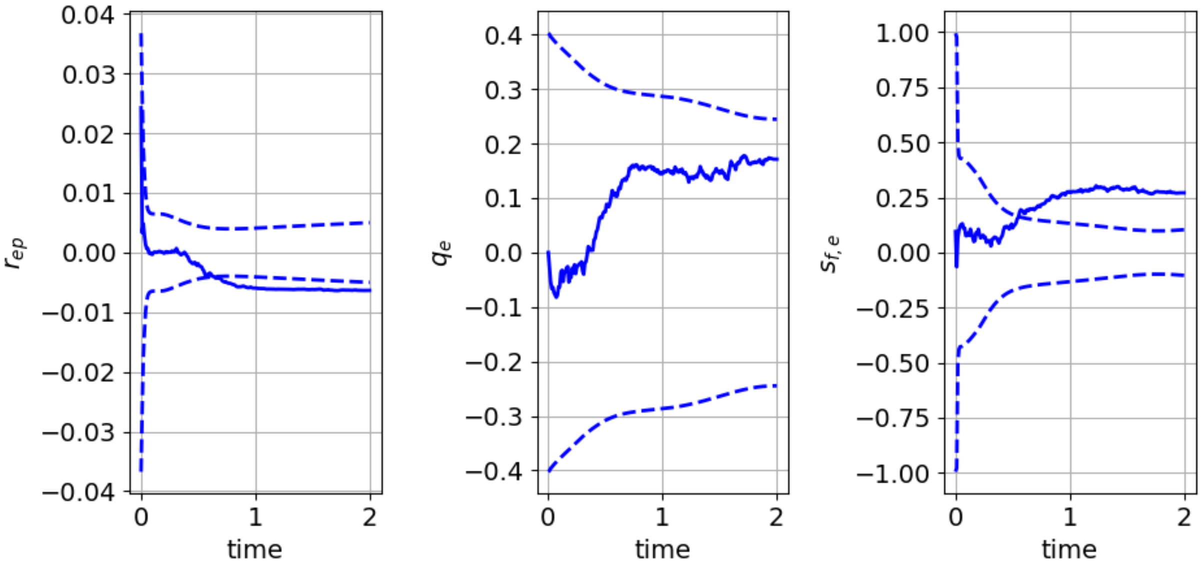

Figure 7.15: Estimation error history with constrained estimator. Weights used are $r_{ep} = -.977, q_e = -5.71, s_{f,e} = -2.0$.

To introduce the nonplanar engagement, first consider a simulation in which a ZS game is played, and each player uses a minimax strategy. The initial states are the same as those of Table 7.5. Fig. 7.14 shows the resulting inertial trajectories and control inputs for each agent. The unperturbed reference trajectories are plotted for comparison. The greatest part of the control energy is expended in the X–Z plane by about an order of magnitude. Both agents experience large perturbations from the reference trajectory, but a small terminal range is maintained.

To demonstrate the utility of constrained estimation, two simulations are conducted: one using constrained estimation, and one using a standard UKF. These engagements use NZS evader weights chosen from the stability interface. Table 7.5 summarizes the set of initial conditions and evader weights used in these engagements. Fig. 7.15 shows the estimation error history when

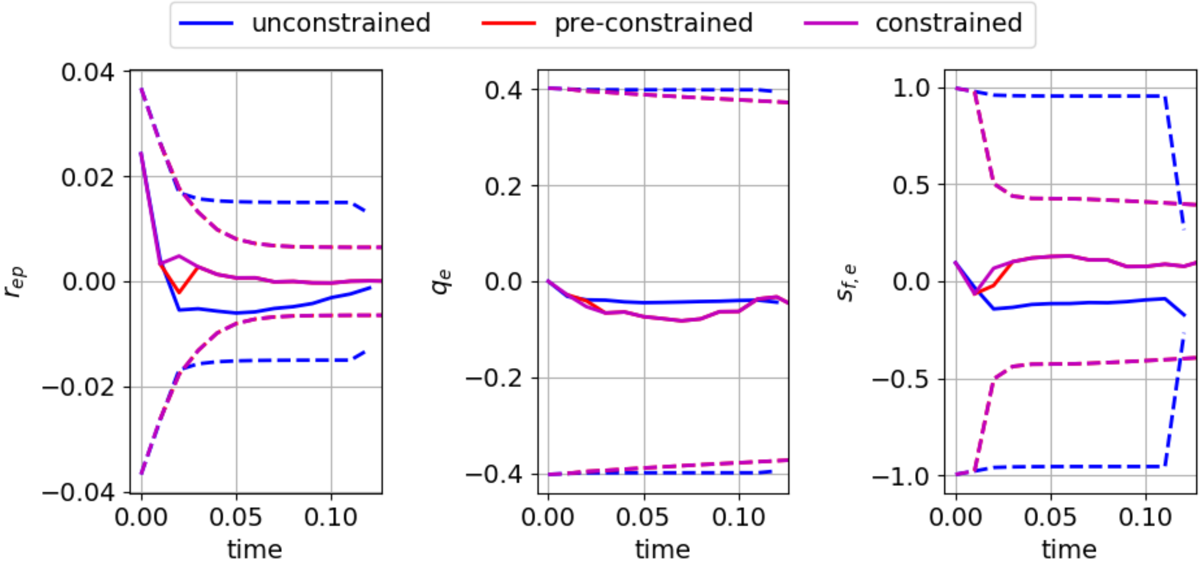

Figure 7.16: Comparison of failed constrained estimator errors with constrained estimator performance. Weights used are $r_{ep} = -.977, q_e = -5.71, s_{f,e} = -2.0$.

the pursuer uses constrained estimation. The relative states are qualitatively similar to those of the nominal simulation and are not plotted. The estimation errors show that a large initial error in r_{ep} is overcome and relatively small terminal errors in each parameter are achieved. There is a noticeable bias in the terminal estimates. Recall that estimation bias is a known feature of the PDF truncation, so it is not surprising the see this result.

Fig. 7.16 shows the estimation error performance when a standard UKF is employed. The simulation terminates after about a dozen time steps because the RDE solution using the estimated parameters is singular. For comparison, the time history of the constrained estimates are also shown. The “pre-constrained” estimate values denote the value of the estimates before PDF truncation is used, and these values match the unconstrained estimate error histories until the first time the PDF is truncated. PDF truncation has two effects: relatively small corrections in each of the state components, and noticeable reductions in the covariance of r_{ep} and $s_{f,e}$. By using constrained estimation, a numerical singularity is avoided and the engagement runs to completion.

Table 7.6 shows the error metrics for Monte Carlo simulations with sampled initial condi-

	Unconstrained estimation			Constrained estimation		
	r_{ep}	q_e	$s_{f,e}$	r_{ep}	q_e	$s_{f,e}$
Mean Square Error	6.88×10^{-5}	0.376	0.0465	8.58×10^{-6}	0.0194	0.0199
Error Covariance	0.00829	0.613	0.215	0.00291	0.139	0.14
Median Abs. Error	0.00131	0.0754	0.0597	0.00131	0.0745	0.0596
Lower Quartile Abs. Error	0.000603	0.0362	0.0266	0.000609	0.0354	0.0258
Upper Quartile Abs. Error	0.00242	0.130	0.119	0.00244	0.130	0.118

Table 7.6: Monte Carlo error summary for 2500 normally distributed evader weights in constrained estimation engagements

	Unconstrained estimation			Constrained estimation		
	r_{ep}	q_e	$s_{f,e}$	r_{ep}	q_e	$s_{f,e}$
Mean Square Error	0.00118	24.8	3.24	0.000114	0.723	0.106
Error Covariance	0.0323	4.72	1.77	0.0093	0.819	0.289
Median Abs. Error	0.00361	0.138	0.193	0.00393	0.124	0.171
Lower Quartile Abs. Error	0.00152	0.0607	0.0824	0.00181	0.0563	0.0832
Upper Quartile Abs. Error	0.0113	0.387	0.462	0.00803	0.24	0.301

Table 7.7: Monte Carlo error summary for 2500 boundary sampled evader weights in constrained estimation engagements

tions and evader weights. Evader weights are nominally Gaussian and are re-sampled such that a nonsingular perfect information game solution exists. 2500 simulations are conducted each with constrained and unconstrained estimation. Out of these, twenty-five of the simulations with unconstrained estimation failed to complete, as did two of the simulations with constrained estimation. Because so few simulations fail, the difference between the constrained and unconstrained estimation errors is minimal. There appear to be some outlier effects causing the unconstrained estimates of q_e and $s_{f,e}$ to have larger MSE and error covariance than the constrained values. However, the median and quartile values match to one or two significant figures between constrained and unconstrained cases, so overall there is a minimal change in the error metrics of a typical engagement.

To more concretely quantify the benefits of constrained estimation in near-singular engagements, an additional Monte Carlo study is conducted. In Table 7.7, the evader weights q_e and $s_{f,e}$ are sampled from a normal distribution and r_{ep} is selected as the corresponding value on the approximate stability interface. All engagements are nearly unstable. Conferring Fig. 7.13, it is obvious that sampling in this fashion selects some values of r_{ep} that lie outside the 3σ bounds of the initial covariance. Inevitably, the filter will diverge or perform poorly in many engagements, regardless of the use of constrained estimation. Out of 2500 simulations conducted for both constrained and unconstrained pursuer estimation, the simulation fails in 1123 cases without the use of constraints and 190 cases with the use of constraints. It is clear that the use of PDF truncation has a significant effect. In terms of the error metrics, the mean-based metrics (MSE and error covariance) are poor for both estimators, most likely due to the presence of outliers. The constrained estimation performs better than unconstrained estimation, but the MSE and error covariance are typically one to two orders of magnitude higher compared to Table 7.6. The median-based metrics are of greater interest, since they are less sensitive to outliers. First, consider the magnitude of the median-based error metrics for boundary sampling versus the previous sampling scheme. Using constrained estimation, errors in r_{ep} increase by about a factor of three. Median errors in q_e increase by about a factor of two, and errors in $s_{f,e}$ by about a factor of three or four. The unconstrained estimation median error metrics follow the same trend.

Continuing the analysis of Table 7.7, there are some apparent trends in the estimation error performance between the unconstrained and constrained cases. Unconstrained estimation achieves a smaller error in the median and lower quartile errors in r_{ep}, as well as the lower quartile error in $s_{f,e}$. For all other median-based metrics, constrained estimation achieves smaller errors. The constrained estimates obtain smaller upper quartile values consistently, suggesting a reduction in outliers and other sources of large errors. In essence, the "worst-case" outcomes are not as bad using the constrained estimates. The results for the median and lower quartile values are more mixed. Note that r_{ep} is treated as the dependent variable in the approximation of the stability manifold. The results indicate that the constrained estimate achieves larger errors in estimating r_{ep} but lower errors in the other variables at the 25% and 50% quartile values. It appears that the action of constraining the estimates increases the estimation accuracy of the independent variables q_e and $s_{f,e}$, presumably by confining the state estimates to the stable regions of the parameter space in which the true values lie. It is interesting that this effect does not apply to estimates of r_{ep}. It is plausible that the choice of approximating r_{ep} as a function of the other two along the boundary induces some implicit bias, or the PDF truncation may simply affect r_{ep} more or less than the other variables for this particular problem. It is difficult to ascribe error performance to one or the other.

This subsection has used constrained estimation to greatly reduce the simulation failure rate in near-singular engagements. When the true engagement weights are nearly singular, the number of failed simulations is reduced significantly using constrained estimation. On the other hand, when the weights are sampled from a near-Gaussian distribution, only about 1% of simulations with unconstrained estimation fail. The utility of the constrained estimation approach is related to how likely the engagement is to be singular; if the opponent weights are known with high confidence to be far from the stability manifold, then there is little to be gained by using the constrained estimation approach. The next subsection considers a case where the evader strategy is subject to uncertainty of form, and the ability of an IMM pursuer to differentiate minimax and FNE evaders is examined.

7.3.3 Identification of OSO and FNE strategies

In Chapter 6, IMM estimation is introduced to simultaneously address parametric uncertainty and uncertainty of form. In the examples of that chapter, IMM estimation does not clearly distinguish minimax and adaptive FNE opponent strategies. This is because the nominal simulation weights are zero-sum, and it can be shown that, at the ZS weight values, the solutions of the generalized RDEs have the same values and first derivatives with respect to the opponent weights. The differences in the two cases are only present in the higher-order derivatives. However, for NZS engagements, the minimax and FNE strategies are distinct and can be differentiated by the IMM estimator, as this subsection demonstrates.

This subsection begins with a discussion of weight selection for engagements. Next, individual simulations with the NZS weights are conducted for both minimax and adaptive FNE evader strategies. In these simulations, the pursuer weights are at the nominal values, and the initial conditions and evader weights are given in Table 7.8. Later in this subsection, Monte Carlo simulations are

$y(0)$	$\dot{y}(0)$	R_{ep}	$Q_e(1,1)$	$Q_e(2,2)$	$S_{f,e}(1,1)$	$S_{f,e}(2,2)$
-0.1	-0.2	-0.444	-0.0921	-0.0683	-8.77	-0.363

Table 7.8: Initial conditions and evader weights for individual simulations showing IMM detection of minimax and adaptive FNE evader strategies

conducted to demonstrate the IMM modal probability accuracy in both scenarios.

A series of planar engagements are conducted to evaluate the ability of an IMM estimator to correctly identify minimax and adaptive FNE opponent strategies. Fourth-order Taylor series approximations to the generalized RDE solutions are used by the adaptive pursuer. Two scenarios are considered: a NZS engagement, and a ZS engagement[6]. In all simulations, the pursuer has the LQDG weights outlined in Section 7.2.1. In the NZS engagement, the evader weights are of the following form:

$$[Q_e] = \text{diag}(0.0, -0.2) \qquad [R_{ep}] = [-0.4] \qquad [S_{f,e}] = \text{diag}(-9.0, -0.4) \tag{7.26}$$

All five evader weight elements are estimated by the pursuer. In the ZS engagement, the evader weights are simply derived from the pursuer weights, but the pursuer still treats all five parameters as unknowns to be estimated. The pursuer's initial covariance on the elements of $[R_{ep}]$, $[Q_e]$, $[S_{f,e}]$ is $\text{diag}(5\times10^{-4}, 1\times10^{-1}, 4\times10^{-2}, 2.7\times10^{-1}, 1\times10^{-3})$. For both adaptive OSO and FNE strategies, the pursuer assumes uncorrelated white noise on the two state and five parameter process models. A continuous-time process noise variance of 10^{-10} is placed on $Q_e(2,2)$ and $S_{f,e}(2,2)$, and a variance of 10^{-4} is placed on all other channels. In FNE engagements, the evader knows only three pursuer weights are nonzero and constrains her estimation accordingly. For the evader's estimation, the initial error variances on r_{pe}, q_p, and $s_{f,p}$ are $2\times10^{-4}, 1, 4.5\times10^{-2}$, and the associated process noise variances are $10^{-4}, 10^{-2}, 10^{-2}$.

Figure 7.17 shows the simulation states, controls, costs-to-go, and IMM performance values for an engagement between a minimax evader and an IMM pursuer. The plotted IMM states are

[6] In Monte Carlo simulations, weights are sampled, and "NZS" and "ZS" refer to the *mean* values of the weights.

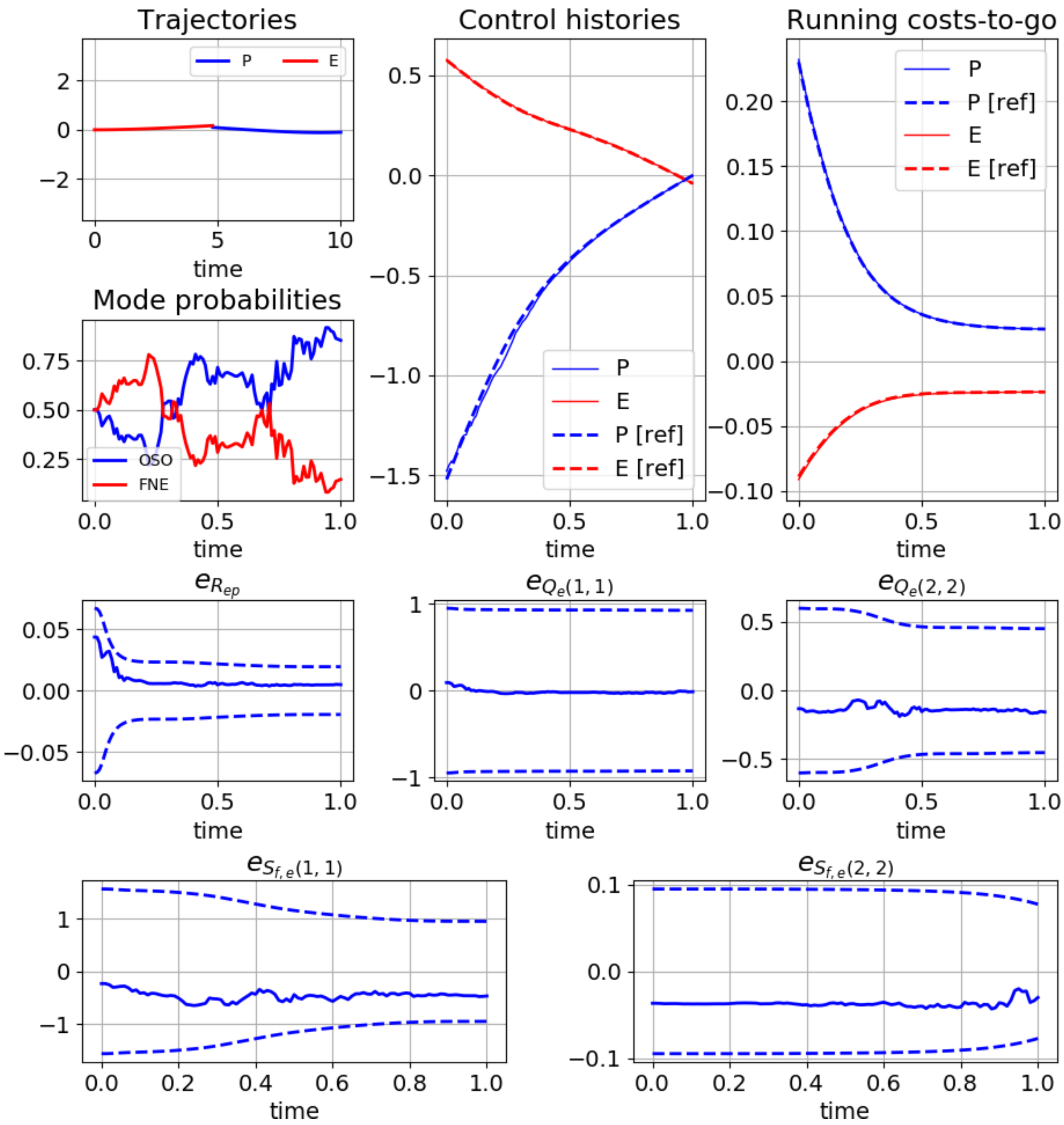

Figure 7.17: Single simulation summary for NZS engagement between minimax evader and IMM pursuer

Evader mode	Correct	Incorrect	Total
Minimax	2496	4	2500
FNE	2015	485	2500

Table 7.9: Summary of detection performance in 2500 Monte Carlo simulations with NZS nominal weights

the modal probabilities and estimation errors in the weights associated with the OSO filter. After a transient period of about 0.3 time units, the IMM pursuer correctly identifies the OSO estimation mode as more probable than the FNE mode. For the control and cost-to-go histories, the perfect information OSO solutions are computed and used to evaluate the OSO feedback control and cost-to-go along the true state history. These values are labeled as "ref" in the figure legends. The actual IMM histories show good agreement with the reference values. The estimation errors do not exceed the 3σ bounds, but it is clear that the parameters $Q_e(1,1)$, $Q_e(2,2)$, and $S_{f,e}(2,2)$ are not very observable, as the covariance bounds change little over the simulation[7].

Fig. 7.18 shows the same simulation summary metrics for an engagement with an adaptive FNE evader. The control and cost-to-go histories are qualitatively similar; however, the modal probabilities clearly show the FNE mode is correctly identified as more probable for most of the simulation. The estimation error histories are similar to the OSO engagement of Fig. 7.17. The clearest difference is in the terminal values of $Q_e(1,1)$ and $S_{f,e}(1,1)$. Where in Fig. 7.17 these values are trending toward zero error, they manifest small trends toward nonzero errors in Fig. 7.18, which is consistent with the bias seen in two-sided adaptive FNE simulations throughout this dissertation. Despite the minimal qualitative difference in the two cases, there is a clear difference in the modal probability plots of Figs. 7.17 and 7.18, demonstrating that the IMM is able to detect differences in the two cases. To further study the ability of the IMM to do this, Monte Carlo simulations are conducted.

In Monte Carlo simulations, initial states and evader weights are sampled according to Latin

[7] Similar performance is observed in simulations conducted with the same initial conditions and reduced model process noise.

