Title
The Linear Quadratic Gaussian Multistage Game with Nonclassical Information Pattern Using a Direct Solution Method

Author
Clemens, Joshua

Publication Date
2018

University of California
Los Angeles

The Linear Quadratic Gaussian Multistage Game with Nonclassical Information Pattern Using a Direct Solution Method

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

by

Joshua William Clemens

2018

Abstract of the Dissertation

The Linear Quadratic Gaussian Multistage Game with Nonclassical Information Pattern Using a Direct Solution Method

by

Joshua William Clemens

Doctor of Philosophy in Aerospace Engineering

University of California, Los Angeles, 2018

Professor Jason L. Speyer, Chair

Game theory has application across multiple fields, spanning from economic strategy to optimal control of an aircraft and missile on an intercept trajectory. The idea of game theory is fascinating in that we can actually mathematically model real-world scenarios and determine optimal decision making. It may not always be easy to mathematically model certain real-world scenarios, nonetheless, game theory gives us an appreciation for the complexity involved in decision making. This complexity is especially apparent when the players involved have access to different information upon which to base their decision making (a nonclassical information pattern).

Here we will focus on the class of adversarial two-player games (sometimes referred to as *pursuit-evasion games*) with nonclassical information pattern. We present a two-sided (simultaneous) optimization solution method for the two-player linear quadratic Gaussian (LQG) multistage game. This direct solution method allows for further interpretation of each player's decision making (strategy) as compared to previously used formal solution methods. In addition to the optimal control strategies, we present a saddle point proof and we derive an expression for the optimal performance index value. We provide some numerical results in order to further interpret the optimal control strategies and to highlight real-world application of this game-theoretic optimal solution.

The dissertation of Joshua William Clemens is approved.

Lieven Vandenberghe

Robert Thomas M'Closkey

James S. Gibson

Jason L. Speyer, Committee Chair

University of California, Los Angeles

2018

To my family . . .

for all your love and support.

Table of Contents

List of Figures

List of Tables

Acknowledgments

This endeavor has been almost ten years in the making. It has been one of perseverance, not only on my part, but those around me, supporting me.

I want to first give my sincere and heartfelt thank you to Dr. Jason Speyer, my advisor and mentor for the past decade. Dr. Speyer's technical guidance, patience, and encouragement propelled me through the program. He made himself available to talk anytime that I requested, which often ended up being Saturday afternoons due to my numerous commitments during the week. Dr. Speyer adjusted to my ever-changing schedule with work and family over a ten-year period, including months of little progress each time my wife and I welcomed a baby into the world. I consider myself blessed to have had the opportunity to work with a giant and pioneer in the field, and I'm honored to call him my mentor and friend.

Thank you to Lockheed Martin for ten years of tuition reimbursement and support, up through executive leadership. Thank you to all of my co-workers and managers for your support and understanding over the many years. Specifically, thank you to my functional leads, Doug and Keith, and to my managers, Larry and Jim, for your unwaivering support during this long endeavor. I realize that my many absences from work, required to complete this degree, were often inconvenient. I would not be at this point today if it weren't for your support. And, thank you to Greg for providing funding during the final dissertation writing - your support afforded me the necessary time to completely focus on research and finish the dissertation.

Most importantly, thank you to my family and friends for your encouragement and support as I chased this dream. Thank you to my parents for the opportunities that you gave me early on in life, and for your constant words of encouragement and support. And, finally, no other person has sacrificed more than my wife, Maria - your love, dedication, and patience has made this dream possible. I'm excited for this next chapter of our lives.

Vita

2004	B.S. Aerospace Engineering, Iowa State University.
2004–2006	Engineer, Lockheed Martin Space Systems Company.
2006–present	Engineer, Lockheed Martin Aeronautics Company.
2007	M.S. Aerospace Engineering, University of Colorado, Boulder.
2011	M.S. Aerospace Engineering, University of California, Los Angeles.

Publications

Joshua W. Clemens and Jason L. Speyer. "The LQG Game with Nonclassical Information Pattern Using a Direct Solution Method." In *2017 American Control Conference*, pp. 418-423, May 2017.

CHAPTER 1

Overview

1.1 History of Dynamic Games

The theory behind dynamic (multistage or differential) games was greatly developed in the 1950s and 1960s, partially motivated by Isaacs' publication [Isa55]. Table 1.1 lists some of this foundational work in game theory. Since Isaacs first proposed the differential game problem much research has been done to further understand the problem and develop its variations. The problem is often cast as a two-player game where the players have competing interests. For this reason, this type of game is often referred to as a *zero-sum game*, where one player is trying to minimize the performance index and the other player is trying to maximize the same performance index.

The information set that each player has available upon which to base his control strategy greatly affects the solution complexity. System noises (process and measurement noise), or information time lag, can greatly alter the optimal control strategies. These result in a nonclassical information pattern (also referred to as information asymmetry), which means that all players do not have access to the same information at each time step upon which to base their strategy. For practical purposes (in terms of actual implementation) it is desired that the information required by any player's optimal strategy be finite-dimensional.

The deterministic two-player linear quadratic (LQ) game (no process or measurement noise) was first solved by Ho, Bryson and Baron ([HBB65]), and particular/special combinations of process and measurement noises were solved by Behn and Ho ([BH68], [Beh68]) and Rhodes and Luenberger ([RL69]). The basis for our research relies heavily on the in-

Table 1.1: A brief history of foundational work in game theory.

Decade	**Names**	**Major Contribution**
1940s	von Neumann and Morgenstern [NM44]	Game Theory
1950s	Nash [Nas50]	Nash Equilibrium
	Shapely [Sha53]	Stochastic games
	Isaacs [Isa55],[Isa65]	Differential games
	Berkovitz and Fleming [BF55], [Ber60]	Differential game solutions using calculus of variations
1960s	Ho, Bryson and Baron [HBB65]	Deterministic LQ differential game solution
	Behn and Ho [BH68], Rhodes and Luenberger [RL69]	Special LQG differential game solutions
	Willman [Wil68], [Wil69]	General LQG differential game solution
	Starr and Ho [SH69]	Non-zero-sum games
	Witsenhausen [Wit68]	Witsenhausen Counterexample
1970s	Witsenhausen [Wit73]	Stochastic control standard form

novative work of Willman ([Wil69], [Wil68]) and his formal analysis of the general (process and measurement noise) linear quadratic Gaussian (LQG) differential game.

1.2 Problem Overview

In game theory parlance, sequential dynamic multi-player optimal control problems (*game* or *team* problems) are generally classified according to Figure 1.1.[1] That is, we can define a particular class of problems by specifying the type of objective and information pattern involved.

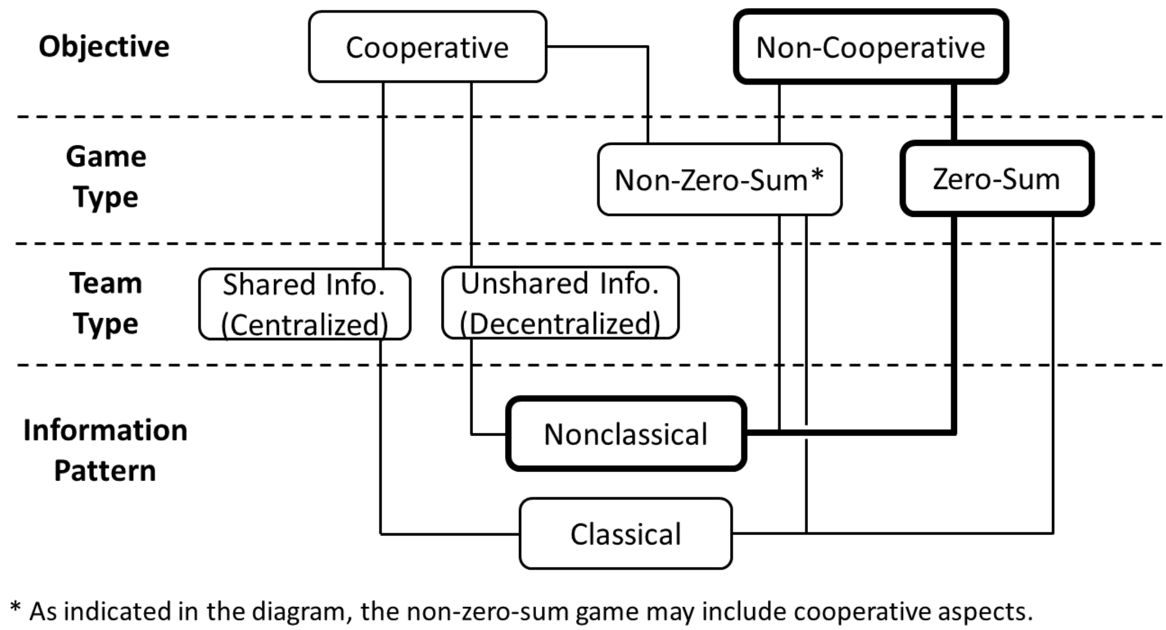

Figure 1.1: Game theory problem classification.

As indicated by the bold outline in the figure above, the type of game considered in this dissertation is the multistage, two-player, zero-sum game (non-cooperative objective) with nonclassical information pattern (unshared, imperfect information). This implies that each player knows only his own measurement history and past controls. He knows nothing else about his opponent other than what he can deduce through his own measurement history and control history. This is the most general LQG two-player game. The restrictions on this solution are multistage linear system dynamics and Gaussian zero-mean, delta-correlated process and measurement noises.

[1] Also, reference [Mah08, Chapter 1] for further discussion of problem classification.

In order to solve this LQG game Willman defined an assumed affine control strategy functional form and an enlarged state-space for each player. He then sequentially substituted the opposing player's assumed control strategy functional form into the enlarged state dynamics and performance index, and solved the resulting one-sided LQG stochastic optimal control problem. Since each player is faced with a one-sided LQG stochastic optimal control problem, the separation principle applies. This allowed Willman to easily derive the optimal control gain and optimal estimator for each player. Furthermore, Willman took advantage of the certainty equivalence principle to interpret and simplify the resulting optimal strategies.

Willman's method is formal and provides no rigorous proof as to when solutions exist or how to calculate the optimal performance index value. Bagchi and Olsder [BO81] re-developed Willman's game with some generalizations and further discussion on solution existence and saddle point uniqueness. With Willman's control strategies the designer numerically iterates back and forth in the hopes of convergence for a particular set of problem parameters. Although formal in nature, Willman's novel technique revealed that the optimal strategies are infinite-dimensional. The solution method presented herein allows for further interpretation of the results as compared to Willman's formal method.

In this dissertation we will reformulate the LQG game problem as a deterministic game problem in the system statistics. We conceived this notion as a combination of the methods employed by Willman, and Behn and Ho, in their aforementioned foundational work.[2] This deterministic framework allows us to use a two-sided (simultaneous) optimization technique to derive the nonclassical information pattern saddle point solution. As will be shown, information asymmetry enters through specification of the respective players' measurement matrix. This allows us to easily look at different information patterns by adjusting the elements of the measurement matrix, thereby controlling the information that each player has available.

[2]We recognize that this method has similarities to Witsenhausen's "standard form" ([Wit73]). Also, this method appears to have similarities to the *designer's approach* ([NMT13] and [Mah08]), a solution approach to decentralized stochastic team optimization problems.

It should be noted that in recent years [NGL14], [GNL14], and [Gup14] have also looked at LQG games with nonclassical information pattern from the viewpoint of what they call *common* and *private* information. Their approach looks at the evolution of information to formulate a game of symmetric information (classical information pattern, called game **G2**) that can be solved, and subsequently used to find a Nash equilibrium of the original asymmetric game (nonclassical information pattern, called game **G1**). This formulation requires a probabilistic (Bayesian) view of the problem, whereas here we will reformulate the stochastic game as a deterministic game. However, we will draw some parallels to the *common* and *private* information concept as it applies to our optimal strategies.

Other recent work in the area of multi-player stochastic optimal control problems with special information patterns include [LL11] and [LL12] (decentralized cooperative control), [PP10] (non-cooperative control, full state feedback), and [Pac16] (non-cooperative control, control sharing pattern).

1.3 Dissertation Overview

In Chapter 2 we completely define the LQG multistage game problem, including the governing dynamics and performance index to be optimized. In Chapter 3 we take a moment to outline the deterministic multistage game solution for later comparison with the stochastic (LQG) multistage game solution. In Chapter 4 we derive the general LQG multistage game solution using a two-sided (simultaneous) optimization technique. This includes a saddle point proof, an expression for the optimal performance index value, and discussion and interpretation of the optimal strategies. We draw parallels between our stochastic optimal solution and the corresponding deterministic game solution from Chapter 3. We also discuss how different information patterns may be modeled within the framework of our solution methodology. In Chapter 5 we perform a numerical study to show the effects of process noise variance and measurement noise variance on the resulting strategies. This allows us to provide additional interpretation of the optimal strategies. In Chapter 6 we look at the performance of the stochastic optimal strategies as applied to a simple missile pursuit-evasion problem. Finally, in Chapter 7 we have some concluding remarks and suggestions for continued research in this area.

CHAPTER 2

Problem Definition

In this chapter we completely define the two-player LQG multistage game optimization problem. We discuss notation, and specify the governing dynamics and performance index that will be used throughout the remainder of this dissertation.

2.1 Notation and Dimensions

In general, subscript p indicates a *pursuer* parameter (the minimizing player) and subscript e indicates an *evader* parameter (the maximizing player). The index i is used to indicate the stage, with the initial stage $i = 0$. Table 2.1 lists the main problem parameters using this notation.

Table 2.1: Parameter definitions for the two-player LQG multistage game.

Parameter	**Description**
$x(i) \in \mathbb{R}^n$	State
$u_p(i) \in \mathbb{R}^m$	Pursuer Control
$u_e(i) \in \mathbb{R}^l$	Evader Control
$w(i) \in \mathbb{R}^n$	Process Noise
$z_p(i) \in \mathbb{R}^p$	Pursuer Measurement
$v_p(i) \in \mathbb{R}^p$	Pursuer Measurement Noise
$z_e(i) \in \mathbb{R}^q$	Evader Measurement
$v_e(i) \in \mathbb{R}^q$	Evader Measurement Noise

Capital letters are used to denote an enlarged dimension matrix (e.g. $X(i)$ is the enlarged state at stage i) and bold-face capital letters are used to denote an enlarged dimension *unconditional* statistical parameter (e.g. $\mathbf{X}(i)$ is the enlarged state second moment at stage i). Greek letters are used to denote system matrices and English letters are used to denote the corresponding enlarged dimension system matrices (e.g. $\Theta_p(i)$ is the pursuer's measurement matrix, and $H_p(i)$ is the pursuer's enlarged measurement matrix).

Lastly, we identify sub-partitions of the enlarged matrices using subscripts denoting (row, column) block element pairs. A colon in the subscript denotes an entire row or column depending on the placement in the subscript pair. For example, $S_2(i)_{i+1,:}$ is the $(i+1)^{th}$ row and all columns of the enlarged matrix $S_2(i)$. In other words, $S_2(i)_{i+1,:}$ is an enlarged row vector.

2.2 State Dynamics

The multistage state dynamics are represented as

$$x(i+1) = x(i) + \Gamma_p(i)u_p(i) - \Gamma_e(i)u_e(i) + w(i) \tag{2.1}$$

for $i = 0, 1, \ldots, N-1$. The problem parameters are the state $x(i) \in \mathbb{R}^n$, pursuer's control matrix $\Gamma_p(i) \in \mathbb{R}^{n \times m}$, pursuer's control $u_p(i) \in \mathbb{R}^m$, evader's control matrix $\Gamma_e(i) \in \mathbb{R}^{n \times l}$, evader's control $u_e(i) \in \mathbb{R}^l$, and process noise $w(i) \in \mathbb{R}^n$.

This is the stochastic multistage form of the deterministic differential game dynamics outlined in [HBB65]. The state x represents the relative dynamics between the pursuer and evader projected to the final stage (i.e. x includes the state transition matrix). This results in a projected-relative state-space that includes only those dynamics that are of interest to the game. Since the state represents the relative dynamics of interest in this form, the pursuer seeks to minimize the final state magnitude (i.e. capture the evader) and the evader seeks to maximize the final state magnitude (i.e. escape from the pursuer).

2.3 Measurement Dynamics

The pursuer and evader measurements are represented as

$$z_p(i) = \Theta_p(i)x(i) + v_p(i) \tag{2.2}$$

$$z_e(i) = \Theta_e(i)x(i) + v_e(i) \tag{2.3}$$

for $i = 0, 1, \ldots, N-1$. The problem parameters are the pursuer's measurement $z_p(i) \in \mathbb{R}^p$, pursuer's measurement matrix $\Theta_p(i) \in \mathbb{R}^{p\times n}$, pursuer's measurement noise $v_p(i) \in \mathbb{R}^p$, evader's measurement $z_e(i) \in \mathbb{R}^q$, evader's measurement matrix $\Theta_e(i) \in \mathbb{R}^{q\times n}$, and evader's measurement noise $v_e(i) \in \mathbb{R}^q$.

2.4 Gaussian Statistics

All random variables are modeled as Gaussian

$$x(0) \sim \mathcal{N}(\bar{x}(0), M(0))$$

$$\begin{bmatrix} w(i) \\ v_p(i) \\ v_e(i) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} W(i) & 0 & 0 \\ 0 & V_p(i) & 0 \\ 0 & 0 & V_e(i) \end{bmatrix}\right).$$

All noises are delta-correlated, and are uncorrelated with the initial state

$$E\left[\begin{bmatrix} w(i) \\ v_p(i) \\ v_e(i) \end{bmatrix}\begin{bmatrix} w^T(j) & v_p^T(j) & v_e^T(j) \end{bmatrix}\right] = \begin{bmatrix} W(i)\delta(i,j) & 0 & 0 \\ 0 & V_p(i)\delta(i,j) & 0 \\ 0 & 0 & V_e(i)\delta(i,j) \end{bmatrix}$$

where $\delta(i,j)$ is the Kronecker delta defined as

$$\delta(i,j) = \begin{cases} 1, & i = j \\ 0, & i \neq j. \end{cases}$$

2.5 Bolza-Type Performance Index

The performance index, J, includes both a weighting on the final state, $Q(N)$, and a weighting on the individual pursuer and evader control energy at each stage, $R_p(i)$ and $R_e(i)$, respectively.

$$J = \frac{1}{2}E\left\{ \|x(N)\|^2_{Q(N)} + \sum_{i=0}^{N-1} \left[\|u_p(i)\|^2_{R_p(i)} - \|u_e(i)\|^2_{R_e(i)} \right] \right\} \tag{2.4}$$

where $Q(N) \geq 0$, $R_p(i) > 0$, and $R_e(i) > 0$. This is the stochastic multistage form of the deterministic differential game performance index outlined in [HBB65]. The expectation operator, E, is required because of the stochastic nature of the problem. The individual players will seek to alter the average value of the performance index in their favor.

2.6 Problem Statement

Find the optimal control strategies, $u_p^o(i)$ and $u_e^o(i)$, resulting from

$$\min_{u_p(i)} \max_{u_e(i)} J(x(N), u_p(i), u_e(i)), \quad i = 0, 1, ..., N-1$$

subject to (2.1), (2.2), and (2.3). The admissible control strategies, U_p and U_e, are the class of Lebesgue square-summable sequences that map from the measurement domain to the control domain: $U_p(i) : Z_p(i) \rightarrow u_p(i)$ and $U_e(i) : Z_e(i) \rightarrow u_e(i)$, where $Z(i)$ is the measurement history (from stage 0 to stage i) for the respective player.

We seek a saddle point solution for this game such that

$$J(u_p^o(i), u_e(i)) \leq J(u_p^o(i), u_e^o(i)) \leq J(u_p(i), u_e^o(i)). \tag{2.5}$$

In other words, if either player unilaterally deviates from his optimal strategy then the performance index improves in favor of the other player (a Nash equilibrium for this zero-sum game). As discussed in Section 4.6, this saddle point condition is verified by showing equivalency of the minimax and maximin solutions.

This is the problem statement for the multistage, two-player, zero-sum game with unshared, imperfect information. A common interpretation of this game is that the pursuer is trying to minimize the distance between the two players (capture the evader), the evader is trying to maximize the distance between the two players (escape), while both players are subjected to energy constraints.

CHAPTER 3

The Deterministic Multistage Game

Ho, Bryson and Baron [HBB65] were the first to solve the deterministic differential game using variational techniques. Here we will present the deterministic multistage game solution for later comparison with the stochastic multistage game solution.

Consider the game described in Chapter 2 with $W(i) = 0$, $V_p(i) = 0$, $V_e(i) = 0$, $\Theta_p(i) = I$ and $\Theta_e(i) = I \; \forall i$. This is the deterministic game. As such, the expectation operator in (2.4) becomes superfluous.

3.1 Standard Derivation

Using standard optimization techniques (a good reference is [BH75]) we form the Hamiltonian with Lagrange multiplier, $\lambda(i)$, as

$$H(i) = \frac{1}{2}\big(u_p^T(i)R_p(i)u_p(i) - u_e^T(i)R_e(i)u_e(i)\big) + \lambda^T(i+1)\big(x(i) + \Gamma_p(i)u_p(i) - \Gamma_e(i)u_e(i)\big).$$

The first-order optimality conditions are

$$\begin{aligned} \frac{\partial H(i)}{\partial u_p(i)} &= 0 \\ &= u_p^T(i)R_p(i) + \lambda^T(i+1)\Gamma_p(i) \\ \Longrightarrow \quad u_p(i) &= -R_p^{-1}(i)\Gamma_p^T(i)\lambda(i+1) \end{aligned} \tag{3.1}$$

and

$$\frac{\partial H(i)}{\partial u_e(i)} = 0$$

$$= -u_e^T(i)R_e(i) - \lambda^T(i+1)\Gamma_e(i)$$

$$\Longrightarrow \; u_e(i) = -R_e^{-1}(i)\Gamma_e^T(i)\lambda(i+1). \tag{3.2}$$

Choose the Lagrange multiplier sequence such that

$$\lambda^T(i) = \frac{\partial H(i)}{\partial x(i)} \;\Longrightarrow\; \lambda(i) = \lambda(i+1) \tag{3.3}$$

and assume the Lagrange multiplier has the form

$$\lambda(i) = S(i)x(i). \tag{3.4}$$

Therefore, using (2.1)

$$\begin{aligned}\lambda(i+1) &= S(i+1)x(i+1) \\ &= S(i+1)\big(x(i) + \Gamma_p(i)u_p(i) - \Gamma_e(i)u_e(i)\big).\end{aligned} \tag{3.5}$$

Substitute (3.1) and (3.2) into (3.5)

$$\lambda(i+1) = S(i+1)\big(x(i) - \Gamma_p(i)R_p^{-1}(i)\Gamma_p^T(i)\lambda(i+1) + \Gamma_e(i)R_e^{-1}(i)\Gamma_e^T(i)\lambda(i+1)\big)$$

and solve explicitly for $\lambda(i+1)$

$$\lambda(i+1) = \big(I + S(i+1)\Gamma_p(i)R_p^{-1}(i)\Gamma_p^T(i) - S(i+1)\Gamma_e(i)R_e^{-1}(i)\Gamma_e^T(i)\big)^{-1}S(i+1)x(i). \tag{3.6}$$

According to (3.3) and (3.4), the sum of all terms multiplying $x(i)$ on the right-hand side of (3.6) is equal to $S(i)$

$$S^d(i) = \big(I + S^d(i+1)\Gamma_p(i)R_p^{-1}(i)\Gamma_p^T(i) - S^d(i+1)\Gamma_e(i)R_e^{-1}(i)\Gamma_e^T(i)\big)^{-1}S^d(i+1) \tag{3.7}$$

where the superscript d has been added as a reminder that we are dealing with the *deterministic* optimal solution. Due to our definition of the Lagrange multiplier the boundary condition for this backward-propagating sequence is $\lambda(N) = Q(N)x(N)$, which implies that

$$S^d(N) = Q(N).$$

Substituting (3.6) into (3.1) and (3.2) yields the optimal deterministic multistage game strategies

$$u_p^{d^o}(i) = -R_p^{-1}(i)\Gamma_p^T(i)S^d(i)x(i) \tag{3.8}$$

$$u_e^{d^o}(i) = -R_e^{-1}(i)\Gamma_e^T(i)S^d(i)x(i). \tag{3.9}$$

3.2 Alternate Derivation

An alternate approach to the deterministic multistage game solution is to form a new governing dynamic equation using an assumed control strategy form. This follows the approach used in [BH68] and [Beh68]. We will outline this approach here for the deterministic game so that it is familiar when applying it to the succeeding stochastic game.