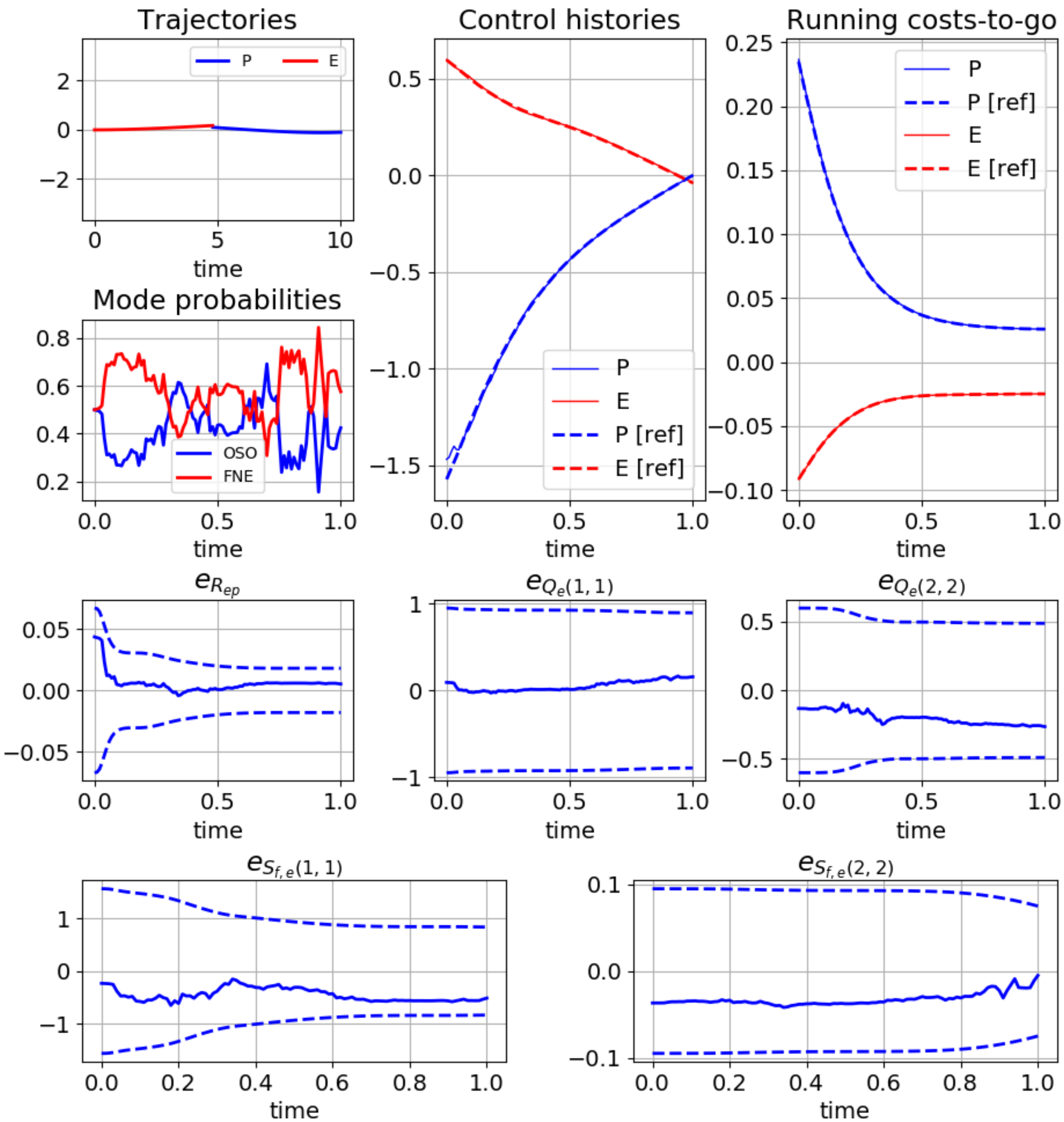

Figure 7.18: Single simulation summary for NZS engagement between adaptive FNE evader and IMM pursuer

Evader mode	Correct	Incorrect	Total
Minimax	1445	1055	2500
FNE	1031	1469	2500

Table 7.10: Summary of detection performance in 2500 Monte Carlo simulations with ZS nominal weights

Hypercube sampling[8]. The pursuer uses an IMM with only OSO and FNE modes. The objective is to evaluate how consistently the IMM correctly identifies the opponent's strategy. The IMM does not generate a binary classification of the opponent's mode, and instead estimates a probability associated with each mode as a function of time. To evaluate the IMM's performance, whichever mode is considered most probable over a majority of the simulation times is taken to the be mode "identified" by the IMM. The NZS engagement is considered first, using sampled initial states and evader weights. Table 7.9 summarizes simulation performance with NZS nominal weights according to this metric. When the evader uses a minimax strategy, it is identified as such in all simulations except four out of 2500. The detection accuracy is not as good when the evader uses the FNE strategy; about 20% of engagements are identified incorrectly. While the IMM demonstrates a bias towards selecting the minimax opponent as most probable, it is clear that the IMM-based identification correctly distinguishes between OSO and FNE strategies in a majority of cases.

Next, a nominally ZS engagement with sampled weights and initial conditions is considered. Table 7.10 displays the error rate in these simulations. In this case, it is apparent that the pursuer is unable to meaningfully differentiate between the minimax and adaptive FNE evader strategies. Regardless of the evader strategy, the IMM pursuer believes the modal evader strategy is minimax in a about 60% of simulations. Any difference between the classification in minimax and FNE simulations is small enough to be statistically insignificant. As expected, in the ZS engagement, the pursuer sees effectively no difference between minimax and FNE evader strategies.

This subsection has considered a simplified engagement in which the evader uses either minimax or adaptive FNE strategies and the pursuer uses an IMM to respond to parametric uncertainty

[8] Latin Hypercube sampling is used because the NZS engagement is more stable with respect to the weights than the nominal engagement used elsewhere in this chapter.

and uncertainty of form. The results indicate that the IMM correctly differentiates between OSO and FNE modes, which was not shown in the NZS engagement of Chapter 6. For ZS engagements, there is no statistically significant difference in the classification performance, regardless of the evader strategy. The next subsection again considers the IMM pursuer, but now in engagements with a deceptive switching evader.

7.3.4 Performance against a deceptive opponent

This subsection considers the performance of a pursuer using IMM estimation against an evader who employs a deceptive strategy involving a switch from a "false" strategy to a true one. First, the goals for each player are reviewed and relevant metrics identified. Second, individual simulation results are presented. Third, Monte Carlo evaluation of performance is conducted.

The goal of these results is to investigate the performance of each agent from an appropriate perspective. The pursuer has the primary objective of responding to each evader strategy as it is displayed in a fashion that approaches perfect information. A secondary objective is to correctly identify each evader mode using the IMM estimator. The evader has the primary objective of improving her performance by employing deception-based strategies, and a secondary objective of evaluating how well the simplified design model of the adaptive pursuer matches the IMM pursuer. Results are presented for both individual and Monte Carlo simulations. The relevant metrics for each are now discussed.

In individual simulations, the pursuer's performance is evaluated qualitatively by comparing the estimation accuracy before and after the evader's switch. Additionally, the mode probability histories are plotted to compare against the true evader modes, which are unknown to the pursuer. The evader's performance is evaluated by comparison of the true closed-loop states and controls of each agent to the reference models used to synthesize switching-based evader strategies. That is, **the objective of studying the evader performance is to validate whether the assumed simplified pursuer model is reflective of the true underlying dynamics**.

Monte Carlo results are summarized primarily by using tabulated metrics of performance. Pursuers are evaluated by their error metrics in estimating opponent parameters. The estimation error metrics are computed separately for the times before and after the switch based on the error between the true parameters and the IMM subfilter that estimates their values. The MSE, error covariance, and absolute error quartiles are used as error metrics. The loss function ℓ is used to evaluate the cost-to-go performance relative to a reference value. The reference value is computed assuming that the pursuer has no a priori knowledge of the switch, but recognizes it instantaneously and knows the true post-switch weights exactly. Because the reference value is defined as such, it is possible for the actual pursuer to achieve a lower cost than the reference. Consequently the quality of the pursuer response is evaluated in terms of having a small magnitude of ℓ. The evader performance is evaluating using ℓ, as well, but in this case the reference cost-to-go is computed using the design model of the pursuer response. The evader loss metric ℓ_e is a metric of how well the design model matches the true pursuer response. The loss metrics are tabulated in terms of their mean, covariance, and quartiles.

In addition to tabulated performance metrics, distribution plots of the mode probability estimates as functions of time are also used to summarize Monte Carlo results. Recall that Chapter 3 introduced an extension of box-whisker plots for distributions that vary with respect to an independent variable. The same plots are used in these results. The goal of plotting the mode probability distributions is to understand which, if any, of the "incorrect" modes are consistently identified as likely by the pursuer. This completes the description of the results and metrics used. Individual simulation results are now discussed.

In the first engagement, the evader uses the gain level switching strategy designed in Section 7.2.5.1. Figures 7.19–7.20 summarize the evader and pursuer performance metrics, respectively. Figures 7.19 shows the predicted and actual state and control histories. The predicted performance under the simplified model used for strategy design is labeled as "switch," and the predicted perfect information outcome using a fixed gain are labeled "fixed." The position histories match the prediction quite well. Recall that the evader's goal in using this strategy is to increase the terminal range, which is achieved as desired. There is a more noticeable mismatch in the $\dot{y}$ and control histories. The convergence of the true IMM pursuer's control to the simplified model's control

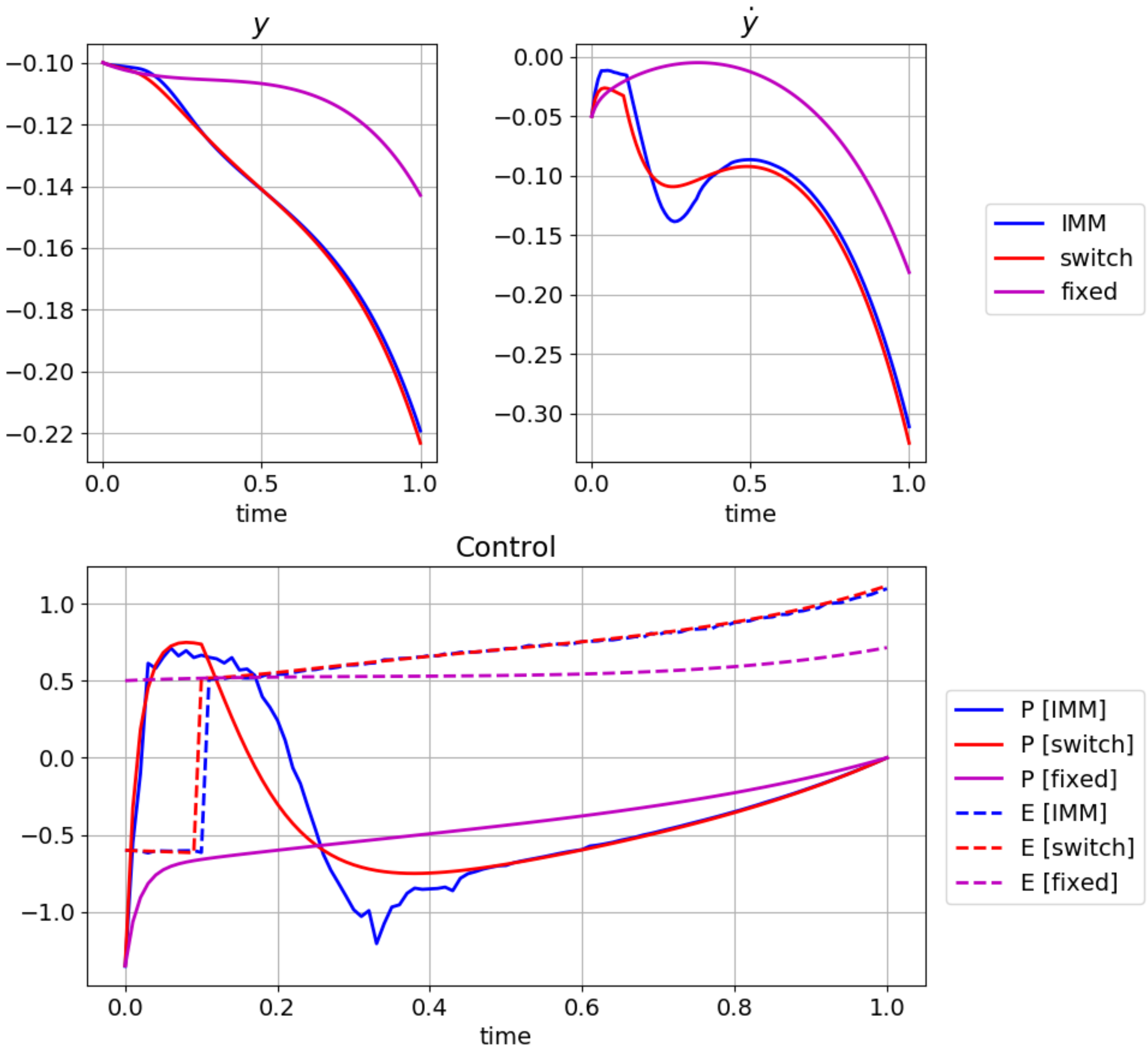

Figure 7.19: Simulation summary results for a gain level-switching evader

has more initial lag and some overshoot, suggesting a second-order convergence model is more appropriate than a first-order one. However, the true closed-loop system and the simplified model appear similar enough to be appropriate for preliminary design.

Continuing with the study of the gain level switching evader engagement, Fig. 7.20 shows the pursuer's estimation history of the evader gain and the modal probabilities. The pursuer's estimate converges to the pre-switch gain within about 5 measurements, but takes almost three times as

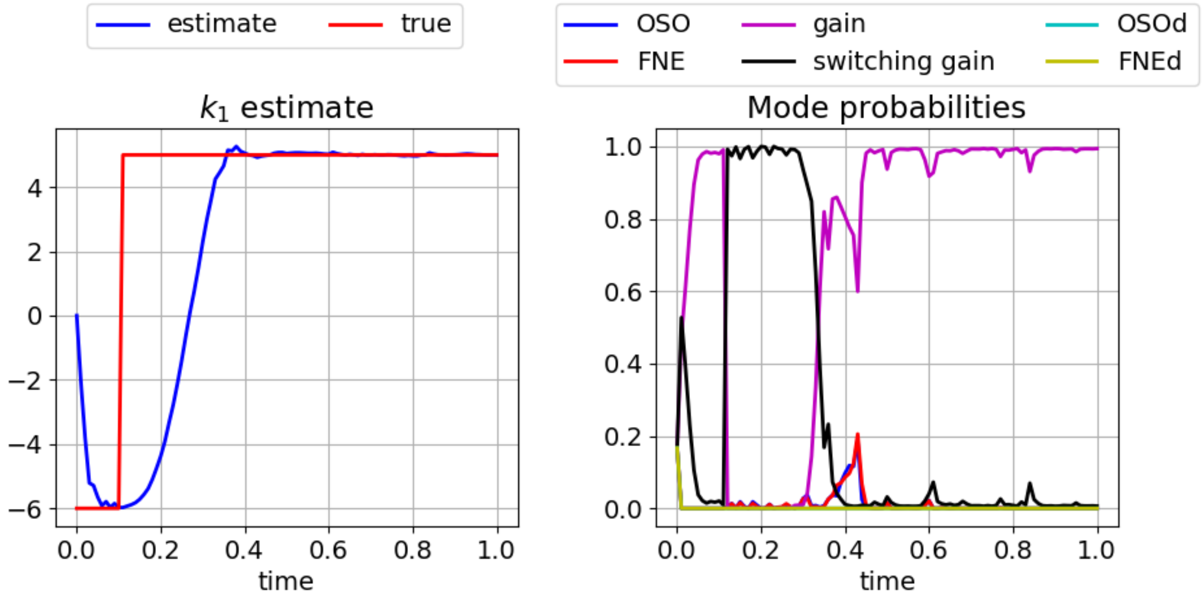

Figure 7.20: Pursuer estimation performance against a gain level-switching evader

long to converge to the post-switch gain. This is an effect of the use of multiple models mixed in the IMM to track the evader gain, and causes the error history to deviate from a first-order model. The modal probabilities show the IMM favoring the "switching gain" mode immediately after the switch and using those estimates. Those state estimates are mixed with the simple "gain" mode, and by the time the estimation error becomes low, the IMM identifies the constant gain mode as more probable. This figure shows the pursuer's estimator works essentially as intended to response to a step change in evader gain. The "rise time" of the post-switch estimate is slower than might be preferred.

Next, a simulation is conducted in which the evader uses the gain-to-LQDG strategy designed in Section 7.2.5.2. Fig. 7.21 shows summary simulation states, controls, and costs-to-go histories. The "design" values of the states and controls under the simplified model are shown for the "switch" and "fixed" cases, as in Fig. 7.19. Consider first the plot of the control histories. The pursuer's true control response initially lags the modeled response, which is denoted as "switch (p) [P]". After the initial lag, the IMM pursuer rises to the correct response more quickly than expected in the evader model. At $t = 0.1$, the evader executes a step change in strategy. After ap-

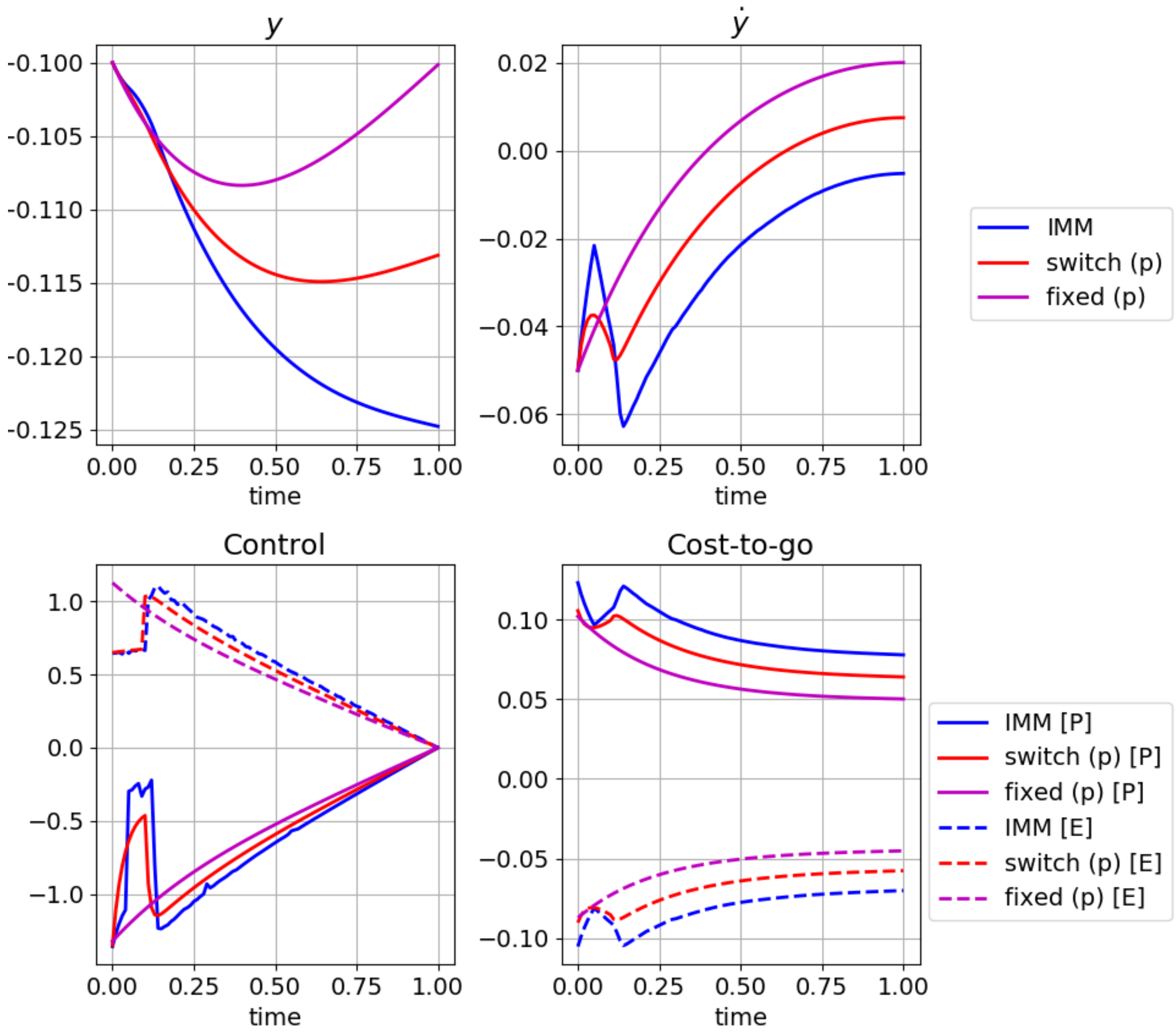

Figure 7.21: Predicted and actual performance for gain-to-LQDG evader engagement with IMM pursuer

proximately two time steps, the pursuer control transitions to the correct LQDG counter-response. Compared to the design model, the transition is delayed by two time steps, but occurs essentially instantaneously after the delay. The effect of this small mismatch between the pursuer control and the modeled pursuer control is seen clearly in the $\dot{y}$ state histories. The curvature of the simulated history (denoted “IMM”) and the predicted history under the switching strategy are approximately the same. However, the initial lag in the pursuer response leads to a larger change in $\dot{y}$ than predicted. When the pursuer corrects his response to match the evader's control, the $\dot{y}$ state is driven to larger amplitudes. Then, the evader switches to his minimax strategy, effectively increasing his feedback gain. This causes a brief transient before the pursuer identifies the new strategy, and the relative $\dot{y}$ magnitude grows again during the transient. After the pursuer switches to the OSO strategy, both the true and predicted $\dot{y}$ state histories have qualitatively similar derivatives. Because of the transient mismatches in $\dot{y}$ states, the overall position histories differ significantly between the simulation with IMM pursuer and the prediction from the simplified model.

The discussion of Fig. 7.21 is continued to note that the actual performance of the evader exceeds the predicted value. Under the design model, the evader's relative loss, compared to his outcome with no switch, is about 3.5%. Under the IMM simulation, the loss is close to 21%, an order of magnitude difference. While it is good for the simulated evader that his expectation was exceeded, the large error in predicted cost is troubling. The fact that the predicted state, control, and cost-to-go histories all had large errors suggests the proposed design model is inadequate to capture the behavior of the IMM pursuer's estimates.

Now, the gain-to-LQDG engagement is considered from the pursuer's perspective. Fig. 7.22 shows the pursuer's estimation states during the simulation. The histories show that the pursuer essentially did everything "correctly," within the limitations of her strategy. The initial evader strategy is correctly identified as being a constant-gain strategy and the estimated gain quickly converges to the true value. Once the evader's switch occurs, the OSO mode is quickly identified as the active mode. The LQDG weight estimates drift some during the constant-gain transient. However, the amount of drift does not destabilize the filter, which converges to near-true values with small biases by the end of the simulation. **The pursuer's performance is limited by her strategy and not by her estimation performance**. The pursuer's strategy is purely reactive to the evader, and does not attempt to anticipate the possibility of a deceptive opponent. The outcome of this simulation suggests a natural avenue for future study. Having discussed individual simulations in detail, Monte Carlo results are now presented.

Monte Carlo simulations are conducted to evaluate the overall play of agents in the presence of

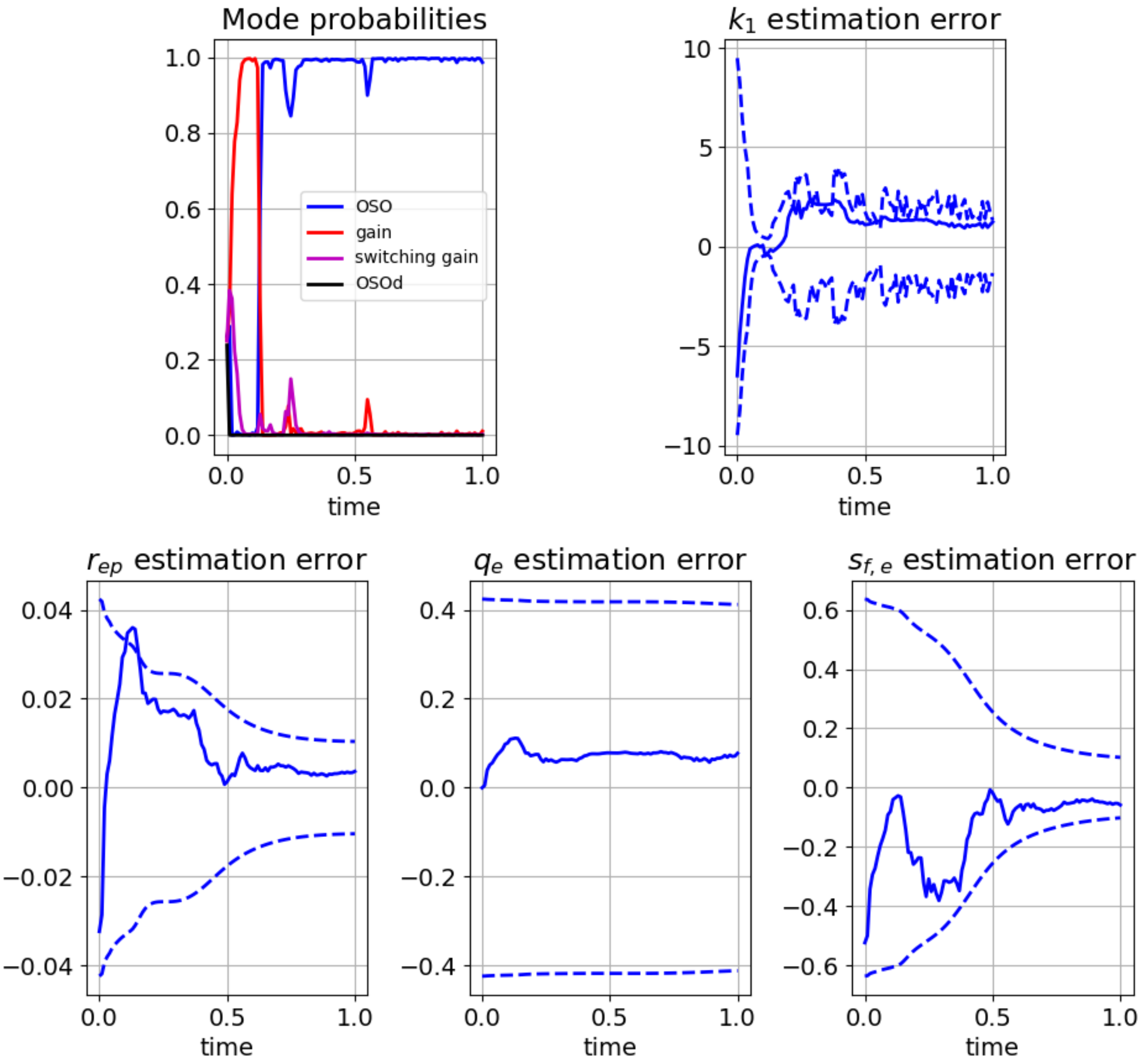

Figure 7.22: Estimation performance of IMM pursuer in engagement with constant-gain-to-LQDG evader

switching-based evader strategies. In these scenarios, each agent uses fixed weights, and uniformly distributed initial states are chosen according to the usual magnitudes using Latin Hypercube sampling. The number of simulations is reduced to 1000, since the weights are deterministic.

Table 7.11 summarizes the Monte Carlo estimation error metrics for an IMM pursuer playing against the gain level switching evader. k_1 and k_2 denote the pre-switch and post-switch gains, respectively. Errors are evaluated for times before and after the switch only. Results indicate the

Metric	k_1	k_2
MSE	9.99	21.6
Error Covariance	2.29	3.96
Median Abs Error	1.19	0.114
Lower Quartile	0.26	0.0287
Upper Quartile	4.33	4.38

Table 7.11: Summary of estimation error metrics for IMM pursuer playing against gain level switching evader

Metric	ℓ_p	ℓ_e
Median	0.608	-0.317
Lower Quartile	0.0801	-0.472
Upper Quartile	0.804	-0.0973

Table 7.12: Summary of loss function metrics for IMM pursuer playing against gain level switching evader

upper quartile for estimation of the pre-switch and post-switch gains are almost identical. However, the lower quartile and median absolute errors are an order of magnitude lower for the post-switch gain. Consider also the much larger MSE in k_2. These facts together suggest that most of the estimation errors in k_2 are lower than for k_1, but there are a noticeable number of outliers and high-error estimate histories for the post-switch gain. None of the simulations failed due to singular control commands. It seems likely that in some rare cases, the gain never correctly converged after the switch. This could be due to any number of factors, such as limited observability stemming from the relative state histories.

The analysis of the gain level switching engagement is continued in Table 7.12, which summarizes the relative loss metrics for pursuer and evader. The results indicate that the typical pursuer performance is noticeably better than predicted under a “perfect adaptation” model. Half of the pursuer loss values are in the range of 8%-80%, indicating improvement over the reference. Paradoxically, this relatively large improvement may be a symptom of the IMM pursuer deviating from expectations. The pursuer can achieve a large cost improvement if she fails to estimate the evader

Metric	k_1	r_{ep}	q_e	$s_{f,e}$
MSE	11.7	0.00151	0.655	0.0924
Error Covariance	2.47	0.0336	0.808	0.302
Median Abs Error	1.33	0.0274	0.439	0.168
Lower Quartile	0.276	0.0157	0.211	0.0755
Upper Quartile	4.68	0.0441	0.768	0.312

Table 7.13: Summary of estimation error metrics for IMM pursuer playing against gain-to-LQDG switching evader

Metric	ℓ_p	ℓ_e
Median	0.0082	0.129
Lower Quartile	-0.634	-0.0227
Upper Quartile	0.183	1.24

Table 7.14: Summary of loss function metrics for IMM pursuer playing against gain-to-LQDG switching evader

gains at all and uses the default minimax strategy, which completely avoids any of the negative effects of the evader's switch. The evader relative loss metrics indicate worse performance than the baseline, although the difference is smaller than for the pursuer. The interquartile range is $[-47\%, -9.7\%]$. The Monte Carlo results for the pursuer and evader show the opposite trend of the individual simulation results, in which the evader outperformed her reference and the pursuer underperformed. This indicates that the performance of the agents under this strategy is both very sensitive to random phenomena and to the initial state, as well.

Next, consider the Monte Carlo simulations conducted in which the evader uses the gain-to-LQDG strategy. Table 7.13 shows the pursuer's estimation performance on the pre-switch and post-switch states. The pre-switch error metrics are all comparable to those of Table 7.11, which is to be expected. The estimates of r_{ep} and $s_{f,e}$ show good consistency between the MSE and median error metrics, suggesting minimal outliers in the estimation. This indicates that the LQDG filters handle the switch robustly and with relatively low errors.

To finish the consideration of the gain-to-LQDG engagements, consider Table 7.14, which

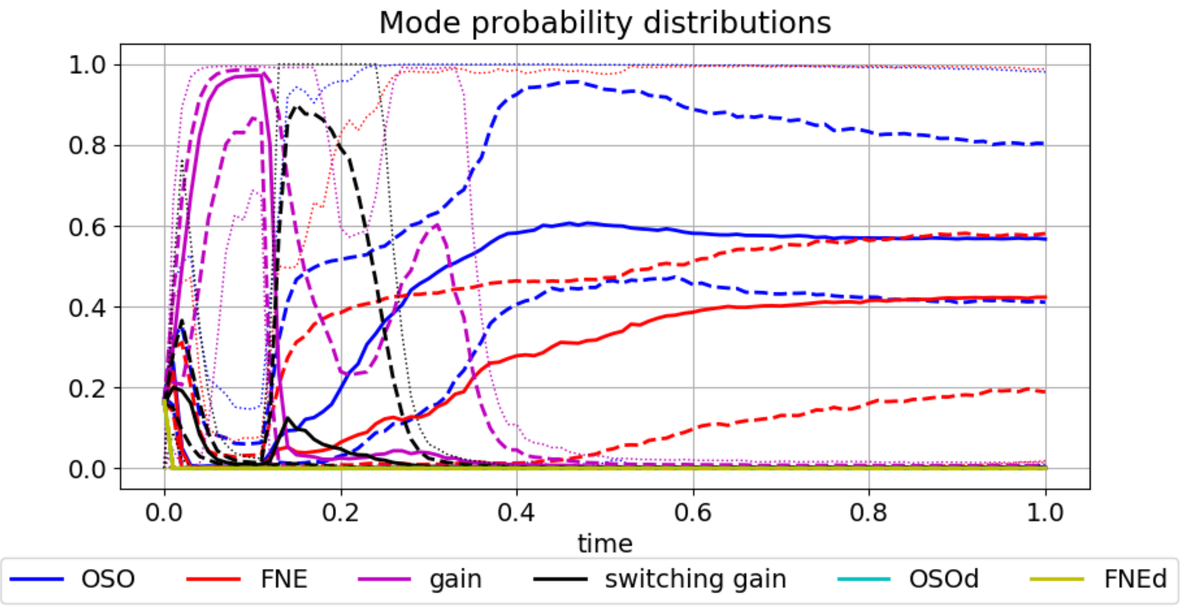

Figure 7.23: Box-whisker time series plot of mode probability distributions across gain-to-LQDG Monte Carlo simulations

displays the relative loss metrics for each player during Monte Carlo. The pursuer's median loss is approximately zero, indicating the pursuer has about a fifty percent chance of doing better or worse than his reference performance. The quartiles are highly asymmetric about the median, indicating that when the pursuer does worse than the reference, he tends to do quite badly. This is consistent with the hypothesis that there might be a relatively small fraction of simulations in which estimates fail to converge post-switch. If that happens, the pursuer might never switch to the LQDG strategies and performance could suffer greatly. The evader's loss metrics are consistent with that hypothesis, as well. The evader has a median of 13%, a lower quartile value of 2.3%, and an upper quartile value of 120%. That is, the evader usually improves his objective function value over the reference, and in many simulations the value improves a great deal. According to this Monte Carlo result, the gain-to-LQDG strategy is quite beneficial for the evader, and the pursuer has a relatively high change of performing poorly against this strategy.

For further consideration of the gain-to-LQDG strategy Monte Carlo results, consider the mode probabilities plotted in Fig. 7.23. This is a box-whisker time-series plot, of the same type introduced in Chapter 3. The solid lines indicate the median probability of each mode across simulations, the dashed lines indicate the quartiles, and the dotted lines indicate the whiskers. What this plot shows is that the gain mode is correctly identified before the switch in most simulations. After the switch, there is a transient of about 0.2 time units during which the OSO, FNE, gain, and switching gain modes frequently achieve relative high probabilities. This result indicates the pursuer's transition to the OSO or FNE modes may be delayed compared to the simulation of Fig. 7.22. This delay could be the source of the poor performance that is observed in Table 7.14.

The current subsection has analyzed the performance of both agents when the evader uses the nonadaptive switching strategies of Section 7.2.5. Individual simulations have shown the pursuer's estimation working as expected. The evader's design model of the pursuer's adaptation predicts the IMM pursuer performance well in the gain level switching engagement, but has large errors in the gain-to-LQDG engagement. In Monte Carlo engagements with gain level switching evader, the IMM pursuer typically outperforms his reference cost and evader performs worse than the reference. When the evader uses the gain-to-LQDG strategy, this trend reverses, and the evader typically outperforms his reference. The two main findings of this subsection are as follow. One, that the IMM estimation typically handles opponent switches smoothly, but transitions between modes slowly in a significant minority of cases. Two, the proposed design model for synthesizing evader strategies does not adequately characterize the full range of IMM pursuer responses, especially in the gain-to-LQDG engagements.

The last subsection of results considers the general performance of an IMM pursuer against evaders using fixed strategies.

7.3.5 General IMM pursuer performance

The last subsection of results considers the general performance of an IMM pursuer against a nonswitching evader. Only individual simulations are considered, with no Monte Carlo results. In performing the analysis, a new possible failure mode associated with the IMM is identified and discussed. In the following results, planar engagements are considered and adaptive agents treat the opponent as having three nonzero LQDG weights. The initial conditions and pursuer weights

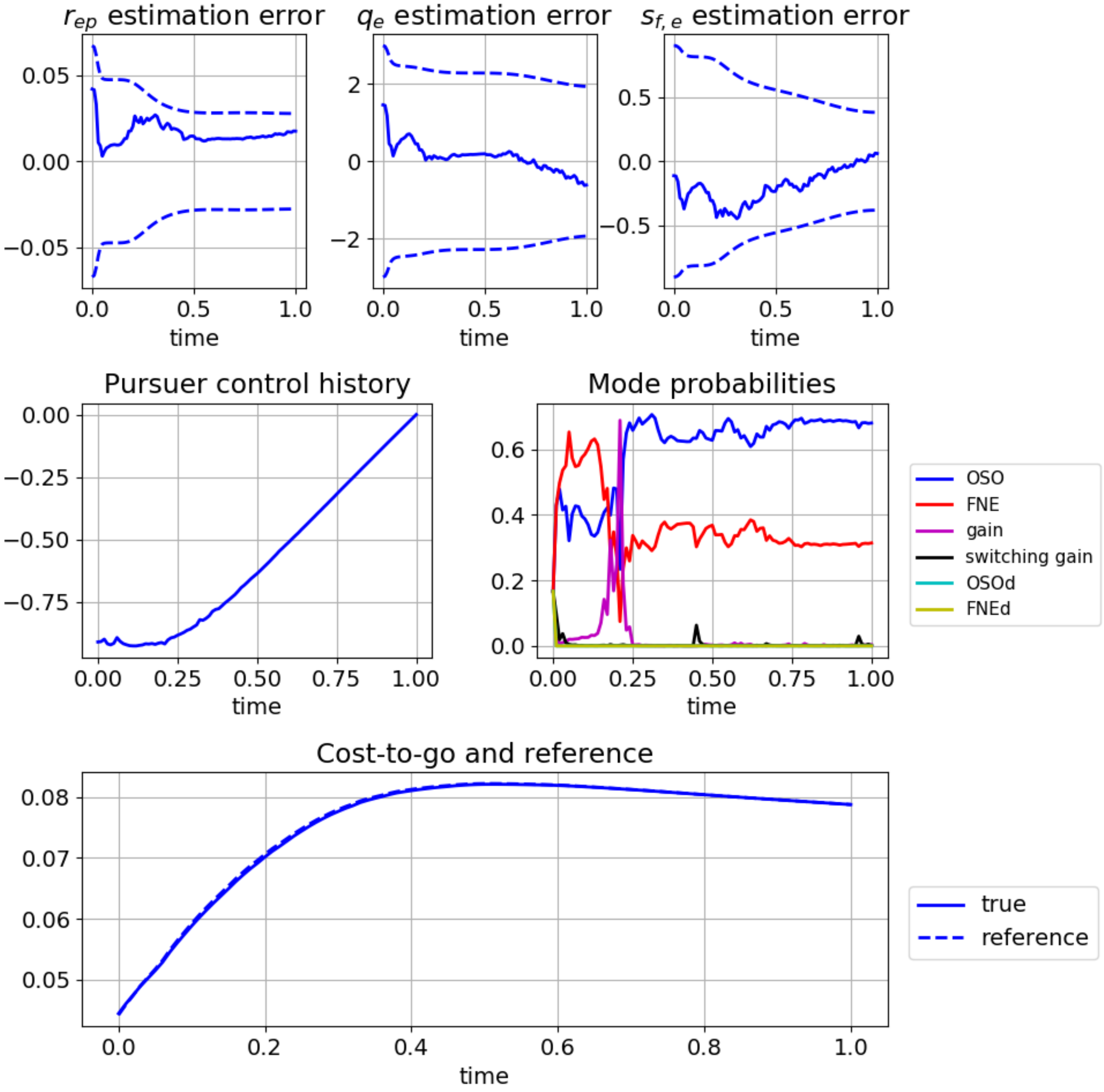

Figure 7.24: Estimation and performance results for a pursuer using IMM against an FNE evader

are the nominal values of Section 7.2.1. The initial covariance on weights $r_{ji}, q_j, s_{f,j}$ is $\text{diag}(5 \times 10^{-4}, 1, 9 \times 10^{-2})$. The pursuer's process noise model on estimated states $y, \dot{y}, r_{ep}, q_e, s_{f,e}, k_e$ is $\text{diag}(10^{-4}, 10^{-4}, 10^{-3}, 10^{-2}, 1, 10)$.

Fig. 7.24 summarizes the IMM pursuer performance against an adaptive FNE evader. The results are indistinguishable from the non-IMM results. The cost-to-go matches the reference value closely. It should be noted that the most probable IMM mode overall is OSO, not FNE, which has a negligible effect on overall performance. The results do not merit much further discussion.

In numerical simulations, the IMM pursuer is often able to achieve similar estimation and cost-to-go performance as is achieved when the opponent's strategy is known. However, there is a new failure mode that can reduce the objective function performance of the adaptive player in this engagement. The pursuer control solution is significantly different between the minimax/FNE and constant evader models. Under the minimax or FNE strategies, the evader gain inevitably approaches a zero terminal value, due to the specification of zero terminal weight on the relative velocity. This is necessarily different from the constant gain model, unless the gain is near zero. Consequently, if the opponent gain is large, the LQDG strategies and LQDG-gain strategy have large control differences. When the IMM-C mixes control values, temporary increases in the modal probability of the LQDG-gain model can cause large changes in the IMM pursuer's control. In practice, this manifests as a step increase in the cost-to-go. An example is now presented to demonstrate.

Fig. 7.25 shows simulation results for an adaptive FNE engagement that demonstrate this new failure mode. The estimation error performance is normal and acceptable. The modal probabilities consistently show the OSO and FNE modes as most probable, which is also correct. However, there are two spikes in the likelihood of either the gain or "switching gain," modes. Each spike corresponds to a step increase in the cost-to-go history. The small increase in the modal probability of the gain modes leads to their associated control being mixed with the OSO/FNE controls, leading to a discontinuity in the commanded control. In this case, the incurred penalty is relatively small; the actual cost is about 2.5% larger than the reference cost.

This new type of failure represents an interesting research challenge. The sudden changes in mode probabilities are, ultimately, a feature of a likelihood-based MM filter working as intended. They become a problem because the control prescriptions are significantly different between the underlying modes. That is, the same quality that makes the IMM-adaptive player well-suited

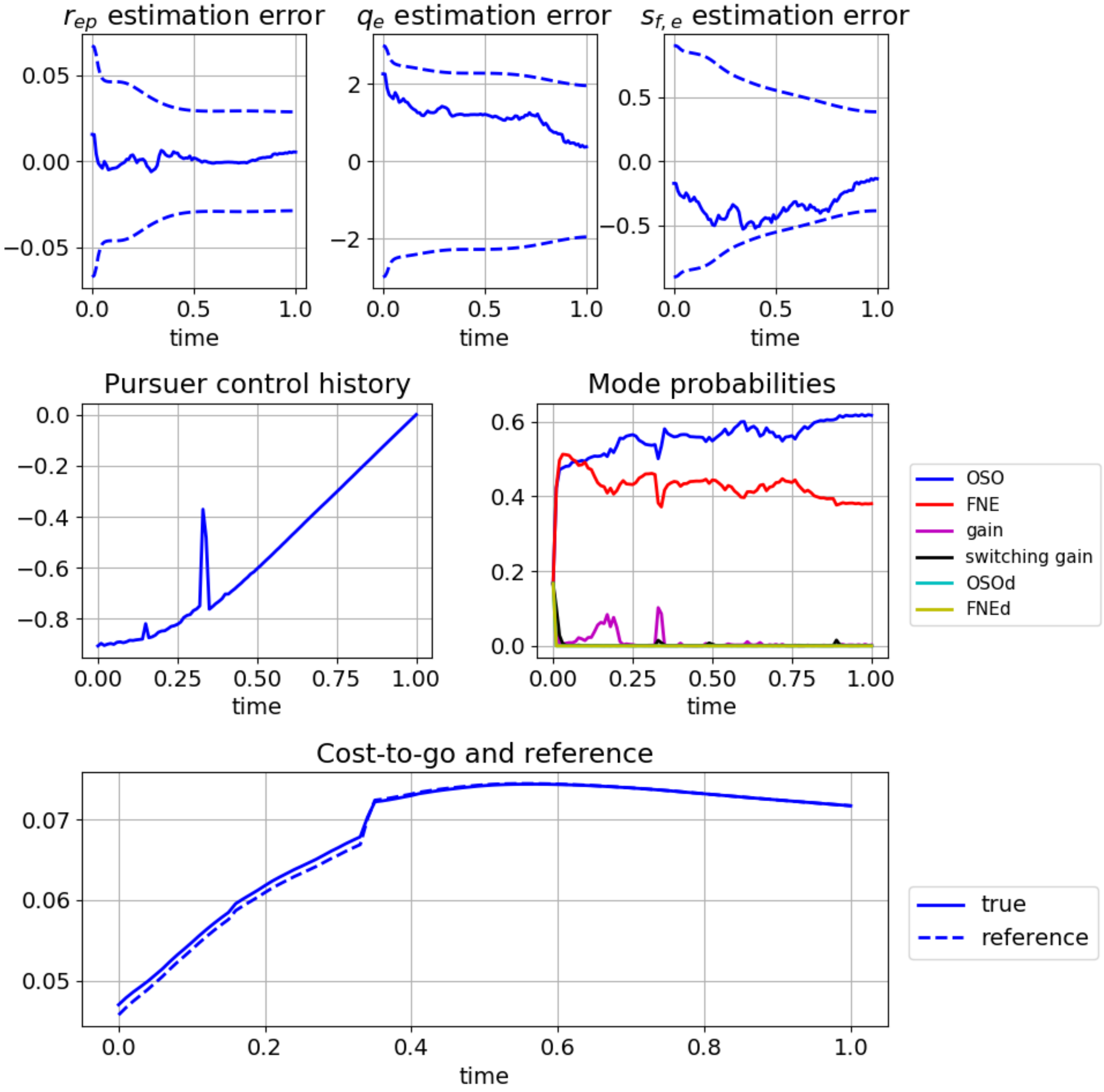

Figure 7.25: Estimation and performance results demonstrating the new failure mode associated with IMM adaptive agents

to abrupt changes in the opponent strategy is degrading its performance against a fixed evader strategy. It is an open research question how a player might retain his advantages against switching opponents while reducing the effects of this new failure mode. It may be useful to conduct an a priori analysis of possible opponent switching times and modulating the prior mode transition probabilities accordingly. For example, opponent switches in this scenario typically offer minimal benefit for times greater than $t = 0.25$. If the prior transition probabilities used in the IMM are scheduled such that they become near-zero after that time, then the occurrence of mode probability spikes might be reduced significantly. This is only a suggested approach, and is not explored in this dissertation.

This subsection has considered individual simulation results for an adaptive FNE evader playing against an IMM pursuer. In most simulations, the IMM pursuer appears to recover the performance of the pure adaptive FNE strategies used in Section 7.3.1. However, in some simulations, small "spikes" in the modal probabilities lead to large changes in the control, which degrade the overall performance. This is partly a result of using the IMM-C approach; the matrix game-based control proposed in Chapter 6 might alleviate this problem. Addressing this new behavior is identified as an area for future study.