Assume that the control strategies are linear functions of the state

$$u_p(i) = -C_p(i)x(i), \quad u_e(i) = C_e(i)x(i). \tag{3.10}$$

Substitute these assumed control strategy forms into the performance index (2.4) and define $\chi(i) \triangleq x(i)x^T(i)$. Since the performance index is a scalar quantity, we can use the *trace* (Tr) operator and write as follows

$$\begin{aligned} J &= \frac{1}{2}Tr\Big(Q(N)x(N)x^T(N) + \sum_{i=0}^{N-1}\big[C_p^T(i)R_p(i)C_p(i)x(i)x^T(i) - C_e^T(i)R_e(i)C_e(i)x(i)x^T(i)\big]\Big) \\ &= \frac{1}{2}Tr\Big(Q(N)\chi(N) + \sum_{i=0}^{N-1}\big[C_p^T(i)R_p(i)C_p(i)\chi(i) - C_e^T(i)R_e(i)C_e(i)\chi(i)\big]\Big). \end{aligned}$$

Now, substitute the assumed control strategy forms (3.10) into the state dynamics (2.1) to get

$$x(i+1) = \big(I - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big)x(i)$$

and subsequently form $\chi(i+1) \triangleq x(i+1)x^T(i+1)$,

$$\chi(i+1) = \big(I - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big)\chi(i)\big(I - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big)^T. \tag{3.11}$$

This is now the new governing dynamic equation. The Hamiltonian with matrix Lagrange multiplier, $\Lambda(i)$, is then

$$\begin{aligned} H(i) = &\frac{1}{2}Tr\Big(C_p^T(i)R_p(i)C_p(i)\chi(i) - C_e^T(i)R_e(i)C_e(i)\chi(i)\Big) \\ &+ Tr\Big(\Lambda^T(i+1)\big(I - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big)\chi(i)\big(I - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big)^T\Big). \end{aligned}$$

The first-order necessary conditions for optimality require that [1]

$$
\begin{aligned}
\frac{\partial H(i)}{\partial C_p(i)} &= 0 \\
&= R_p(i)C_p(i)\chi(i) + \Gamma_p^T(i)\big(\Lambda_1(i+1) + \Lambda_1^T(i+1)\big)\big(I - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big)\chi(i) \\
\frac{\partial H(i)}{\partial C_e(i)} &= 0 \\
&= -R_e(i)C_e(i)\chi(i) - \Gamma_e^T(i)\big(\Lambda_1(i+1) + \Lambda_1^T(i+1)\big)\big(I - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big)\chi(i) \\
\Lambda^T(i) &= \frac{\partial H(i)}{\partial \chi(i)} \\
&= \frac{1}{2}\big(C_p^T(i)R_p(i)C_p(i) - C_e^T(i)R_e(i)C_e(i)\big) \\
&\quad + \big(I - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big)^T\Lambda^T(i+1)\big(I - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big).
\end{aligned}
$$

Define $S^d(i+1) \triangleq \Lambda(i+1) + \Lambda^T(i+1)$ and solve for the control strategies

$$
\begin{aligned}
C_p(i) &= R_p^{-1}(i)\Gamma_p^T(i)S^d(i+1)\big(I - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big) \\
C_e(i) &= -R_e^{-1}(i)\Gamma_e^T(i)S^d(i+1)\big(I - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big).
\end{aligned}
$$

Calculate $S^d(i)$ using the above expression for $C_p(i)$ and $C_e(i)$

$$
\begin{aligned}
S^d(i) &\triangleq \Lambda(i) + \Lambda^T(i) \\
&= C_p^T(i)R_p(i)C_p(i) - C_e^T(i)R_e(i)C_e(i) \\
&\quad + \big(I - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big)^T S^d(i+1)\big(I - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big) \\
&= S^d(i+1)\big(I - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big).
\end{aligned}
$$

Therefore,

$$
\begin{aligned}
C_p(i) &= R_p^{-1}(i)\Gamma_p^T(i)S^d(i) && (3.12) \\
C_e(i) &= -R_e^{-1}(i)\Gamma_e^T(i)S^d(i) && (3.13) \\
S^d(i) &= \big(I + S^d(i+1)\Gamma_p(i)R_p^{-1}(i)\Gamma_p^T(i) - S^d(i+1)\Gamma_e(i)R_e^{-1}(i)\Gamma_e^T(i)\big)^{-1}S^d(i+1) \\
S^d(N) &= Q(N) && (3.14)
\end{aligned}
$$

[1] We use the *trace* differential properties detailed in Appendix B.2.2 to find these partial derivatives.

where, again, the superscript d has been added as a reminder that we are dealing with the *deterministic* optimal solution.

Returning to our assumed control strategy forms (3.10), we can now write the optimal deterministic multistage game strategies as

$$u_p^{do}(i) = -R_p^{-1}(i)\Gamma_p^T(i)S^d(i)x(i) \tag{3.15}$$

$$u_e^{do}(i) = -R_e^{-1}(i)\Gamma_e^T(i)S^d(i)x(i). \tag{3.16}$$

Notice that (3.14), (3.15) and (3.16) are the same as we previously derived in (3.7), (3.8) and (3.9).

3.3 Solution Existence

The above equations ((3.14), (3.15), and (3.16)) constitute a saddle point solution to the deterministic multistage game as long as the *convexity* and *no conjugate point* conditions are satisfied.

The *convexity* conditions are easily derived from $\frac{\partial^2 H(i)}{(\partial C_p(i))^2} > 0$ and $\frac{\partial^2 H(i)}{(\partial C_e(i))^2} < 0$:

$$\begin{aligned} R_p(i) + \Gamma_p^T(i)S^d(i+1)\Gamma_p(i) > 0 \\ -R_e(i) + \Gamma_e^T(i)S^d(i+1)\Gamma_e(i) < 0, \qquad \forall i. \end{aligned}$$

These conditions ensure the Hamiltonian is convex with respect to the pursuer's control and concave with respect to the evader's control.

The *no conjugate point* condition ensures that $S^d(i)$ does not have a finite escape time. We can guarantee that $S^d(i)$ is bounded if

$$\|S^d(i)\| \leq \|S^d(i+1)\|, \qquad \forall i.$$

Using (3.14) and the Cauchy-Schwarz inequality,

$$\|S^d(i)\| \leq \|\left(I + S^d(i+1)\Gamma_p(i)R_p^{-1}(i)\Gamma_p^T(i) - S^d(i+1)\Gamma_e(i)R_e^{-1}(i)\Gamma_e^T(i)\right)^{-1}\|\|S^d(i+1)\|$$

$$\implies \|\big(I + S^d(i+1)\Gamma_p(i)R_p^{-1}(i)\Gamma_p^T(i) - S^d(i+1)\Gamma_e(i)R_e^{-1}(i)\Gamma_e^T(i)\big)^{-1}\| \leq 1$$

$$\implies \|I + S^d(i+1)\Gamma_p(i)R_p^{-1}(i)\Gamma_p^T(i) - S^d(i+1)\Gamma_e(i)R_e^{-1}(i)\Gamma_e^T(i)\| \geq 1.$$

Therefore, since $S^d(N) = Q(N) \geq 0$, a sufficient condition for *no conjugate point* is

$$\Gamma_p(i)R_p^{-1}(i)\Gamma_p^T(i) \geq \Gamma_e(i)R_e^{-1}(i)\Gamma_e^T(i), \quad \forall i. \tag{3.17}$$

This is analogous to saying that the pursuer is more controllable than the evader as in [HBB65].

3.4 Optimal Performance Index

The optimal performance index value, J^{d^o}, may be found using the relationships derived above, including

$$S^d(i)\chi(i) = S^d(i+1)\big(I - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big)\chi(i).$$

Adding and subtracting this optimal game relationship to the performance index yields

$$J^{d^o} = \frac{1}{2}Tr\Big(S^d(N)\chi(N) + \sum_{i=0}^{N-1}\big[C_p^T(i)R_p(i)C_p(i)\chi(i) - C_e^T(i)R_e(i)C_e(i)\chi(i) + S^d(i)\chi(i) - S^d(i+1)\big(I - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big)\chi(i)\big]\Big).$$

Pull $S^d(0)\chi(0)$ outside of summation and substitute in for $\chi(i+1)$ (3.11), $C_p(i)$ (3.12), and $C_e(i)$ (3.13) as follows

$$\begin{aligned} J^{d^o} &= \frac{1}{2}Tr\Big(S^d(0)\chi(0) + \sum_{i=0}^{N-1}\big[C_p^T(i)R_p(i)C_p(i)\chi(i) - C_e^T(i)R_e(i)C_e(i)\chi(i) \\ &\quad + S^d(i+1)\chi(i+1) - S^d(i+1)\big(I - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big)\chi(i)\big]\Big) \\ &= \frac{1}{2}Tr\Big(S^d(0)\chi(0) + \sum_{i=0}^{N-1}\big[C_p^T(i)R_p(i)C_p(i)\chi(i) - C_e^T(i)R_e(i)C_e(i)\chi(i) \\ &\quad + S^d(i+1)\big(I - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big)\chi(i)\big(I - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big)^T \\ &\quad - S^d(i+1)\big(I - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big)\chi(i)\big]\Big) \end{aligned}$$

$$
\begin{aligned}
&= \frac{1}{2} Tr\Big(S^d(0)\chi(0) + \sum_{i=0}^{N-1} \big[C_p^T(i) R_p(i) C_p(i) \chi(i) - C_e^T(i) R_e(i) C_e(i) \chi(i) \\
&\quad + \big(-\Gamma_p(i) C_p(i) - \Gamma_e(i) C_e(i) \big)^T S^d(i+1) \big(I - \Gamma_p(i) C_p(i) - \Gamma_e(i) C_e(i) \big) \chi(i) \big] \Big) \\
&= \frac{1}{2} Tr\Big(S^d(0)\chi(0) + \sum_{i=0}^{N-1} \big[C_p^T(i) \Gamma_p^T(i) S^d(i) \chi(i) + C_e^T(i) \Gamma_e^T(i) S^d(i) \chi(i) \\
&\quad + \big(-\Gamma_p(i) C_p(i) - \Gamma_e(i) C_e(i) \big)^T S^d(i) \chi(i) \big] \Big).
\end{aligned}
$$

Canceling terms yields

$$
\begin{aligned}
J^{d^o} &= \frac{1}{2} Tr\Big(S^d(0)\chi(0) \Big) \\
&= \frac{1}{2} x^T(0) S^d(0) x(0).
\end{aligned}
\tag{3.18}
$$

CHAPTER 4

The LQG Multistage Game

Now, let's return to the general stochastic problem statement of Chapter 2. Similar to our solution approach to the deterministic game in Section 3.2, let's define each player's admissible control strategy to be an affine function of their measurement history, [1]

$$\begin{aligned} u_p(i) &= -b_p(i) - \sum_{j=0}^{i} K_p(i)_j z_p(j) \\ u_e(i) &= b_e(i) + \sum_{j=0}^{i} K_e(i)_j z_e(j) \end{aligned} \tag{4.1}$$

where $b_p \in \mathbb{R}^m$, $K_p(i)_j \in \mathbb{R}^{m\times p}$, $b_e \in \mathbb{R}^l$, $K_e(i)_j \in \mathbb{R}^{l\times q}$. Note that $K_p(i)$ and $K_e(i)$ are kernels transforming from the measurement domain to the control domain.

Willman's ([Wil69], [Wil68]) approach to this problem is to solve a one-sided optimization problem for each player using the assumed strategy form for the opposing player. This approach allows Willman to take advantage of the separation principle, resulting in the standard LQG solution for each player. However, Willman's approach is formal and lacks proof and insight as to when solutions may or may not exist. Using Willman's approach, a numerical solution is found by iterating back and forth between the optimal strategy forms in the hopes of convergence.

Instead of Willman's one-sided (iterative) approach, we propose using a two-sided (simultaneous) optimization approach in order to directly solve for the saddle point strategies. Due to the unshared, imperfect information premise of the game, and the inability to make use of the separation principle, it becomes non-obvious as to how each player will optimize the

[1] This assumption will be validated in Section 4.6.

performance index relative to their respective information set. As a result, we reformulate the problem and take the expectation prior to performing the optimization. This allows us to treat the problem as a deterministic optimization problem. As will be shown, there is a richness of information present in the optimal strategies. Furthermore, we can easily see that these general strategies become the well-known strategies for the special case of the two-player deterministic game.

4.1 Solution Derivation

Each player must optimize the performance index given their individual information set. In this sense, the problem can be viewed as a non-zero-sum game with effectively two performance indexes

$$u_p^0(i) = \underset{u_p(i)}{\operatorname{argmin}} \, (J|Z_p(i))$$

$$u_e^0(i) = \underset{u_e(i)}{\operatorname{argmax}} \, (J|Z_e(i)).$$

Viewing the zero-sum game as a non-zero-sum game in this fashion is essentially Willman's approach with his assumed strategy forms and one-sided, iterative, optimization technique. Rhodes and Luenberger [RL69] also recognized the non-zero-sum aspect of this information pattern. Even though the conditional expectation of the performance index will differ for each player, we will show that the unconditional expectation of the performance index for the zero-sum game is agreed upon and known *a priori*.

In order to transform what can be thought of as a non-zero-sum game into a zero-sum game we define an enlarged infinite-dimensional state-space (a Hilbert space). The admissible control strategy forms (4.1) are restricted to the space ℓ_2 (Lebesgue square-summable sequences).

4.1.1 Enlarged State-Space

Define an enlarged state vector, $X(i) \in \mathbb{R}^{(N+1)(n+p+q)}$, comprised of state and measurement noise histories up to and including stage i. The $X(i)$ sub-partitions are

$$X(i)_j = \begin{cases} x(j), & j \leq i \\ 0, & i < j \leq N \\ v_p(j-N-1), & N+1 \leq j \leq N+1+i \\ 0, & N+1+i < j \leq 2N+1 \\ v_e(j-2N-2), & 2N+2 \leq j \leq 2N+2+i \\ 0, & 2N+2+i < j \leq 3N+2 \end{cases}$$

for $i = 0, 1, ..., N$, $j = 0, 1, ..., 3N+2$.

Define an enlarged process noise vector, $Y(i) \in \mathbb{R}^{n+p+q}$, comprised of the state process noise at the current stage (i) and measurement noises at the next stage $(i+1)$. The $Y(i)$ sub-partitions are

$$Y(i)_j = \begin{cases} w(i), & j = 0 \\ v_p(i+1), & j = 1 \\ v_e(i+1), & j = 2 \end{cases}$$

for $i = 0, 1, ..., N-1$. Note that the enlarged process noise vector is still zero-mean and delta-correlated.

Define enlarged system, control, and measurement matrices with the following sub-partitions

$$F(i)_{j,k} = \begin{cases} I_n, & j = k \leq i \\ I_n, & j = i+1, k = i \\ I_p, & N+1 \leq j = k \leq N+1+i \\ I_q, & 2N+2 \leq j = k \leq 2N+2+i \\ 0, & otherwise \end{cases} \tag{4.2}$$

$$F(i) \in \mathbb{R}^{(N+1)(n+p+q) \times (N+1)(n+p+q)},$$

$$G_p(i)_j = \begin{cases} \Gamma_p(i), & j = i+1 \\ 0, & otherwise \end{cases} \tag{4.3}$$

$$G_p(i) \in \mathbb{R}^{(N+1)(n+p+q)\times m},$$

$$G_e(i)_j = \begin{cases} \Gamma_e(i), & j = i+1 \\ 0, & otherwise \end{cases} \tag{4.4}$$

$$G_e(i) \in \mathbb{R}^{(N+1)(n+p+q)\times l},$$

$$G_y(i)_{j,k} = \begin{cases} I_n, & j = i+1, k = 0 \\ I_p, & j = (N+1) + (i+1), k = 1 \\ I_q, & j = (2N+2) + (i+1), k = 2 \\ 0, & otherwise \end{cases} \tag{4.5}$$

$$G_y(i) \in \mathbb{R}^{(N+1)(n+p+q)\times(n+p+q)},$$

$$H_p(i)_{j,k} = \begin{cases} \Theta_p(j), & j = k \leq i \\ I_p, & j \leq i, k = N+1+j \\ 0, & otherwise \end{cases} \tag{4.6}$$

$$H_p(i) \in \mathbb{R}^{(i+1)p\times(N+1)(n+p+q)},$$

$$H_e(i)_{j,k} = \begin{cases} \Theta_e(j), & j = k \leq i \\ I_q, & j \leq i, k = 2N+2+j \\ 0, & otherwise \end{cases} \tag{4.7}$$

$$H_e(i) \in \mathbb{R}^{(i+1)q\times(N+1)(n+p+q)}$$

for $i = 0, 1, ..., N-1$. Then, with appropriately sized system, control, and measurement matrices we may write the enlarged state-space as

$$\begin{aligned} X(i+1) &= F(i)X(i) + G_p(i)u_p(i) - G_e(i)u_e(i) + G_y(i)Y(i) \\ Z_p(i) &= H_p(i)X(i) \\ Z_e(i) &= H_e(i)X(i) \end{aligned} \tag{4.8}$$

for $i = 0, 1, ..., N-1$, where $Z_p(i) \in \mathbb{R}^{(i+1)p}$ and $Z_e(i) \in \mathbb{R}^{(i+1)q}$ are each player's measurement history up to and including stage i. This state-space is functionally identically to (2.1), (2.2), and (2.3).

To illustrate these definitions let's consider how the enlarged state-space would appear at $i = 1$ for a two-stage ($N = 2$) scalar game:

$$\underbrace{\begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ v_p(0) \\ v_p(1) \\ v_p(2) \\ v_e(0) \\ v_e(1) \\ v_e(2) \end{bmatrix}}_{X(2)} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}}_{F(1)} \underbrace{\begin{bmatrix} x(0) \\ x(1) \\ 0 \\ v_p(0) \\ v_p(1) \\ 0 \\ v_e(0) \\ v_e(1) \\ 0 \end{bmatrix}}_{X(1)} + \underbrace{\begin{bmatrix} 0 \\ 0 \\ \Gamma_p(1) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{G_p(1)} u_p(1) - \underbrace{\begin{bmatrix} 0 \\ 0 \\ \Gamma_e(1) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{G_e(1)} u_e(1)$$

$$+ \underbrace{\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{G_y(1)} \underbrace{\begin{bmatrix} w(1) \\ v_p(2) \\ v_e(2) \end{bmatrix}}_{Y(1)}$$

$$\underbrace{\begin{bmatrix} z_p(0) \\ z_p(1) \end{bmatrix}}_{Z_p(1)} = \underbrace{\begin{bmatrix} \Theta_p(0) & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & \Theta_p(1) & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}}_{H_p(1)} X(1)$$

$$\underbrace{\begin{bmatrix} z_e(0) \\ z_e(1) \end{bmatrix}}_{Z_e(1)} = \underbrace{\begin{bmatrix} \Theta_e(0) & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & \Theta_e(1) & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}}_{H_e(1)} X(1).$$

4.1.2 Governing Dynamics

With the enlarged state-space now defined, the admissible control strategies (4.1) may be rewritten in matrix form as

$$\begin{aligned} u_p(i) &= -b_p(i) - K_p(i)Z_p(i) \\ &= -b_p(i) - K_p(i)H_p(i)X(i) \\ u_e(i) &= b_e(i) + K_e(i)Z_e(i) \\ &= b_e(i) + K_e(i)H_e(i)X(i) \end{aligned} \tag{4.9}$$

where $K_p(i) \in \mathbb{R}^{m\times(i+1)p}$ and $K_e(i) \in \mathbb{R}^{l\times(i+1)q}$. Using the admissible control strategy forms (4.9) in the enlarged state dynamics (4.8) yields

$$X(i+1) = \tilde{F}(i)X(i) - G_p(i)b_p(i) - G_e(i)b_e(i) + G_y(i)Y(i) \tag{4.10}$$

where

$$\tilde{F}(i) \triangleq \big(F(i) - G_p(i)K_p(i)H_p(i) - G_e(i)K_e(i)H_e(i)\big). \tag{4.11}$$

At this point we may also define the enlarged mean state, mean-square state (second moment), covariance, and process noise variance as follows

$$\begin{aligned} \bar{\mathbf{X}}(i) &\triangleq E[X(i)] \\ \mathbf{X}(i) &\triangleq E[X(i)X^T(i)] \\ \mathbf{P}(i) &\triangleq \mathbf{X}(i) - \bar{\mathbf{X}}(i)\bar{\mathbf{X}}^T(i) \\ \mathbf{Y}(i) &\triangleq E[Y(i)Y^T(i)]. \end{aligned} \tag{4.12}$$

Taking the above unconditional expectations of (4.10) results in the governing statistical dynamic equations. The mean state sequence is

$$
\begin{aligned}
\bar{\mathbf{X}}(i+1) &= \tilde{F}(i)\bar{\mathbf{X}}(i) - G_p(i)b_p(i) - G_e(i)b_e(i) \\
\bar{\mathbf{X}}(0)_j &= \begin{cases} \bar{x}(0), & j = 0 \\ 0, & otherwise \end{cases}
\end{aligned}
\tag{4.13}
$$

where $\bar{\mathbf{X}}(i) \in \mathbb{R}^{(N+1)(n+p+q)}$. The mean-square state sequence is

$$
\begin{aligned}
\mathbf{X}(i+1) = \; & \tilde{F}(i)\mathbf{X}(i)\tilde{F}^T(i) \\
& - \tilde{F}(i)\bar{\mathbf{X}}(i)b_p^T(i)G_p^T(i) - \tilde{F}(i)\bar{\mathbf{X}}(i)b_e^T(i)G_e^T(i) \\
& - G_p(i)b_p(i)\bar{\mathbf{X}}^T(i)\tilde{F}^T(i) + G_p(i)b_p(i)b_p^T(i)G_p^T(i) + G_p(i)b_p(i)b_e^T(i)G_e^T(i) \\
& - G_e(i)b_e(i)\bar{\mathbf{X}}^T(i)\tilde{F}^T(i) + G_e(i)b_e(i)b_p^T(i)G_p^T(i) + G_e(i)b_e(i)b_e^T(i)G_e^T(i) \\
& + G_y(i)\mathbf{Y}(i)G_y^T(i) \\
= \; & \tilde{F}(i)\mathbf{X}(i)\tilde{F}^T(i) - \tilde{F}(i)\bar{\mathbf{X}}(i)\bar{\mathbf{X}}^T(i)\tilde{F}^T(i) \\
& + \Big(\tilde{F}(i)\bar{\mathbf{X}}(i) - G_p(i)b_p(i) - G_e(i)b_e(i)\Big)\Big(\tilde{F}(i)\bar{\mathbf{X}}(i) - G_p(i)b_p(i) - G_e(i)b_e(i)\Big)^T \\
& + G_y(i)\mathbf{Y}(i)G_y^T(i) \\
= \; & \tilde{F}(i)\Big(\mathbf{X}(i) - \bar{\mathbf{X}}(i)\bar{\mathbf{X}}^T(i)\Big)\tilde{F}^T(i) + \bar{\mathbf{X}}(i+1)\bar{\mathbf{X}}^T(i+1) + G_y(i)\mathbf{Y}(i)G_y^T(i)
\end{aligned}
$$

and writing in terms of the covariance yields

$$
\begin{aligned}
\mathbf{X}(i+1) &= \tilde{F}(i)\mathbf{P}(i)\tilde{F}^T(i) + \bar{\mathbf{X}}(i+1)\bar{\mathbf{X}}^T(i+1) + G_y(i)\mathbf{Y}(i)G_y^T(i) \\
\mathbf{X}(0)_{j,k} &= \begin{cases} M(0) + \bar{x}(0)\bar{x}^T(0), & j = k = 0 \\ V_p(0), & j = k = N+1 \\ V_e(0), & j = k = 2N+2 \\ 0, & otherwise \end{cases}
\end{aligned}
\tag{4.14}
$$

where $\mathbf{X}(i) \in \mathbb{R}^{(N+1)(n+p+q)\times(N+1)(n+p+q)}$. It is now easy to see that the covariance sequence is

$$\mathbf{P}(i+1) = \tilde{F}(i)\mathbf{P}(i)\tilde{F}^T(i) + G_y(i)\mathbf{Y}(i)G_y^T(i)$$

$$\mathbf{P}(0)_{j,k} = \begin{cases} M(0), & j = k = 0 \\ V_p(0), & j = k = N+1 \\ V_e(0), & j = k = 2N+2 \\ 0, & otherwise \end{cases} \tag{4.15}$$

where $\mathbf{P}(i) \in \mathbb{R}^{(N+1)(n+p+q)\times(N+1)(n+p+q)}$.