This section presents numerical results evaluating the performance of various techniques introduced in Chapters 4–6. These results include engagements between simple adaptive agents subject only to parametric uncertainty, as well as engagements with agents subject to uncertainty in the form of the opponent's strategy. Various nonadaptive strategies are considered including switching-based strategies. Individual simulation results and Monte Carlo studies are conducted to quantify performance. From all of these results, three primary conclusions are identified. First, the estimation part of the adaptive gameplay generally works as intended. This applies to the simple adaptive gameplay under cost-based loop closure, constrained estimation, and IMM estimation. Second, the design models suggested for synthesizing switching strategies are not fully capturing the adaptation of IMM agent models. Third, the use of IMM estimation can lead to undesired behaviors that reduce the benefits of adaptation, and these behaviors should be mitigated.

This chapter has considered a missile interception engagement as an example of a realistic problem that can be conceptualized as a differential game. Linearized relative equations of motion are derived to study LQDGs between various adaptive and nonadaptive agents. Numerical simulations including both single simulation runs and Monte Carlo simulations are conducted to describe and quantify the performance of the different advances proposed in previous chapters of the dissertation. The overall findings, while suggesting some avenues for future research, demonstrate that the new adaptive gameplay techniques generally perform as expected. The numerical results display substantially more robust performance than the previous numerical examples of Chapter 3. The following chapter presents the conclusions of the dissertation.

8. CONCLUSIONS

This chapter summarizes the content of the dissertation, outlines the main conclusions, and suggests avenues for future research. This dissertation studies agents in differential games subject to incomplete information about their opponent's objective function. Linear quadratic differential games (LQDGs) are chosen as the focus of study. These games offer sufficient realism to model some real-world problems of interest and, under modest assumptions, lead to linear feedback solutions that satisfy properties of robustness, optimality, or Nash equilibrium. Chapter 1 introduces LQDGs and the primary solutions that are used in later chapters to study uncertainty. These are the minimax, one-sided optimal (OSO), and Feedback Nash Equilibrium (FNE) strategies. The generalized Riccati Differential Equations (RDEs) that govern each case are introduced. Chapter 2 begins by introducing the one-sided adaptive gameplay concept originally presented in Ref. [6]. This approach is extended to games in which both agents play adaptively. An adaptive FNE strategy assuming a perfect information opponent is proposed as an analog of the minimax strategy for such two-sided games. These adaptive strategies are applied to an orbital pursuit-evasion game in Chapter 3. The Hill-Clohessy-Wiltshire equations of relative motion are used to obtain linear relative dynamics between agents in the vicinity of a circular orbit. The numerical results of this application demonstrate some of the merits of the proposed adaptive gameplay, but also highlight several significant challenges that are addressed by subsequent chapters.

Chapter 4 mainly addresses incremental improvements to the existing adaptive gameplay structure. A statistically rigorous procedure is outlined for determining when an adaptive agent can safely compute closed-loop controls using her estimates of the opponent weight(s). Automatic differentiation is introduced to compute truncated Taylor series approximations to the generalized RDEs, simplifying the computation of neighboring solutions for estimation purposes at a cost of greater memory demands. In Chapter 5, constrained estimation is employed to prevent the computation of singular solutions to the generalized RDEs. A novel procedure for numerically approximating the limits on the opponent parameters corresponding to nonsingular engagements is demonstrated. Chebyshev polynomial basis functions are used to approximate the stability interface. A probability density function truncation procedure is implemented for constrained estimation. A two-filter structure for detecting opponent parameters for which the RDEs become singular is proposed and evaluated in Monte Carlo tests. In Chapter 6, the differential game uncertainty is expanded to encompass uncertainty of the form of the opponent's strategy, in addition to the parametric uncertainty studied earlier. Multiple-model (MM) estimation is identified as an alternative to pure learning-based adaptive methods for this type of problem. MM estimates exploit domain knowledge to estimate the state of a target with hybrid dynamics, and are suitable for time-sensitive engagements of relatively short duration. This chapter also identifies opponent deception as a vulnerability of purely adaptive game strategies. Interacting Multiple Model (IMM) estimation is suggested as an estimator that is appropriate if the opponent may change her strategy during the engagement. Some initial numerical results are presented for a problem with scalar relative dynamics.

A detailed missile interception problem is used in Chapter 7 to study different features of the innovations outlined in Chapters 4–6. A large number of scenarios with different initial conditions, opponent weights, and adaptive game strategies are tested in both individual simulations and Monte Carlo analyses. The results demonstrate that the adaptive gameplay techniques, which include cost-based loop closure, constrained estimation, and IMM estimation, generally perform as expected. The control performance in general is substantially more robust than the previous numerical examples of Chapter 3.

The primary conclusions of the dissertation can be summarized as follows:

1. The suggested extension of adaptive gameplay to "two-sided" engagements, and the proposed estimation techniques of later chapters, generally achieve the objective of initially responding cautiously while uncertainty is high, and approaching the limits of perfect information as measurements converge.

2. To first order, the suggested adaptive OSO and FNE strategies are identical in simulations that are nominally zero-sum. If the expected values of an adaptive opponent's weights are

the zero-sum values, then the adaptive OSO structure can be used even when the opponent is also playing adaptively. This can achieve a small computational savings, although if the opponent weights are far from the zero-sum values, a bias is incurred.

3. Observability of opponent weights and uniqueness of estimation solutions is a significant challenge in adaptive gameplay with parametric uncertainty. These related issues arise in many of the examples studied in Chapters 3 and 7.

4. The use of AD to obtain the truncated Taylor series solution approximations is promising for differential games subject to parametric uncertainty. In particular, the marriage of this technique with Kalman Filter-based estimation is appealing. The Taylor series degree can be chosen to achieve small errors within the radius of convergence of the nonlinear Kalman Filter.

5. Purely adaptive agents have new vulnerabilities, such as the class of deceptive, switching-based strategies identified in Chapter 6. These strategies have a natural connection to Stackelberg (leader–follower) games that might be exploited in the future.

These conclusions and the results in this dissertation suggest a number of research challenges and other directions future study may take. Some relevant activities are enumerated below:

1. Application of the estimation-based adaptive gameplay to non-LQDG engagements should be considered. In particular, it is of interest to study numerical solutions of the Hamilton-Jacobi-Isaacs partial differential equation subject to parametric uncertainty. These solutions yield feedback controls and are widely used in the differential games literature.

2. For adaptive players, game-theoric analysis of switching opponents should be conducted to derive appropriate mitigating strategies to play against deceptive players. The matrix game-based control introduced in Chapter 6 may be useful here both to add robustness and to identify switching times that are beneficial to the opponent.

3. Singular engagement detection, introduced in Chapter 5, can be extended to dynamically handle the tradeoff between performance and robustness in control problems. This can be done by changing the set of weights a player uses to ensure the estimated opponent weights remain in the stable region of the parameter space, but lie relatively near the stability manifold. This could have the effect of ensuring that a robust control design adjusts itself to become more and less robust as disturbance magnitudes become larger and smaller, respectively. The connection between LQDGs and H_∞ control is quite significant here.

4. The truncated Taylor series can be exploited additionally in uncertain differential games where nonlinear Kalman Filters are used. The series approximation can quantify the observability and uniqueness of various parameters, and can be used to choose an appropriate initial covariance in the absence of prior knowledge of the opponent weights. This concept is suggested by Figure 7.2, but not pursued in the dissertation.

5. The concept of deceptive strategies to oppose adaptive agents is intriguing and merits further consideration. To enable this research, the use of adaptive agents that use nonlinear observers in place of nonlinear Kalman Filters is suggested. Under this model, the adaptive agent's estimation dynamics take on a more tractable analytical form. In particular, the model reference adaptive control literature seems an appealing place from which to begin further study of this topic.

REFERENCES

[1] Y. Bar-Shalom, X.-R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation: Theory, Algorithms, and Software*. New York: Wiley-Interscience, 2001.

[2] T. Başar and G. Olsder, *Dynamic Noncooperative Game Theory, 2nd Edition*. Classics in Applied Mathematics, Society for Industrial and Applied Mathematics, 1998.

[3] J. L. Crassidis and J. L. Junkins, *Optimal estimation of dynamic systems*. CRC press, 2011.

[4] E. A. Wan and R. V. D. Merwe, "The unscented Kalman filter for nonlinear estimation," in *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No.00EX373)*, pp. 153–158.

[5] R. Isaacs, *Differential Games: A Mathematical Theory with Applications to Warfare and Pursuit, Control and Optimization*. John Wiley & Sons, Inc, 1965.

[6] K. D. Aures-Cavalieri, *Incomplete information pursuit-evasion games with applications to spacecraft rendezvous and missile defense*. Dissertation, 2014.

[7] A. Starr, "Nonzero-sum differential games: concepts and models," report, Harvard University, 1969.

[8] J. Von Neumann and O. Morgenstern, *Theory of Games and Economic Behaviour*. Princeton University Press, 1944.

[9] P. Bernhard, "Linear-quadratic, two-person, zero-sum differential games: Necessary and sufficient conditions," *Journal of Optimization Theory and Applications*, vol. 27, no. 1, pp. 51–69, 1979.

[10] F. L. Lewis, D. L. Vrabie, and V. L. Syrmos, *Optimal Control*. John Wiley & Sons, Inc, 3rd ed., 2012.

[11] T. Başar and P. Bernhard, *H_∞-Optimal Control and Related Minimax Design Problems: A Dynamic Game Approach*. Modern Birkhauser Classics, Boston: Birkhauser, 2008.

[12] N. Satak, *Behavior learning in differential games and reorientation maneuvers*. Dissertation, 2013.

[13] Y. Ho, A. Bryson, and S. Baron, "Differential games and optimal pursuit-evasion strategies," *IEEE Transactions on Automatic Control*, vol. 10, no. 4, pp. 385–389, 1965.

[14] D. L. Lukes and D. L. Russell, "A global theory for linear-quadratic differential games," *Journal of Mathematical Analysis and Applications*, vol. 33, no. 1, pp. 96–123, 1971.

[15] D. Lukes, "Equilibrium feedback control in linear games with quadratic costs," *SIAM Journal on Control*, vol. 9, no. 2, pp. 234–252, 1971.

[16] M. C. Delfour, "Linear quadratic differential games: Saddle point and riccati differential equation," *SIAM Journal on Control and Optimization*, vol. 46, no. 2, pp. 750–774, 2007.

[17] J. A. Bryson and A. Carrier, *A comparison of control synthesis using differential games (H-infinity) and LQR*. Guidance, Navigation, and Control and Co-located Conferences, American Institute of Aeronautics and Astronautics, 1989.

[18] H. Abou-Kandil, G. Freiling, and G. Jank, "Necessary conditions for constant solutions of coupled Riccati equations in Nash games," *Systems & Control Letters*, vol. 21, no. 4, pp. 295–306, 1993.

[19] X. Wang and J. B. Cruz, *Nash Equilibrium for 2nd-order Two-Player Non-Zero Sum LQ Games with Executable Decentralized Control Strategies*, pp. 1960–1965. Proceedings of the 45th IEEE Conference on Decision and Control, 2006.

[20] A. E. Bryson and Y.-C. Ho, *Applied optimal control: optimization, estimation, and control*. Hemisphere Publishing Corporation, rev. printing. ed., 1975.

[21] J. Engwerda, *Linear Quadratic Differential Games: An Overview*, pp. 1–34. Boston: Birkhäuser Boston, 2009.

[22] H. Abou-Kandil, G. Freiling, V. Ionescu, and G. Jank, *Matrix Riccati Equations in Control and Systems Theory*. Systems & Control: Foundations & Applications, Birkhäuser, Basel, 2003.

[23] G. Anderson and V. Samant, *Differential game guidance laws for intercept missiles*. Guidance, Navigation, and Control and Co-located Conferences, American Institute of Aeronautics and Astronautics, 1981.

[24] G. M. Anderson, "Comparison of optimal control and differential game intercept missile guidance laws," *Journal of Guidance, Control, and Dynamics*, vol. 4, no. 2, pp. 109–115, 1981.

[25] O. Belapolsky and J. Ben-Asher, *On Two Formulations of Linear Quadratic Optimal Guidance*. Guidance, Navigation, and Control and Co-located Conferences, American Institute of Aeronautics and Astronautics, 2007.

[26] V. Shaferman and T. Shima, *Linear Quadratic Differential Games Guidance Law for Imposing a Terminal Intercept Angle*. Guidance, Navigation, and Control and Co-located Conferences, American Institute of Aeronautics and Astronautics, 2008.

[27] R. Fonod and T. Shima, "Multiple model adaptive evasion against a homing missile," *Journal of Guidance, Control, and Dynamics*, vol. 39, no. 7, pp. 1578–1592, 2016.

[28] V. Turetsky and T. Shima, "Target evasion from a missile performing multiple switches in guidance law," *Journal of Guidance, Control, and Dynamics*, vol. 39, no. 10, pp. 2364–2373, 2016.

[29] J. Shinar, V. Y. Glizer, and V. Turetsky, "A pursuit-evasion game with hybrid pursuer dynamics," *European Journal of Control*, vol. 15, no. 6, pp. 665–684, 2009.

[30] F. Amato, M. Mattei, and A. Pironti, "Guaranteeing cost strategies for linear quadratic differential games under uncertain dynamics," *Automatica*, vol. 38, no. 3, pp. 507–515, 2002.

[31] N. Satak and J. E. Hurtado, *A Framework for Behavior Learning in Differential Games*. AIAA SciTech Forum, National Harbor, Maryland, 2014.

[32] N. Satak and J. E. Hurtado, *Cost-Strategy Recognition Method for Behavior Learning: Two Players Learning Simultaneously*. AIAA SciTech Forum, National Harbor, Maryland, 2014.

[33] F. Köpf, J. Inga, S. Rothfuß, M. Flad, and S. Hohmann, "Inverse reinforcement learning for identification in linear-quadratic dynamic games," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 14902–14908, 2017.

[34] K. G. Vamvoudakis and F. L. Lewis, "Multi-player non-zero-sum games: Online adaptive learning solution of coupled Hamilton-Jacobi equations," *Automatica*, vol. 47, no. 8, pp. 1556–1569, 2011.

[35] P. Menon and A.J.Calise, "Interception, evasion, rendezvous and velocity-to-be-gained guidance for spacecraft," pp. 334–341, AIAA Guidance, Navigation, and Control and Co-located Conferences, 1987.

[36] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.

[37] K. Alfriend, S. R. Vadali, P. Gurfil, J. How, and L. Breger, *Spacecraft formation flying: dynamics, control and navigation*. Butterworth-Heinemann, 2009.

[38] R. H. Woodward, "Pursuit-evasion games between two spacecraft in near-Earth orbit," Master's thesis, Air Force Institute of Technology, 1972.

[39] G. Anderson, "A closed-form solution for the barrier in pursuit-evasion problem between two low thrust orbital spacecraft and its application," 13th Aerospace Sciences Meeting, January 1975.

[40] P. Menon, A.J.Calise, and S. Leung, "Guidance laws for spacecraft pursuit-evasion and rendezvous," pp. 688–697, Guidance, Navigation and Control Conference, Guidance, Navigation, and Control and Co-located Conferences, 1988.

[41] M. Vasile and F. B. Zazzera, "Direct multiphase optimisation of multiobjective trajectory design problems," AAS/AIAA Space Flight Mechanics Meeting, January 2002.

[42] M. Pontani and B. A. Conway, “Numerical solution of the three-dimensional orbital pursuit-evasion game,” *Journal of Guidance, Control, and Dynamics*, vol. 32, pp. 474–487, March–April 2017.

[43] E. P. Blasch, K. Plam, and D. Shen, “Orbital satellite pursuit-evasion game-theoretical control,” pp. 1007–1012, IEEE International Conference on Information Sciences, Signal Processing and Applications, 2012.

[44] W. T. Hafer, *Sensitivity methods applied to orbital pursuit-evasion*. Dissertation, 2014.

[45] A. Jagat and A. J. Sinclair, “Optimization of spacecraft pursuit-evasion game trajectories in the Euler-Hill reference frame,” AIAA Space Forum, August 2014.

[46] A. Jagat and A. J. Sinclair, “Nonlinear control for spacecraft pursuit-evasion game using state-dependent Riccati equation method,” *IEEE Transactions on Aerospace and Electronic Systems*, 2017. pre-print.

[47] J. D. Hunter, “Matplotlib: A 2d graphics environment,” *Computing In Science & Engineering*, vol. 9, no. 3, pp. 90–95, 2007.

[48] M. D. McKay, R. J. Beckman, and W. J. Conover, “A comparison of three methods for selecting values of input variables in the analysis of output from a computer code,” *Technometrics*, vol. 21, no. 2, pp. 239–245, 1979.

[49] M. Baudin, M. Christopoulou, Y. Collette, and J.-M. Martinez, “pyDOE: The experimental design package for python,” 2009–. [Online; accessed January 28, 2019].

[50] A. G. Baydin, B. A. Pearlmutter, A. A. Radul, and J. M. Siskind, “Automatic differentiation in machine learning: a survey,” *Journal of Machine Learning Research*, vol. 18, no. 153, p. 43, 2018.

[51] A. Bani Younes, J. Turner, M. Majji, and J. Junkins, *High-order state feedback gain sensitivity calculations using computational differentiation*. 2012.

[52] R. D. Neidinger, “Introduction to automatic differentiation and MATLAB object-oriented programming,” *SIAM Review*, vol. 52, no. 3, pp. 545–563, 2010.

[53] S. F. Walter and L. Lehmann, “Algorithmic differentiation in Python with AlgoPy,” *Journal of Computational Science*, vol. 4, no. 5, pp. 334 – 344, 2013.

[54] A. Griewank, J. Utke, and A. Walther, “Evaluating higher derivative tensors by forward propagation of univariate taylor series,” *Math. Comput.*, vol. 69, no. 231, pp. 1117–1130, 2000.

[55] J. H. Mathews and K. D. Fink, *Runge-Kutta Methods*. Upper Saddle River, N.J.: Pearson, 2004.

[56] Mathworks, “ODE event location.” `https://www.mathworks.com/help/matlab/math/ode-event-location.html`. Accessed February 2, 2019.

[57] SciPy, “scipy.integrate.solve_ivp.” `https://docs.scipy.org/doc/scipy/reference/generated/scipy.integrate.solve_ivp.html`. Accessed February 2, 2019.

[58] A. H. A. A. Bani Younes, *Orthogonal polynomial approximation in higher dimensions: Applications in astrodynamics*. Dissertation, 2013.

[59] P. Vachhani, S. Narasimhan, and R. Rengaswamy, “Robust and reliable estimation via unscented recursive nonlinear dynamic data reconciliation,” *Journal of Process Control*, vol. 16, no. 10, pp. 1075–1086, 2006.

[60] D. Simon and D. L. Simon, “Constrained Kalman filtering via density function truncation for turbofan engine health estimation,” *International Journal of Systems Science*, vol. 41, no. 2, pp. 159–171, 2010.

[61] O. Straka, J. Duník, and M. Šimandl, “Truncation nonlinear filters for state estimation with nonlinear inequality constraints,” *Automatica*, vol. 48, no. 2, pp. 273–286, 2012.

[62] B. O. S. Teixeira, L. A. B. Torres, L. A. Aguirre, and D. S. Bernstein, *Unscented filtering for interval-constrained nonlinear systems*, pp. 5116–5121. 2008 47th IEEE Conference on Decision and Control, 2008.

[63] D. Simon, "Kalman filtering with state constraints: a survey of linear and nonlinear algorithms," *IET Control Theory & Applications*, vol. 4, no. 8, pp. 1303–1318, 2010.

[64] D. Simon, *Optimal state estimation: Kalman, H_∞, and nonlinear approaches*. Hoboken, New Jersey: John Wiley & Sons, Inc., 2006.

[65] S. Tully, G. Kantor, and H. Choset, *Inequality constrained Kalman filtering for the localization and registration of a surgical robot*, pp. 5147–5152. 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2011.

[66] E. Mazor, A. Averbuch, Y. Bar-Shalom, and J. Dayan, "Interacting multiple model methods in target tracking: a survey," *IEEE Transactions on aerospace and electronic systems*, vol. 34, no. 1, pp. 103–123, 1998.

[67] Y. Oshman, J. Shinar, and S. Weizman, *Using a multiple model adaptive estimator in a random evasion missile/aircraft encounter*. Guidance, Navigation, and Control and Co-located Conferences, American Institute of Aeronautics and Astronautics, 1999.

[68] H. Wen-hua, C. Xing-lin, and W. Zhi-peng, *Maneuver target interception based on multiple model adaptive estimator*, pp. 185–190. 2010 3rd International Symposium on Systems and Control in Aeronautics and Astronautics.

[69] H. A. P. Blom, *An efficient filter for abruptly changing systems*, pp. 656–658. The 23rd IEEE Conference on Decision and Control.

[70] M. Yeddanapudi, Y. Bar-Shalom, and K. Pattipati, "IMM estimation for multitarget-multisensor air traffic surveillance," *Proceedings of the IEEE*, vol. 85, no. 1, pp. 80–96, 1997.

[71] S. Lee and I. Hwang, *Interacting Multiple Model Estimation for Spacecraft Maneuver Detection and Characterization*. AIAA SciTech Forum, American Institute of Aeronautics and Astronautics, 2015.

[72] D. Lee, C. Liu, Y.-W. Liao, and J. K. Hedrick, “Parallel interacting multiple model-based human motion prediction for motion planning of companion robots,” *IEEE Transactions on Automation Science and Engineering*, vol. 14, no. 1, pp. 52–61, 2017.

[73] K. Granström, P. Willett, and Y. Bar-Shalom, “Systematic approach to IMM mixing for unequal dimension states,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 51, no. 4, pp. 2975–2986, 2015.

[74] C. Rago, R. Prasanth, R. K. Mehra, and R. Fortenbaugh, *Failure detection and identification and fault tolerant control using the IMM-KF with applications to the Eagle-Eye UAV*, vol. 4 of *Proceedings of the 37th IEEE Conference on Decision and Control (Cat. No.98CH36171)*, pp. 4208–4213. 1998.

[75] L. Mihaylova and E. Semerdjiev, “An interacting multiple model algorithm for stochastic systems control,” *Information & Security, An International Journal*, vol. 2, 1999.

[76] P. S. Maybeck and R. D. Stevens, *Reconfigurable flight control via multiple model adaptive control methods*, vol. 6 of *29th IEEE Conference on Decision and Control*, pp. 3351–3356. 1990.

[77] J. Shinar, M. Guelman, and A. Green, “An optimal guidance law for a planar pursuit-evasion game of kind,” *Computers & Mathematics with Applications*, vol. 18, no. 1, pp. 35–44, 1989.

[78] E. Garcia, D. W. Casbeer, and M. Pachter, “Active target defence differential game: fast defender case,” *IET Control Theory & Applications*, vol. 11, no. 17, pp. 2985–2993, 2017.

[79] E. Garcia, D. W. Casbeer, and M. Pachter, “Active target defense using first order missile models,” *Automatica*, vol. 78, no. Supplement C, pp. 139–143, 2017.

[80] A. Ratnoo and T. Shima, “Guidance strategies against defended aerial targets,” *Journal of Guidance, Control, and Dynamics*, vol. 35, July–August 2012.

[81] N. F. Palumbo, R. A. Blauwkamp, and J. M. Lloyd, “Modern homing missile guidance theory and techniques,” *Johns Hopkins APL Tech. Dig*, vol. 29, no. 1, pp. 42–59, 2010.

[82] N. F. Palumbo, R. A. Blauwkamp, and J. M. Lloyd, *Basic Principles of Homing Guidance*, vol. 29. 11100 Johns Hopkins Road, Laurel, MD 20723-6099: The Johns Hopkins University Applied Physics Laboratory, 2010.

[83] J. Stigter, D. Joubert, and J. Molenaar, “Observability of complex systems: Finding the gap,” *Scientific reports*, vol. 7, no. 1, p. 16566, 2017.

[84] J. D. Stigter and J. Molenaar, “A fast algorithm to assess local structural identifiability,” *Automatica*, vol. 58, pp. 118–124, 2015.

[85] J. D. Stigter, M. B. Beck, and J. Molenaar, “Assessing local structural identifiability for environmental models,” *Environmental Modelling & Software*, vol. 93, pp. 398–408, 2017.

[86] A. F. Genovese, “The interacting multiple model algorithm for accurate state estimation of maneuvering targets,” *Johns Hopkins Applied Technical Digest*, vol. 22, no. 4, pp. 614–623, 2001.

[87] T. Yuan, Y. Bar-Shalom, P. Willett, E. Mozeson, S. Pollak, and D. Hardiman, “A multiple IMM estimation approach with unbiased mixing for thrusting projectiles,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 4, pp. 3250–3267, 2012.

[88] B. Li, F. Pang, C. Liang, X. Chen, and Y. Liu, *Improved interactive multiple model filter for maneuvering target tracking*, pp. 7312–7316. Proceedings of the 33rd Chinese Control Conference, 2014.

APPENDIX A

OBSERVABILITY ANALYSIS

The problem of estimating opponent weights in a LQDG is significantly nonlinear. Observability analysis should be conducted to confirm that any proposed estimation state is theoretically observable. This appendix evaluates the observability of the estimation problems of Chapters 3 and 7.

The most common approach for evaluating nonlinear observability in the dynamics and controls literature is algebraic analysis using Lie derivatives. However, nonlinear observability analysis with Lie derivatives depends on an observability matrix of Lie derivatives having full rank. For systems of relatively small state-space, analysis can be conducted by hand, but if the system dimension is larger than three or four, manual analysis becomes extremely tedious. While robust computer symbolic algebra software is relatively common, the computational time or memory requirements required for symbolic nonlinear observability analysis are still unacceptably large in many systems of practical interest[83]. The Lie derivatives also require explicit Jacobians of the governing first-order ODE exist. This is incompatible with the use of the terminal values of the opponent Riccati matrix as estimated states. Hence, a numerical observability test is required.

A recent series of papers[84, 85, 83] offers a solution that is reasonably simple to evaluate and complements the computation of Taylor series in the approximate solution of the RDEs presented in Chapter 4. This method is suitable for numerically analyzing the observability of state spaces with hundreds of dimensions. In this approach, a small number [1] of random initial conditions and parameter values are chosen. The system states and first-order sensitivity derivatives of the states with respect to time-invariant parameters and initial states are then numerically evaluated as functions of time. A stacked time history of the sensitivity derivatives of the outputs is then formed. The rank of the output sensitivity matrix, determined by its singular value decomposition (SVD),

[1] Ten is suggested in [84].

characterizes whether a unique basis for the sensitivity of the outputs to the parameters and initial states exists. If the output sensitivity matrix is full rank, the system as parameterized is theoretically locally structurally observable. If the matrix is not of full rank, then the nonzero elements of the associated right singular vectors indicate which parameters are correlated. Then, the standard Lie derivative analysis can be computed only on the correlated parameters. By limiting the analysis to these parameters, the symbolic computation is simplified, and a basis for the nullspace of the nonlinear observability matrix can be computed in reduced time. The nullspace basis, in turn, enables a minimal parameterization of the system to be readily identified.

The remainder of this appendix is divided into three sections. The first summarizes the nonlinear observability analysis procedure. The second and third present results evaluating the observability of the orbital PE and missile interception examples considered in Chapters 3 and 7.

A.1 Analysis procedure

The method of Stigter and Molenaar [84] is used in this dissertation for observability analysis in the LQDG game. The method is thought to be of interest in dynamics systems and estimation for uncommon models with high dimensional state spaces. The rest of this section describes the approach in detail.

An ODE system $\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{p})$ is assumed to consist of n states $\boldsymbol{x}$ and n_p unknown but constant parameters $\boldsymbol{p}$. The method of Stigter and Molenaar treats both the system constants, such as mass properties, and initial conditions as parameters to be estimated. The system is assumed to have an output $\boldsymbol{y} \in \mathbb{R}^m$, which in this dissertation is assumed to be a linear function of the states. The first-order sensitivity derivative of $\boldsymbol{x}$ with respect to a parameter p_i is simply $\boldsymbol{x}_{,p_i}$, and in general has no closed-form solution. However, its governing form is simply a known ODE:

$$\frac{d}{dt}\left(\boldsymbol{x}_{,p_i}\right) = [\boldsymbol{f}_{,\boldsymbol{x}}]\boldsymbol{x}_{,p_i} + \boldsymbol{f}_{,p_i} \tag{A.1}$$

In Stigter and Molenaar, the time histories of the sensitivity derivatives are computed by numerical integration. Sensitivity derivatives with respect to system parameters satisfy $\boldsymbol{x}_{,p_i}(0) = 0$, and derivatives with respect to initial conditions satisfy $x_{j,p_i}(0) = \delta_{ij}$; otherwise they are propagated identically. However, in this dissertation, automatic differentiation is used to recover the first-order sensitivities. The procedure for one simulation is now described:

1. Take the given set of randomly chosen parameters and initial conditions and simulate the time history of $\boldsymbol{x}(t)$ and its $n \times n_p$ sensitivity derivatives matrix $[\boldsymbol{x}_{,\boldsymbol{p}}]$
2. Rescale the sensitivity derivative history to obtain a relative sensitivity matrix, which improves the accuracy of the rank calculation. Commonly, $x_{j,p_i}(t)$ is multiplied by either $\frac{p_i}{x_j(t)}$ or $\frac{p_i}{\bar{x}_j}$ if $x_j(t)$ is zero. $\bar{x}_j$ is some representative value of x_j.
3. Compute the full output sensitivity derivative history by taking $[\boldsymbol{y}_{k,\boldsymbol{p}}] = [H_k][\boldsymbol{x}_{k,\boldsymbol{p}}], k = 1, \ldots, K$
4. Extract a small set of sensitivity derivative values for the SVD computation. The authors of Ref. [85] suggest choosing N evenly spaced times across the simulation horizon, with N the smallest integer that satisfies $Nm > n_p$.
5. Form an $Nm \times n_p$ relative output sensitivity matrix (ROSM) by stacking the selected values of $[\boldsymbol{y}_{k,\boldsymbol{p}}]$ along their first dimension.
6. Compute the SVD of the ROSM. Record the singular values and the right singular vectors.

The procedure is repeated for several simulations with randomly chosen initial conditions to avoid cases where the system might be unobservable along a particular trajectory. The SVD outputs across all simulations are then assessed. This is typically done by a human user, although it could be automated. The analysis is straightforward, and it is most easily understood by reviewing sample results. A simple numerical example is now introduced, and is used to complete the discussion of the numerical observability analysis.

The numerical example considers the LQDG with scalar dynamics used throughout the dissertation. Here, the modeled opponent weights include a redundant term. That is, the estimating player i assumes his opponent uses the following cost function:

$$J_j = \frac{1}{2}s_{f,j}x(T_f)^2 + \frac{1}{2}\int_0^{T_f}\left(q_j x^2 + r_{jj}u_j^2 + r_{ji}u_i^2\right)\,\mathrm{d}t \tag{A.2}$$

The estimated state consists of $x, r_{ji}, q_j, s_{f,j}$, and r_{jj}. Player i assumes the following closed-loop dynamics:

$$\dot{x} = (a - r_{jj}^{-1}s_j(q_j, r_{jj}, r_{ji}, s_{f,j}))x + u_i \tag{A.3}$$

$s_j(t)$ is computed according to the scalar RDE and subject to the boundary condition $s_j(T_f) = s_{f,j}$:

$$\dot{s}_j = -2as_j - q_j + s_j^2(r_{jj}^{-1} + r_{ji}^{-1}) \tag{A.4}$$

To evaluate the observability, it is most convenient to propagate both x and s_j and their sensitivity derivatives in time. Hence the state vector is of size 2, and there are five effective parameters: $\boldsymbol{\theta} \equiv (r_{ji}, q_j, r_{jj}, x_0, s_{f,j})$. $s_f(t)$ is propagated first in reverse time, and then $x(t)$ is propagated in forward time. For simplicity, agent i is assumed to use zero control over the integration. Automatic differentiation is employed to automatically extract the first-order sensitivity derivatives. The parameters are sampled from a normal distribution with the following mean and covariance:

$$E[\boldsymbol{\theta}] = \begin{bmatrix} -.916 \\ -3 \\ 1 \\ 0 \\ -3 \end{bmatrix} \qquad \mathrm{cov}[\boldsymbol{\theta}] = \mathrm{diag}\begin{bmatrix} 5.76\times 10^{-4} & .444 & 1.11\times 10^{-3} & 44.4 & .444 \end{bmatrix} \tag{A.5}$$

Ten sets of the parameters are chosen randomly. For each set, the six-step procedure is followed to generate the singular values and associated singular vectors of the ROSM. Once the singular values and singular vectors are recorded, the so-called “observability signature” is generated. The signature consists of two types of plots. The observability signature for this problem is shown in

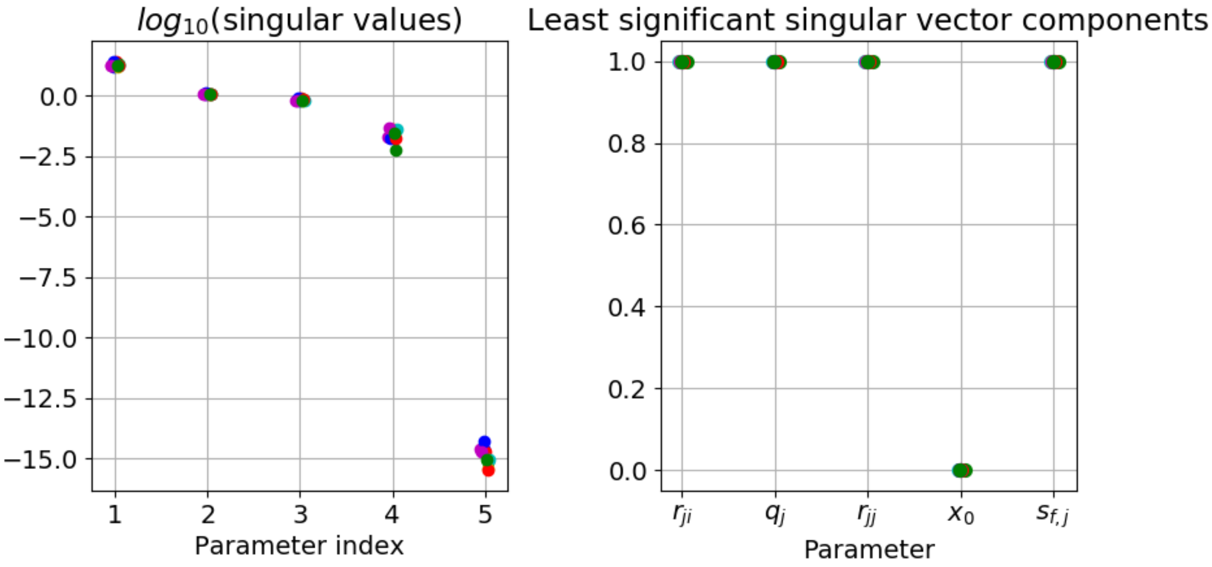

Figure A.1: Observability signature for ten simulations of the example problem with scalar dynamics

Fig. A.1. The left plot is simply a plot of the components of the ordered singular values for each simulation. The right plot is a plot of the right singular vector associated with the least significant singular value(s)[2]. Note that random Gaussian noise is added to the x coordinates in each plot to make the results from the ten simulations more distinct. From observation of the left plot, it should be clear that the ROSM has one singular value that is numerically zero. This indicates the system has one unobservable state. The associated right singular vector components are plotted in the right hand plot. This plot shows that the unobservable "mode" of the system is correlated with the parameters r_{ji}, q_j, r_{jj}, and $s_{f,j}$. This finding reflects what is already known: the LQDG game weights are unique only up to a scale factor.

For the scalar LQDG example problem, a minimal, observable system representation can be obtained trivially by normalizing the cost function by any one weight to eliminate its degree of freedom. If a minimal representation is not readily identified, it can be obtained procedurally as follows. An alternate solution exists based on the nonlinear observability matrix. The observability

[2] If more than one singular value is numerically zero, then the singular vector plot should be generated for each such singular value.

matrix is computed symbolically using Lie derivatives for the full set of parameters. Then, the nullspace of a reduced observability matrix is computed from the columns of the observability matrix corresponding only to the correlated parameters. Symbolic algebra is not avoided, but the computational complexity of the nullspace operation is greatly reduced. The nullspace, in general, forms a vector of values $c_i, i = 1, \ldots, N_c$ for the N_c correlated parameters. A nonunique minimal parameterization of the system satisfies the following PDE:

$$0 = \sum_{i=1}^{N_c} c_i \frac{\partial \phi}{\partial p_i} \tag{A.6}$$

The solutions to ϕ in Eq. A.6 are new parameters that collect terms to simplify the governing equations. However, the solution to the PDE is not unique, and any number of possible parameterizations can be used to reduce the system.

This completes the discussion of the Stigter method for numerical observability analysis. The remainder of this appendix applies the nonlinear observability analysis to the orbital PE game of Chapter 3 and the missile interception game of Chapter 7.

A.2 Orbital pursuit-evasion game

The orbital PE game is described in detail in Chapter 3. The problem is considered from the perspective of an adaptive pursuer. There are five unknown evader parameters in addition to the four relative states to estimate:

$$\boldsymbol{\theta} \equiv \begin{bmatrix} r_e & q_{1,e} & q_{2,e} & s_{1,e} & s_{2,e} \end{bmatrix}^T \tag{A.7}$$

The problem dynamics are defined by Eq. 3.10. To evaluate the sensitivity derivatives, automatic differentiation is used to first propagate the generalized RDE solutions backward from T_f to $t = 0.0$. Then the relative state is propagated forward from $t = 0$ to T_f. Euler integration is used for propagation with a time step of $\Delta t = 0.005$. The pursuer weights and mean evader weights are the same as in Table 3.3. The evader weights are sampled with the given mean and 10% of the covariance reported in Table 3.2. It is found that a larger covariance produces too many singular

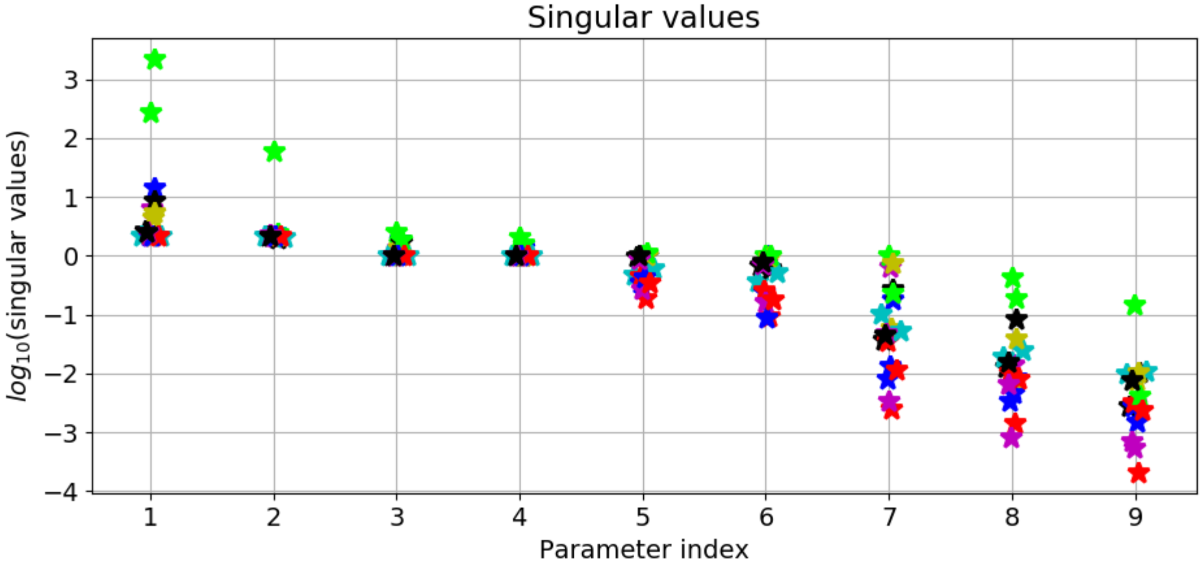

Figure A.2: Observability signature for the orbital pursuit-evasion game of Chapter 3

engagements. Initial conditions are drawn from independent uniform distributions with the positions bounded by $[-5, 5]$ and velocity components by $[-0.5, 0.5]$. Simple random sampling is used and eighteen nonsingular engagement weights are chosen.

The system is simulated and the relative output sensitivity matrix is computed for the chosen initial values. The resulting observability signature is plotted in Figure A.2. The signature clearly indicates that all the singular values exceed machine precision by several orders of magnitude and satisfy the requirements for local structural observability. This figure is generated for an adaptive OSO engagement. The observability signature is similar when adaptive FNE engagement is used. This demonstrates the orbital PE game satisfies the theoretical requirements for observability. The observability of the missile interception problem is now considered.

A.3 Missile interception problem

Chapter 7 considers a missile interception game. Both planar and nonplanar engagements are considered, but weights are assumed to be symmetric in the sense that the Y and Z components of weights are always equal. Thus, observability can be demonstrated by considering only the

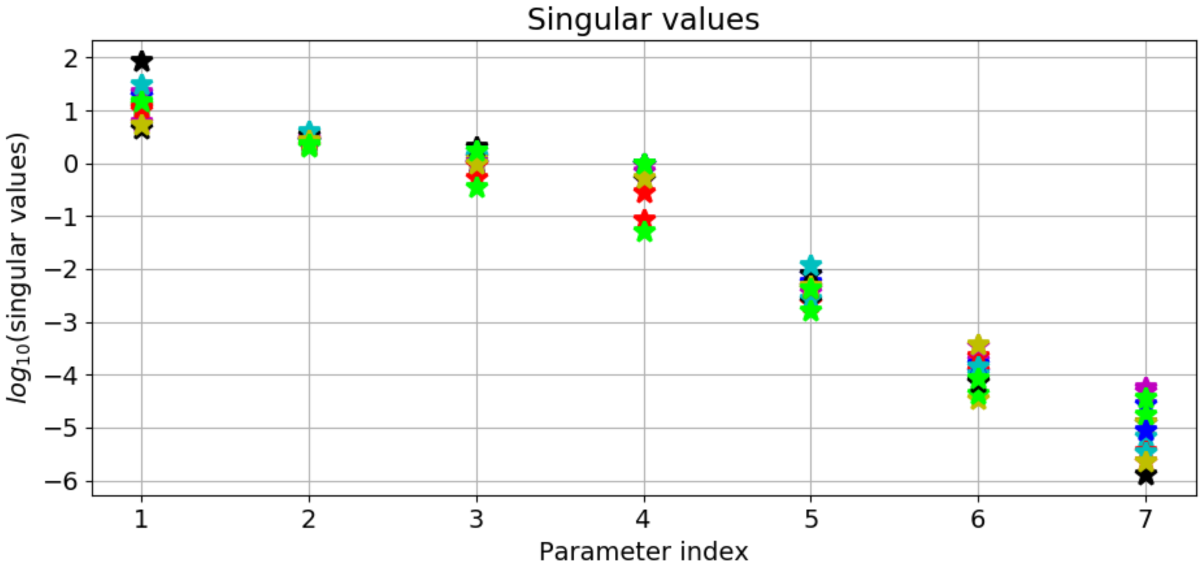

Figure A.3: Observability signature for the planar missile interception game of Chapter 7

planar engagement. The estimation problem of Section 7.3.3 is considered because this is the most general problem, in which the pursuer does not assume any of the evader's weights are zero. The adaptive pursuer estimates five evader parameters $\boldsymbol{\theta}$ in addition to the two relative states:

$$\boldsymbol{\theta} \equiv \begin{bmatrix} R_{ep}(1,1) & Q_e(1,1) & Q_e(2,2) & S_{f,e}(1,1) & S_{f,e}(2,2) \end{bmatrix}^T \tag{A.8}$$

The standard problem weights of Section 7.2.1 are used for the observability analysis. Uniform initial states, all bounded by ± 0.1, are assumed. The evader weights are sampled as unconstrained Gaussian random variables. The mean is given from the zero-sum weights, and the covariance from the pursuer's initial covariance of Section 7.3.3. Simple random samples are generated for fifteen simulations. As for the orbital PE game, Euler integration is used, and the sample time is reduced to 0.001.

The resulting observability signature for an adaptive FNE problem is shown in Fig. A.3. Results are qualitatively similar when an adaptive OSO problem is considered. All singular values are clearly much greater than machine precision, demonstrating the missile interception game is locally structurally identifiable.

This appendix has presented the algorithm for nonlinear observability analysis based on the SVD of a relative output sensitivity matrix. Using this method, it has been shown that particular values of the orbital PE game and missile interception game satisfy local structural identifiability. It is assumed, based on this analysis, that all variants of those problems considered in the dissertation satisfy the same identifiability property.

APPENDIX B

ESTIMATION ALGORITHMS

This appendix summarizes estimation algorithms used throughout the dissertation. For convenience, the governing equations have been collected here. The following sections present the algorithms associated with the following filters: nonlinear estimation with the Extended Kalman Filter and Unscented Kalman Filter; Constrained estimation via probability density function (PDF) truncation; and the Multiple-Model Adaptive Estimator (MMAE) and Interacting Multiple Model (IMM) estimation.