As stated, the driving noises are Gaussian and the admissible strategies are affine. Therefore, the governing statistics remain Gaussian which means that the mean (4.13) and covariance (4.15) are sufficient to describe the $X(i)$ probability density function.

4.1.3 Necessary Conditions

Let's return to the performance index (2.4). First, we define an enlarged final state weighting matrix

$$\tilde{Q}(N)_{j,k} = \begin{cases} Q(N), & j = k = N \\ 0, & otherwise \end{cases}$$

where $\tilde{Q}(N) \in \mathbb{R}^{(N+1)(n+p+q)\times(N+1)(n+p+q)}$. We can now substitute the assumed affine strategies (4.9) into the performance index (2.4) using the enlarged state, $X(i)$. Furthermore, since the performance index is a scalar quantity, we can use the *trace* (Tr) operator without affecting the result. We can then use the cyclic property to rearrange the order of matrix multiplication, which then allows us to take the unconditional expectation of the performance index using the previously defined enlarged statistical parameters. This procedure yields

$$J = \frac{1}{2}E\Bigg\{X^T(N)\tilde{Q}(N)X(N) + \sum_{i=0}^{N-1}\Big[\big(b_p(i) + K_p(i)H_p(i)X(i)\big)^T R_p(i)\big(b_p(i) + K_p(i)H_p(i)X(i)\big)$$
$$- \big(b_e(i) + K_e(i)H_e(i)X(i)\big)^T R_e(i)\big(b_e(i) + K_e(i)H_e(i)X(i)\big)\Big]\Bigg\}$$

$$
\begin{aligned}
&= \frac{1}{2}Tr\Big(\tilde{Q}(N)\mathbf{P}(N) + \tilde{Q}(N)\bar{\mathbf{X}}(N)\bar{\mathbf{X}}^T(N)\Big) \\
&\quad + \frac{1}{2}\sum_{i=0}^{N-1} Tr\Big(\big(b_p(i) + K_p(i)H_p(i)\bar{\mathbf{X}}(i)\big)^T R_p(i)\big(b_p(i) + K_p(i)H_p(i)\bar{\mathbf{X}}(i)\big) \\
&\quad - \big(b_e(i) + K_e(i)H_e(i)\bar{\mathbf{X}}(i)\big)^T R_e(i)\big(b_e(i) + K_e(i)H_e(i)\bar{\mathbf{X}}(i)\big) \\
&\quad + H_p^T(i)K_p^T(i)R_p(i)K_p(i)H_p(i)\mathbf{P}(i) - H_e^T(i)K_e^T(i)R_e(i)K_e(i)H_e(i)\mathbf{P}(i)\Big)
\end{aligned}
$$

where we used the fact that $\mathbf{X}(N) = \mathbf{P}(N) + \bar{\mathbf{X}}(N)\bar{\mathbf{X}}^T(N)$. Some of these terms are scalar quantities, making the *trace* operator superfluous. We can therefore simplify the performance index expression as

$$
\begin{aligned}
J &= \frac{1}{2}\bar{\mathbf{X}}^T(N)\tilde{Q}(N)\bar{\mathbf{X}}(N) + \frac{1}{2}Tr\Big(\tilde{Q}(N)\mathbf{P}(N)\Big) \\
&\quad + \frac{1}{2}\sum_{i=0}^{N-1}\Big[\big(b_p(i) + K_p(i)H_p(i)\bar{\mathbf{X}}(i)\big)^T R_p(i)\big(b_p(i) + K_p(i)H_p(i)\bar{\mathbf{X}}(i)\big) \\
&\quad - \big(b_e(i) + K_e(i)H_e(i)\bar{\mathbf{X}}(i)\big)^T R_e(i)\big(b_e(i) + K_e(i)H_e(i)\bar{\mathbf{X}}(i)\big) \\
&\quad + Tr\Big(H_p^T(i)K_p^T(i)R_p(i)K_p(i)H_p(i)\mathbf{P}(i) - H_e^T(i)K_e^T(i)R_e(i)K_e(i)H_e(i)\mathbf{P}(i)\Big)\Big].
\end{aligned} \tag{4.16}
$$

Since the mean (4.13) and covariance (4.15) are the governing dynamics, we can append these dynamic constraints to the performance index using the column vector Lagrange multiplier $\lambda_1(i) \in \mathbb{R}^{(N+1)(n+p+q)}$ and symmetric matrix Lagrange multiplier $\Lambda_2(i) \in \mathbb{R}^{(N+1)(n+p+q)\times(N+1)(n+p+q)}$. This augmented performance index, $\bar{J}$, is

$$
\begin{aligned}
\bar{J} &= \frac{1}{2}\bar{\mathbf{X}}^T(N)\tilde{Q}(N)\bar{\mathbf{X}}(N) + \frac{1}{2}Tr\Big(\tilde{Q}(N)\mathbf{P}(N)\Big) \\
&\quad + \frac{1}{2}\sum_{i=0}^{N-1}\Big[\big(b_p(i) + K_p(i)H_p(i)\bar{\mathbf{X}}(i)\big)^T R_p(i)\big(b_p(i) + K_p(i)H_p(i)\bar{\mathbf{X}}(i)\big) \\
&\quad - \big(b_e(i) + K_e(i)H_e(i)\bar{\mathbf{X}}(i)\big)^T R_e(i)\big(b_e(i) + K_e(i)H_e(i)\bar{\mathbf{X}}(i)\big) + 2\lambda_1^T(i+1)\big(f_1(i) - \bar{\mathbf{X}}(i+1)\big) \\
&\quad + Tr\Big(H_p^T(i)K_p^T(i)R_p(i)K_p(i)H_p(i)\mathbf{P}(i) - H_e^T(i)K_e^T(i)R_e(i)K_e(i)H_e(i)\mathbf{P}(i) \\
&\quad + 2\Lambda_2^T(i+1)\big(f_2(i) - \mathbf{P}(i+1)\big)\Big)\Big]
\end{aligned}
$$

where $f_1(i)$ and $f_2(i)$ are the right-hand side of (4.13) and (4.15), respectively. We define the Hamiltonian, $H(i)$, as

$$
H(i) \triangleq \frac{1}{2}\big(b_p(i) + K_p(i)H_p(i)\bar{\mathbf{X}}(i)\big)^T R_p(i)\big(b_p(i) + K_p(i)H_p(i)\bar{\mathbf{X}}(i)\big)
$$

$$
\begin{aligned}
&- \frac{1}{2}\big(b_e(i) + K_e(i)H_e(i)\bar{\mathbf{X}}(i)\big)^T R_e(i)\big(b_e(i) + K_e(i)H_e(i)\bar{\mathbf{X}}(i)\big) + \lambda_1^T(i+1)f_1(i) \\
&+ \frac{1}{2}Tr\Big(H_p^T(i)K_p^T(i)R_p(i)K_p(i)H_p(i)\mathbf{P}(i) - H_e^T(i)K_e^T(i)R_e(i)K_e(i)H_e(i)\mathbf{P}(i) \\
&+ 2\Lambda_2^T(i+1)f_2(i)\Big)
\end{aligned} \tag{4.17}
$$

for $i = 0, 1, \ldots, N-1$. Rewriting the augmented performance index using the Hamiltonian results in

$$
\begin{aligned}
\bar{J} &= \frac{1}{2}\bar{\mathbf{X}}^T(N)\tilde{Q}(N)\bar{\mathbf{X}}(N) + \frac{1}{2}Tr\Big(\tilde{Q}(N)\mathbf{P}(N)\Big) \\
&\quad + \sum_{i=0}^{N-1}\Big[H(i) - \lambda_1^T(i+1)\bar{\mathbf{X}}(i+1) - Tr\Big(\Lambda_2^T(i+1)\mathbf{P}(i+1)\Big)\Big] \\
&= \frac{1}{2}\bar{\mathbf{X}}^T(N)\tilde{Q}(N)\bar{\mathbf{X}}(N) - \lambda_1^T(N)\bar{\mathbf{X}}(N) + Tr\Big(\frac{1}{2}\tilde{Q}(N)\mathbf{P}(N) - \Lambda_2^T(N)\mathbf{P}(N)\Big) \\
&\quad + \sum_{i=1}^{N-1}\Big[H(i) - \lambda_1^T(i)\bar{\mathbf{X}}(i) - Tr\Big(\Lambda_2^T(i)\mathbf{P}(i)\Big)\Big] + H(0).
\end{aligned}
$$

The first-order variation of $\bar{J}$ with respect to changes in the control variables ($b_p(i)$, $K_p(i)$, $b_e(i)$, $K_e(i)$) and state variables ($\bar{\mathbf{X}}(i)$, $\mathbf{P}(i)$) produces the necessary conditions for optimality. In order to write the $\bar{J}$ differential we convert all matrices to vectors using the *vectorization* (*vec*) column stacking operator

$$
\begin{aligned}
d\bar{J} &= \frac{1}{2}(d\bar{\mathbf{X}}(N))^T\tilde{Q}(N)\bar{\mathbf{X}}(N) + \frac{1}{2}\bar{\mathbf{X}}^T(N)\tilde{Q}(N)d\bar{\mathbf{X}}(N) - \lambda_1^T(N)d\bar{\mathbf{X}}(N) \\
&+ Tr\Big(\frac{1}{2}\tilde{Q}(N)d\mathbf{P}(N) - \Lambda_2^T(N)d\mathbf{P}(N)\Big) \\
&+ \sum_{i=1}^{N-1}\Bigg[\frac{\partial H(i)}{\partial b_p(i)}db_p(i) + \frac{\partial H(i)}{\partial vec(K_p(i))}dvec(K_p(i)) + \frac{\partial H(i)}{\partial b_e(i)}db_e(i) + \frac{\partial H(i)}{\partial vec(K_e(i))}dvec(K_e(i)) \\
&+ \frac{\partial H(i)}{\partial \bar{\mathbf{X}}(i)}d\bar{\mathbf{X}}(i) + \frac{\partial H(i)}{\partial vec(\mathbf{P}(i))}dvec(\mathbf{P}(i)) - \lambda_1^T(i)d\bar{\mathbf{X}}(i) - Tr\Big(\Lambda_2^T(i)d\mathbf{P}(i)\Big)\Bigg] \\
&+ \frac{\partial H(0)}{\partial b_p(0)}db_p(0) + \frac{\partial H(0)}{\partial vec(K_p(0))}dvec(K_p(0)) + \frac{\partial H(0)}{\partial b_e(0)}db_e(0) + \frac{\partial H(0)}{\partial vec(K_e(0))}dvec(K_e(0)) \\
&+ \frac{\partial H(0)}{\partial \bar{\mathbf{X}}(0)}d\bar{\mathbf{X}}(0) + \frac{\partial H(0)}{\partial vec(\mathbf{P}(0))}dvec(\mathbf{P}(0)).
\end{aligned}
$$

Replace the remaining *trace* terms using the *vectorization* operator (reference Appendix B.1.4)

$$
d\bar{J} = \big(\bar{\mathbf{X}}^T(N)\tilde{Q}(N) - \lambda_1^T(N)\big)d\bar{\mathbf{X}}(N) + vec\big(\frac{1}{2}\tilde{Q}(N) - \Lambda_2(N)\big)^T dvec(\mathbf{P}(N))
$$

$$
\begin{aligned}
&+\sum_{i=1}^{N-1}\Bigg[\frac{\partial H(i)}{\partial b_p(i)}db_p(i)+\frac{\partial H(i)}{\partial vec(K_p(i))}dvec(K_p(i))+\frac{\partial H(i)}{\partial b_e(i)}db_e(i)+\frac{\partial H(i)}{\partial vec(K_e(i))}dvec(K_e(i))\\
&+\Big(\frac{\partial H(i)}{\partial \bar{\mathbf{X}}(i)}-\lambda_1^T(i)\Big)d\bar{\mathbf{X}}(i)+\Big(\frac{\partial H(i)}{\partial vec(\mathbf{P}(i))}-vec(\Lambda_2(i))^T\Big)dvec(\mathbf{P}(i))\Bigg]\\
&+\frac{\partial H(0)}{\partial b_p(0)}db_p(0)+\frac{\partial H(0)}{\partial vec(K_p(0))}dvec(K_p(0))+\frac{\partial H(0)}{\partial b_e(0)}db_e(0)+\frac{\partial H(0)}{\partial vec(K_e(0))}dvec(K_e(0))\\
&+\frac{\partial H(0)}{\partial \bar{\mathbf{X}}(0)}d\bar{\mathbf{X}}(0)+\frac{\partial H(0)}{\partial vec(\mathbf{P}(0))}dvec(\mathbf{P}(0)).
\end{aligned}
$$

Using this expression for $d\bar{J}$ we choose the Lagrange multipliers such that

$$
\begin{aligned}
\lambda_1^T(i)&=\frac{\partial H(i)}{\partial \bar{\mathbf{X}}(i)},\quad \lambda_1^T(N)=\bar{\mathbf{X}}^T(N)\tilde{Q}(N)\\
\Lambda_2^T(i)&=\frac{\partial H(i)}{\partial \mathbf{P}(i)},\quad \Lambda_2^T(N)=\frac{1}{2}\tilde{Q}(N)
\end{aligned}
\tag{4.18}
$$

for $i=0,1,\ldots,N-1$. In addition, at an extremum, variations with respect to the control produce $d\bar{J}=0$, therefore the controls must satisfy

$$
\begin{aligned}
\frac{\partial H(i)}{\partial b_p(i)}&=0,\quad \frac{\partial H(i)}{\partial K_p(i)}=0\\
\frac{\partial H(i)}{\partial b_e(i)}&=0,\quad \frac{\partial H(i)}{\partial K_e(i)}=0
\end{aligned}
\tag{4.19}
$$

for $i=0,1,\ldots,N-1$.

We now define two matrices $S_1(i)$, $S_2(i)\in\mathbb{R}^{(N+1)(n+p+q)\times(N+1)(n+p+q)}$, such that

$$
\begin{aligned}
&\lambda_1(i)=S_1(i)\bar{\mathbf{X}}(i),\quad S_1(N)=\tilde{Q}(N)\\
&S_2(i)\triangleq\Lambda_2(i)+\Lambda_2^T(i)=2\Lambda_2(i)=2\Lambda_2^T(i),\quad S_2(N)=\tilde{Q}(N)
\end{aligned}
\tag{4.20}
$$

for $i=0,...,N-1$. Then, with the above first-order necessary conditions satisfied we can find an expression for the augmented performance index differential due to differential changes in initial conditions

$$
\begin{aligned}
d\bar{J}&=\frac{\partial H(0)}{\partial \bar{\mathbf{X}}(0)}d\bar{\mathbf{X}}(0)+\frac{\partial H(0)}{\partial vec(\mathbf{P}(0))}dvec(\mathbf{P}(0))\\
&=\lambda_1^T(0)d\bar{\mathbf{X}}(0)+vec(\Lambda_2(0))^Tdvec(\mathbf{P}(0))\\
&=\bar{\mathbf{X}}^T(0)S_1^T(0)d\bar{\mathbf{X}}(0)+Tr\Big(\Lambda_2^T(0)d\mathbf{P}(0)\Big)
\end{aligned}
$$

$$= \bar{\mathbf{X}}^T(0)S_1^T(0)d\bar{\mathbf{X}}(0) + \frac{1}{2}Tr\Big(S_2(0)d\mathbf{P}(0)\Big). \tag{4.21}$$

It is apparent that $\lambda_1(0)$ and $\Lambda_2(0)$ are the performance index gradient with respect to the mean state and covariance initial conditions. Therefore, $S_1(0)$ and $S_2(0)$ are also influence functions relating a change in $\bar{\mathbf{X}}(0)$ or $\mathbf{P}(0)$, respectively, to a change in J.

4.1.4 Optimization

As will be shown, the first-order necessary conditions ((4.18) and (4.19)) produce a *primary* and a *secondary* set of equations. The *primary* set of equations follows from the $\Lambda_2(i)$, $K_p(i)$, and $K_e(i)$ necessary conditions. The *secondary* set of equations follows from the $\lambda_1(i)$, $b_p(i)$, and $b_e(i)$ necessary conditions. The *primary* set of equations is independent of the *secondary* set of equations; as such, we shall start by deriving the *primary* set of equations.

Primary Set of Equations

The *primary* set of equations follows from the $\Lambda_2(i)$, $K_p(i)$, and $K_e(i)$ necessary conditions. We use the *trace* differential properties detailed in Appendix B.2.2 to find the following partial derivatives.

$$\begin{aligned}
\Lambda_2^T(i) &= \frac{\partial H(i)}{\partial \mathbf{P}(i)} \\
\implies \Lambda_2(i) &= \frac{1}{2}\big(H_p^T(i)K_p^T(i)R_p(i)K_p(i)H_p(i) - H_e^T(i)K_e^T(i)R_e(i)K_e(i)H_e(i)\big) \\
&\quad + \tilde{F}^T(i)\Lambda_2(i+1)\tilde{F}(i)
\end{aligned}$$

and using our definition of $S_2(i)$ in (4.20) we may write

$$\begin{aligned}
S_2(i) &= H_p^T(i)K_p^T(i)R_p(i)K_p(i)H_p(i) - H_e^T(i)K_e^T(i)R_e(i)K_e(i)H_e(i) + \tilde{F}^T(i)S_2(i+1)\tilde{F}(i) \\
S_2(N) &= \tilde{Q}(N).
\end{aligned} \tag{4.22}$$

Before proceeding we define two matrices, $L_p(i)$ and $L_e(i)$, that represent the pursuer's and evader's enlarged Kalman gain matrix, respectively. Appendix A.3 details the derivation and interpretation of these Kalman gain matrices. In particular, because these Kalman gain matrices are formed using the enlarged state-space, they produce a current state estimate, as well as a smoothed estimate of all past states using the available information through stage i,

$$\begin{aligned} L_p(i) &\triangleq \mathbf{P}(i)H_p^T(i)\big(H_p(i)\mathbf{P}(i)H_p^T(i)\big)^{-1} \\ L_e(i) &\triangleq \mathbf{P}(i)H_e^T(i)\big(H_e(i)\mathbf{P}(i)H_e^T(i)\big)^{-1}. \end{aligned} \tag{4.23}$$

Continuing with the first-order necessary conditions

$$\begin{aligned} \frac{\partial H(i)}{\partial K_p(i)} &= 0 \\ &= H_p(i)\bar{\mathbf{X}}(i)b_p^T(i)R_p(i) + H_p(i)\bar{\mathbf{X}}(i)\bar{\mathbf{X}}^T(i)H_p^T(i)K_p^T(i)R_p(i) \\ &\quad + H_p(i)\mathbf{P}(i)H_p^T(i)K_p^T(i)R_p(i) - H_p(i)\bar{\mathbf{X}}(i)\lambda_1^T(i+1)G_p(i) \\ &\quad - H_p(i)\mathbf{P}(i)\tilde{F}^T(i)S_2(i+1)G_p(i). \end{aligned}$$

Take the transpose and rearrange to arrive at

$$\begin{aligned} &R_p(i)K_p(i)H_p(i)\mathbf{P}(i)H_p^T(i) \\ &= G_p^T(i)S_2(i+1)\tilde{F}(i)\mathbf{P}(i)H_p^T(i) \\ &\quad - R_p(i)\big(b_p(i) - R_p^{-1}(i)G_p^T(i)\lambda_1(i+1) + K_p(i)H_p(i)\bar{\mathbf{X}}(i)\big)\bar{\mathbf{X}}^T(i)H_p^T(i). \end{aligned}$$

As shown below in (4.32), the second term on the right-hand side of the above equation is identically zero due to the $b_p(i)$ first-order necessary condition. Furthermore, since by definition $R_p(i) > 0$ (i.e. invertible), and, as discussed in Appendix A.3, $H_p(i)\mathbf{P}(i)H_p^T(i)$ is invertible, we can write

$$\begin{aligned} K_p(i) &= R_p^{-1}(i)G_p^T(i)S_2(i+1)\tilde{F}(i)\mathbf{P}(i)H_p^T(i)\big(H_p(i)\mathbf{P}(i)H_p^T(i)\big)^{-1} \\ &= R_p^{-1}(i)G_p^T(i)S_2(i+1)\tilde{F}(i)L_p(i). \end{aligned} \tag{4.24}$$

Similarly,

$$\frac{\partial H(i)}{\partial K_e(i)} = 0$$

$$
\begin{aligned}
= & - H_e(i)\bar{\mathbf{X}}(i)b_e^T(i)R_e(i) - H_e(i)\bar{\mathbf{X}}(i)\bar{\mathbf{X}}^T(i)H_e^T(i)K_e^T(i)R_e(i) \\
& - H_e(i)\mathbf{P}(i)H_e^T(i)K_e^T(i)R_e(i) - H_e(i)\bar{\mathbf{X}}(i)\lambda_1^T(i+1)G_e(i) \\
& - H_e(i)\mathbf{P}(i)\tilde{F}^T(i)S_2(i+1)G_e(i).
\end{aligned}
$$

Take the transpose and rearrange to arrive at

$$
\begin{aligned}
& R_e(i)K_e(i)H_e(i)\mathbf{P}(i)H_e^T(i) \\
= & - G_e^T(i)S_2(i+1)\tilde{F}(i)\mathbf{P}(i)H_e^T(i) \\
& - R_e(i)\big(b_e(i) + R_e^{-1}(i)G_e^T(i)\lambda_1(i+1) + K_e(i)H_e(i)\bar{\mathbf{X}}(i)\big)\bar{\mathbf{X}}^T(i)H_e^T(i).
\end{aligned}
$$

As shown below in (4.33), the second term on the right-hand side of the above equation is identically zero due to the $b_e(i)$ first-order necessary condition. Furthermore, since by definition $R_e(i) > 0$ (i.e. invertible), and, as discussed in Appendix A.3, $H_e(i)\mathbf{P}(i)H_e^T(i)$ is invertible, we can write

$$
\begin{aligned}
K_e(i) &= -R_e^{-1}(i)G_e^T(i)S_2(i+1)\tilde{F}(i)\mathbf{P}(i)H_e^T(i)\big(H_e(i)\mathbf{P}(i)H_e^T(i)\big)^{-1} \\
&= -R_e^{-1}(i)G_e^T(i)S_2(i+1)\tilde{F}(i)L_e(i).
\end{aligned} \tag{4.25}
$$

We can now find another expression for $S_2(i)$ that will be useful later on. Substitute (4.24) and (4.25) into (4.22) and rearrange to get

$$
\begin{aligned}
S_2(i) = & H_p^T(i)K_p^T(i)G_p^T(i)S_2(i+1)\tilde{F}(i)\big(L_p(i)H_p(i) - I\big) \\
& + H_e^T(i)K_e^T(i)G_e^T(i)S_2(i+1)\tilde{F}(i)\big(L_e(i)H_e(i) - I\big) \\
& + F^T(i)S_2(i+1)\tilde{F}(i) \\
S_2(N) = & \tilde{Q}(N).
\end{aligned} \tag{4.26}
$$

We can also rearrange (4.24) and (4.25) into a more explicit form. First, we recognize from (4.23) that $H_p(i)L_p(i) = I_{(i+1)p}$ and $H_e(i)L_e(i) = I_{(i+1)q}$. We then substitute $\tilde{F}(i)$ (4.11) into (4.24) to get

$$
\begin{aligned}
K_p(i) = & R_p^{-1}(i)G_p^T(i)S_2(i+1)\big(F(i) - G_p(i)K_p(i)H_p(i) - G_e(i)K_e(i)H_e(i)\big)L_p(i) \\
= & - R_p^{-1}(i)G_p^T(i)S_2(i+1)G_p(i)K_p(i) \\
& + R_p^{-1}(i)G_p^T(i)S_2(i+1)\big(F(i) - G_e(i)K_e(i)H_e(i)\big)L_p(i).
\end{aligned} \tag{4.27}
$$

Furthermore, we know that $G_p(i)$ is a sparse matrix, with only the $(i+1)^{th}$ sub-partition being non-zero and equal to $\Gamma_p(i)$. Therefore, we can simplify the $K_p(i)$ expression using subscripts on $S_2(i+1)$ as

$$\begin{aligned} K_p(i) = &-R_p^{-1}(i)\Gamma_p^T(i)S_2(i+1)_{i+1,i+1}\Gamma_p(i)K_p(i) \\ &+R_p^{-1}(i)\Gamma_p^T(i)S_2(i+1)_{i+1,:}\big(F(i)-G_e(i)K_e(i)H_e(i)\big)L_p(i). \end{aligned} \tag{4.28}$$

Solving explicitly for $K_p(i)$ and using the *matrix inversion lemma* (reference Appendix B.1.1) we get

$$\begin{aligned} K_p(i) = R_p^{-1}(i)\Gamma_p^T(i)\big(I+S_2(i+1)_{i+1,i+1}\Gamma_p(i)R_p^{-1}(i)\Gamma_p^T(i)\big)^{-1} \\ * S_2(i+1)_{i+1,:}\big(F(i)-G_e(i)K_e(i)H_e(i)\big)L_p(i). \end{aligned} \tag{4.29}$$

Following the same procedure we can rearrange (4.25) as

$$\begin{aligned} K_e(i) = &-R_e^{-1}(i)\Gamma_e^T(i)\big(I-S_2(i+1)_{i+1,i+1}\Gamma_e(i)R_e^{-1}(i)\Gamma_e^T(i)\big)^{-1} \\ &* S_2(i+1)_{i+1,:}\big(F(i)-G_p(i)K_p(i)H_p(i)\big)L_e(i). \end{aligned} \tag{4.30}$$

Equations (4.15), (4.26), (4.29), and (4.30) form a two-point boundary value problem where $\mathbf{P}(0)$ and $S_2(N)$ are given.

Secondary Set of Equations

The *secondary* set of equations follows from the $\lambda_1(i)$, $b_p(i)$, and $b_e(i)$ necessary conditions

$$\begin{aligned} \lambda_1^T(i) &= \frac{\partial H(i)}{\partial \bar{\mathbf{X}}(i)} \\ \Longrightarrow \lambda_1(i) &= H_p^T(i)K_p^T(i)R_p(i)b_p(i) - H_e^T(i)K_e^T(i)R_e(i)b_e(i) \\ &\quad + H_p^T(i)K_p^T(i)R_p(i)K_p(i)H_p(i)\bar{\mathbf{X}}(i) - H_e^T(i)K_e^T(i)R_e(i)K_e(i)H_e(i)\bar{\mathbf{X}}(i) \\ &\quad + \tilde{F}^T(i)\lambda_1(i+1) \end{aligned} \tag{4.31}$$

$$\begin{aligned} &\frac{\partial H(i)}{\partial b_p(i)} = 0 \\ \Longrightarrow\ & b_p(i) = R_p^{-1}(i)G_p^T(i)\lambda_1(i+1) - K_p(i)H_p(i)\bar{\mathbf{X}}(i) \end{aligned} \tag{4.32}$$

$$\frac{\partial H(i)}{\partial b_e(i)} = 0$$

$$\implies b_e(i) = -R_e^{-1}(i)G_e^T(i)\lambda_1(i+1) - K_e(i)H_e(i)\bar{\mathbf{X}}(i). \tag{4.33}$$

Substitute (4.32) and (4.33) into (4.31)

$$\begin{aligned}
\lambda_1(i) = {} & H_p^T(i)K_p^T(i)G_p^T(i)\lambda_1(i+1) - H_p^T(i)K_p^T(i)R_p(i)K_p(i)H_p(i)\bar{\mathbf{X}}(i) \\
& + H_e^T(i)K_e^T(i)G_e^T(i)\lambda_1(i+1) + H_e^T(i)K_e^T(i)R_e(i)K_e(i)H_e(i)\bar{\mathbf{X}}(i) \\
& + H_p^T(i)K_p^T(i)R_p(i)K_p(i)H_p(i)\bar{\mathbf{X}}(i) - H_e^T(i)K_e^T(i)R_e(i)K_e(i)H_e(i)\bar{\mathbf{X}}(i) \\
& + \tilde{F}^T(i)\lambda_1(i+1) \\
= {} & F^T(i)\lambda_1(i+1).
\end{aligned}$$

Using our definition of $S_1(i)$ in (4.20) we may rewrite this optimal λ_1 relationship as

$$S_1(i)\bar{\mathbf{X}}(i) = F^T(i)S_1(i+1)\bar{\mathbf{X}}(i+1). \tag{4.34}$$

Now, substitute the optimal $b_p(i)$ (4.32) and $b_e(i)$ (4.33) into the $\bar{\mathbf{X}}(i+1)$ expression (4.13)

$$\begin{aligned}
\bar{\mathbf{X}}(i+1) = {} & \tilde{F}(i)\bar{\mathbf{X}}(i) - G_p(i)R_p^{-1}(i)G_p^T(i)\lambda_1(i+1) + G_p(i)K_p(i)H_p(i)\bar{\mathbf{X}}(i) \\
& + G_e(i)R_e^{-1}(i)G_e^T(i)\lambda_1(i+1) + G_e(i)K_e(i)H_e(i)\bar{\mathbf{X}}(i) \\
= {} & F(i)\bar{\mathbf{X}}(i) - G_p(i)R_p^{-1}(i)G_p^T(i)\lambda_1(i+1) + G_e(i)R_e^{-1}(i)G_e^T(i)\lambda_1(i+1) \qquad (4.35)
\end{aligned}$$

which means that

$$\begin{aligned}
\lambda_1(i+1) &= S_1(i+1)\bar{\mathbf{X}}(i+1) \\
&= S_1(i+1)\big(F(i)\bar{\mathbf{X}}(i) - G_p(i)R_p^{-1}(i)G_p^T(i)\lambda_1(i+1) + G_e(i)R_e^{-1}(i)G_e^T(i)\lambda_1(i+1)\big) \\
&= \big(I + S_1(i+1)G_p(i)R_p^{-1}(i)G_p^T(i) - S_1(i+1)G_e(i)R_e^{-1}(i)G_e^T(i)\big)^{-1}S_1(i+1)F(i)\bar{\mathbf{X}}(i).
\end{aligned} \tag{4.36}$$

This now allows us to rewrite the right-hand side of (4.34) as

$$\begin{aligned}
& S_1(i)\bar{\mathbf{X}}(i) \\
& = F^T(i)\big(I + S_1(i+1)G_p(i)R_p^{-1}(i)G_p^T(i) - S_1(i+1)G_e(i)R_e^{-1}(i)G_e^T(i)\big)^{-1}S_1(i+1)F(i)\bar{\mathbf{X}}(i)
\end{aligned}$$

from which we can easily see the backward-propagating equation for $S_1(i)$ is

$$\begin{aligned}
&S_1(i) = F^T(i)\big(I + S_1(i+1)G_p(i)R_p^{-1}(i)G_p^T(i) - S_1(i+1)G_e(i)R_e^{-1}(i)G_e^T(i)\big)^{-1}S_1(i+1)F(i)\\
&S_1(N) = \tilde{Q}(N)
\end{aligned} \tag{4.37}$$

for $i = 0, ..., N-1$. Furthermore, using (4.36) the optimal $b_p(i)$ (4.32) and $b_e(i)$ (4.33) become

$$\begin{aligned}
b_p(i) &= K_p^d(i)\bar{\mathbf{X}}(i) - K_p(i)H_p(i)\bar{\mathbf{X}}(i)\\
K_p^d(i) &\triangleq R_p^{-1}(i)G_p^T(i)\big(I + S_1(i+1)G_p(i)R_p^{-1}(i)G_p^T(i) - S_1(i+1)G_e(i)R_e^{-1}(i)G_e^T(i)\big)^{-1}\\
&\quad * S_1(i+1)F(i)
\end{aligned} \tag{4.38}$$

and

$$\begin{aligned}
b_e(i) &= K_e^d(i)\bar{\mathbf{X}}(i) - K_e(i)H_e(i)\bar{\mathbf{X}}(i)\\
K_e^d(i) &\triangleq -R_e^{-1}(i)G_e^T(i)\big(I + S_1(i+1)G_p(i)R_p^{-1}(i)G_p^T(i) - S_1(i+1)G_e(i)R_e^{-1}(i)G_e^T(i)\big)^{-1}\\
&\quad * S_1(i+1)F(i).
\end{aligned} \tag{4.39}$$

The above equations can be greatly simplified due to the sparse structure of the enlarged matrices. Recall that per the enlarged matrix definitions at the terminal boundary condition $S_1(N)_{N,N} = Q(N)$, $G_p(N-1)_N = \Gamma_p(N-1)$, and $G_e(N-1)_N = \Gamma_e(N-1)$ with all other sub-partitions being zero. Also, $F(N-1)_{N,N-1} = I_n$. Therefore, returning to (4.37) we can, through induction starting at the terminal boundary condition, write

$$S_1(i)_{j,k} = \begin{cases} \big(I_n + S_1(i+1)_{i+1,i+1}\Gamma_p(i)R_p^{-1}(i)\Gamma_p^T(i) & \\ -S_1(i+1)_{i+1,i+1}\Gamma_e(i)R_e^{-1}(i)\Gamma_e^T(i)\big)^{-1}S_1(i+1)_{i+1,i+1}, & j = k = i \\ 0, & otherwise \end{cases} \tag{4.40}$$

$$S_1(N)_{j,k} = \begin{cases} Q(N), & j = k = N \\ 0, & otherwise. \end{cases}$$

That is, $S_1(i)$ has only one non-zero sub-partition, $S_1(i)_{i,i}$, which is exactly the same as the deterministic game $S^d(i)$ backward-propagating sequence (3.14). As a result of the sparse $S_1(i)$ matrix, returning to (4.35) and substituting $\lambda_1(i+1) = S_1(i+1)\bar{\mathbf{X}}(i+1)$ we can also write

$$\bar{\mathbf{X}}(i+1)_j = \begin{cases} \bar{\mathbf{X}}(i)_j, & j \leq i \\ \big(I_n + \Gamma_p(i)R_p^{-1}(i)\Gamma_p^T(i)S_1(i+1)_{i+1,i+1} & \\ -\Gamma_e(i)R_e^{-1}(i)\Gamma_e^T(i)S_1(i+1)_{i+1,i+1}\big)^{-1}\bar{\mathbf{X}}(i)_i, & j = i+1 \\ 0, & otherwise \end{cases}$$
$$\bar{\mathbf{X}}(0)_j = \begin{cases} \bar{x}(0), & j = 0 \\ 0, & otherwise. \end{cases} \tag{4.41}$$

Definition 4.1. *The corresponding deterministic game is the noiseless game with* $x(0) = \bar{x}(0)$.

Remark 4.1. *It is important to recognize that the mean state sequence,* $\bar{\mathbf{X}}(i)_i$, *is just the state propagation for the corresponding deterministic game. In other words, the forward-propagating mean state sequence is the corresponding deterministic game state trajectory.*

We can also further simplify the optimal $K_p^d(i)$ (4.38) and $K_e^d(i)$ (4.39)

$$\begin{aligned} K_p^d(i) &= R_p^{-1}(i)\Gamma_p^T(i)\big(I_n + S_1(i+1)_{i+1,i+1}\Gamma_p(i)R_p^{-1}(i)\Gamma_p^T(i) \\ &\quad - S_1(i+1)_{i+1,i+1}\Gamma_e(i)R_e^{-1}(i)\Gamma_e^T(i)\big)^{-1}S_1(i+1)_{i+1,i+1}\bar{\mathbf{X}}(i)_i \\ &= R_p^{-1}(i)\Gamma_p^T(i)S_1(i)_{i,i}\bar{\mathbf{X}}(i)_i \\ &= C_p(i)\bar{\mathbf{X}}(i)_i \end{aligned}$$

and

$$\begin{aligned} K_e^d(i) &= -R_e^{-1}(i)\Gamma_e^T(i)\big(I_n + S_1(i+1)_{i+1,i+1}\Gamma_p(i)R_p^{-1}(i)\Gamma_p^T(i) \\ &\quad - S_1(i+1)_{i+1,i+1}\Gamma_e(i)R_e^{-1}(i)\Gamma_e^T(i)\big)^{-1}S_1(i+1)_{i+1,i+1}\bar{\mathbf{X}}(i)_i \\ &= -R_e^{-1}(i)\Gamma_e^T(i)S_1(i)_{i,i}\bar{\mathbf{X}}(i)_i \\ &= C_e(i)\bar{\mathbf{X}}(i)_i \end{aligned}$$

where $C_p(i)$ and $C_e(i)$ are exactly the same as (3.12) and (3.13), respectively, as derived for the deterministic game. The optimal $b_p(i)$ (4.38) and $b_e(i)$ (4.39) may now be expressed as

$$b_p(i) = C_p(i)\bar{\mathbf{X}}(i)_i - K_p(i)H_p(i)\bar{\mathbf{X}}(i) \tag{4.42}$$

$$b_e(i) = C_e(i)\bar{\mathbf{X}}(i)_i - K_e(i)H_e(i)\bar{\mathbf{X}}(i). \tag{4.43}$$

We can rewrite (4.40) using $C_p(i)$ and $C_e(i)$

$$\begin{aligned} S_1(i)_{j,k} &= \begin{cases} S_1(i+1)_{i+1,i+1}\big(I_n - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big), & j = k = i \\ 0, & otherwise \end{cases} \\ S_1(N)_{j,k} &= \begin{cases} Q(N), & j = k = N \\ 0, & otherwise. \end{cases} \end{aligned} \tag{4.44}$$

We can also rewrite the enlarged mean state sequence (4.41) by starting with (4.13) and substituting in (4.42) and (4.43)

$$\bar{\mathbf{X}}(i+1)_j = \begin{cases} \bar{\mathbf{X}}(i)_j, & j \le i \\ \big(I_n - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big)\bar{\mathbf{X}}(i)_i, & j = i+1 \\ 0, & otherwise \end{cases} \tag{4.45}$$

$$\bar{\mathbf{X}}(0)_j = \begin{cases} \bar{x}(0), & j = 0 \\ 0, & otherwise. \end{cases}$$

Finally, using (4.44) we can also express $C_p(i)$ and $C_e(i)$ as

$$C_p(i) = R_p^{-1}(i)\Gamma_p^T(i)S_1(i+1)_{i+1,i+1}\big(I_n - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big) \tag{4.46}$$

$$C_e(i) = -R_e^{-1}(i)\Gamma_e^T(i)S_1(i+1)_{i+1,i+1}\big(I_n - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big). \tag{4.47}$$

For later convenience we define the *mean control strategy*, $\bar{u}_p(i)$ and $\bar{u}_e(i)$, for each player as follows. We take the unconditional expectation of the assumed control strategy forms (4.9) and then use the optimal $b_p(i)$ (4.42) and $b_e(i)$ (4.43) to arrive at

$$\bar{u}_p(i) \triangleq E[u_p(i)] = -b_p(i) - K_p(i)H_p(i)\bar{\mathbf{X}}(i) = -C_p(i)\bar{\mathbf{X}}(i)_i \tag{4.48}$$

$$\bar{u}_e(i) \triangleq E[u_e(i)] = b_e(i) + K_e(i)H_e(i)\bar{\mathbf{X}}(i) = C_e(i)\bar{\mathbf{X}}(i)_i. \tag{4.49}$$

So, on average, the players use their *corresponding deterministic game* control strategy.

4.2 Solution Summary

To solve the LQG multistage game:

1. Solve the two-point boundary value problem involving the forward-propagating sequence $\mathbf{P}(i)$ (4.15) with initial boundary condition $\mathbf{P}(0)$, and the backward-propagating sequence $S_2(i)$ (4.26) with terminal boundary condition $S_2(N)$ in conjunction with $K_p(i)$ (4.29) and $K_e(i)$ (4.30).

$$\mathbf{P}(i+1) = \tilde{F}(i)\mathbf{P}(i)\tilde{F}^T(i) + G_y(i)\mathbf{Y}(i)G_y^T(i)$$
$$\mathbf{P}(0)_{j,k} = \begin{cases} M(0), & j=k=0 \\ V_p(0), & j=k=N+1 \\ V_e(0), & j=k=2N+2 \\ 0, & otherwise \end{cases} \tag{4.15}$$

$$\begin{aligned} S_2(i) = {} & H_p^T(i)K_p^T(i)G_p^T(i)S_2(i+1)\tilde{F}(i)\big(L_p(i)H_p(i) - I\big) \\ & + H_e^T(i)K_e^T(i)G_e^T(i)S_2(i+1)\tilde{F}(i)\big(L_e(i)H_e(i) - I\big) \\ & + F^T(i)S_2(i+1)\tilde{F}(i), \quad S_2(N) = \tilde{Q}(N) \end{aligned} \tag{4.26}$$

$$\begin{aligned} K_p(i) = {} & R_p^{-1}(i)\Gamma_p^T(i)\big(I + S_2(i+1)_{i+1,i+1}\Gamma_p(i)R_p^{-1}(i)\Gamma_p^T(i)\big)^{-1} \\ & * S_2(i+1)_{i+1,:}\big(F(i) - G_e(i)K_e(i)H_e(i)\big)L_p(i) \end{aligned} \tag{4.29}$$

$$\begin{aligned} K_e(i) = {} & -R_e^{-1}(i)\Gamma_e^T(i)\big(I - S_2(i+1)_{i+1,i+1}\Gamma_e(i)R_e^{-1}(i)\Gamma_e^T(i)\big)^{-1} \\ & * S_2(i+1)_{i+1,:}\big(F(i) - G_p(i)K_p(i)H_p(i)\big)L_e(i) \end{aligned} \tag{4.30}$$

2. Solve the *corresponding deterministic game* involving the backward-propagating sequence $S^d(i)$ (3.14) with terminal boundary condition $S^d(N)$. Form $C_p(i)$ (3.12) and $C_e(i)$ (3.13).

$$\begin{aligned} S^d(i) = {} & \big(I + S^d(i+1)\Gamma_p(i)R_p^{-1}(i)\Gamma_p^T(i) \\ & - S^d(i+1)\Gamma_e(i)R_e^{-1}(i)\Gamma_e^T(i)\big)^{-1}S^d(i+1), \quad S^d(N) = Q(N) \end{aligned} \tag{3.14}$$

$$C_p(i) = R_p^{-1}(i)\Gamma_p^T(i)S^d(i) \tag{3.12}$$

$$C_e(i) = -R_e^{-1}(i)\Gamma_e^T(i)S^d(i) \tag{3.13}$$

3. Using the result of step 2, forward-propagate $\bar{\mathbf{X}}(i)$ (4.45) with initial boundary condition $\bar{\mathbf{X}}(0)$.

$$\boxed{\begin{aligned} \bar{\mathbf{X}}(i+1)_j &= \begin{cases} \bar{\mathbf{X}}(i)_j, & j \le i \\ \big(I_n - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big)\bar{\mathbf{X}}(i)_i, & j = i+1 \\ 0, & otherwise \end{cases} \\ \bar{\mathbf{X}}(0)_j &= \begin{cases} \bar{x}(0), & j = 0 \\ 0, & otherwise \end{cases} \end{aligned}} \tag{4.45}$$

4. Using the results of steps 1-3, form $b_p(i)$ (4.42) and $b_e(i)$ (4.43).

$$b_p(i) = C_p(i)\bar{\mathbf{X}}(i)_i - K_p(i)H_p(i)\bar{\mathbf{X}}(i) \tag{4.42}$$

$$b_e(i) = C_e(i)\bar{\mathbf{X}}(i)_i - K_e(i)H_e(i)\bar{\mathbf{X}}(i) \tag{4.43}$$

5. For any realization of the game, each player will play their optimal strategy (4.9) using the optimal gains from steps 1 and 4.

$$\boxed{\begin{aligned} u_p(i) &= -b_p(i) - K_p(i)Z_p(i) \\ &= -b_p(i) - K_p(i)H_p(i)X(i) \\ u_e(i) &= b_e(i) + K_e(i)Z_e(i) \\ &= b_e(i) + K_e(i)H_e(i)X(i) \end{aligned}} \tag{4.9}$$

4.3 Solving the Two-Point Boundary Value Problem

To solve the two-point boundary value problem (TPBVP) in Section 4.2, Step 1, we need to find an explicit expression for $K_p(i)$ and $K_e(i)$ as a function of $\mathbf{P}(i)$, $S_2(i)$, and the problem parameters ($R_p(i)$, $R_e(i)$, etc.). We first substitute (4.30) into (4.29) to obtain

$$\begin{aligned} K_p(i) = {} & R_p^{-1}(i)\Gamma_p^T(i)\big(I + S_2(i+1)_{i+1,i+1}\Gamma_p(i)R_p^{-1}(i)\Gamma_p^T(i)\big)^{-1}S_2(i+1)_{i+1,:}F(i)L_p(i) \\ & + R_p^{-1}(i)\Gamma_p^T(i)\big(I + S_2(i+1)_{i+1,i+1}\Gamma_p(i)R_p^{-1}(i)\Gamma_p^T(i)\big)^{-1} \\ & * S_2(i+1)_{i+1,i+1}\Gamma_e(i)R_e^{-1}(i)\Gamma_e^T(i)\big(I - S_2(i+1)_{i+1,i+1}\Gamma_e(i)R_e^{-1}(i)\Gamma_e^T(i)\big)^{-1} \end{aligned}$$

$$
\begin{aligned}
&* S_2(i+1)_{i+1,:} F(i) L_e(i) H_e(i) L_p(i) \\
&- R_p^{-1}(i)\Gamma_p^T(i)\big(I + S_2(i+1)_{i+1,i+1}\Gamma_p(i)R_p^{-1}(i)\Gamma_p^T(i)\big)^{-1} \\
&* S_2(i+1)_{i+1,i+1}\Gamma_e(i)R_e^{-1}(i)\Gamma_e^T(i)\big(I - S_2(i+1)_{i+1,i+1}\Gamma_e(i)R_e^{-1}(i)\Gamma_e^T(i)\big)^{-1} \\
&* S_2(i+1)_{i+1,i+1}\Gamma_p(i)K_p(i)H_p(i)L_e(i)H_e(i)L_p(i).
\end{aligned}
$$

We use the *matrix inversion lemma* to re-order some matrix multiplications, then multiply the first term by an identity and combine with the second term, and finally add/subtract additional $K_p(i)$ terms to get

$$
\begin{aligned}
K_p(i) = {} & R_p^{-1}(i)\Gamma_p^T(i)\big(I + S_2(i+1)_{i+1,i+1}\Gamma_p(i)R_p^{-1}(i)\Gamma_p^T(i)\big)^{-1} \\
&* \big(I - S_2(i+1)_{i+1,i+1}\Gamma_e(i)R_e^{-1}(i)\Gamma_e^T(i)\big)^{-1}\big(S_2(i+1)_{i+1,:}F(i)L_p(i) \\
&- S_2(i+1)_{i+1,i+1}\Gamma_e(i)R_e^{-1}(i)\Gamma_e^T(i)S_2(i+1)_{i+1,:}F(i)\big(I - L_e(i)H_e(i)\big)L_p(i)\big) \\
&- R_p^{-1}(i)\Gamma_p^T(i)\big(I + S_2(i+1)_{i+1,i+1}\Gamma_p(i)R_p^{-1}(i)\Gamma_p^T(i)\big)^{-1} \\
&* \big(I - S_2(i+1)_{i+1,i+1}\Gamma_e(i)R_e^{-1}(i)\Gamma_e^T(i)\big)^{-1}S_2(i+1)_{i+1,i+1}\Gamma_e(i)R_e^{-1}(i)\Gamma_e^T(i) \\
&* S_2(i+1)_{i+1,i+1}\Gamma_p(i)K_p(i)\big(H_p(i)L_e(i)H_e(i)L_p(i) - I\big) \\
&- R_p^{-1}(i)\Gamma_p^T(i)\big(I + S_2(i+1)_{i+1,i+1}\Gamma_p(i)R_p^{-1}(i)\Gamma_p^T(i)\big)^{-1} \\
&* \big(I - S_2(i+1)_{i+1,i+1}\Gamma_e(i)R_e^{-1}(i)\Gamma_e^T(i)\big)^{-1}S_2(i+1)_{i+1,i+1}\Gamma_e(i)R_e^{-1}(i)\Gamma_e^T(i) \\
&* S_2(i+1)_{i+1,i+1}\Gamma_p(i)K_p(i).
\end{aligned}
$$

Now, moving the last $K_p(i)$ term to the left-hand side, and after extensive use of the *matrix inversion lemma*, and some algebra, we find the following simplified expression

$$
\begin{aligned}
K_p(i) = {} & R_p^{-1}(i)\Gamma_p^T(i)\big(I + S_2(i+1)_{i+1,i+1}\Gamma_p(i)R_p^{-1}(i)\Gamma_p^T(i) - S_2(i+1)_{i+1,i+1}\Gamma_e(i)R_e^{-1}(i)\Gamma_e^T(i)\big)^{-1} \\
&* \big(S_2(i+1)_{i+1,:}F(i)L_p(i) - S_2(i+1)_{i+1,i+1}\Gamma_e(i)R_e^{-1}(i)\Gamma_e^T(i)S_2(i+1)_{i+1,:} \\
&* F(i)\big(I - L_e(i)H_e(i)\big)L_p(i)\big) \\
&+ R_p^{-1}(i)\Gamma_p^T(i)\big(I + S_2(i+1)_{i+1,i+1}\Gamma_p(i)R_p^{-1}(i)\Gamma_p^T(i) - S_2(i+1)_{i+1,i+1}\Gamma_e(i)R_e^{-1}(i)\Gamma_e^T(i)\big)^{-1} \\
&* S_2(i+1)_{i+1,i+1}\Gamma_e(i)R_e^{-1}(i)\Gamma_e^T(i)S_2(i+1)_{i+1,i+1}\Gamma_p(i)K_p(i)H_p(i)\big(I - L_e(i)H_e(i)\big)L_p(i)
\end{aligned}
\tag{4.50}
$$

where we have also used the fact that $H_p(i)L_p(i) = I_{(i+1)p}$ to re-write the last term. Note that (4.50) is a discrete Sylvester equation. Likewise, we could substitute (4.29) into (4.30) to find the discrete Sylvester form of $K_e(i)$.