B.1 Nonlinear Kalman Filtering

The Kalman Filter algorithm is straightforward to implement and it obtains state estimates using a recursive operation with no iteration. This makes the Kalman Filter attractive for systems that operate in real time. However, the Kalman Filter algorithm is exact only for the linear systems with Gaussian forcing terms. Numerous extensions of the Kalman Filter to nonlinear systems with non-Gaussian noise exist. The Extended Kalman Filter (EKF) and the Unscented Kalman Filter (UKF) are well-known extensions for systems with nonlinear dynamics and measurements. The reader is assumed familiar with estimation, and is directly to references such as Ref. [3] if additional background is needed. This sections reviews the two estimation approaches qualitatively.

The EKF uses the estimated state process and measurement Jacobians to replace the linear coefficients of the standard Kalman Filter algorithm. The state estimate and measurement expectation are evaluated using the nonlinear equations, and the covariance is propagated and updated using the Jacobians. This leads to an algorithm that is first-order accurate with respect to the statistical moments of the propagated system. The EKF is a direct extension of the linear Kalman Filter, and is commonly used throughout engineering domains. Because it truncates the higher-order statistical moments, it may perform poorly when the uncertainty is high, or if the time between measurements is quite large[3]. Furthermore, the need to compute analytical system Jacobians can

Model	$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{w}, t),\ \boldsymbol{w}(t) \sim \mathcal{N}(\boldsymbol{0}, [Q(t)])$ $\tilde{\boldsymbol{y}}_k = \boldsymbol{h}_k(\boldsymbol{x}, t) + \boldsymbol{v}_k,\ \boldsymbol{v}_k \sim \mathcal{N}(\boldsymbol{0}, [R_k])$
Initialization	$\hat{\boldsymbol{x}}(t_0) = \hat{\boldsymbol{x}}_0$ $[P_0] = \mathrm{cov}(\hat{\boldsymbol{x}}_0)$
Propagation	$\dot{\hat{\boldsymbol{x}}} = \boldsymbol{f}(\hat{\boldsymbol{x}}, \boldsymbol{u}, \boldsymbol{0}, t)$ $[\dot{P}] = [F(t)][P] + [P][F(t)]^T + [G(t)][Q(t)][G(t)]^T$ $[F(t)] \equiv \left.\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}\right\|_{\hat{\boldsymbol{x}}(t)},\ [G(t)] \equiv \left.\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{w}}\right\|_{\hat{\boldsymbol{x}}(t)}$
Update	$[K_k] = [P_k^-][H_k]^T([H_k][P_k^-][H_k]^T + [R_k])^{-1}$ $\hat{\boldsymbol{x}}_k^+ = \hat{\boldsymbol{x}}_k^- + [K_k](\tilde{\boldsymbol{y}}_k - \boldsymbol{h}(\hat{\boldsymbol{x}}_k^-, t))$ $[P_k^+] = ([\mathbb{I}_{n\times n}] - [K_k][H_k])[P_k^-]$ $[H_k] \equiv \left.\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}\right\|_{\hat{\boldsymbol{x}}_k^-}$

Table B.1: Summary of continuous-discrete Extended Kalman Filter algorithm adapted from Ref. [3]

be tedious, and is not possible for all systems.

The UKF is an equivalent algorithm to the EKF, in that it extends the linear Kalman Filter to systems with nonlinear process and measurement models. It obtains theoretically better convergence properties than the EKF, at a slightly increased typical computational cost. For every iteration of the UKF, the state and covariance estimate are replaced by a particle approximation. The particle approximation and associated weights are deterministically chosen and the particles are referred to as sigma points. Each particle is propagated through the nonlinear dynamics. The weighted statistical mean and covariances of the propagated particles and their measurement residuals are used to directly implement the Kalman update without the need to compute explicit system Jacobians.

The properties of the EKF and UKF have been widely studied, and interested readers are directed to sources such as Refs. [3, 4], which include more details. The algorithms for each estimator are collected in this section, for reference. Table B.1 summarizes the EKF. The algorithm re-

Model	$\boldsymbol{x}_{k+1} = \boldsymbol{f}_k(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{w}_k),\ \boldsymbol{w}_k \sim \mathcal{N}(\mathbf{0}, [Q_k])$ $\tilde{\boldsymbol{y}}_k = \boldsymbol{h}_k(\boldsymbol{x}) + \boldsymbol{v}_k,\ \boldsymbol{v}_k \sim \mathcal{N}(\mathbf{0}, [R_k])$
Initialization	$\hat{\boldsymbol{x}}(t_0) = \hat{\boldsymbol{x}}_0$ $[P_0] = \text{cov}(\hat{\boldsymbol{x}}_0)$
Reset	$\hat{\boldsymbol{x}}_k^a = \begin{bmatrix} \hat{\boldsymbol{x}}_k^T & \mathbf{0}_{n_w}^T & \mathbf{0}_{n_v}^T \end{bmatrix}^T$ $[P_k^a] = \text{blockdiag}([P_k], [Q_k], [R_k])$
Sigma point calculation	$\boldsymbol{\chi}_k = \begin{bmatrix} \boldsymbol{x}_k^a & \boldsymbol{x}_k^a + \text{chol}((L+\lambda)[P_k^a]) & \boldsymbol{x}_k^a - \text{chol}((L+\lambda)[P_k^a]) \end{bmatrix}$
Propagation	$\boldsymbol{\chi}_{i,k+1}^{x,-} = \boldsymbol{f}_k(\boldsymbol{\chi}_{i,k}^{x}, \boldsymbol{u}_k, \boldsymbol{\chi}_{i,k}^{w}), i = 0, \ldots, 2L$ $\hat{\boldsymbol{x}}_{k+1}^- \equiv \sum_{i=0}^{2L} W_i^{(m)} \boldsymbol{\chi}_{i,k+1}^{x,-}$ $[P_{k+1}^-] \equiv \sum_{i=0}^{2L} W_i^{(c)} (\boldsymbol{\chi}_{i,k+1}^{x,-} - \hat{\boldsymbol{x}}_{k+1}^-)(\boldsymbol{\chi}_{i,k+1}^{x,-} - \hat{\boldsymbol{x}}_{k+1}^-)^T$
Update	$\boldsymbol{\Upsilon}_{i,k}^- = \boldsymbol{h}_k(\boldsymbol{\chi}_{i,k}^{x,-}) + \boldsymbol{\chi}_{i,k+1}^{v},\ i = 0, \ldots, 2L$ $\hat{\boldsymbol{y}}_k^- \equiv \sum_{i=0}^{2L} W_i^{(m)} \boldsymbol{\Upsilon}_{i,k}^-$ $[P_{yy_k}] = \sum_{i=0}^{2L} W_i^{(c)} (\boldsymbol{\Upsilon}_{i,k}^- - \hat{\boldsymbol{y}}_k^-)(\boldsymbol{\Upsilon}_{i,k}^- - \hat{\boldsymbol{y}}_k^-)^T$ $[P_{xy_k}] = \sum_{i=0}^{2L} W_i^{(c)} (\boldsymbol{\chi}_{i,k}^{x,-} - \hat{\boldsymbol{x}}_k^-)(\boldsymbol{\Upsilon}_{i,k}^- - \hat{\boldsymbol{y}}_k^-)^T$ $[K_k] \equiv [P_{xy_k}][P_{yy_k}]^{-1}$ $\boldsymbol{x}_k^+ = \boldsymbol{x}_k^- + [K_k](\tilde{\boldsymbol{y}}_k - \hat{\boldsymbol{y}}_k^-)$ $[P_k^+] = [P_k^-] - [K_k][P_{yy_k}][K_k]^T$

Table B.2: Summary of discrete Unscented Kalman Filter algorithm adapted from Ref. [4]

quires knowledge of the process and measurement model Jacobians. Assuming these are available, it is straightforward to implement the EKF, and the algorithm requires little further explanation.

The UKF is summarized in Table B.2. The operations are somewhat more complicated than the EKF's and some explanation is merited. The governing model and initialization are identical between the EKF and UKF, save for the discrete process model in the UKF. At every iteration of the UKF, an augmented state and covariance $\hat{\boldsymbol{x}}_k^a$ and $[P_k^a]$ are computed using the current estimates and noise covariances. The augmented state is of dimension $L = n + n_w + n_v$, where n, n_w, and n_v denote the length of the state, process noise, and measurement noise vectors. The

Cholesky decomposition of the augmented covariance is used to generate $2L+1$ “sigma points” whose weighted mean and covariance match the prior mean and covariance exactly. $\boldsymbol{\chi}^{x}_{i,k}$ denotes the elements of the ith sigma point corresponding to the state elements, and the superscripts w and v denote the elements corresponding to process and measurement noise elements. Each sigma point is individually propagated through the nonlinear dynamics, and the propagated mean and covariance are computed as weighted sums using the mean weights $W_i^{(m)}$ and the covariance weights $W_i^{(c)}$. In a similar fashion, predicted measurement values $\boldsymbol{\Upsilon}^{-}_{i,k}$ are generated for each sigma point, and the associated values of the measurement expectation, measurement covariance $[P_{yy_k}]$, and state-measurement cross-correlation matrix $[P_{xy_k}]$ are computed as weighted sums. The Kalman gain and updated state and covariance are directly evaluated without recourse to any Jacobians.

UKF weights are presented for completeness. The weights used in the mean and covariance approximations are obtained as follows:

$$
\begin{aligned}
W_0^{(c)} = \frac{\lambda}{L+\lambda} + (1-\alpha^2+\beta) \\
W_0^{(m)} = \frac{\lambda}{L+\lambda} \\
W_i^{(c)} = W_i^{(m)} = \frac{\lambda}{2(L+\lambda)} \\
\lambda \equiv \alpha^2(L+\kappa) - L
\end{aligned}
$$

α determines the spread of the sigma points about the mean and is commonly set to a small value such as 10^{-3}. κ is a “secondary” scaling parameter and is commonly set to zero. β is a factor that incorporates prior knowledge of the distribution, and $\beta = 2$ is optimal for Gaussian distributions.

This completes the presentation of the basic EKF and UKF algorithms. The PDF truncation algorithm is presented next.

B.2 Constrained estimation

Constrained estimation is discussed in Chapter 5 as a method to reduce control singularities due to estimation errors and to identify singular engagements. This section reviews a PDF truncation algorithm for constraining state estimates. There are three subsections. First, the algorithm for PDF truncation is presented. Second, a brief description of recursive and nonrecursive implementations of this algorithm is given. Finally, the application of this approach to nonlinearly constrained problems is considered.

B.2.1 Density function truncation for constrained estimation

This subsection outlines a method for PDF truncation based on an algorithm described in Simon and Simon [64] and refined in Ref. [65]. For a system with *linear* inequality constraints, Ref. [64] presents an exact PDF truncation algorithm. The approach can be approximately extended to nonlinear dynamics or measurements by the usual EKF approximation with Jacobians. The procedure uses matrix decomposition of the covariance to perform the PDF truncation in a one-dimensional fashion by operating on a transformation of the unconstrained state. The primary focus of this subsection is a description of the algorithm. The procedure begins with the posterior state estimate and covariance, $\hat{\boldsymbol{x}}_k^+$ and $[P_k^+]$, produced by the standard, unconstrained Kalman update step. The state is subject to s linear constraints with upper and lower bounds as follows:

$$\boldsymbol{a} \leq [\Phi]\boldsymbol{x} \leq \boldsymbol{b} \tag{B.1}$$

In Eq. B.1, $[\Phi] \in \mathbb{R}^{s \times n}$. The algorithm truncates the PDF given by $\mathcal{N}(\hat{\boldsymbol{x}}_k^+, [P_k^+])$ by applying constraints one at a time. The linear constraints may be time-varying, in which case the subscript k can be appended to all terms, but this is omitted to simplify the notation. The constrained state mean and covariance are denoted $\tilde{\boldsymbol{x}}_k^+$ and $[\tilde{P}_k^+]$, and are initialized to $\tilde{\boldsymbol{x}}_k^+ = \hat{\boldsymbol{x}}_k^+$ and $[\tilde{P}_k^+] = [P_k^+]$. The algorithm then iterates over the s constraints according to the following procedure. For each $i = 1, 2, \ldots, s$, compute the Jordan canonical decomposition of $[P_k^+]$ such that $[T][W][T]^T = [P_k^+]$, in which $[T]$ is orthogonal and $[W]$ diagonal. Let $\boldsymbol{\phi}_i$ indicate the ith row of $[\Phi]$. An orthogonal matrix $[\rho]$ is then computed by Gram-Schmidt orthogonalization such that it satisfies the following relationship:

$$[\rho][W]^{1/2}[T]^T\boldsymbol{\phi}_i = \begin{bmatrix} \left(\boldsymbol{\phi}_i^T[\tilde{P}_k^+]\boldsymbol{\phi}_i\right)^{1/2} & 0 & \ldots & 0 \end{bmatrix}^T \tag{B.2}$$

An appropriate Gram-Schmidt algorithm is given in Ref. [64]. The matrices are used to effect a coordinate change from the original problem states $\boldsymbol{x}_k$ to new coordinates $\boldsymbol{z}_k^{(i)}$:

$$\boldsymbol{z}_k^{(i)} \equiv [\rho][W]^{-1/2}[T]^T(\boldsymbol{x}_k - \tilde{\boldsymbol{x}}_k^+) \tag{B.3}$$

Note the transformation of Eq. B.3 need not be evaluated, but is included here for clarity. The effect of the coordinate transformation is to ensure that the elements of $\boldsymbol{x}_k$ associated with constraint i become the first dimension of the coordinates $\boldsymbol{z}_k^{(i)}$; the PDF can then be truncated by performing operations only on a scalar density associated with the first element of $\boldsymbol{z}_k^{(i)}$. The upper and lower bounds a_i and b_i must also be transformed. When the transformation of Eq. B.3 is applied to the ith equation of Eq. B.1, constraints take the following form, in which $\boldsymbol{z}_k^{(i)}(1)$ indicates the first element of $\boldsymbol{z}_k^{(i)}$:

$$\frac{a_i - \boldsymbol{\phi}_i^T \tilde{\boldsymbol{x}}_k^+}{\sqrt{\boldsymbol{\phi}_i^T [\tilde{P}_k^+] \boldsymbol{\phi}_i}} \leq \boldsymbol{z}_k^{(i)}(1) \leq \frac{b_i - \boldsymbol{\phi}_i^T \tilde{\boldsymbol{x}}_k^+}{\sqrt{\boldsymbol{\phi}_i^T [\tilde{P}_k^+] \boldsymbol{\phi}_i}} \tag{B.4}$$

From the coordinate transformation of Eq. B.3, it can be verified that the variable $\boldsymbol{z}_k^{(i)}$ has identity covariance. Since the elements of $\boldsymbol{z}_k^{(i)}$ are uncorrelated, the truncated PDF can be evaluated using simple one-dimensional analysis. Defining the lower and upper bounds of $\boldsymbol{z}_k^{(i)}(1)$ to be c_i and d_i, respectively, the Gaussian PDF of $\boldsymbol{z}_k^{(i)}(1)$ is truncated to have zero value below c_i and above d_i. The resulting truncated function is then re-normalized, so that it is a valid PDF, and its expectation and covariance become the values of the constrained $\boldsymbol{z}_k^{(i)}$, which is denoted $\tilde{\boldsymbol{z}}_k^{(i)}$. Under this model, the mean and covariance of the random variable associated with $\tilde{\boldsymbol{z}}_k^{(i)}(1)$ are given as follow:

$$\mu \equiv \alpha(e^{-\frac{c_i^2}{2}} - e^{-\frac{d_i^2}{2}}) \tag{B.5}$$

$$\sigma^2 \equiv \alpha(e^{-\frac{1}{2}c_i^2}(c_i - 2\mu) - e^{-\frac{1}{2}d_i^2}(d_i - 2\mu)) + \mu^2 + 1 \tag{B.6}$$

$$\alpha \equiv \frac{\sqrt{2}}{\sqrt{\pi}(\operatorname{erf}(d_i/\sqrt{2}) - \operatorname{erf}(c_i/\sqrt{2}))} \tag{B.7}$$

The truncated variable $\tilde{\boldsymbol{z}}_k^{(i)}$ is initialized to its expectation, which is given below with its covariance:

$$\tilde{\boldsymbol{z}}_k^{(i)} = \begin{bmatrix} \mu & 0 & 0 & \dots & 0 \end{bmatrix}^T \tag{B.8}$$

$$[\tilde{Z}] = \operatorname{cov}(\tilde{\boldsymbol{z}}_k^{(i)}) = \operatorname{diag}(\sigma^2, 1, 1, \dots, 1) \tag{B.9}$$

By inverting the original transformation of Eq. B.3, the truncated state and covariance in the original state space are updated to the following values:

$$\tilde{\boldsymbol{x}}_k^+ \leftarrow [T][W]^{1/2}[\rho]^T \tilde{\boldsymbol{z}}_k^{(i)} + \tilde{\boldsymbol{x}}_k^+ \tag{B.10}$$

$$[\tilde{P}_k^+] \leftarrow [T][W]^{1/2}[\rho]^T [\tilde{Z}][\rho][W]^{1/2}[T]^T \tag{B.11}$$

The value of i is then incremented by one, and the process repeats until $i > s$. This completes the description of the PDF truncation procedure outlined in Ref. [64].

B.2.2 Recursive and nonrecursive constrained estimates

In Chapter 5, a simple example problem described by Simon [60] is used to show that blindly applying the PDF truncation algorithm after each measurement leads to degenerate behavior of the estimator. Simon identifies two possible remedies and labels them as “recursive” and “nonrecursive” implementations. The recursive filter can be obtained by only applying the truncation process to constraints in Eq. B.1 that the posterior unconstrained mean violates. If a posterior state mean satisfies all constraints, no truncation is performed. This is qualitatively similar to other methods of constrained estimation, and is a potential computational savings as well. With the nonrecursive filter, the state estimate used by the UKF is never altered, but the state estimate used for external calculation and logging is computed by performing the truncation operation on the unconstrained mean and covariance. The method ensures that the filter continues to operate using the approximate maximum likelihood estimate. On the other hand, performing operations with the unconstrained estimate may not be possible, as in the exact LQDG adaptive game, where there is no finite-valued control solution if the constraints are violated. When the Taylor series approximation to the Riccati solution is used, the nonrecursive algorithm can be applied with no numerical problems.

B.2.3 Application to nonlinear inequality constraints

The algorithm outlined thus far can be applied to the nonlinearly constrained problem simply by expanding the constraint about the unconstrained posterior and using the Jacobian to obtain a first-order accurate solution. A modified algorithm used by Ref. [65] operates in "constraint space" rather than in state-space. Since, for the LQDG problem, the number of constraints is always less than the number of states, this approach reduces the computational costs of some linear algebra operations and may be appealing. The approach is now summarized. Let the nonlinearly constrained system be given as follows:

$$\boldsymbol{a} \leq \boldsymbol{p}(\boldsymbol{x}) \leq \boldsymbol{b} \tag{B.12}$$

As before, the vectors $\boldsymbol{a}, \boldsymbol{b}$ are both of dimension s. To first order, the unconstrained value of the constraint function has mean $\boldsymbol{\mu}_p \equiv \boldsymbol{p}(\hat{\boldsymbol{x}}_k^+)$ and covariance $[O][P_k^+][O]^T$, where $[O]$ is the Jacobian of $\boldsymbol{p}$ evaluated at $\hat{\boldsymbol{x}}_k^+$. The PDF truncation algorithm can then be applied directly to the random variable having distribution $\mathcal{N}(\boldsymbol{\mu}_p, [O][P_k^+][O]^T)$, resulting in a truncation which is accurate to first order, and operates on a lower-dimensional space than the original algorithm by Simon. However, the resulting truncated variable in constraint space must be converted back into state space, a process which in general is not unique. The solution used in Ref. [65] is to treat the constraint space truncated variable as a pseudo-measurement acting on the state space. The pseudo-measurement noise is chosen to satisfy the following expression:

$$[R] = ([P_k^+]^{-1} - [P_k^+]^{-1}[\tilde{P}_k^+][P_k^+]^{-1})^{-1} - [P_k^+] \tag{B.13}$$

The Kalman gain associated with the pseudo-measurement is given from the standard EKF equation as follows:

$$[K_k] = [P_k^+][O]^T(R + [O][P_k^+][O])^{-1} \tag{B.14}$$

The covariance update from the pseudo-measurement follows the standard equation of the EKF, using $[O]$ as the pseudo-measurement Jacobian. The mean update of the EKF is modified to produce a truncated mean that is consistent with the truncation in constraint space:

$$\tilde{\boldsymbol{x}}_k^+ = \hat{\boldsymbol{x}}_k^+ + [K_k]([O][K_k])^{-1}(\tilde{\boldsymbol{z}}_k - \hat{\boldsymbol{z}}_k) \tag{B.15}$$

This completes the description of the Tully algorithm for PDF truncation of nonlinearly constrained state estimates, and finishes the discussion of the PDF truncation algorithm. In the next section, multiple-model estimation algorithms are discussed.

B.3 Multiple-model estimation

This section presents the multiple-model adaptive estimator (MMAE) and Interacting Multiple Model (IMM) estimator developed in the radar tracking field. The IMM is used in this dissertation to identify opponent strategies from a set of candidate approaches if the opponent may be using a strategy based on switching his strategy during a differential game. If the opponent is expected not to make a switch, then the MMAE should properly be used. The MMAE is exact for linear hybrid systems that do not switch modes. This section begins by introducing the MMAE, which has a simple algorithm and helps motivate the IMM. Afterwards the full IMM algorithm is presented and summarized.