We define the first term in (4.50) as $C_p^S(i)$, the left matrix multiplier on $K_p(i)$ in the second term as $A_p^S(i)$, and the right matrix multiplier on $K_p(i)$ in the second term as $B_p^S(i)$, where the superscript S is a reminder that these are discrete Sylvester equation matrices. We now write (4.50) using this notation as

$$K_p(i) - A_p^S(i)K_p(i)B_p^S(i) = C_p^S(i). \tag{4.51}$$

As shown in Appendix B.1.3, the solution to this equation may be written as

$$vec(K_p(i)) = [I_{(i+1)p} \otimes I_m - (B_p^S(i))^T \otimes A_p^S(i)]^{-1} vec(C_p^S(i)), \tag{4.52}$$

assuming $[I_{(i+1)p} \otimes I_m - (B_p^S(i))^T \otimes A_p^S(i)]$ is nonsingular.

We can now detail our solution process to solve this TPBVP:

1. Take an initial guess at the $S_2(i)$ sequence for $i = 0, 1, \ldots, N-1$ (use the terminal boundary condition for $S_2(N)$). We call this guess $S_2^G(i)$.

2. Solve for $L_p(i)$ and $L_e(i)$ (4.23) using $\mathbf{P}(i)$ (starting with $i = 0$ and the $\mathbf{P}(0)$ initial boundary condition).

3. Plug the result of Step 2, along with $S_2^G(i)$, into (4.52) to find $K_p(i)$.[2]

4. Plug the result of Step 3, along with $S_2^G(i)$, into (4.30) to find $K_e(i)$.

5. Plug the results of Steps 3 and 4 into (4.15) to find $\mathbf{P}(i+1)$.

6. Repeat Steps 2-5 for $i = 0, 1, \ldots, N-1$.

7. Using the results of Steps 2-6, backward-propagate $S_2(i)$ for $i = N-1, N-2, \ldots, 0$ using (4.26).

[2]Within the MATLAB environment we use the *dlyap* function to solve for $K_p(i)$.

8. Compare $S_2(i)$ from Step 7 with $S_2^G(i)$. If the two sequences are within some tolerance, then exit the loop. Otherwise, update $S_2^G(i)$ based on $S_2(i)$ and return to Step 2.

In terms of the initial guess for $S_2^G(i)$ in Step 1, we have had success using the deterministic game $S_1(i)$ sequence (4.44). Although we have demonstrated the ability to solve non-scalar, multiple stage (e.g. $N = 50$), problems using this method, it is still quite sensitive to the $S_2^G(i)$ initial guess and update in Step 8. A more robust and efficient method to solve this TPBVP is certainly desired/warranted.

The MATLAB code that we developed to solve this TPBVP using the method outlined above is included in Appendix C.

4.4 Interpretation of Optimal Strategies

Now, let's take a closer at the optimal strategies and interpret our results. Referring to Appendix A, define each player's enlarged state estimate

$$\begin{aligned}
\hat{X}_p(i) \triangleq E[X(i)|Z_p(i)] &= \ell_p(i) + L_p(i)Z_p(i) \\
&= \big(I - L_p(i)H_p(i)\big)\bar{\mathbf{X}}(i) + L_p(i)H_p(i)X(i) \\
\hat{X}_e(i) \triangleq E[X(i)|Z_e(i)] &= \ell_e(i) + L_e(i)Z_e(i) \\
&= \big(I - L_e(i)H_e(i)\big)\bar{\mathbf{X}}(i) + L_e(i)H_e(i)X(i).
\end{aligned}$$

Using (4.29) and (4.42) in (4.9) we can write the pursuer's optimal strategy as

$$\begin{aligned}
u_p^o(i) =& -C_p(i)\bar{\mathbf{X}}(i)_i + K_p(i)H_p(i)\bar{\mathbf{X}}(i) - K_p(i)H_p(i)X(i) \\
=& -C_p(i)\bar{\mathbf{X}}(i)_i - K_p(i)H_p(i)\big(X(i) - \bar{\mathbf{X}}(i)\big) \\
=& -C_p(i)\bar{\mathbf{X}}(i)_i \\
& - R_p^{-1}(i)\Gamma_p^T(i)\big(I + S_2(i+1)_{i+1,i+1}\Gamma_p(i)R_p^{-1}(i)\Gamma_p^T(i)\big)^{-1} \\
& * S_2(i+1)_{i+1,:}\big(F - G_e(i)K_e(i)H_e(i)\big)L_p(i)H_p(i)\big(X(i) - \bar{\mathbf{X}}(i)\big)
\end{aligned}$$

$$
\begin{aligned}
= & -C_p(i)\bar{\mathbf{X}}(i)_i \\
& - R_p^{-1}(i)\Gamma_p^T(i)\big(I + S_2(i+1)_{i+1,i+1}\Gamma_p(i)R_p^{-1}(i)\Gamma_p^T(i)\big)^{-1} \\
& * S_2(i+1)_{i+1,:}\big(F(i) - G_e(i)K_e(i)H_e(i)\big)\big(\hat{X}_p(i) - \bar{\mathbf{X}}(i)\big).
\end{aligned}
\tag{4.53}
$$

Likewise, using (4.30) and (4.43) in (4.9) we can write the evader's optimal strategy as

$$
\begin{aligned}
u_e^o(i) = & \, C_e(i)\bar{\mathbf{X}}(i)_i \\
& - R_e^{-1}(i)\Gamma_e^T(i)\big(I - S_2(i+1)_{i+1,i+1}\Gamma_e(i)R_e^{-1}(i)\Gamma_e^T(i)\big)^{-1} \\
& * S_2(i+1)_{i+1,:}\big(F(i) - G_p(i)K_p(i)H_p(i)\big)\big(\hat{X}_e(i) - \bar{\mathbf{X}}(i)\big).
\end{aligned}
\tag{4.54}
$$

We also define

$$
\begin{aligned}
\tilde{C}_p(i) \triangleq & \, R_p^{-1}(i)\Gamma_p^T(i)\big(I + S_2(i+1)_{i+1,i+1}\Gamma_p(i)R_p^{-1}(i)\Gamma_p^T(i)\big)^{-1} \\
& * S_2(i+1)_{i+1,:}\big(F(i) - G_e(i)K_e(i)H_e(i)\big) \\
K_p(i) = & \, \tilde{C}_p(i)L_p(i)
\end{aligned}
\tag{4.55}
$$

and

$$
\begin{aligned}
\tilde{C}_e(i) \triangleq & -R_e^{-1}(i)\Gamma_e^T(i)\big(I - S_2(i+1)_{i+1,i+1}\Gamma_e(i)R_e^{-1}(i)\Gamma_e^T(i)\big)^{-1} \\
& * S_2(i+1)_{i+1,:}\big(F(i) - G_p(i)K_p(i)H_p(i)\big) \\
K_e(i) = & \, \tilde{C}_e(i)L_e(i)
\end{aligned}
\tag{4.56}
$$

so that we can rewrite (4.53) and (4.54) in more compact form as

$$
u_p^o(i) = -C_p(i)\bar{\mathbf{X}}(i)_i - \tilde{C}_p(i)\big(\hat{X}_p(i) - \bar{\mathbf{X}}(i)\big) \tag{4.57}
$$

$$
u_e^o(i) = C_e(i)\bar{\mathbf{X}}(i)_i + \tilde{C}_e(i)\big(\hat{X}_e(i) - \bar{\mathbf{X}}(i)\big). \tag{4.58}
$$

The optimal closed loop system, using strategies (4.53) and (4.54), is shown graphically in Fig. 4.1. Again, we delineate the deterministic and stochastic aspects of each player's strategy. These forms of the optimal strategies allow for an interesting interpretation: Each player plays a certainty equivalent term assuming the *corresponding deterministic game* state trajectory ($\bar{\mathbf{X}}(i)_i$), plus an error term that is his best estimate of the actual state deviation from the *corresponding deterministic game* state trajectory.

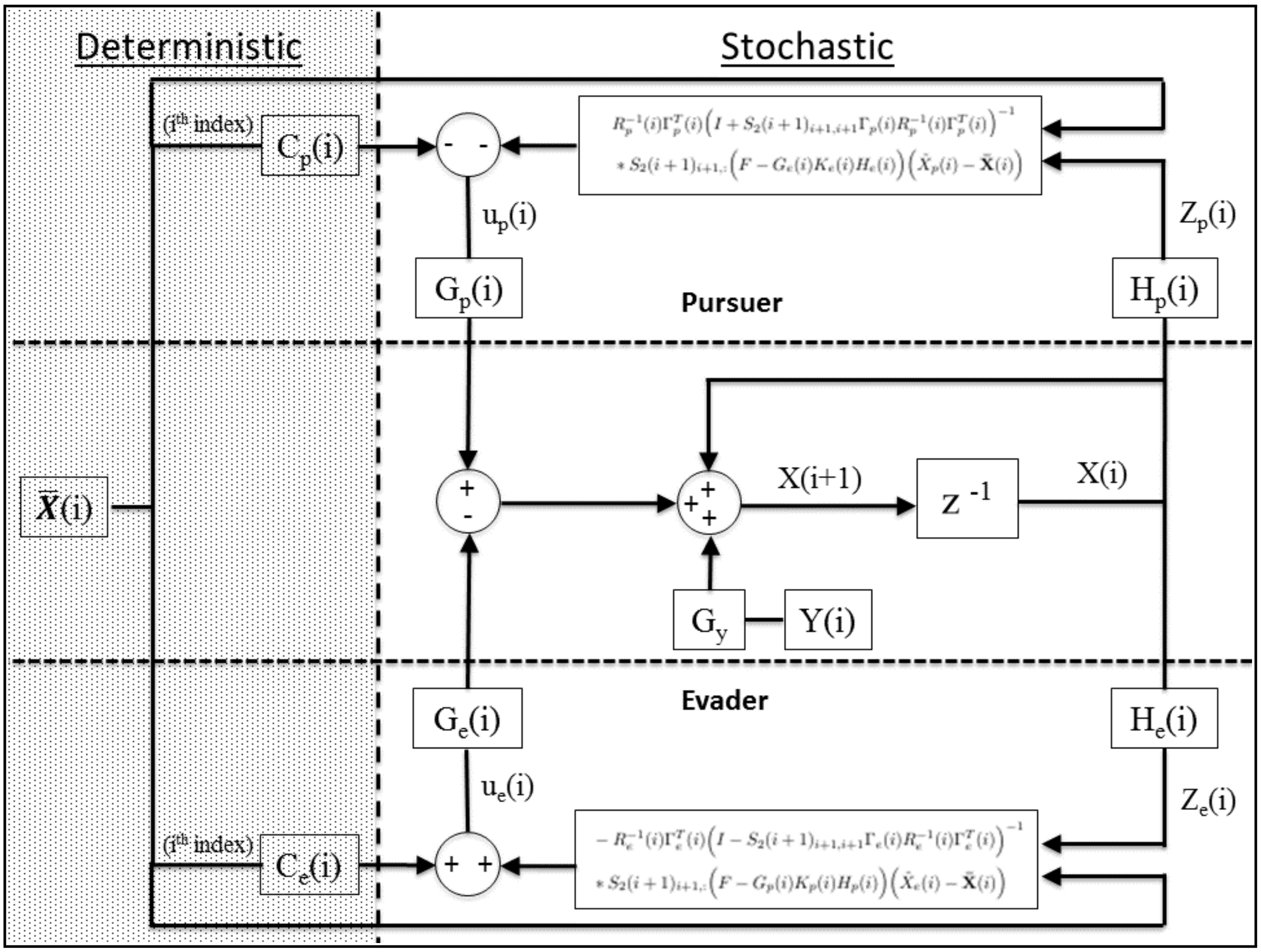

Figure 4.1: Optimal strategies for the LQG multistage game.

Finally, it is easy to see from (4.53) and (4.54) that for the special case where all noises are zero and $x(0) = \bar{x}(0)$ ($\hat{X}_p(i) = \hat{X}_e(i) = \bar{\mathbf{X}}(i)$) we have exactly the deterministic game solution (3.15) and (3.16).

Remark 4.2. *In the parlance of* common *and* private *information employed by [NGL14], [GNL14] and [Gup14], we can say that the* common *information is* $\bar{\mathbf{X}}(i)_i$ *(known by both players) and the* private *information is the unshared measurement history,* $Z_p(i)$ *(or* $\hat{X}_p(i)$*) and* $Z_e(i)$ *(or* $\hat{X}_e(i)$*). Therefore, each player plays his certainty equivalent control with the* common *information, plus an error term operating on the difference between his* private *information and the* common *information.*

Note that in general the error term appearing in these strategies could be infinite-dimensional since it includes smoothing. This naturally leads to possible sub-optimal strategies whereby we could limit the amount of smoothing that each player is allowed to incorporate in his strategy.

4.5 Solution Existence

Due to the fact that each player includes a certainty equivalent term, the solution existence requirements for the deterministic multistage game as discussed in Section 3.3 still apply. These requirements ensure that $C_p(i)$ and $C_e(i)$ exist. However, we have additional solution existence requirements for the stochastic game in order to ensure that $K_p(i)$ and $K_e(i)$ exist.

Specifically, from $\frac{\partial^2 H(i)}{(\partial K_p(i))^2} > 0$ and $\frac{\partial^2 H(i)}{(\partial K_e(i))^2} < 0$ we find that the stochastic game *convexity* conditions are

$$R_p(i) + \Gamma_p^T(i) S_2(i+1)_{i+1,i+1} \Gamma_p(i) > 0$$
$$-R_e(i) + \Gamma_e^T(i) S_2(i+1)_{i+1,i+1} \Gamma_e(i) < 0, \qquad \forall i.$$

These are analogous to the deterministic game *convexity* conditions, with the only difference being the reliance on $S_2(i+1)$ instead of $S^d(i+1)$ (or, equivalently, $S_1(i+1)$). In addition, the stochastic game *no conjugate point* condition requires that $S_2(i)$ remain bounded.

Remark 4.3. *The fact that we must satisfy additional* convexity *and* no conjugate point *conditions that are a function of* $S_2(i+1)$ *for the stochastic game is intuitive:* $S_2(i+1)$ *contains information related to the nonclassical information pattern through* $H_p(i)$ *and* $H_e(i)$*, and propagation of system noise variances through* $\mathbf{P}(i)$*.*

4.6 Saddle Point Proof

The LQG multistage game optimal control strategies must satisfy the saddle point condition (2.5). This section follows the procedure used in [BH75] to prove optimality of the two-sided deterministic game solution; here we will prove optimality of the two-sided stochastic game solution.[3]

As outlined in [BH75] for the deterministic game, the saddle point condition is validated by solving the two one-sided optimization problems formed by substituting the opposing player's optimal strategy, (4.53) or (4.54), into the performance index (2.4) and enlarged state dynamics (4.8). If the resulting strategies from these two one-sided optimization problems are equivalent to (4.53) and (4.54), then the saddle point condition is satisfied.

Theorem 4.1. *The LQG multistage game optimal control strategies as derived in* (4.53) *and* (4.54) *form a saddle point solution as defined in* (2.5)*.*

Proof. Consider the two one-sided optimization problems formed by substituting the opposing player's optimal strategy, (4.53) or (4.54), into the performance index (2.4) and enlarged state dynamics (4.8). We will use compact notation and write these strategies as

$$u_p^o(i) = -b_p^o(i) - K_p^o(i)H_p(i)X(i)$$
$$u_e^o(i) = b_e^o(i) + K_e^o(i)H_e(i)X(i)$$

where $b_p^o(i)$ (4.42), $K_p^o(i)$ (4.29), $b_e^o(i)$ (4.43), and $K_e^o(i)$ (4.30) are defined above.

[3] Behn and Ho [BH68] also used this method to validate the saddle point condition for their specific problem.

Game #1, denoted by superscript "(1)", is formed by substituting the evader's optimal two-sided strategy, $u_e^o(i)$, and solving for the pursuer's optimal one-sided strategy, $u_p^{(1)}(i)$:

$$\begin{aligned} J^{(1)} = \min_{u_p^{(1)}(i)} \frac{1}{2} E\Bigg\{ \|X^{(1)}(N)\|^2_{\tilde{Q}(N)} + \sum_{i=0}^{N-1} \Big[\|u_p^{(1)}(i)\|^2_{R_p(i)} \\ - (b_e^o(i))^T R_e(i) b_e^o(i) - (X^{(1)}(i))^T H_e^T(i)(K_e^o(i))^T R_e(i) K_e^o(i) H_e(i) X^{(1)}(i) \\ - (b_e^o(i))^T R_e(i) K_e^o(i) H_e(i) X^{(1)}(i) - (X^{(1)}(i))^T H_e^T(i)(K_e^o(i))^T R_e(i) b_e^o(i) \Big] \Bigg\} \end{aligned} \tag{4.59}$$

subject to

$$\begin{aligned} X^{(1)}(i+1) &= \big(F(i) - G_e(i)K_e^o(i)H_e(i)\big)X^{(1)}(i) + G_p(i)u_p^{(1)}(i) - G_e(i)b_e^o(i) + G_y(i)Y(i) \\ Z_p^{(1)}(i) &= H_p(i)X^{(1)}(i). \end{aligned} \tag{4.60}$$

This, however, is just a standard one-sided LQG optimization problem with the enlarged state dynamics driven by delta-correlated Gaussian noise, $Y(i)$. Therefore, we know that the separation and certainty equivalence principles apply. That is, we can solve the equivalent optimization problem subject to the *a priori* known form of the pursuer's enlarged state estimation dynamics

$$\hat{X}_p^{(1)}(i+1) = \big(F(i) - G_e(i)K_e^o(i)H_e(i)\big)\hat{X}_p^{(1)}(i) + G_p(i)u_p^{(1)}(i) - G_e(i)b_e^o(i) + \tilde{Y}_p^{(1)}(i) \tag{4.61}$$

where $\tilde{Y}_p^{(1)}(i)$ is the pursuer's delta-correlated innovations sequence.

Due to the linear Hamiltonian terms resulting from $b_e^o(i)$, we assume the Lagrange multiplier will have the form

$$\lambda_2^{(1)}(i) = S_2^{(1)}(i)\hat{X}_p^{(1)}(i) + \beta_p^{(1)}(i). \tag{4.62}$$

Performing the optimization we find the optimal control strategy as

$$\begin{aligned} u_p^{(1)}(i) = & -\tilde{C}_p^{(1)}(i)\hat{X}_p^{(1)}(i) - R_p^{-1}(i)G_p^T(i)\big(I + S_2^{(1)}(i+1)G_p(i)R_p^{-1}(i)G_p^T(i)\big)^{-1} \\ & * \big(\beta_p^{(1)}(i+1) - S_2^{(1)}(i+1)G_e(i)b_e^o(i)\big) \end{aligned} \tag{4.63}$$

where

$$\tilde{C}_p^{(1)}(i) \triangleq R_p^{-1}(i)G_p^T(i)\big(I + S_2^{(1)}(i+1)G_p(i)R_p^{-1}(i)G_p^T(i)\big)^{-1} \\ * S_2^{(1)}(i+1)\big(F(i) - G_e(i)K_e^o(i)H_e(i)\big).$$

The backward-propagating Lagrange multiplier sequences used in the optimal control strategy are

$$\begin{aligned} S_2^{(1)}(i) = & -H_e^T(i)(K_e^o(i))^T R_e(i)K_e^o(i)H_e(i) + \big(F(i) - G_e(i)K_e^o(i)H_e(i)\big)^T \\ & * S_2^{(1)}(i+1)\big(F(i) - G_p(i)\tilde{C}_p^{(1)}(i) - G_e(i)K_e^o(i)H_e(i)\big) \end{aligned} \tag{4.64}$$

and

$$\begin{aligned} \beta_p^{(1)}(i) = & -H_e^T(i)(K_e^o(i))^T R_e(i)b_e^o(i) - \big(F(i) - G_e(i)K_e^o(i)H_e(i)\big)^T \\ & * \big(I + S_2^{(1)}(i+1)G_p(i)R_p^{-1}(i)G_p^T(i)\big)^{-1}\big(S_2^{(1)}(i+1)G_e(i)b_e^o(i) - \beta_p^{(1)}(i+1)\big) \end{aligned} \tag{4.65}$$

with boundary conditions $S_2^{(1)}(N) = \tilde{Q}(N)$ and $\beta_p^{(1)}(N) = 0$.

Remark 4.4. *The pursuer's one-sided optimal control strategy, consisting of* (4.63)*,* (4.64)*, and* (4.65)*, is identical in form to Willman's result ([Wil68, Appendix A (A28), (A31), and (A32)]).*

At this point it is non-obvious as to whether or not this one-sided optimal control strategy (4.63) is equivalent to the two-sided optimal control strategy (4.53). To see this equivalency, we recall that due to the LQG structure of this one-sided optimization problem we know *a priori* that the optimal estimator will be an affine function of the measurement history. We can therefore rewrite the pursuer's enlarged state estimate (4.61) using a more general expression,

$$\begin{aligned} \hat{X}_p^{(1)}(i) &\triangleq \ell_p^{(1)}(i) + L_p^{(1)}(i)Z_p^{(1)}(i) \\ &= \ell_p^{(1)}(i) + L_p^{(1)}(i)H_p(i)X^{(1)}(i), \end{aligned}$$

where $\ell_p^{(1)}(i)$ and $L_p^{(1)}(i)$ will be defined shortly. Also, define

$$K_p^{(1)}(i) \triangleq \tilde{C}_p^{(1)}(i)L_p^{(1)}(i) \tag{4.66}$$

and

$$b_p^{(1)}(i) \triangleq \tilde{C}_p^{(1)}(i)\ell_p^{(1)}(i) + R_p^{-1}(i)G_p^T(i)\big(I + S_2^{(1)}(i+1)G_p(i)R_p^{-1}(i)G_p^T(i)\big)^{-1} \\ * \big(\beta_p^{(1)}(i+1) - S_2^{(1)}(i+1)G_e(i)b_e^o(i)\big) \tag{4.67}$$

so that we can rewrite (4.63) as

$$u_p^{(1)}(i) = -b_p^{(1)}(i) - K_p^{(1)}(i)H_p(i)X^{(1)}(i).$$

Using this form for $u_p^{(1)}(i)$ we can express the enlarged mean state sequence (as defined in (4.12)) for *Game #1* as

$$\bar{\mathbf{X}}^{(1)}(i+1) = \big(F(i) - G_p(i)K_p^{(1)}(i)H_p(i) - G_e(i)K_e^o(i)H_e(i)\big)\bar{\mathbf{X}}^{(1)}(i) - G_p(i)b_p^{(1)}(i) - G_e(i)b_e^o(i) \tag{4.68}$$

and the enlarged covariance sequence (also defined in (4.12)) for *Game #1* as

$$\mathbf{P}^{(1)}(i+1) = \big(F(i) - G_p(i)K_p^{(1)}(i)H_p(i) - G_e(i)K_e^o(i)H_e(i)\big)\mathbf{P}^{(1)}(i) \\ * \big(F(i) - G_p(i)K_p^{(1)}(i)H_p(i) - G_e(i)K_e^o(i)H_e(i)\big)^T + G_y(i)\mathbf{Y}(i)G_y^T(i). \tag{4.69}$$

Now, using these enlarged state statistics we can define the terms in the pursuer's enlarged state estimate (as shown in Appendix A.3)

$$\ell_p^{(1)}(i) = \big(I - L_p^{(1)}(i)H_p(i)\big)\bar{\mathbf{X}}^{(1)}(i) \\ L_p^{(1)}(i) = \mathbf{P}^{(1)}(i)H_p^T(i)\big(H_p(i)\mathbf{P}^{(1)}(i)H_p^T(i)\big)^{-1}. \tag{4.70}$$

A very important property that we will take advantage of is $H_p(i)L_p^{(1)}(i) = I$. This means that

$$\begin{aligned} L_p^{(1)}(i)H_p(i)\Big(\hat{X}_p^{(1)}(i) &= \ell_p^{(1)}(i) + L_p^{(1)}(i)H_p(i)X^{(1)}(i)\Big) \\ L_p^{(1)}(i)H_p(i)\hat{X}_p^{(1)}(i) &= \big(L_p^{(1)}(i)H_p(i) - L_p^{(1)}(i)H_p(i)\big)\bar{\mathbf{X}}^{(1)}(i) + L_p^{(1)}(i)H_p(i)X^{(1)}(i) \\ &= L_p^{(1)}(i)H_p(i)X^{(1)}(i) \\ &= \hat{X}_p^{(1)}(i) - \ell_p^{(1)}(i). \end{aligned}$$

To summarize, we have the following two equivalent expressions [4]

$$\hat{X}_p^{(1)}(i) = \ell_p^{(1)}(i) + L_p^{(1)}(i) H_p(i) \hat{X}_p^{(1)}(i) \tag{4.71}$$

or, rearranging,

$$\big(L_p^{(1)}(i) H_p(i) - I\big) \hat{X}_p^{(1)}(i) + \ell_p^{(1)}(i) = 0. \tag{4.72}$$

Using (4.66) and (4.71) in (4.63),

$$\begin{aligned} u_p^{(1)}(i) = &- \tilde{C}_p^{(1)}(i) \big(\ell_p^{(1)}(i) + L_p^{(1)}(i) H_p(i) \hat{X}_p^{(1)}(i)\big) \\ &- R_p^{-1}(i) G_p^T(i) \big(I + S_2^{(1)}(i+1) G_p(i) R_p^{-1}(i) G_p^T(i)\big)^{-1} \big(\beta_p^{(1)}(i+1) - S_2^{(1)}(i+1) G_e(i) b_e^o(i)\big) \\ = &- \tilde{C}_p^{(1)}(i) \bar{\mathbf{X}}^{(1)}(i) + K_p^{(1)}(i) H_p(i) \bar{\mathbf{X}}^{(1)}(i) - K_p^{(1)}(i) H_p(i) \hat{X}_p^{(1)}(i) \\ &- R_p^{-1}(i) G_p^T(i) \big(I + S_2^{(1)}(i+1) G_p(i) R_p^{-1}(i) G_p^T(i)\big)^{-1} \big(\beta_p^{(1)}(i+1) - S_2^{(1)}(i+1) G_e(i) b_e^o(i)\big) \\ = &- b_p^{(1)}(i) - K_p^{(1)}(i) H_p(i) \hat{X}_p^{(1)}(i). \end{aligned} \tag{4.73}$$

We can rewrite the assumed Lagrange multiplier form (4.62) by subtracting a zero-quantity using (4.72),

$$\begin{aligned} \lambda_2^{(1)}(i) = &\, S_2^{(1)}(i) \hat{X}_p^{(1)}(i) + \beta_p^{(1)}(i) \\ &- H_p^T(i) (K_p^{(1)}(i))^T G_p^T(i) S_2^{(1)}(i+1) \big(F(i) - G_p(i) K_p^{(1)}(i) H_p(i) - G_e(i) K_e^o(i) H_e(i)\big) \\ &* \Big(\big(L_p^{(1)}(i) H_p(i) - I\big) \hat{X}_p^{(1)}(i) + \ell_p^{(1)}(i)\Big). \end{aligned} \tag{4.74}$$

Note that we can write another expression for $K_p^{(1)}(i)$ (4.66), again using the fact that $H_p(i) L_p^{(1)}(i) = I$,

$$\begin{aligned} K_p^{(1)}(i) &= \tilde{C}_p^{(1)}(i) L_p^{(1)}(i) \\ &= R_p^{-1}(i) G_p^T(i) \big(I + S_2^{(1)}(i+1) G_p(i) R_p^{-1}(i) G_p^T(i)\big)^{-1} \\ &\quad * S_2^{(1)}(i+1) \big(F(i) - G_e(i) K_e^o(i) H_e(i)\big) L_p^{(1)}(i) \\ &= R_p^{-1}(i) G_p^T(i) S_2^{(1)}(i+1) \big(F(i) - G_p(i) K_p^{(1)}(i) H_p(i) - G_e(i) K_e^o(i) H_e(i)\big) L_p^{(1)}(i). \end{aligned} \tag{4.75}$$

[4]Reference Appendix A.3 for another derivation of this property using the definition of the conditional mean.

Using (4.73), (4.74), and (4.75) we can write a new expression for $S_2^{(1)}(i)$ (4.64). We will step through this process in order to gain an appreciation for how we arrive at the final $S_2^{(1)}(i)$ expression. The Lagrange multiplier is chosen to satisfy a Hamiltonian first-order necessary condition such that

$$\begin{aligned}
\lambda_2^{(1)}(i) &= \big(F(i) - G_e(i)K_e^o(i)H_e(i)\big)^T \lambda_2^{(1)}(i+1) \\
&\qquad - H_e^T(i)(K_e^o(i))^T R_e(i) b_e^o(i) - H_e^T(i)(K_e^o(i))^T R_e(i) K_e^o(i) H_e(i) \hat{X}_p^{(1)}(i) \\
&= \big(F(i) - G_e(i)K_e^o(i)H_e(i)\big)^T \big(S_2^{(1)}(i+1)\hat{X}_p^{(1)}(i+1) + \beta_p^{(1)}(i+1)\big) \\
&\qquad - H_e^T(i)(K_e^o(i))^T R_e(i) b_e^o(i) - H_e^T(i)(K_e^o(i))^T R_e(i) K_e^o(i) H_e(i) \hat{X}_p^{(1)}(i) \\
&= \big(F(i) - G_e(i)K_e^o(i)H_e(i)\big)^T \Big(S_2^{(1)}(i+1)\Big(\big(F(i) - G_p(i)K_p^{(1)}(i)H_p(i) \\
&\qquad - G_e(i)K_e^o(i)H_e(i)\big)\hat{X}_p^{(1)}(i) - G_p(i)b_p^{(1)}(i) - G_e(i)b_e^o(i)\Big) + \beta_p^{(1)}(i+1)\Big) \\
&\qquad - H_e^T(i)(K_e^o(i))^T R_e(i) b_e^o(i) - H_e^T(i)(K_e^o(i))^T R_e(i) K_e^o(i) H_e(i) \hat{X}_p^{(1)}(i).
\end{aligned} \tag{4.76}$$

Equating (4.76) to (4.74) and collecting all terms multiplying $\hat{X}_p^{(1)}(i)$, using (4.75) in the process, we find that

$$\begin{aligned}
S_2^{(1)}(i) &= \big(F(i) - G_e(i)K_e^o(i)H_e(i)\big)^T S_2^{(1)}(i+1)\big(F(i) - G_p(i)K_p^{(1)}(i)H_p(i) - G_e(i)K_e^o(i)H_e(i)\big) \\
&\qquad - H_e^T(i)(K_e^o(i))^T R_e(i) K_e^o(i) H_e(i) \\
&\qquad + H_p^T(i)(K_p^{(1)}(i))^T G_p^T(i) S_2^{(1)}(i+1)\big(F(i) - G_p(i)K_p^{(1)}(i)H_p(i) - G_e(i)K_e^o(i)H_e(i)\big) \\
&\qquad * \big(L_p^{(1)}(i)H_p(i) - I\big) \\
&= H_p^T(i)(K_p^{(1)}(i))^T G_p^T(i) S_2^{(1)}(i+1)\big(F(i) - G_p(i)K_p^{(1)}(i)H_p(i) - G_e(i)K_e^o(i)H_e(i)\big) \\
&\qquad * L_p^{(1)}(i)H_p(i) - H_e^T(i)(K_e^o(i))^T R_e(i) K_e^o(i) H_e(i) \\
&\qquad + \big(F(i) - G_p(i)K_p^{(1)}(i)H_p(i) - G_e(i)K_e^o(i)H_e(i)\big)^T S_2^{(1)}(i+1) \\
&\qquad * \big(F(i) - G_p(i)K_p^{(1)}(i)H_p(i) - G_e(i)K_e^o(i)H_e(i)\big) \\
&= H_p^T(i)(K_p^{(1)}(i))^T R_p(i) K_p^{(1)}(i) H_p(i) - H_e^T(i)(K_e^o(i))^T R_e(i) K_e^o(i) H_e(i) \\
&\qquad + \big(F(i) - G_p(i)K_p^{(1)}(i)H_p(i) - G_e(i)K_e^o(i)H_e(i)\big)^T S_2^{(1)}(i+1) \\
&\qquad * \big(F(i) - G_p(i)K_p^{(1)}(i)H_p(i) - G_e(i)K_e^o(i)H_e(i)\big).
\end{aligned} \tag{4.77}$$

Again, equating (4.76) to (4.74) and this time collecting all other terms not multiplying

$\hat{X}_p^{(1)}(i)$ we find that

$$
\begin{aligned}
\beta_p^{(1)}(i) = &\big(F(i) - G_e(i)K_e^o(i)H_e(i)\big)^T \Big(S_2^{(1)}(i+1)\big(-G_p(i)b_p^{(1)}(i) - G_e(i)b_e^o(i)\big) + \beta_p^{(1)}(i+1)\Big) \\
&- H_e^T(i)(K_e^o(i))^T R_e(i)b_e^o(i) + H_p^T(i)(K_p^{(1)}(i))^T G_p^T(i)S_2^{(1)}(i+1) \\
&* \big(F(i) - G_p(i)K_p^{(1)}(i)H_p(i) - G_e(i)K_e^o(i)H_e(i)\big)\ell_p^{(1)}(i).
\end{aligned}
\tag{4.78}
$$

Remark 4.5. *Before proceeding, we pause to note that we've now derived a Game #1 two-point boundary value problem with governing equations* $K_p^{(1)}(i)$, $\mathbf{P}^{(1)}(i+1)$, *and* $S_2^{(1)}(i)$ *((4.66), (4.69) and (4.77), respectively). These Game #1 governing equations are identical to those we derived using the primary set of equations for the two-sided solution in Section 4.1.4 ((4.29), (4.14) and (4.22)). Furthermore, the Game #1 initial boundary condition is* $\mathbf{P}^{(1)}(0) = \mathbf{P}(0)$ *and the terminal boundary condition is* $S_2^{(1)}(N) = S_2(N)$. *Therefore, since the governing equations and boundary conditions are identical we can conclude that the parameters derived via one-sided optimization are identical to those derived via two-sided optimization:*

$$
\begin{aligned}
&K_p^{(1)}(i) = K_p^o(i), \\
&\mathbf{P}^{(1)}(i) = \mathbf{P}(i) \qquad \Longrightarrow \; L_p^{(1)}(i) = L_p(i), \\
&S_2^{(1)}(i) = S_2(i).
\end{aligned}
\tag{4.79}
$$

We will use these interim results as we proceed with the remainder of the proof.

It now remains to determine how $b_p^{(1)}(i)$ relates to $b_p^o(i)$. Returning to (4.78) and using the interim results above to drop the superscript "(1)" where appropriate, we substitute for

$\ell_p^{(1)}(i)$ using (4.70) and do some rearranging

$$\begin{aligned}
\beta_p^{(1)}(i) &= \big(F(i) - G_e(i)K_e^o(i)H_e(i)\big)^T S_2(i+1)\big(-G_p(i)b_p^{(1)}(i) - G_e(i)b_e^o(i)\big) \\
&\quad + \big(F(i) - G_e(i)K_e^o(i)H_e(i)\big)^T \beta_p^{(1)}(i+1) \\
&\quad - H_e^T(i)(K_e^o(i))^T R_e(i)b_e^o(i) + H_p^T(i)(K_p^o(i))^T G_p^T(i)S_2(i+1) \\
&\quad * \big(F(i) - G_p(i)K_p^o(i)H_p(i) - G_e(i)K_e^o(i)H_e(i)\big)\big(I - L_p(i)H_p(i)\big)\bar{\mathbf{X}}^{(1)}(i) \\
&= \big(F(i) - G_e(i)K_e^o(i)H_e(i)\big)^T S_2(i+1)\big(-G_p(i)b_p^{(1)}(i) - G_e(i)b_e^o(i)\big) \\
&\quad + \big(F(i) - G_e(i)K_e^o(i)H_e(i)\big)^T \beta_p^{(1)}(i+1) \\
&\quad - H_e^T(i)(K_e^o(i))^T R_e(i)b_e^o(i) + H_p^T(i)(K_p^o(i))^T G_p^T(i)S_2(i+1) \\
&\quad * \big(F(i) - G_p(i)K_p^o(i)H_p(i) - G_e(i)K_e^o(i)H_e(i)\big)\bar{\mathbf{X}}^{(1)}(i) \\
&\quad - H_p^T(i)(K_p^o(i))^T R_p(i)K_p^o(i)H_p(i)\bar{\mathbf{X}}^{(1)}(i).
\end{aligned}$$

Using $\bar{\mathbf{X}}^{(1)}(i+1)$ (4.68) we can rewrite the first term as follows

$$\begin{aligned}
\beta_p^{(1)}(i) &= \big(F(i) - G_e(i)K_e^o(i)H_e(i)\big)^T S_2(i+1)\Big(\bar{\mathbf{X}}^{(1)}(i+1) \\
&\quad - \big(F(i) - G_p(i)K_p^o(i)H_p(i) - G_e(i)K_e^o(i)H_e(i)\big)\bar{\mathbf{X}}^{(1)}(i)\Big) \\
&\quad + \big(F(i) - G_e(i)K_e^o(i)H_e(i)\big)^T \beta_p^{(1)}(i+1) \\
&\quad - H_e^T(i)(K_e^o(i))^T R_e(i)b_e^o(i) + H_p^T(i)(K_p^o(i))^T G_p^T(i)S_2(i+1) \\
&\quad * \big(F(i) - G_p(i)K_p^o(i)H_p(i) - G_e(i)K_e^o(i)H_e(i)\big)\bar{\mathbf{X}}^{(1)}(i) \\
&\quad - H_p^T(i)(K_p^o(i))^T R_p(i)K_p^o(i)H_p(i)\bar{\mathbf{X}}^{(1)}(i) \\
&= - H_p^T(i)(K_p^o(i))^T R_p(i)K_p^o(i)H_p(i)\bar{\mathbf{X}}^{(1)}(i) \\
&\quad - \big(F(i) - G_p(i)K_p^o(i)H_p(i) - G_e(i)K_e^o(i)H_e(i)\big)^T S_2(i+1) \\
&\quad * \big(F(i) - G_p(i)K_p^o(i)H_p(i) - G_e(i)K_e^o(i)H_e(i)\big)\bar{\mathbf{X}}^{(1)}(i) \\
&\quad - H_e^T(i)(K_e^o(i))^T R_e(i)b_e^o(i) \\
&\quad + \big(F(i) - G_e(i)K_e^o(i)H_e(i)\big)^T \big(S_2(i+1)\bar{\mathbf{X}}^{(1)}(i+1) + \beta_p^{(1)}(i+1)\big).
\end{aligned}$$

And now, using $S_2(i)$ (4.77), we can rewrite the first two terms and rearrange to get the following expression

$$\beta_p^{(1)}(i) = \big(-S_2(i) - H_e^T(i)(K_e^o(i))^T R_e(i)K_e^o(i)H_e(i)\big)\bar{\mathbf{X}}^{(1)}(i)$$

$$
\begin{aligned}
& - H_e^T(i)(K_e^o(i))^T R_e(i) b_e^o(i) \\
& + \big(F(i) - G_e(i)K_e^o(i)H_e(i)\big)^T \big(S_2(i+1)\bar{\mathbf{X}}^{(1)}(i+1) + \beta_p^{(1)}(i+1)\big) \\
S_2(i)\bar{\mathbf{X}}^{(1)}(i) + \beta_p^{(1)}(i) = & - H_e^T(i)(K_e^o(i))^T R_e(i) K_e^o(i) H_e(i)\bar{\mathbf{X}}^{(1)}(i) - H_e^T(i)(K_e^o(i))^T R_e(i) b_e^o(i) \\
& + \big(F(i) - G_e(i)K_e^o(i)H_e(i)\big)^T \big(S_2(i+1)\bar{\mathbf{X}}^{(1)}(i+1) + \beta_p^{(1)}(i+1)\big).
\end{aligned}
$$

Define

$$
\begin{aligned}
\lambda_1^{(1)}(i) &\triangleq S_2(i)\bar{\mathbf{X}}^{(1)}(i) + \beta_p^{(1)}(i), \\
\lambda_1^{(1)}(N) &= S_2(N)\bar{\mathbf{X}}^{(1)}(N) + \beta_p^{(1)}(N) \\
&= \tilde{Q}(N)\bar{\mathbf{X}}^{(1)}(N)
\end{aligned}
\tag{4.80}
$$

so that we can now write

$$
\begin{aligned}
\lambda_1^{(1)}(i) = & - H_e^T(i)(K_e^o(i))^T R_e(i) K_e^o(i) H_e(i)\bar{\mathbf{X}}^{(1)}(i) - H_e^T(i)(K_e^o(i))^T R_e(i) b_e^o(i) \\
& + \big(F(i) - G_e(i)K_e^o(i)H_e(i)\big)^T \lambda_1^{(1)}(i+1).
\end{aligned}
\tag{4.81}
$$

We now return to $b_p^{(1)}(i)$ (4.67) and we substitute for $\ell_p^{(1)}(i)$ using (4.70) and do some rearranging

$$
\begin{aligned}
b_p^{(1)}(i) = & - K_p^o(i)H_p(i)\bar{\mathbf{X}}^{(1)}(i) \\
& + R_p^{-1}(i)G_p^T(i)\big(I + S_2(i+1)G_p(i)R_p^{-1}(i)G_p^T(i)\big)^{-1} \\
& * S_2(i+1)\big(F(i) - G_e(i)K_e^o(i)H_e(i)\big)\bar{\mathbf{X}}^{(1)}(i) \\
& + R_p^{-1}(i)G_p^T(i)\big(I + S_2(i+1)G_p(i)R_p^{-1}(i)G_p^T(i)\big)^{-1}\big(\beta_p^{(1)}(i+1) - S_2(i+1)G_e(i)b_e^o(i)\big).
\end{aligned}
$$

Add and subtract a term in order to get the following form

$$
\begin{aligned}
b_p^{(1)}(i) = & - K_p^o(i)H_p(i)\bar{\mathbf{X}}^{(1)}(i) \\
& + R_p^{-1}(i)G_p^T(i)\big(I + S_2(i+1)G_p(i)R_p^{-1}(i)G_p^T(i)\big)^{-1} S_2(i+1)G_p(i)K_p^o(i)H_p(i)\bar{\mathbf{X}}^{(1)}(i) \\
& + R_p^{-1}(i)G_p^T(i)\big(I + S_2(i+1)G_p(i)R_p^{-1}(i)G_p^T(i)\big)^{-1} \\
& * S_2(i+1)\big(F(i) - G_p(i)K_p^o(i)H_p(i) - G_e(i)K_e^o(i)H_e(i)\big)\bar{\mathbf{X}}^{(1)}(i) \\
& + R_p^{-1}(i)G_p^T(i)\big(I + S_2(i+1)G_p(i)R_p^{-1}(i)G_p^T(i)\big)^{-1}\big(\beta_p^{(1)}(i+1) - S_2(i+1)G_e(i)b_e^o(i)\big).
\end{aligned}
$$

Using $\bar{\mathbf{X}}^{(1)}(i+1)$ (4.68) we can rewrite the third term and rearrange

$$\begin{aligned}
b_p^{(1)}(i) = & -K_p^o(i)H_p(i)\bar{\mathbf{X}}^{(1)}(i) \\
& + R_p^{-1}(i)G_p^T(i)\big(I + S_2(i+1)G_p(i)R_p^{-1}(i)G_p^T(i)\big)^{-1}S_2(i+1)G_p(i)K_p^o(i)H_p(i)\bar{\mathbf{X}}^{(1)}(i) \\
& + R_p^{-1}(i)G_p^T(i)\big(I + S_2(i+1)G_p(i)R_p^{-1}(i)G_p^T(i)\big)^{-1} \\
& * S_2(i+1)\big(\bar{\mathbf{X}}^{(1)}(i+1) + G_p(i)b_p^{(1)}(i) + G_e(i)b_e^o(i)\big) \\
& + R_p^{-1}(i)G_p^T(i)\big(I + S_2(i+1)G_p(i)R_p^{-1}(i)G_p^T(i)\big)^{-1}\big(\beta_p^{(1)}(i+1) - S_2(i+1)G_e(i)b_e^o(i)\big) \\
= & -K_p^o(i)H_p(i)\bar{\mathbf{X}}^{(1)}(i) \\
& + R_p^{-1}(i)G_p^T(i)\big(I + S_2(i+1)G_p(i)R_p^{-1}(i)G_p^T(i)\big)^{-1} \\
& * S_2(i+1)G_p(i)\big(b_p^{(1)}(i) + K_p^o(i)H_p(i)\bar{\mathbf{X}}^{(1)}(i)\big) \\
& + R_p^{-1}(i)G_p^T(i)\big(I + S_2(i+1)G_p(i)R_p^{-1}(i)G_p^T(i)\big)^{-1}\big(S_2(i+1)\bar{\mathbf{X}}^{(1)}(i+1) + \beta_p^{(1)}(i+1)\big) \\
= & -K_p^o(i)H_p(i)\bar{\mathbf{X}}^{(1)}(i) \\
& + R_p^{-1}(i)G_p^T(i)\big(I + S_2(i+1)G_p(i)R_p^{-1}(i)G_p^T(i)\big)^{-1} \\
& * S_2(i+1)G_p(i)\big(b_p^{(1)}(i) + K_p^o(i)H_p(i)\bar{\mathbf{X}}^{(1)}(i)\big) \\
& + R_p^{-1}(i)G_p^T(i)\big(I + S_2(i+1)G_p(i)R_p^{-1}(i)G_p^T(i)\big)^{-1}\lambda_1^{(1)}(i+1).
\end{aligned}$$

Solving explicitly for $b_p^{(1)}(i)$, and using the *matrix inversion lemma* in the process (Appendix B.1.1), we obtain a greatly simplified expression

$$b_p^{(1)}(i) = R_p^{-1}(i)G_p^T(i)\lambda_1^{(1)}(i+1) - K_p^o(i)H_p(i)\bar{\mathbf{X}}^{(1)}(i). \tag{4.82}$$

Finally, let's rewrite $\lambda_1^{(1)}(i)$ (4.81). We add and subtract a term and then use (4.82) to arrive at

$$\begin{aligned}
\lambda_1^{(1)}(i) = & -H_e^T(i)(K_e^o(i))^T R_e(i)K_e^o(i)H_e(i)\bar{\mathbf{X}}^{(1)}(i) - H_e^T(i)(K_e^o(i))^T R_e(i)b_e^o(i) \\
& + \big(F(i) - G_p(i)K_p^o(i)H_p(i) - G_e(i)K_e^o(i)H_e(i)\big)^T\lambda_1^{(1)}(i+1) \\
& + \big(G_p(i)K_p^o(i)H_p(i)\big)^T\lambda_1^{(1)}(i+1) \\
= & -H_e^T(i)(K_e^o(i))^T R_e(i)K_e^o(i)H_e(i)\bar{\mathbf{X}}^{(1)}(i) - H_e^T(i)(K_e^o(i))^T R_e(i)b_e^o(i) \\
& + \big(F(i) - G_p(i)K_p^o(i)H_p(i) - G_e(i)K_e^o(i)H_e(i)\big)^T\lambda_1^{(1)}(i+1)
\end{aligned}$$

$$
\begin{aligned}
&+ H_p^T(i)(K_p^o(i))^T\big(R_p(i)b_p^{(1)}(i) + R_p(i)K_p^o(i)H_p(i)\bar{\mathbf{X}}^{(1)}(i)\big)\\
&= H_p^T(i)(K_p^o(i))^T R_p(i)b_p^{(1)}(i) - H_e^T(i)(K_e^o(i))^T R_e(i)b_e^o(i)\\
&\quad + H_p^T(i)(K_p^o(i))^T R_p(i)K_p^o(i)H_p(i)\bar{\mathbf{X}}^{(1)}(i) - H_e^T(i)(K_e^o(i))^T R_e(i)K_e^o(i)H_e(i)\bar{\mathbf{X}}^{(1)}(i)\\
&\quad + \big(F(i) - G_p(i)K_p^o(i)H_p(i) - G_e(i)K_e^o(i)H_e(i)\big)^T\lambda_1^{(1)}(i+1).
\end{aligned}
\tag{4.83}
$$

Remark 4.6. *We have now derived a second Game #1 two-point boundary value problem with governing equations $b_p^{(1)}(i)$, $\bar{\mathbf{X}}^{(1)}(i+1)$, and $\lambda_1^{(1)}(i)$ ((4.82), (4.68) and (4.83), respectively). Given the results of Remark 4.5, these Game #1 governing equations are identical to those we derived using the secondary set of equations for the two-sided solution in Section 4.1.4 ((4.32), (4.13) and (4.31)). Furthermore, the Game #1 initial boundary condition is $\bar{\mathbf{X}}^{(1)}(0) = \bar{\mathbf{X}}(0)$ and the terminal boundary condition is $\lambda_1^{(1)}(N) = \lambda_1(N)$. Therefore, since the governing equations and boundary conditions are identical we can conclude that the parameters derived via one-sided optimization are identical to those derived via two-sided optimization:*

$$
\begin{aligned}
b_p^{(1)}(i) &= b_p^o(i),\\
\bar{\mathbf{X}}^{(1)}(i) &= \bar{\mathbf{X}}(i) \qquad \Longrightarrow \; \ell_p^{(1)}(i) = \ell_p(i),\\
\lambda_1^{(1)}(i) &= \lambda_1(i).
\end{aligned}
\tag{4.84}
$$

In summary, for *Game #1* we have shown that the pursuer's one-sided optimal strategy is equivalent to the two-sided optimal strategy. That is, $b_p^{(1)}(i) = b_p^o(i)$ and $K_p^{(1)}(i) = K_p^o(i)$ which means that $u_p^{(1)}(i) = u_p^o(i)$.