B.3.1 Multiple Model Adaptive Estimation algorithm

If an observed system is known to remain in one mode for all time, then the MMAE of Ref. [1, p. 442–443] is an efficient solution. While this estimator is not used in the dissertation, it is simple to understand and useful in introducing multiple model estimation. The operation of a two-mode estimator is shown in Fig. B.1. A filter is implemented for each of r candidate modes. The filters operate totally independently and exchange no information. When a measurement is made, each filter computes the likelihood of the measurement given the propagated state, which is the function $\Gamma_k^{(\ell)} = p(\tilde{\boldsymbol{y}}_k; \hat{\boldsymbol{y}}_k^-, [H_k][P_k^-][H_k]^T + [R_k])$. Based on the likelihood of each filter, the

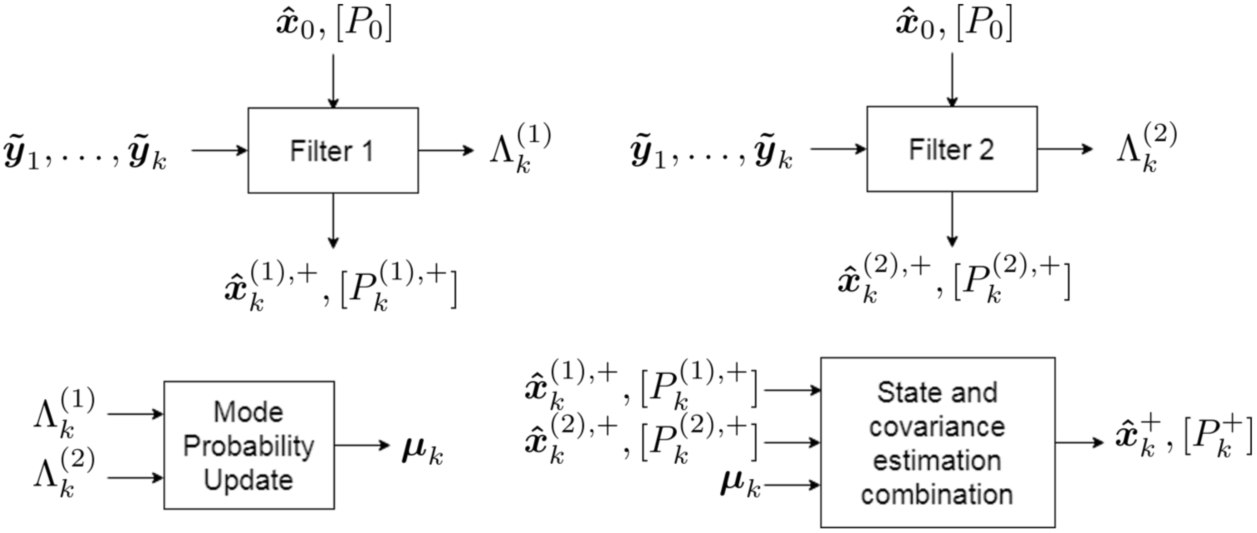
$\hat{\boldsymbol{x}}_0, [P_0]$
$\tilde{\boldsymbol{y}}_1, \ldots, \tilde{\boldsymbol{y}}_k$
Filter 1
$\Lambda_k^{(1)}$
$\hat{\boldsymbol{x}}_k^{(1),+}, [P_k^{(1),+}]$
$\hat{\boldsymbol{x}}_0, [P_0]$
$\tilde{\boldsymbol{y}}_1, \ldots, \tilde{\boldsymbol{y}}_k$
Filter 2
$\Lambda_k^{(2)}$
$\hat{\boldsymbol{x}}_k^{(2),+}, [P_k^{(2),+}]$
$\Lambda_k^{(1)}$
$\Lambda_k^{(2)}$
Mode Probability Update
$\boldsymbol{\mu}_k$
$\hat{\boldsymbol{x}}_k^{(1),+}, [P_k^{(1),+}]$
$\hat{\boldsymbol{x}}_k^{(2),+}, [P_k^{(2),+}]$
$\boldsymbol{\mu}_k$
State and covariance estimation combination
$\hat{\boldsymbol{x}}_k^{+}, [P_k^{+}]$

Figure B.1: Diagrammatic representation of a two-mode static multiple model estimator, adapted from Ref. [1, p. 442].

mode probability for each filter is computed according to Bayes' rule:

$$\mu_k^{(\ell)} = \frac{\Gamma_k^{(\ell)} \mu_{k-1}^{(\ell)}}{\sum_j^n \Gamma_k^{(j)} \mu_{k-1}^{(j)}} \tag{B.16}$$

In the final stage, the mode-conditioned state and covariance are computed by a particle average weighted by the mode probabilities; this produces a merged state and covariance, which is used by whatever process requires state estimates. This procedure is simple, but is analytically exact if the mode describing the target dynamics is in the set of modes and has been active during all measurements[1].

The MMAE is an appropriate choice of estimator for a game environment if an opponent can be reasonably assumed to use a fixed mode for the duration of the game. This is the case, for example, in the missile endgame scenario considered in Ref. [27]. In that work, an aircraft targeted by a missile uses an MMAE to identify the missile's guidance law and improve the chance of evasion. This assumption is appropriate for the missile endgame scenario, which typically has a duration on the order of seconds. However, it is felt that for general games, the possibility that the opponent might switch his strategy is an important one to consider. Consequently, this dissertation uses the

IMM, rather than the MMAE, for gameplay against an opponent with an a priori unknown strategy. It should be noted that the computational complexity of the two approaches is similar, as the largest computational cost is associated with propagating and updating r estimators.

B.3.2 Interacting Multiple Model estimation

The IMM algorithm is now presented. The IMM estimator can be thought of as an ensemble of "standard" sequential estimators with a few additional steps wrapping each estimator. This approach is flexible, and allows estimators of different types to be mixed freely. In this dissertation, only EKFs and UKFs are used. First, the basic IMM steps are discussed, followed by a detailed discussion of IMM modes with non-overlapping states, and then a radar tracking example is presented.

The basic algorithm and notation presented here are taken from Refs. [1, pp. 454–457][86]. The algorithm can be summarized in five steps:

1. Calculation of the mixing probabilities

2. State and covariance mixing

3. Standard filter propagation and update

4. Mode probability update

5. Estimate and covariance combination

The governing equations for each step are now detailed.

B.3.2.1 Calculation of the mixing probabilities

The mixing probabilities conditioned on the system measurements to index $k-1$ are given as $\mu_{k-1}^{(i|j)+}$ below:

$$\mu_{k-1}^{(i|j)+} = \frac{p_{ij}\mu_{k-1}^{(i)}}{\bar{c}_j} \tag{B.17}$$

In Eq. B.17, $\mu_{k-1}^{(i|j)+}$ is the probability that mode i was in effect at $k-1$ given that mode j is in effect at k, conditioned on measurements up to and including time $k-1$. p_{ij} is an element of an a priori matrix that defines the probability of switching from model i to model j. $\mu_{k-1}^{(i)}$ is the mode probability that mode i is active at $k-1$. $\bar{c}_j$ is a normalizing factor computed as follows:

$$\bar{c}_j = \sum_{i=1}^{r} p_{ij} \mu_{k-1}^{(i)} \;\; j = 1, \ldots, r \tag{B.18}$$

B.3.2.2 Mixing

In this step, the individual filter estimates are averaged with weights according to the mixing probabilities. For each mode j in the ensemble, the prior estimate at the beginning of iteration k is computed as follows:

$$\hat{\boldsymbol{x}}_{k-1}^{(j)}{}^{+} = \sum_{i=1}^{r} \hat{\boldsymbol{x}}_{k-1}^{(i)} \mu_{k-1}^{(i|j)+} \tag{B.19}$$

The covariance for each mode is computed in a similar fashion:

$$[P_{k-1}^{(j)}]^{+} = \sum_{i=1}^{r} \mu_{k-1}^{(i|j)+} \left([P_{k-1}^{(i)}] + (\hat{\boldsymbol{x}}_{k-1}^{(i)} - \hat{\boldsymbol{x}}_{k-1}^{(j)}{}^{+})(\hat{\boldsymbol{x}}_{k-1}^{(i)} - \hat{\boldsymbol{x}}_{k-1}^{(j)}{}^{+})^{T} \right) \tag{B.20}$$

B.3.2.3 Standard filter propagation and update

For each mode j, the mixed state and covariance $\hat{\boldsymbol{x}}_{k-1}^{(j)}{}^{+}$, $[P_{k-1}^{(j)}]^{+}$ are taken as the prior information and the standard estimation steps of propagation and update are performed. The linear Kalman filter or its nonlinear approximations can be used here. Additionally, likelihoods associated with each mode are computed. Assuming approximately Gaussian statistics, likelihoods $\Gamma_k^{(j)}$ are computed using the innovations and innovation covariance:

$$\Gamma_k^{(j)} = p(\tilde{\boldsymbol{y}}_k - \hat{\boldsymbol{y}}_k^{(j)}, [H_k][P_k^{(j)}]^{-}[H_k]^{T} + [R_k]) \tag{B.21}$$

B.3.2.4 Mode probability update

The probability that the target is in mode j at index k is denoted as $\mu_k^{(j)}$. The mode probabilities are updated according to the following equations:

$$\mu_k^{(j)} = \frac{1}{c}\Gamma_k^{(j)}\bar{c}_j \tag{B.22}$$

$$c \equiv \sum_{j=1}^{r} \Gamma_k^{(j)}\bar{c}_j \tag{B.23}$$

B.3.2.5 Estimation and covariance combination

This step is not necessary for the sequential IMM algorithm, but is required to compute a state estimate for feedback control. According to the standard algorithm, the model-conditioned state and covariance estimates are combined according to the following equations:

$$\hat{\boldsymbol{x}}_k = \sum_{j=1}^{r} \hat{\boldsymbol{x}}_k^{(j)+}\mu_k^{(j)} \tag{B.24}$$

$$[P_k] = \sum_{j=1}^{r} \mu_k^{(j)}\left([P_k^{(j)}]^{+} + (\hat{\boldsymbol{x}}_k^{(j)+} - \hat{\boldsymbol{x}}_k)(\hat{\boldsymbol{x}}_k^{(j)+} - \hat{\boldsymbol{x}}_k)^T\right) \tag{B.25}$$

The complete algorithm is summarized in Table B.3. As the next subsection discusses, this estimate combination is not appropriate when the state vectors contain unlike elements, and should be modified in that case. As presented thus far, the IMM algorithm is appropriate for mixing estimates from filters that estimate the same state variables. However, one of the great strengths of multiple model estimation is the flexibility to use ensemble models having unlike state components. This is critical in the application to differential games, since the parameterized uncertainties in different opponent modes do not have a one-to-one correspondence between different modes. An additional state vector augmentation before mixing is required to avoid biased estimates, and this is the topic of the next subsection.

Model	$\dot{\boldsymbol{x}} = \boldsymbol{f}^{(m)}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{w}, t),\ m \in \{m_1, \dots, m_r\},\ \boldsymbol{w} \sim \mathcal{N}(\boldsymbol{0}, [Q])$ $\tilde{\boldsymbol{y}}_k = \boldsymbol{h}_k(\boldsymbol{x}, \boldsymbol{v}_k, t),\ \boldsymbol{v}_k \sim \mathcal{N}(\boldsymbol{0}, [R_k])$
Initialization	$\hat{\boldsymbol{x}}^{(\ell)}(t_0) = \hat{\boldsymbol{x}}_0^{(\ell)}$ $\text{cov}(\hat{\boldsymbol{x}}_0^{(\ell)}) = [P_0^{(\ell)}]$ $Pr(m_\ell)\Big\|_{t=0} = \mu_0^\ell$ $Pr(m(k) = m_h \| m(k-1) = m_\ell) = p_{h\ell}$
Calculate mixing probabilities	$\mu_{k-1}^{(i\|j)^+} = \frac{p_{ij}\mu_{k-1}^{(i)}}{\sum_{\ell=1}^r p_{\ell j}\mu_{k-1}^{(\ell)}}, j = 1, \dots, r$
State mixing	$\hat{\boldsymbol{x}}_{k-1}^{(j)^{\ +}} = \sum_{i=1}^r \hat{\boldsymbol{x}}_{k-1}^{(i)} \mu_{k-1}^{(i\|j)^+}$ $[P_{k-1}^{(j)}]^+ = \sum_{i=1}^r \mu_{k-1}^{(i\|j)^+} \left([P_{k-1}^{(i)}] + (\hat{\boldsymbol{x}}_{k-1}^{(i)} - \hat{\boldsymbol{x}}_{k-1}^{(j)^{\ +}})(\hat{\boldsymbol{x}}_{k-1}^{(i)} - \hat{\boldsymbol{x}}_{k-1}^{(j)^{\ +}})^T\right)$
Filter propagation and update	$(\hat{\boldsymbol{x}}_{k-1}^{(\ell)^{\ +}}, [P_{k-1}^{(\ell)}]^+, \tilde{\boldsymbol{y}}_k) \to (\hat{\boldsymbol{x}}_k^{(\ell)^+}, [P_k^{(\ell)}]^+)$ $\Gamma_k^{(\ell)} = p(\tilde{\boldsymbol{y}}_k - \hat{\boldsymbol{y}}_k^{(\ell)}, [H_k][P_k^{(\ell)}]^-[H_k]^T + [R_k])$
Mode probability update	$\mu_k^{(j)} = \frac{1}{c}\Gamma_k^{(j)}\bar{c}_j$ $c \equiv \sum_{j=1}^r \Gamma_k^{(j)}\bar{c}_j$
Mixed state output	$\hat{\boldsymbol{x}}_k = \sum_{j=1}^r \hat{\boldsymbol{x}}_k^{(j)^+} \mu_k^{(j)}$ $[P_k] = \sum_{j=1}^r \mu_k^{(j)} \left([P_k^{(j)}]^+ + (\hat{\boldsymbol{x}}_k^{(j)^+} - \hat{\boldsymbol{x}}_k)(\hat{\boldsymbol{x}}_k^{(j)^+} - \hat{\boldsymbol{x}}_k)^T\right)$

Table B.3: Algorithm summary for Interacting Multiple Model estimation

B.3.3 Ensemble models of different state dimensions

The IMM algorithm can in principle support filters having different state vectors, and various examples in the literature can be found (e.g, Refs [86, 72]). As pointed out by Granström et al[73], the different state vectors are often handled in an ad hoc fashion by padding the lower-dimensional states with zeros, which leads to biased estimates when the mixed state output is computed. A theoretically unbiased estimate is obtained instead by augmenting the lower-dimensional states with the mean and covariance extracted from the larger state estimate(s). Such an approach is outlined in Ref. [87]. Ref. [73] goes on to present a method for augmenting states with an arbitrary distribution, which may improve filter performance in some arenas; for the purposes of this dissertation, consideration is restricted to approximately Gaussian distributions, for which it is sufficient to augment states with the mean and covariance from other estimators.

As a concrete example, consider an opponent in a LQDG that is assumed to play either a minimax or FNE game solution. For simplicity, let the state vector have dimension 1. Filter 1 becomes the filter for the OSO game, in which the estimated states are $\hat{\boldsymbol{x}}^{(1)} \equiv [\hat{\boldsymbol{x}}, \hat{\boldsymbol{\theta}}^{(1)^T}]^T$. Similarly, Filter 2 is the filter for the FNE game with $\hat{\boldsymbol{x}}^{(2)} \equiv [\hat{\boldsymbol{x}}, \hat{\boldsymbol{\phi}}^{(2)^T}]^T$. $\hat{\boldsymbol{\theta}}$ is of length 3 and contains weight elements $q_j, r_{ji}, s_{f,j}$ corresponding to j's minimax strategy, while $\hat{\boldsymbol{\phi}}$ contains the three weight elements corresponding to j's FNE strategy. Each filter operates as normal with an estimate vector length of dimension four. For the purposes of state and covariance mixing, an augmented state of dimension seven is defined:

$$\hat{\boldsymbol{x}}^{(a)} \equiv \begin{bmatrix} \hat{\boldsymbol{x}} \\ \hat{\boldsymbol{\theta}} \\ \hat{\boldsymbol{\phi}} \end{bmatrix} \tag{B.26}$$

To obtain unbiased mixed state estimates, the augmented mean and covariance defined below are used for each filter:

$$\hat{\boldsymbol{x}}^{(1,a)} \equiv \begin{bmatrix} \hat{\boldsymbol{x}}^{(1)} \\ \hat{\boldsymbol{\theta}}^{(1)} \\ \hat{\boldsymbol{\phi}}^{(2)} \end{bmatrix}, \ \mathrm{cov}(\hat{\boldsymbol{x}}^{(1,a)}) = \begin{bmatrix} p_{xx}^{(1)} & p_{xr}^{(1)} & p_{xq}^{(1)} & p_{xs}^{(1)} & 0 & 0 & 0 \\ p_{rx}^{(1)} & p_{rr}^{(1)} & p_{rq}^{(1)} & p_{rs}^{(1)} & 0 & 0 & 0 \\ p_{qx}^{(1)} & p_{qr}^{(1)} & p_{qq}^{(1)} & p_{qs}^{(1)} & 0 & 0 & 0 \\ p_{sx}^{(1)} & p_{sr}^{(1)} & p_{sq}^{(1)} & p_{ss}^{(1)} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & p_{rr}^{(2)} & p_{rq}^{(2)} & p_{rs}^{(2)} \\ 0 & 0 & 0 & 0 & p_{qr}^{(2)} & p_{qq}^{(2)} & p_{qs}^{(2)} \\ 0 & 0 & 0 & 0 & p_{sr}^{(2)} & p_{sq}^{(2)} & p_{ss}^{(2)} \end{bmatrix} \tag{B.27}$$

$$\hat{\boldsymbol{x}}^{(2,a)} \equiv \begin{bmatrix} \hat{\boldsymbol{x}}^{(2)} \\ \hat{\boldsymbol{\theta}}^{(1)} \\ \hat{\boldsymbol{\phi}}^{(2)} \end{bmatrix}, \; \mathrm{cov}(\hat{\boldsymbol{x}}^{(2,a)}) = \begin{bmatrix} p_{xx}^{(2)} & 0 & 0 & 0 & p_{xr}^{(2)} & p_{xq}^{(2)} & p_{xs}^{(2)} \\ 0 & p_{rr}^{(1)} & p_{rq}^{(1)} & p_{rs}^{(1)} & 0 & 0 & 0 \\ 0 & p_{qr}^{(1)} & p_{qq}^{(1)} & p_{qs}^{(1)} & 0 & 0 & 0 \\ 0 & p_{sr}^{(1)} & p_{sq}^{(1)} & p_{ss}^{(1)} & 0 & 0 & 0 \\ p_{rx}^{(2)} & 0 & 0 & 0 & p_{rr}^{(2)} & p_{rq}^{(2)} & p_{rs}^{(2)} \\ p_{qx}^{(2)} & 0 & 0 & 0 & p_{qr}^{(2)} & p_{qq}^{(2)} & p_{qs}^{(2)} \\ p_{sx}^{(2)} & 0 & 0 & 0 & p_{sr}^{(2)} & p_{sq}^{(2)} & p_{ss}^{(2)} \end{bmatrix} \tag{B.28}$$

The augmented state and covariance, which contain the parameter estimates from all r filters, are used in the mixing processes of Eqs. B.19, B.20, B.24, and B.25. Note that the particle-like covariance terms in Eq. B.25 for the terms not common across filters will produce zero contribution to the mixed covariance, so some redundant calculations in the covariance mixing process can be eliminated.

This completes the presentation of the background of MM estimation and the IMM algorithm. The next subsection presents a concrete example of the IMM estimator.

B.3.4 IMM example

As a numerical example of IMM estimation, consider the one-dimensional motion of a target having a maximum speed constraint. The nominal target model is a constant-acceleration model, with disturbing zero mean white noise at the acceleration level. This model can track the position, velocity, and acceleration-level states reasonably well given position measurements of sufficient quality. However, by using a bank of filters that also incorporates a constant-velocity model, lower tracking error can be achieved.

The target dynamics are given simply by a double-integrator model. The following open-loop velocity profile is followed:

$$x_2(t) = \min(V_m, \max(-V_m, 3\sin\left(\frac{t}{3}\right))) \tag{B.29}$$

Two filters are used in the IMM filter bank. Filter 1 is a constant-acceleration model with the following process model:

$$\begin{bmatrix} \dot{x}_1{}^{(1)} \\ \dot{x}_2{}^{(1)} \\ \dot{x}_3{}^{(1)} \end{bmatrix} = \begin{bmatrix} x_2{}^{(1)} \\ x_3{}^{(1)} \\ w_1{}^{(1)} \end{bmatrix}, \quad w_1{}^{(1)} \sim \mathcal{N}(0, q_1) \tag{B.30}$$

Similarly, Filter 2 is a constant-velocity model with a disturbing white-noise term acting on the velocity derivative. Note that Filter 2 has one fewer state than Filter 1:

$$\begin{bmatrix} \dot{x}_1{}^{(2)} \\ \dot{x}_2{}^{(2)} \end{bmatrix} = \begin{bmatrix} x_2{}^{(2)} \\ w_1{}^{(2)} \end{bmatrix}, \quad w_1{}^{(2)} \sim \mathcal{N}(0, q_2) \tag{B.31}$$

Both filters have position-level measurements with the same discrete-time measurement model:

$$\tilde{y}_k = \tilde{y}(t_k) = x_1(t_k) + v_k, \;\; v_k \sim \mathcal{N}(0, r) \tag{B.32}$$

In practice, the performance of the IMM estimate is highly dependent on the tuning of the filters. Filter 1 is tuned to produce acceptably fast tracking of the acceleration-level state, and Filter 2 is tuned to provide low-variance estimates when the target velocity is saturated. This induces a lag in the response of Filter 2. The effect of this lag is that the estimated states of Filter 1 are initially used when the target velocity saturates. Once Filter 2's states converge to the true velocity the IMM algorithm favors those estimates, which tend to have lower errors because they more accurately reflect the saturated dynamics.

Both filters are initialized with identity covariance of appropriate dimension. Each mode is assumed equally probable initially. The prior mode transition probability matrix is populated as $p_{ij} = .94\delta_{ij} + 0.03$. The other numerical values used are collected in Table B.4.

Estimation performance for one execution of this scenario is shown in Fig. B.2. The mode probability estimate histories are shown in Figs. B.3. This figure shows the true and estimated states, as well as the estimation error performance. Recall that x_3, corresponding to the accelera-

$x_1(0)$	$x_2(0)$	$x_3(0)$	Δt	T_f	q_1	q_2	r
0	0	0	0.1	6π	1.0	0.1	0.01

Table B.4: Numerical values used in IMM numerical example

tion, is not estimated by Filter 2, so the plotted values for the IMM estimate of x_3 are simply Filter 2's estimates. The results show good tracking of the position and velocity terms, and acceptable tracking of the acceleration under the circumstances. During this simulation, the mean squared errors for each state are $(0.00508, 0.0518, 0.238)$. For comparison, Filter 1's mean squared errors are $(0.00568, 0.0593, 0.238)$ and Filter 2's are $(0.00586, 0.0581)$. Thus, for the common states, the IMM estimate accuracy outperforms either of the two component filters for this simulation.

It is instructive to review the estimation error and covariance plots of Fig. B.2. The error plots show the IMM covariance as well as the component filter covariances. When the velocity is constant, Filter 2 is used more, and the IMM covariance approaches the Filter 2 covariance. When the velocity is changing, then Filter 1 is used more, and the Filter 2 covariance, due to the state mixing, approaches the Filter 1 covariance.

In reviewing the mode probability histories of Fig. B.3, it is clear that the likelihood-based probability estimates are quite noisy. This is typical of modal probability histories estimated by an IMM for a single simulation, as other literature examples show[88, 71]. This high noise has the effect of introducing fluctuations in the IMM covariance estimates, which exhibit a "jitter" compared to either Filter 1 or Filter 2 alone. This symptom can be remedied by altering the prior transition probabilities to increase p_{ii} and reduce $p_{ij}, j \neq i$, which has the effect of reducing the state mixing. However, for this particular example, this alteration has the effect of increasing the mean squared estimation error.