In a similar manner we can construct *Game #2*, denoted with superscript "(2)", by substituting the pursuer's optimal two-sided strategy, $u_p^o(i)$, and solving for the evader's optimal one-sided strategy, $u_e^{(2)}(i)$:

$$J^{(2)} = \max_{u_e^{(2)}(i)} \frac{1}{2} E\Bigg\{ \|X^{(2)}(N)\|^2_{\tilde{Q}(N)} + \sum_{i=0}^{N-1} \Big[-\|u_e^{(2)}(i)\|^2_{R_e(i)}$$
$$(b_p^o(i))^T R_p(i) b_p^o(i) + (X^{(2)}(i))^T H_p^T(i)(K_p^o(i))^T R_p(i) K_p^o(i) H_p(i) X^{(2)}(i)$$
$$+ (b_p^o(i))^T R_p(i) K_p^o(i) H_p(i) X^{(2)}(i) + (X^{(2)}(i))^T H_p^T(i)(K_p^o(i))^T R_p(i) b_p^o(i) \Big] \Bigg\} \tag{4.85}$$

subject to

$$X^{(2)}(i+1) = \big(F(i) - G_p(i)K_p^o(i)H_p(i)\big)X^{(2)}(i) - G_p(i)b_p^o(i) - G_e(i)u_e^{(2)}(i) + G_y(i)Y(i)$$
$$Z_e^{(2)}(i) = H_e(i)X^{(2)}(i). \tag{4.86}$$

As before with *Game #1*, this is just a standard one-sided LQG optimization problem with the enlarged state dynamics driven by delta-correlated Gaussian noise, $Y(i)$.

Remark 4.7. *Due to the symmetry of structure between* Game #1 *and* Game #2*, and based on Remark 4.4, we can deduce that the evader's one-sided optimal control strategy resulting from direct optimization of* (4.85) *will be identical in form to Willman's result ([Wil68, Appendix A]).*

Following the same algebraic manipulations as we did for *Game #1* (with some sign changes) it can be shown that the corresponding *Game #2* parameters, $b_e^{(2)}(i)$ and $K_e^{(2)}(i)$, form two separate two-point boundary value problems with identical boundary conditions and governing equations as compared to the primary and secondary equations of Section 4.1.4. Therefore, due to the symmetry of structure between *Game #1* and *Game #2* we prove by deduction that the evader's one-sided optimal strategy is equivalent to the two-sided optimal strategy: $u_e^{(2)}(i) = u_e^o(i)$.

■

In conclusion, the optimal control strategies (4.53) and (4.54) form a saddle point solution for the zero-sum LQG multistage game with nonclassical information pattern. Neither player can unilaterally change his control strategy for a better outcome; this is the definition of a saddle point equilibrium or, equivalently for this class of games, the Nash equilibrium.

In this saddle point proof we made no *a priori* assumption regarding the form of $u_p^{(1)}(i)$ or $u_e^{(2)}(i)$. As such, the fact that these end up being affine strategies validates that our affine strategy assumption in (4.9) is appropriate, as we have now proven that an affine strategy is the best strategy out of all possible strategies (linear and nonlinear) for this game.

Furthermore, from stochastic optimal control theory we know that for the one-sided LQG problem we get identical optimal solutions using a variational optimization technique (a necessary condition for global optimality) or a dynamic programming optimization technique (a sufficient condition for global optimality). This means that the LQG solution using either technique yields a globally optimal solution. We derived $u_p^{(1)}(i)$ and $u_e^{(2)}(i)$ using a variational technique, which means these are globally optimal strategies for their respective one-sided LQG problems. Therefore, since we have shown these globally optimal strategies are equivalent to $u_p^o(i)$ and $u_e^o(i)$, we can state that we have found globally optimal saddle point strategies for the zero-sum LQG multistage game with nonclassical information pattern. As such, the optimal performance index value (as derived in Section 4.7) is unique.

Remark 4.8. *We make no statement here regarding uniqueness of the optimal affine strategies (i.e. nonlinear saddle point strategies may exist). Regardless, as discussed in [BH68, Section V-A], each player can rest assured that by playing these optimal affine strategies they will at least achieve the calculated expected performance index value, even if their opponent plays a nonlinear strategy.*

4.7 Optimal Performance Index

We can now use the derived optimal relationships to find an expression for the optimal performance index value. First, we start with the form of the performance index in (4.16), and substitute in our *mean control strategy* definition from (4.48) and (4.49),

$$
\begin{aligned}
J &= \frac{1}{2}\bar{\mathbf{X}}^T(N)\tilde{Q}(N)\bar{\mathbf{X}}(N) + \frac{1}{2}Tr\Big(\tilde{Q}(N)\mathbf{P}(N)\Big) \\
&\quad + \frac{1}{2}\sum_{i=0}^{N-1}\Big[\big(b_p(i) + K_p(i)H_p(i)\bar{\mathbf{X}}(i)\big)^T R_p(i)\big(b_p(i) + K_p(i)H_p(i)\bar{\mathbf{X}}(i)\big) \\
&\quad - \big(b_e(i) + K_e(i)H_e(i)\bar{\mathbf{X}}(i)\big)^T R_e(i)\big(b_e(i) + K_e(i)H_e(i)\bar{\mathbf{X}}(i)\big) \\
&\quad + Tr\Big(H_p^T(i)K_p^T(i)R_p(i)K_p(i)H_p(i)\mathbf{P}(i) - H_e^T(i)K_e^T(i)R_e(i)K_e(i)H_e(i)\mathbf{P}(i)\Big)\Big] \\
&= \frac{1}{2}\bar{\mathbf{X}}^T(N)_N Q(N)\bar{\mathbf{X}}(N)_N + \frac{1}{2}\sum_{i=0}^{N-1}\Big[\bar{u}_p^T(i)R_p(i)\bar{u}_p(i) - \bar{u}_e^T(i)R_e(i)\bar{u}_e(i)\Big] \\
&\quad + \frac{1}{2}Tr\Big(\tilde{Q}(N)\mathbf{P}(N)\Big) + \frac{1}{2}\sum_{i=0}^{N-1}\Big[Tr\Big(H_p^T(i)K_p^T(i)R_p(i)K_p(i)H_p(i)\mathbf{P}(i) \\
&\quad - H_e^T(i)K_e^T(i)R_e(i)K_e(i)H_e(i)\mathbf{P}(i)\Big)\Big] \\
&= \frac{1}{2}\bar{\mathbf{X}}^T(N)_N Q(N)\bar{\mathbf{X}}(N)_N + \frac{1}{2}\sum_{i=0}^{N-1}\Big[\bar{\mathbf{X}}^T(i)_i C_p^T(i)R_p(i)C_p(i)\bar{\mathbf{X}}(i)_i \\
&\quad - \bar{\mathbf{X}}^T(i)_i C_e^T(i)R_e(i)C_e(i)\bar{\mathbf{X}}(i)_i\Big] \\
&\quad + \frac{1}{2}Tr\Big(\tilde{Q}(N)\mathbf{P}(N)\Big) + \frac{1}{2}\sum_{i=0}^{N-1}\Big[Tr\Big(H_p^T(i)K_p^T(i)R_p(i)K_p(i)H_p(i)\mathbf{P}(i) \\
&\quad - H_e^T(i)K_e^T(i)R_e(i)K_e(i)H_e(i)\mathbf{P}(i)\Big)\Big].
\end{aligned}
$$

Notice that the first two terms are related to the mean state and *mean control strategy*, whereas the last two terms are related to the variance about this mean behavior. We now add/subtract equalities for $S_1(i)_{i,i}$ (4.44) and $S_2(i)$ (4.22)

$$
\begin{aligned}
J^o &= \frac{1}{2}\bar{\mathbf{X}}^T(N)_N Q(N)\bar{\mathbf{X}}(N)_N + \frac{1}{2}\sum_{i=0}^{N-1}\Big[\bar{\mathbf{X}}^T(i)_i C_p^T(i)R_p(i)C_p(i)\bar{\mathbf{X}}(i)_i \\
&\quad - \bar{\mathbf{X}}^T(i)_i C_e^T(i)R_e(i)C_e(i)\bar{\mathbf{X}}(i)_i + \bar{\mathbf{X}}^T(i)_i S_1(i)_{i,i}\bar{\mathbf{X}}(i)_i \\
&\quad - \bar{\mathbf{X}}^T(i)_i S_1(i+1)_{i+1,i+1}\big(I_n - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big)\big)\bar{\mathbf{X}}(i)_i\Big] \\
&\quad + \frac{1}{2}Tr\Big(\tilde{Q}(N)\mathbf{P}(N)\Big) + \frac{1}{2}\sum_{i=0}^{N-1}\Big[Tr\Big(H_p^T(i)K_p^T(i)R_p(i)K_p(i)H_p(i)\mathbf{P}(i) \\
&\quad - H_e^T(i)K_e^T(i)R_e(i)K_e(i)H_e(i)\mathbf{P}(i) + S_2(i)\mathbf{P}(i) - H_p^T(i)K_p^T(i)R_p(i)K_p(i)H_p(i)\mathbf{P}(i)
\end{aligned}
$$

$$+ H_e^T(i)K_e^T(i)R_e(i)K_e(i)H_e(i)\mathbf{P}(i) - \tilde{F}^T(i)S_2(i+1)\tilde{F}(i)\mathbf{P}(i)\Big)\Big].$$

Move the terminal boundary conditions inside the summations (remember that $S_1(N)_{N,N} = Q(N)$ and $S_2(N) = \tilde{Q}(N)$) and pull the $S_1(0)_{0,0}$ and $S_2(0)$ terms outside the summations

$$\begin{aligned}
J^o = {} & \frac{1}{2}\bar{\mathbf{X}}^T(0)_0 S_1(0)_{0,0}\bar{\mathbf{X}}(0)_0 + \frac{1}{2}\sum_{i=0}^{N-1}\Big[\bar{\mathbf{X}}^T(i)_i C_p^T(i)R_p(i)C_p(i)\bar{\mathbf{X}}(i)_i \\
& - \bar{\mathbf{X}}^T(i)_i C_e^T(i)R_e(i)C_e(i)\bar{\mathbf{X}}(i)_i + \bar{\mathbf{X}}^T(i+1)_{i+1}S_1(i+1)_{i+1,i+1}\bar{\mathbf{X}}(i+1)_{i+1} \\
& - \bar{\mathbf{X}}^T(i)_i S_1(i+1)_{i+1,i+1}\big(I_n - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big)\bar{\mathbf{X}}(i)_i\Big] \\
& + \frac{1}{2}Tr\Big(S_2(0)\mathbf{P}(0)\Big) + \frac{1}{2}\sum_{i=0}^{N-1}\Big[Tr\Big(S_2(i+1)\mathbf{P}(i+1) - \tilde{F}^T(i)S_2(i+1)\tilde{F}(i)\mathbf{P}(i)\Big)\Big].
\end{aligned}$$

Substitute in $\bar{\mathbf{X}}(i+1)_{i+1}$ (4.45), $\mathbf{P}(i+1)$ (4.15), $C_p(i)$ (4.46), and $C_e(i)$ (4.47)

$$\begin{aligned}
J^o = {} & \frac{1}{2}\bar{\mathbf{X}}^T(0)_0 S_1(0)_{0,0}\bar{\mathbf{X}}(0)_0 \\
& + \frac{1}{2}\sum_{i=0}^{N-1}\Big[\bar{\mathbf{X}}^T(i)_i C_p^T(i)R_p(i)R_p^{-1}(i)\Gamma_p^T(i)S_1(i+1)_{i+1,i+1}\big(I_n - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big)\bar{\mathbf{X}}(i)_i \\
& + \bar{\mathbf{X}}^T(i)_i C_e^T(i)R_e(i)R_e^{-1}(i)\Gamma_e^T(i)S_1(i+1)_{i+1,i+1}\big(I_n - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big)\bar{\mathbf{X}}(i)_i \\
& + \bar{\mathbf{X}}^T(i)_i\big(I_n - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big)^T S_1(i+1)_{i+1,i+1}\big(I_n - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big)\bar{\mathbf{X}}(i)_i \\
& - \bar{\mathbf{X}}^T(i)_i S_1(i+1)_{i+1,i+1}\big(I_n - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big)\bar{\mathbf{X}}(i)_i\Big] \\
& + \frac{1}{2}Tr\Big(S_2(0)\mathbf{P}(0)\Big) + \frac{1}{2}\sum_{i=0}^{N-1}\Big[Tr\Big(S_2(i+1)\tilde{F}(i)\mathbf{P}(i)\tilde{F}^T(i) + S_2(i+1)G_y(i)\mathbf{Y}(i)G_y^T(i) \\
& - \tilde{F}^T(i)S_2(i+1)\tilde{F}(i)\mathbf{P}(i)\Big)\Big]
\end{aligned}$$

and cancel terms (the first summation is identically zero)

$$J^o = \frac{1}{2}\bar{\mathbf{X}}^T(0)_0 S_1(0)_{0,0}\bar{\mathbf{X}}(0)_0 + \frac{1}{2}Tr\Big(S_2(0)\mathbf{P}(0)\Big) + \frac{1}{2}\sum_{i=0}^{N-1}Tr\Big(S_2(i+1)G_y(i)\mathbf{Y}(i)G_y^T(i)\Big). \tag{4.87}$$

Recall that for the *corresponding deterministic game* $x(0) = \bar{\mathbf{X}}(0)_0$ and $S^d(0) = S_1(0)_{0,0}$. Therefore, the first term in the stochastic game optimal performance index is simply the *corresponding deterministic game* optimal performance index value (3.18), J^{d^o}, [5]

$$J^o = J^{d^o} + \frac{1}{2} Tr\Big(S_2(0)\mathbf{P}(0) + \sum_{i=0}^{N-1} S_2(i+1) G_y(i) \mathbf{Y}(i) G_y^T(i) \Big). \tag{4.88}$$

It's now easy to see that if all the variances are zero (no noise), we are left with the *corresponding deterministic game* optimal performance index value. In general, whether or not the stochastic game optimal performance index increases/decreases relative to the *corresponding deterministic game* optimal performance index depends on the problem parameters, affecting the definiteness of $S_2(i)$. One conclusion that we can readily make is that process noise at the final stage is advantageous to the evader (increasing the performance index) since $S_2(N) \geq 0$.

In (4.21) we wrote the augmented performance index differential due solely to differential changes in initial conditions. Now, given (4.88) we can write the *total* optimal performance index differential as

$$dJ^o = \bar{x}^T(0) S^d(0) d\bar{x}(0) + \frac{1}{2} Tr\Big(S_2(0) d\mathbf{P}(0) + \sum_{i=0}^{N-1} G_y^T(i) S_2(i+1) G_y(i) d\mathbf{Y}(i) \Big). \tag{4.89}$$

4.8 Modeling Other Information Patterns

The solution methodology that we've outlined in this chapter provides great flexibility for investigating various information patterns. In fact, modeling different information patterns is as simple as specifying the appropriate enlarged measurement matrices for the pursuer and evader, $H_p(i)$ and $H_e(i)$, respectively.

For example, we can model a stochastic game with perfect information as follows. Instead of using the measurement matrices in (4.6) and (4.7), we specify the j^{th} sub-partition as having three rows, with total dimension $\mathbb{R}^{(n+p+q)\times(N+1)(n+p+q)}$. Then, we can define the

[5] Note the similarities between this optimal performance index value and the special cases (in continuous time) from [BH75, Eqn. 14.2.16] and [BH68, Eqn. 73].

measurement matrices for this game as

$$H_p(i)_{j,k} = \begin{cases} I_n, & j \le i, k = j \quad \text{Row 1 of } j^{th} \text{ sub-partition} \\ I_p, & j \le i, k = N+1+j \quad \text{Row 2 of } j^{th} \text{ sub-partition} \\ I_q, & j \le i, k = 2N+2+j \quad \text{Row 3 of } j^{th} \text{ sub-partition} \\ 0, & otherwise \end{cases}$$

$$H_e(i) = H_p(i) \in \mathbb{R}^{(i+1)(n+p+q)\times(N+1)(n+p+q)}$$

for $i = 0, 1, ..., N-1$. Therefore, at stage $i = 1$ for a two-stage ($N = 2$) scalar game the enlarged measurement matrices would appear as

$$H_p(1) = H_e(1) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

where the first three rows are the measurement at stage 0, and the last three rows are the measurement at stage 1 (current stage).

As another example, we can model a game where the pursuer has perfect information and the evader has imperfect information. With no process noise this becomes the Behn and Ho game [BH68]. In fact, we can compare the case where the perfect-information pursuer has a one-stage delay in his knowledge of the evader's estimation error (which happens naturally through the dynamics) versus a no-delay information pattern where we give the pursuer immediate access to the evader's measurement at the current stage (and hence the evader's current estimation error).[6]

[6]Note that the one-stage delay information pattern requires special dimension and matrix invertibility restrictions in order for the pursuer to determine the evader's estimation error as discussed in [BH68, Section IV] and [RL69, Section V].

For this game the evader's measurement matrix still appears as in (4.7). The perfect-information pursuer one-stage delay information pattern appears as

$$H_p(i)_{j,k} = \begin{cases} I_n, & j = k \le i \\ 0, & otherwise \end{cases}$$

$$H_p(i) \in \mathbb{R}^{(i+1)n \times (N+1)(n+p+q)}$$

for $i = 0, 1, ..., N-1$.[7] To model the perfect-information pursuer no-delay information pattern (superscript ND) we specify the j^{th} sub-partition as having two rows, with total dimension $\mathbb{R}^{(n+q)\times(N+1)(n+p+q)}$. We can therefore give the pursuer immediate access to the evader's current measurement by using

$$H_p^{ND}(i)_{j,k} = \begin{cases} I_n, & j = k \le i & \text{Row 1 of } j^{th} \text{ sub-partition} \\ \Theta_e(j), & j = k \le i & \text{Row 2 of } j^{th} \text{ sub-partition} \\ I_q, & j \le i, k = 2N+2+j & \text{Row 2 of } j^{th} \text{ sub-partition} \\ 0, & otherwise \end{cases}$$

$$H_p^{ND}(i) \in \mathbb{R}^{(i+1)(n+q)\times(N+1)(n+p+q)}$$

for $i = 0, 1, ..., N-1$. Therefore, at stage $i = 1$ for a two-stage ($N = 2$) scalar game the perfect-information pursuer no-delay information pattern enlarged measurement matrix would appear as

$$H_p^{ND}(1) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \Theta_e(0) & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \Theta_e(1) & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

where the first two rows are the pursuer's measurement at stage 0 (the second row being the evader's measurement at stage 0), and the last two rows are the pursuer's measurement at stage 1 (the last row being the evader's measurement at stage 1).

[7]Note that we have kept the enlarged state dimension the same (i.e. the $H_p(i)$ column dimension is still $(N+1)(n+p+q)$). Since the pursuer is assumed to have perfect information, his measurement noise variance will be zero and the corresponding elements of his enlarged measurement matrix will be zero. Therefore, we could have reduced the enlarged state dimension to $(N+1)(n+q)$.

Note that we could easily reverse this information pattern and model a game where the evader has perfect information and the pursuer has imperfect information.

4.9 Summary

In summary, we've derived the general LQG multistage game solution using a two-sided (simultaneous) optimization technique. We discussed the methodology used to solve the resulting two-point boundary value problem, we provided interpretation of the optimal strategies, and we showed that these optimal strategies satisfy the saddle point condition. In addition, we have shown that our general solution to the LQG multistage game may be used to study various information patterns by simply modifying the enlarged measurement matrices.

In the next chapter we will study the performance of these optimal strategies in the presence of noise by way of a numerical study.

CHAPTER 5

Effects of Noise: Some Numerical Results

Some interesting strategy characteristics arise as a function of the process and measurement noise variances. In order to study the effects of noise, we consider a simple, scalar, twenty-stage ($N = 20$) game using the optimal strategies as outlined in Section 4.2. We will characterize the noise impacts on the *stochastic* part of the strategies, that is the $S_2(i)$ (4.26) Lagrange multiplier sequence, and the $K_p(i)$ (4.29) and $K_e(i)$ (4.30) kernels. As such, we can assume that $\bar{x}(0) = 0$ for this study since it only affects the *deterministic* part of the strategies.

5.1 Analysis Setup

Referring to the multistage state dynamics (2.1), the time-invariant parameters we will use are

$$\Gamma_p(i) = 1, \quad \Gamma_e(i) = 1$$
$$\Theta_p(i) = 1, \quad \Theta_e(i) = 1.$$

The initial state statistics are $x(0) \sim \mathcal{N}(0, 10)$. The process noise variance $W(i)$, pursuer measurement noise variance $V_p(i)$, and evader measurement noise variance $V_e(i)$ will be denoted on each figure as we study the effects of these individual variances.

Referring to the performance index (2.4), the time-invariant parameters we will use are

$$Q(N) = 0.5$$
$$R_p(i) = [0.5, 0.95], \quad R_e(i) = 1.$$

Note that we will investigate two different values for $R_p(i)$. We have chosen $R_p(i) < R_e(i)$ so that the deterministic *no conjugate point* condition (3.17) is satisfied.