This completes the discussion of the multiple-model estimation algorithms, and finishes the appendix of estimation algorithms.

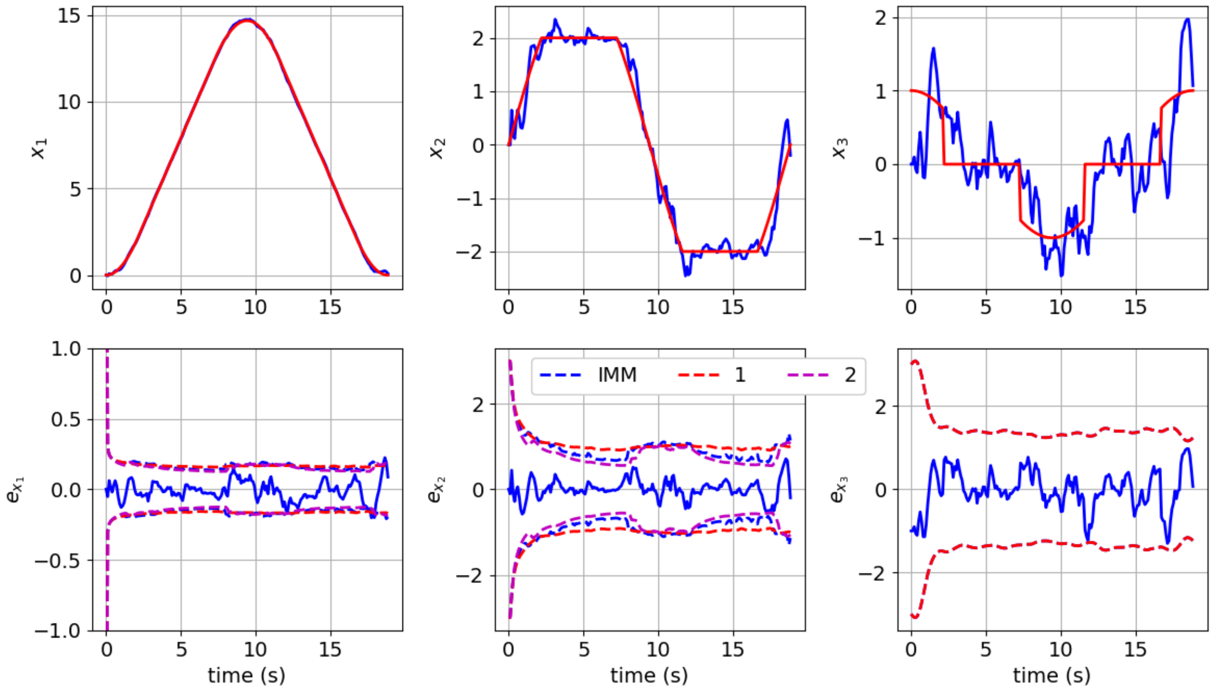

Figure B.2: IMM estimation performance for example tracking problem. Top: true state (red) and estimated state (blue). Bottom: estimation errors with 3σ bounds. For reference, the individual filter 3σ bounds are also plotted.

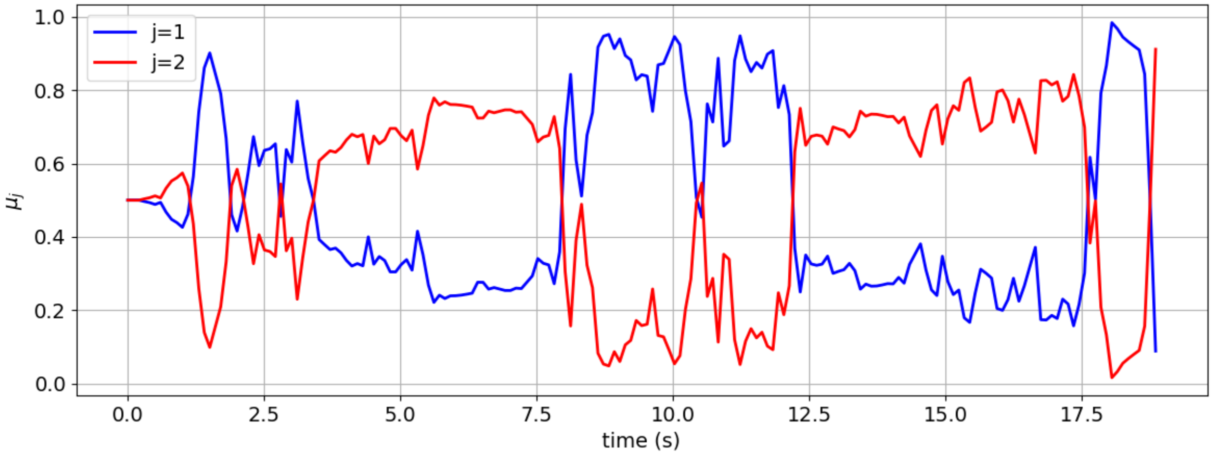

Figure B.3: Mode probability estimates from IMM example tracking problem

APPENDIX C

CHEBYSHEV POLYNOMIAL FUNCTION APPROXIMATION

Chapter 5 proposes a procedure for evaluating LQDG engagement stability. A core component of this procedure is a numerical approximation to the engagement stability boundary. Chebyshev polynomial basis functions are used to approximate this stability boundary because they are relatively simple to use and scale reasonably well to function approximation in high dimensions. The dissertation of Ref. [58] demonstrates the suitability for least-squares-based function approximation with Chebyshev polynomials in high dimensions. The approach works by evaluating the function at grid points corresponding to the Chebyshev-Gauss-Lobato (CGL) points and choosing a least-squares weight matrix derived from the Chebyshev orthogonality property. By exploiting sparsity in the resulting high-order tensor coefficients, the linear least-squares solution for the Chebyshev basis function coefficients can be obtained from a 2D matrix product of dense matrices.

This appendix outlines the vectorized method of Chebyshev basis function approximation via weighted least-squares. The problem of Chebyshev function approximation is to find a linear combination of basis functions that approximately satisfies the following relationship:

$$f(\boldsymbol{x}) \approx \sum_{i=1}^{K} a_i \alpha_i(\boldsymbol{x}) \tag{C.1}$$

In Eq. C.1, the basis functions $\alpha_i(\boldsymbol{x})$ are chosen to be products of univariate Chebyshev functions. The weighted least-squares problem is obtained by requiring the coefficients minimize the following cost function, in which Einstein summation notation is employed:

$$J = \min_{\boldsymbol{a}} W_{kg} \left(f(\boldsymbol{x}_k) - \sum_{i=1}^{K} a_i \alpha_i(\boldsymbol{x}_k) \right) \left(f(\boldsymbol{x}_g) - \sum_{i=1}^{K} a_i \alpha_i(\boldsymbol{x}_g) \right) \tag{C.2}$$

The independent variables $\boldsymbol{x}$ are taken to be of dimension n.

To present the method of Ref. [58], some preliminary terms must be defined. A set of lower and upper bounds on each independent variable must be provided, and the bounds are denoted by l_i, u_i for each dimension $i = 1, 2, \ldots, n$. For a given approximation problem, the maximum degree of polynomial to be used, N, and the number of CGL nodes to use, denoted by $M+1$, are specified by the user.[1] The approach outlined here must satisfy $M \leq N$. While in general, the values of N and M need not be the same for every dimension of the problem, the presented solution assumes uniformity across all dimensions for simplicity. Let ϕ-space denote the n-dimensional space in which the Chebyshev basis functions are defined, with $\phi_i \in [-1, 1]$ being satisfied. The mapping from ϕ-space to parameter space is evaluated as follows for each dimension i:

$$x_i = l_i + \frac{1}{2}(\phi_i + 1)(u_i - l_i) \tag{C.3}$$

The unique $M + 1$ unique CGL nodes for each axis are computed according to the discrete cosine sampling scheme given below:

$$\phi_i(k) = \cos\left(\frac{\pi k}{M}\right), k = 0, \ldots, M; i = 1, \ldots, n \tag{C.4}$$

Weight matrices $[W_k] \in \mathbb{R}^{(M+1)\times(M+1)}$ associated with the least-squares estimation problem are defined as follows:

$$[W_k] \equiv \text{diag}(0.5, 1, 1, \ldots, 1, 0.5),\ k = 1, \ldots, n \tag{C.5}$$

Similarly, n matrices of dimension $(N+1) \times (M+1)$ associated with the values of the Chebyshev polynomials at the CGL nodes are computed as follows. Note that $T_j(\phi)$ is the jth Chebyshev polynomial evaluated in ϕ-space, and $[B]^{1/2}$ is used to denote the Cholesky decomposition of the

[1] This slightly cumbersome notation is chosen so that M and N become interchangeable if $M = N$ is specified, as is commonly done in Ref [58].

matrix $[B]$.

$$[C_x]_k \equiv \frac{1}{M} \begin{bmatrix} \frac{1}{2}T_0(\phi_k(0)) & T_0(\phi_k(1)) & \dots & T_0(\phi_k(M-1)) & \frac{1}{2}T_0(\phi_k(M)) \\ T_1(\phi_k(0)) & 2T_1(\phi_k(1)) & \dots & 2T_1(\phi_k(M-1)) & T_(\phi_k(M)) \\ & & \vdots & & \\ T_N(\phi_k(0)) & 2T_N(\phi_k(1)) & \dots & 2T_N(\phi_k(M-1)) & T_N(\phi_k(M)) \end{bmatrix} [W_k]^{-1/2} \quad \text{(C.6)}$$

Finally, let $\mathbf{1}_j$ represent a vector of ones of length j, and let the operator $\mathrm{stack}(\boldsymbol{v}, g)$ yield a gm-length vector formed by vertically concatenating the $m-$length vector $\boldsymbol{v}$ g times. Let the vector $\boldsymbol{x}_i$ be the $(M+1)$-length vector of unique values of the ith dimension CGL nodes, expressed in parameter space according to Eq. C.3. A 2D array $[X]$ of unique combinations of the CGL nodes in parameter space is formed with columns given from the following expression:

$$\boldsymbol{X}_i = \mathrm{stack}(\boldsymbol{x}_i \otimes \mathbf{1}_{(M+1)^{N+1-i}}, (M+1)^{i-1}), i = 1, \dots, n \quad \text{(C.7)}$$

The defined terms are then processed into 2D arrays for final computation. The Kronecker product plays a large role. The vector $\boldsymbol{f} \in \mathbb{R}^{(M+1)^{(N+1)}}$ is then populated by evaluating the underlying function at the point in parameter space given by each row in $[X]$. This vector represents the true function values in the least-squares fitting problem after factorization by the Chebyshev orthogonality properties. The matrices $[C_x]_k$ are the values of the Chebyshev basis functions at the CGL nodes associated with dimension k. These matrices are the result of Kronecker-factorizing the dense least-squares matrix $[C]$, which is computed as follows:

$$[C] \equiv [C_x]_1 \otimes [C_x]_2 \otimes \ldots \otimes [C_x]_n \quad \text{(C.8)}$$

Similarly, the term $[W_f]$, which arises after factoring and vectorizing the weight matrices in the least-squares operation, is computed as follows:

$$[W_f] \equiv [W_1] \otimes [W_2] \otimes \ldots \otimes [W_n] \quad \text{(C.9)}$$

The equation for the vector solution of the Chebyshev coefficients is given as follows:

$$\boldsymbol{a} = [C][W_f]^{1/2}\boldsymbol{f} \tag{C.10}$$

The vector of coefficients, $\boldsymbol{a}$, denote the coefficients of terms in the Chebyshev polynomial function approximation to the underlying function. $\boldsymbol{a}$ can be re-arranged into an n-d array, each dimension of size $N+1$, according to the following procedure: first, $\boldsymbol{a}$ is reshaped to an n-d array, each dimension of size N, using column-major operations; then the dimensions of the resulting array are permuted such that dimension 1 becomes dimension n, dimension 2 becomes dimension $n-1$, etc. The latter operation is analogous to a transpose in two dimensions. The resulting array $[A]$ has $(N+1)^n$ elements, and each entry is associated with a particular product of Chebyshev basis functions. For example, for $n = 3$, $A_{i,j,k}$ is the coefficient associated with the product of Chebyshev polynomials evaluated at point (ϕ_1, ϕ_2, ϕ_3) in ϕ-space: $T_{i-1}(\phi_1)T_{j-1}(\phi_2)T_{k-1}(\phi_3)$. The full function approximation is evaluated by summation over all the independent elements of $[A]$, and can also be conveniently evaluated by vectorization[58]. This completes the description of the algorithm for Chebyshev basis function approximation.

APPENDIX D

MONTE CARLO CONVERGENCE STUDIES FOR MISSILE INTERCEPTION GAMES

It is good practice to confirm the statistical significance of results by repeating Monte Carlo trials with two samples of the same distribution and observing the change in performance metrics. In this dissertation, repetition studies are not conducted for all Monte Carlo tests. Rather, one repetition trial is conducted for the major classes of problems in Chapter 7. The primary goal is to ensure that subsequent Monte Carlo trials use enough samples to obtain statistical significance.

This appendix summarizes Monte Carlo repetition studies conducted to validate convergence in Chapter 7. Four types of engagements are considered. First is the simple planar engagement in which both players use adaptive FNE strategies, reported in Section 7.3.1. Second is the non-planar engagement used in the constrained estimation tests of Section 7.3.2. Third is the planar engagement of Section 7.3.3 used to evaluate the ability of the IMM estimator to differentiate OSO and FNE opponent strategies. Finally, the Monte Carlo evaluation of play against switching-based evader strategies of Section 7.3.4 is evaluated.

D.1 Simple adaptive LQDG engagements

The results of Chapter 7 begin by re-considering engagements in which adaptive agents play subject only to parametric uncertainty in opponent strategies. To validate the numerical convergence of these results, two sets of 2500 Monte Carlo simulations are conducted in which both agents use the adaptive FNE strategy. These settings use the default (10^{-6}) state measurement variance. The tabulated results are error metrics and median relative loss metrics for each agent.

Table D.1 compares the error metrics for the two Monte Carlo simulations. Metrics based on the quartiles of absolute error all match to two significant figures, except for the evader upper quartile metric. The Mean Square Error (MSE) metrics agree only to the first significant figure. From this analysis, the pursuer and evader error distributions are extremely similar. The evader absolute errors are slightly larger at all quartile levels. However, only the upper quartile increase

	MSE	Median Abs Err	Lower Quartile	Upper Quartile
		2500 simulations		
Pursuer	0.0342	0.0258	0.00641	0.0808
Evader	0.0263	0.0263	0.00643	0.0845
		Replication case		
Pursuer	0.0359	0.0257	0.00636	0.0808
Evader	0.0273	0.0263	0.0064	0.0829

Table D.1: Comparison of error metrics for two sets of 2500 simulations with adaptive FNE agents

	Median Loss	Lower Quartile	Upper Quartile
		2500 simulations	
Pursuer	-0.000638	-0.0029	0.00146
Evader	$4.07{\times}10^{-5}$	-0.00283	0.00277
		Replication case	
Pursuer	-0.000579	-0.00304	0.00146
Evader	-0.00012	-0.00291	0.00274

Table D.2: Comparison of loss metrics for two sets of 2500 simulations with adaptive FNE agents

is statistically significant for the number of simulations completed. The MSE metric is larger for pursuer than for evader. This suggests that pursuer errors above the 75th percentile grow more quickly than evader errors, increasing the MSE.

Table D.2 compares the loss metrics for the two scenarios. The metrics show two significant figure agreement on the upper quartile, one significant figure agreement on the lower quartile, and limited agreement on the median value. The pursuer median values agree to one significant figure, but the evader median values do not even agree in sign. It can be confidentally concluded that the loss metrics are clustered between small values of loss that are slightly biased toward negative values. 50% of pursuer loss values lie on the interval $[-.0030, .0014]$ and 50% of evader loss values lie on $[-.003, .0027]$. The median pursuer value is $-.0006$, and the median evader value is near zero.

	Unconstrained estimation			Constrained estimation		
	r_{ep}	q_e	$s_{f,e}$	r_{ep}	q_e	$s_{f,e}$
2500 simulations						
Mean Square Error	6.88×10^{-5}	0.376	0.0465	8.58×10^{-6}	0.0194	0.0199
Standard Error	0.00829	0.613	0.215	0.00291	0.139	0.14
Median Abs. Error	0.00131	0.0754	0.0597	0.00131	0.0745	0.0596
Lower Quartile Abs. Error	0.000603	0.0362	0.0266	0.000609	0.0354	0.0258
Upper Quartile Abs. Error	0.00242	0.130	0.119	0.00244	0.130	0.118
Replication case						
Mean Square Error	1.22×10^{-5}	0.0657	0.027	9.2×10^{-6}	0.0166	0.0201
Standard Error	0.00347	0.256	0.164	0.00301	0.129	0.141
Median Abs. Error	0.00133	0.0781	0.0635	0.00134	0.0775	0.0631
Lower Quartile Abs. Error	0.000612	0.0385	0.0286	0.00063	0.0364	0.028
Upper Quartile Abs. Error	0.00248	0.132	0.122	0.00249	0.132	0.122

Table D.3: Monte Carlo estimation error metrics in validation study for evader weights sampled from a stable set in 2500 constrained estimation engagements

D.2 Constrained estimation engagements

The constrained estimation engagements compare the estimation error and failure rate of simulations when the pursuer uses constrained and unconstrained estimation. The results reported in the text in Table 7.6 display error metrics for 2500 Monte Carlo simulations, and are reproduced in the top part of Table D.3. Using unconstrained estimation, the mean-based error metrics, MSE and standard error, do not achieve convergence to one significant figure-agreement for the unconstrained estimation. The median-based error metrics typically agree to one significant figure, and some agree to two. These two facts suggest that enough large outliers are recorded to prevent convergence of the mean-based metrics, which could indicate either a failure of the filter to converge. When constrained estimation is used, both mean- and median-based metrics converge to either one or two significant figures. This suggests constrained estimation reduces the number of high-error outliers during execution of the filter.

Table D.4 compares the number of failed simulations for the case of normally distributed evader weights. Simulation failures are extremely uncommon and are not expected to exhibit strong statistical convergence. These results indicate that the rates of failures converge in their order-of-

	2500 simulations	**Replication case**
Unconstrained estimation	25	17
Constrained estimation	2	1

Table D.4: Replication study of simulation failures

	2500 simulations			**Replication case**		
Evader mode	Correct	Incorrect	Total	Correct	Incorrect	Total
Minimax	2496	4	2500	2491	9	2500
FNE	2015	485	2500	2028	472	2500

Table D.5: Summary of detection performance in 2500 Monte Carlo simulations with NZS nominal weights

magnitude to 10^{-2} for unconstrained estimaton and to 10^{-4} for constrained estimation. This is suggestive of a statistically significant difference in the two populations, although it would be prudent to conduct a larger study with more unstable cases to confirm this trend.

D.3 OSO/FNE differentiation

This series of engagements evaluates the ability of a pursuer using an IMM estimation algorithm to distinguish evaders using the minimax and adaptive FNE strategies. Table D.5 displays Monte Carlo results for a nominally NZS engagement and a validation experiment. The upper set of values is reported in the main text as Table 7.9. The original set of results indicate that incorrect identifications are made quite rarely when the evader uses a minimax strategy. Based on the repeated results, these outcomes occur extremely rarely, in less than 1% of simulations. When the evader uses the adaptive FNE strategy, the IMM pursuer is more likely to make errors in identification. Based on the repeated experiment, the error rate is 19% with two significant figures of confidence.

D.4 Switching-based evader strategies

The effectiveness of deceptive evader strategies is evaluated in Monte Carlo simulations. These simulations tabulate the estimation error metrics for the pursuer as well as loss function metrics

	2500 simulations		**Replication case**	
P Metric	L_i	l_i	L_i	l_i
Mean	0.0201	1.35	0.0207	1.05
Standard Error	0.0248	13.2	0.0251	5.6
Median	0.0106	0.608	0.0119	0.598
Lower Quartile	0.000725	0.0801	0.000897	0.127
Upper Quartile	0.0329	0.804	0.0344	0.805
E Metric	L_i	l_i	L_i	l_i
Mean	-0.0278	2.39	-0.029	0.537
Standard Error	0.0274	54	0.0274	12.2
Median	-0.0257	-0.317	-0.0254	-0.328
Lower Quartile	-0.0455	-0.472	-0.0486	-0.481
Upper Quartile	-0.00717	-0.0973	-0.00832	-0.119

Table D.6: Convergence study of loss function metrics with evader using gain level switching strategy

for pursuer and evader. Additionally, plots of the loss function and mode probability distributions across simulations are generated. This section presents a validation study of 1000 simulations conducted to confirm convergence of evader gain level switching results. 1000 simulations are used instead of 2500 because the evader and pursuer weight values are fixed, and only initial conditions are sampled.

Table D.6 shows the convergence study results for the loss functions values obtained by each agent when the evader uses the gain level switching strategy. The non-normalized loss function metrics show agreement to two significant figures in mean, standard error, and median. The quartile values demonstrate less convergence, but typically one or two significant figures. The normalized loss metrics display typically one significant figure of agreement, except in the mean and standard error metrics. Each of these metrics is derived from the mean and is sensitive to outliers. The median losses agree to two significant figures for both pursuer and evader. The quartile values agree to either one or two significant figures for each player. Based on these findings, 50% of the pursuer performance metrics lie between 10% and 80% with a median value of 60%. 50% of evader metrics lie between -50% and -10% with a median of -30%.

Table D.7 shows the estimation error results of the same convergence study. Errors are com-

	2500 simulations		**Replication case**	
Metric	k_1	k_2	k_1	k_2
Mean	10.0	21.5	9.99	21.6
Standard Error	2.30	3.93	2.29	3.96
Median Absolute Error	1.20	0.117	1.19	0.114
Lower Quartile	0.265	0.0274	0.26	0.0287
Upper Quartile	4.32	4.5	4.33	4.38

Table D.7: Convergence study of estimation error with gain level switching evader strategy

puted separately for the pre-switch and post-switch periods of the engagement. Most error metrics agree to two significant figures between the original and replication cases. Some upper and lower quartile values agree to only one figure. The mean and standard error metrics are particularly consistent, suggesting that outlier effects in these engagements are relatively small. Of course, the estimation errors shown here do not have any useful diagnostic information about the mode a pursuer believes the evader is using; all that can be said is that the estimator itself performs consistently between the two Monte Carlo samples.

This appendix has summarized convergence studies conducted for a representative subset of Monte Carlo results. 2500 simulations are found to be adequate to obtain one to two significant figures of agreement between convergence trials. Certain metrics demonstrate less consistent convergence when mean-based metrics are evaluated. The significant figures in these results should be kept in mind in reviewing the missile interception results of Chapter 7.