We include the following types of plots and associated notation:

- In order to characterize the effects of noise, we will compare the maximum singular value of $S_2(i)$ (4.26) (associated with the stochastic game) to the maximum singular value of $S^d(i)$ (3.14) (associated with the deterministic game). The maximum singular value can be thought of as the maximum gain of a matrix as shown in Appendix B.1.2. Therefore, by comparing the maximum singular values we can characterize the relative system gains.

- In addition, we further characterize the effects of noise by plotting the optimal performance index value, J^o (4.88).

- $K_p(i)$ (4.29) and $K_e(i)$ (4.30) plots show the last six elements of these kernels at stage i. That is, at stage i we show the gains applied to the last six measurements: $K_p(i)_{i-5:i}$ and $K_e(i)_{i-5:i}$. These kernels are not shown for every stage of the game, but, rather, in intervals so that we can discern the overall behavior and decision making without making the plots overly-complicated.

- $\tilde{C}_p(i)$ (4.55) and $\tilde{C}_e(i)$ (4.56) plots show the last six elements of these control gains operating on the state estimate at stage i. That is, at stage i we show the control gains applied to the last six state estimates (current state estimate, $\hat{x}(i)$, and smoothed state estimates, $\hat{x}(k|i)$, $k = i-5, i-4, ..., i-1$). We use this as a comparison with the $K_p(i)$ and $K_e(i)$ plots in order to distinguish between control gain behavior and estimator gain behavior.

- $K_p^{ce}(i)$ and $K_e^{ce}(i)$ are the *certainty equivalent sub-optimal centralized solution kernels*. These kernels are generated using $C_p(i)$ (3.12) and $C_e(i)$ (3.13) applied to a sub-optimal centralized Kalman filter for each player. The sub-optimal centralized Kalman filter for each player is designed *assuming* that his opponent uses a common state estimate.

Referring to Appendix A.1 we can show this explicitly using the traditional Kalman filter equations as

$$\begin{aligned}
u_p^{ce}(i) &= -C_p(i)\hat{x}_p(i)\\
\hat{x}_p(i) &= \bar{x}_p(i) + M_p(i)\Theta_p^T(i)\big(\Theta_p(i)M_p(i)\Theta_p^T(i) + V_p(i)\big)^{-1}\big(z_p(i) - \Theta_p(i)\bar{x}_p(i)\big)\\
P_p(i) &= M_p(i) - M_p(i)\Theta_p^T(i)\big(\Theta_p(i)M_p(i)\Theta_p^T(i) + V_p(i)\big)^{-1}\Theta_p(i)M_p(i)\\
\bar{x}_p(i+1) &= \big(I_n - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big)\hat{x}_p(i)\\
M_p(i+1) &= \big(I_n - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big)P_p(i)\big(I_n - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big)^T + W(i)
\end{aligned} \tag{5.1}$$

and

$$\begin{aligned}
u_e^{ce}(i) &= C_e(i)\hat{x}_e(i)\\
\hat{x}_e(i) &= \bar{x}_e(i) + M_e(i)\Theta_e^T(i)\big(\Theta_e(i)M_e(i)\Theta_e^T(i) + V_e(i)\big)^{-1}\big(z_e(i) - \Theta_e(i)\bar{x}_e(i)\big)\\
P_e(i) &= M_e(i) - M_e(i)\Theta_e^T(i)\big(\Theta_e(i)M_e(i)\Theta_e^T(i) + V_e(i)\big)^{-1}\Theta_e(i)M_e(i)\\
\bar{x}_e(i+1) &= \big(I_n - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big)\hat{x}_e(i)\\
M_e(i+1) &= \big(I_n - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big)P_e(i)\big(I_n - \Gamma_p(i)C_p(i) - \Gamma_e(i)C_e(i)\big)^T + W(i)
\end{aligned} \tag{5.2}$$

where $\bar{x}_p(0) = \bar{x}_e(0) = \bar{x}(0)$ and $M_p(0) = M_e(0) = M(0)$. We can also write the state estimates, $\hat{x}_p(i)$ and $\hat{x}_e(i)$, as affine functions of the measurement histories to get the *certainty equivalent sub-optimal centralized strategies* in the following form

$$\begin{aligned}
u_p^{ce}(i) &= -b_p^{ce}(i) - \sum_{j=0}^{i} K_p^{ce}(i)_j z_p(j)\\
u_e^{ce}(i) &= b_e^{ce}(i) + \sum_{j=0}^{i} K_e^{ce}(i)_j z_e(j)
\end{aligned} \tag{5.3}$$

where $b_p^{ce} \in \mathbb{R}^m$, $K_p^{ce}(i)_j \in \mathbb{R}^{m\times p}$, $b_e^{ce} \in \mathbb{R}^l$, $K_e^{ce}(i)_j \in \mathbb{R}^{l\times q}$. If $\bar{x}(0) = 0$ then $b_p^{ce}(i) = 0$ and $b_e^{ce}(i) = 0, \forall i$. The centralized Kalman filter assumption produces a sub-optimal filter/solution, but it avoids the infinite recursion that we see in the stochastic optimal solution. We use these *certainty equivalent sub-optimal centralized solution kernels* here as a comparison with the stochastic optimal solution kernels.

In order to specify different strategy combinations and compare the respective performance we define three terms for use throughout Chapters 5 and 6.

Definition 5.1. *The LQD game is the corresponding deterministic multistage game (noiseless game with $x(0) = \bar{x}(0)$).*

Definition 5.2. *The LQG1 game is the stochastic multistage game where each player uses their* certainty equivalent sub-optimal centralized strategy*. The terms P-LQG1 and E-LQG1 will be used to indicate when only the pursuer or evader, respectively, is playing their LQG1 strategy.*

Definition 5.3. *The LQG2 game is the stochastic multistage game where each player uses their stochastic optimal strategy. The terms P-LQG2 and E-LQG2 will be used to indicate when only the pursuer or evader, respectively, is playing their LQG2 strategy.*

We use the solution process outlined in Section 4.2 to solve the two-point boundary value problem as outlined in Section 4.3 for the various combinations of process and measurement noises below.

5.2 No Measurement Noise

We first consider the game where measurement noises are zero: $V_p(i) = 0$ and $V_e(i) = 0$. Because the player's have access to the exact same information upon which to base their strategies, we should surmise that they will use their deterministic game strategies, and, indeed, this is the case. As we can see in Fig. 5.1 and Fig. 5.2 the stochastic optimal strategies (LQG2), $K_p(i)$ and $K_e(i)$, are exactly equal to their respective certainty equivalent sub-optimal centralized strategies (LQG1), $K_p^{ce}(i)$ and $K_e^{ce}(i)$ (the sub-optimal centralized strategies are indicated by the underlying gray markers). Each player makes a perfect measurement at each stage, and, as such, at each stage of the game the kernels include only one non-zero element associated with $C_p(i)$ or $C_e(i)$ for the respective player.

This behavior is also shown explicitly in Fig. 5.3 and Fig. 5.4 where the control gains $\tilde{C}_p(i)$

and $\tilde{C}_e(i)$ have only one non-zero gain operating on the current state and that gain is exactly equal to $C_p(i)$ or $C_e(i)$ (indicated by the underlying gray markers) for the respective player. Therefore, we have $K_p(i)_i = \tilde{C}_p(i)_i = C_p(i)$ (Fig. 5.1 and Fig. 5.3) and $K_e(i)_i = \tilde{C}_e(i)_i = C_e(i)$ (Fig. 5.2 and Fig. 5.4).

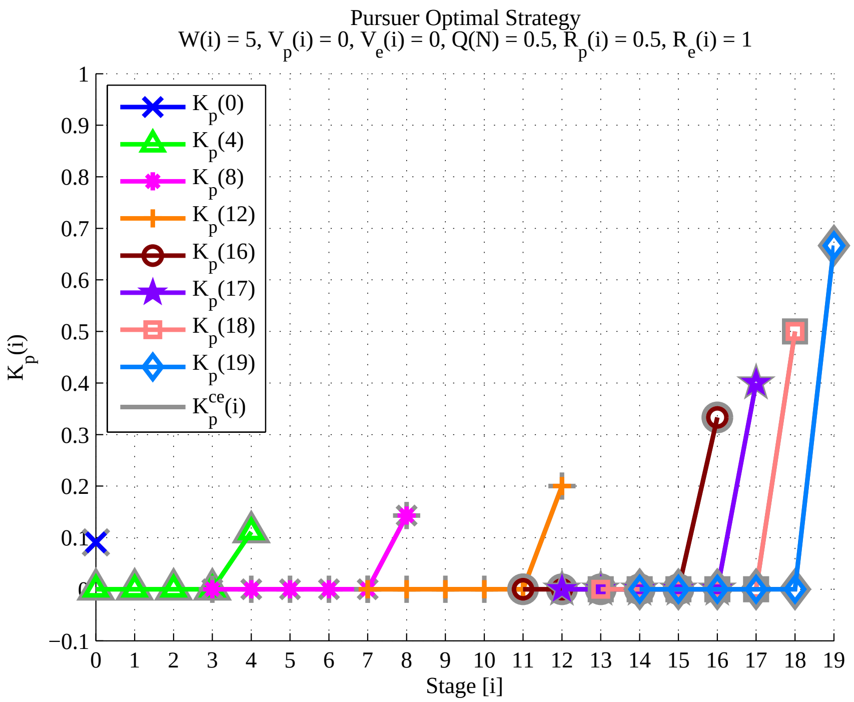

Figure 5.1: Pursuer's optimal strategy for a perfect information game.

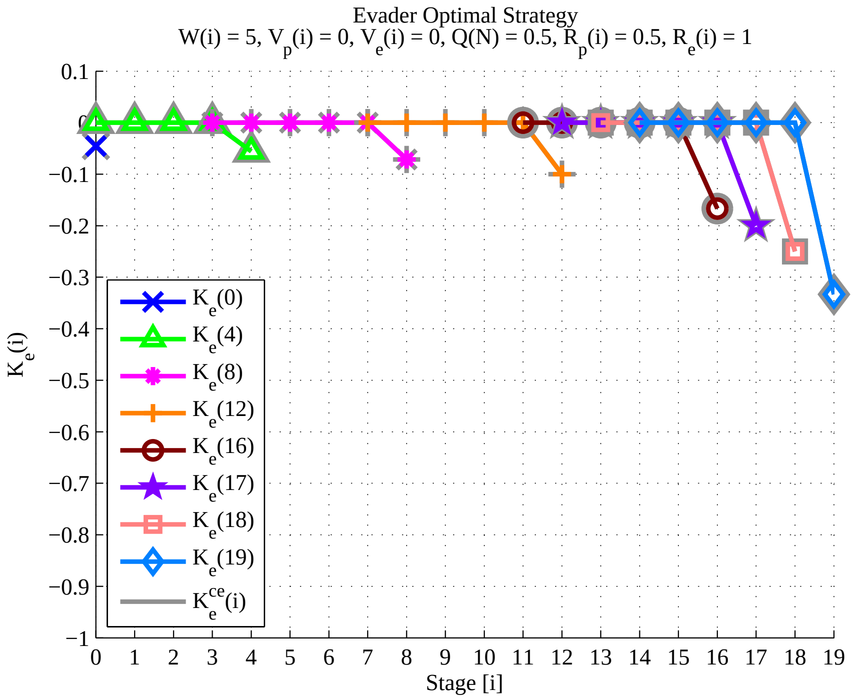

Figure 5.2: Evader's optimal strategy for a perfect information game.

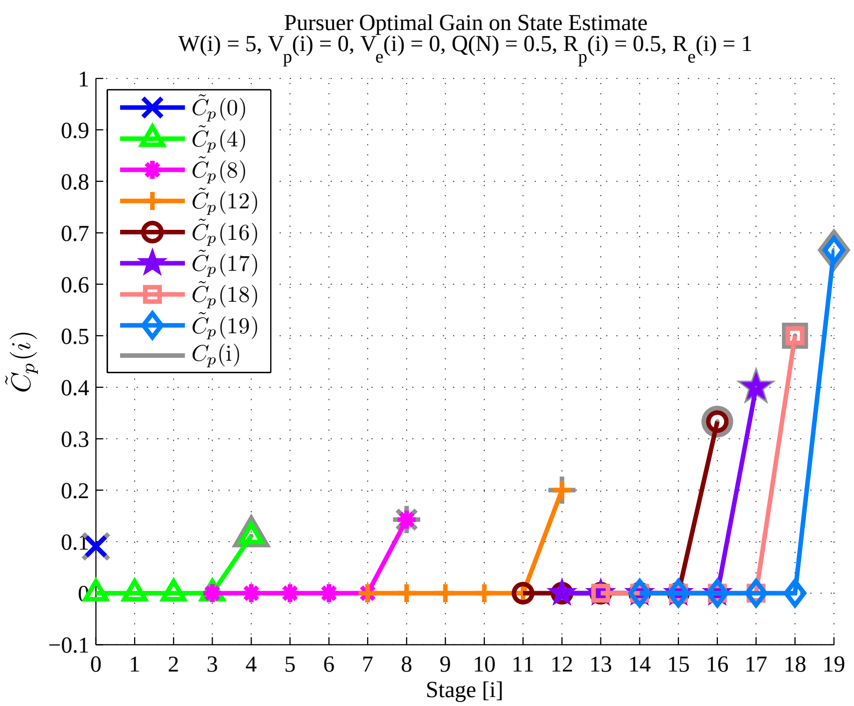

Figure 5.3: Pursuer's optimal gain for a perfect information game.

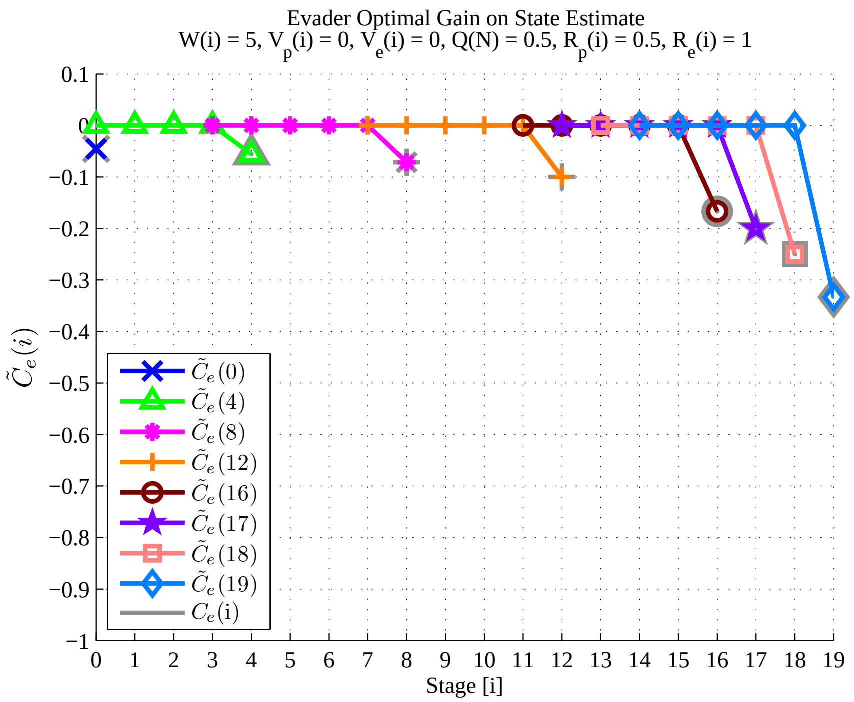

Figure 5.4: Evader's optimal gain for a perfect information game.

5.3 Process and Measurement Noise Effects

We will study the effects of process and measurement noise variances by observing the impact these variances have on the Lagrange multiplier sequence, $S_2(i)$, and the optimal performance index value, J^o. We generated the plots contained in this section using the following measurement noise variance combinations; one player's variance was fixed while the other player's variance changed. We vary the pursuer measurement noise variance as

$$\begin{aligned} V_p(i) &= [0.1, 1, 5, 10, 15, 20, 30, 40, 60, 80, 100] \\ V_e(i) &= 0.1 \end{aligned} \tag{5.4}$$

and we vary the evader measurement noise variance as

$$\begin{aligned} V_p(i) &= 0.1 \\ V_e(i) &= [0.1, 1, 5, 10, 15, 20, 30, 40, 60, 80, 100]. \end{aligned} \tag{5.5}$$

To begin, let's look at the case where $W(i) = 5$, $R_p(i) = 0.5$, and we sweep through the pursuer and evader measurement noise variance combinations in (5.4) and (5.5). In Fig. 5.5 we can see the measurement noise variance effects on the stochastic system gain, $\bar{\sigma}(S_2(i))$, relative to the deterministic system gain, $\bar{\sigma}(S^d(i))$.[1] The larger magnitude V_p variation lines are associated with relatively larger pursuer measurement noise variances, and the smaller magnitude V_e variation lines are associated with relatively larger evader measurement noise variances. That is, as the pursuer measurement noise variance increases the system gain increases, but as the evader measurement noise variance increases the system gain decreases.

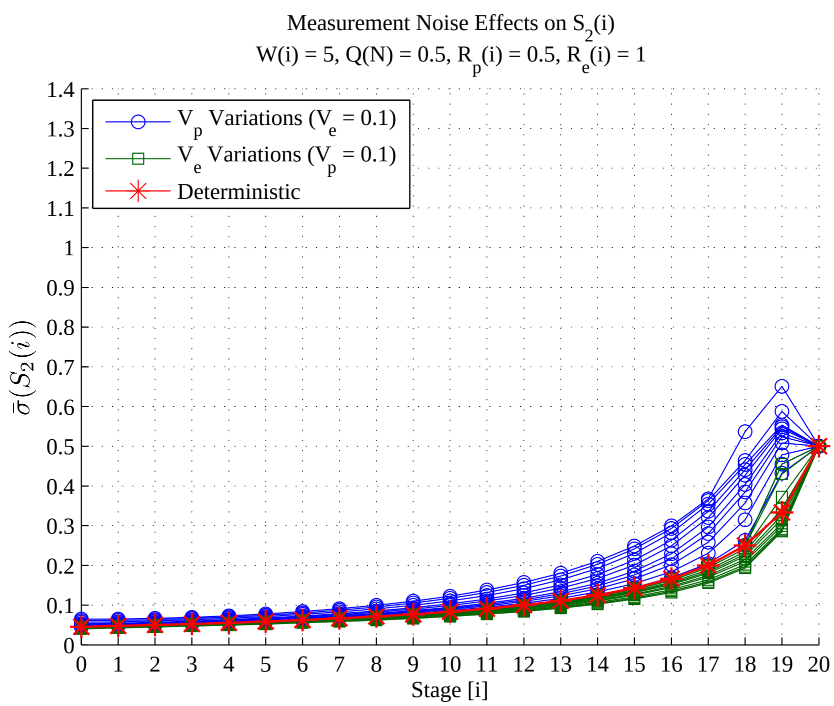

Figure 5.5: Measurement noise effects on $S_2(i)$ with $W(i) = 5$ and $R_p(i) = 0.5$.

This behavior is also reflected in the magnitude of the LQG2 strategies, $K_p(i)$ and $K_e(i)$, relative to the LQG1 strategies, $K_p^{ce}(i)$ and $K_e^{ce}(i)$. For the case where $V_p(i) = 100$ and $V_e(i) = 0.1$ we can see from Fig. 5.6 and Fig. 5.7 the relatively larger LQG2 gains as compared with the LQG1 gains. Furthermore, we can see that the overall behavior of $K_e(i)$

[1] The relatively large y-axis scale is for ease of comparison with larger magnitude plots that follow.

is significantly different than $K_e^{ce}(i)$ in Fig. 5.7. The reason for this behavior becomes more apparent, and we will discuss it further, using the limiting noise cases that follow.

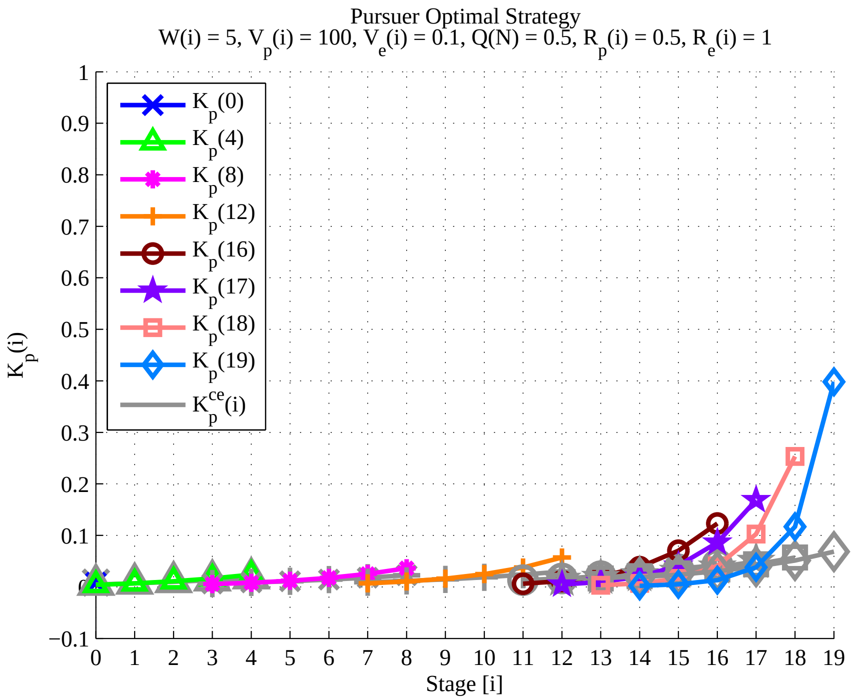

Figure 5.6: Pursuer's optimal strategy for $W(i) = 5$, $V_p(i) = 100$, and $V_e(i) = 0.1$.

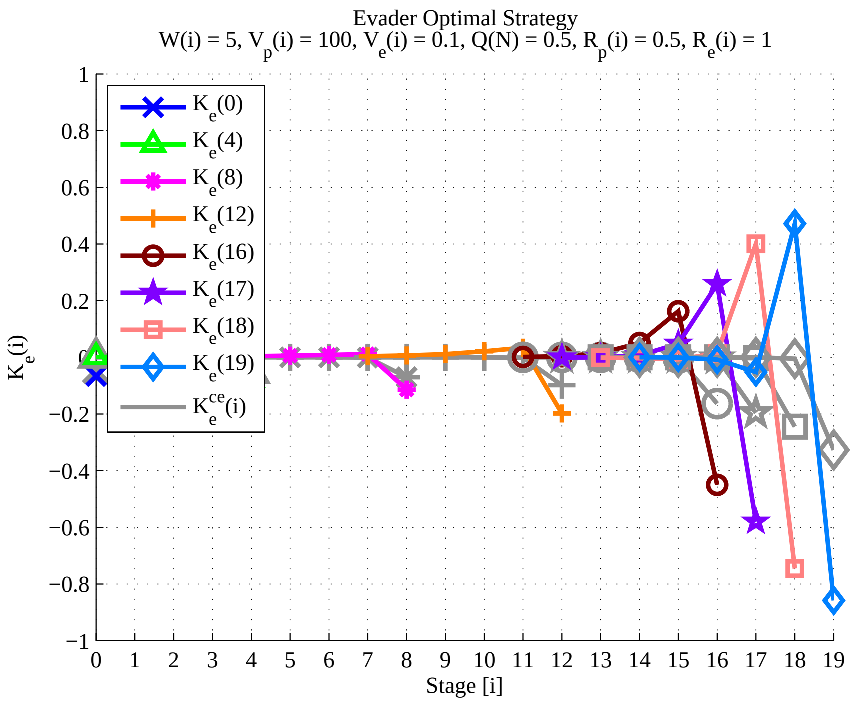

Figure 5.7: Evader's optimal strategy for $W(i) = 5$, $V_p(i) = 100$, and $V_e(i) = 0.1$.

For the case where $V_p(i) = 0.1$ and $V_e(i) = 100$ we can see from Fig. 5.8 and Fig. 5.9 the relatively smaller LQG2 gains as compared with the LQG1 gains. In fact, it appears that as $V_e(i)$ increases $K_e(i)$ trends towards zero (along with $K_e^{ce}(i)$). That is, the evader control becomes open-loop. As we can see from (4.43) as $K_e(i) \to 0$, $b_e(i) \to C_e(i)\bar{\mathbf{X}}(i)_i$, which means that $u_e^o(i) = C_e(i)\bar{\mathbf{X}}(i)_i$. Therefore, as $V_e(i)$ increases the evader uses his deterministic gain operating on the *corresponding deterministic game* state trajectory.

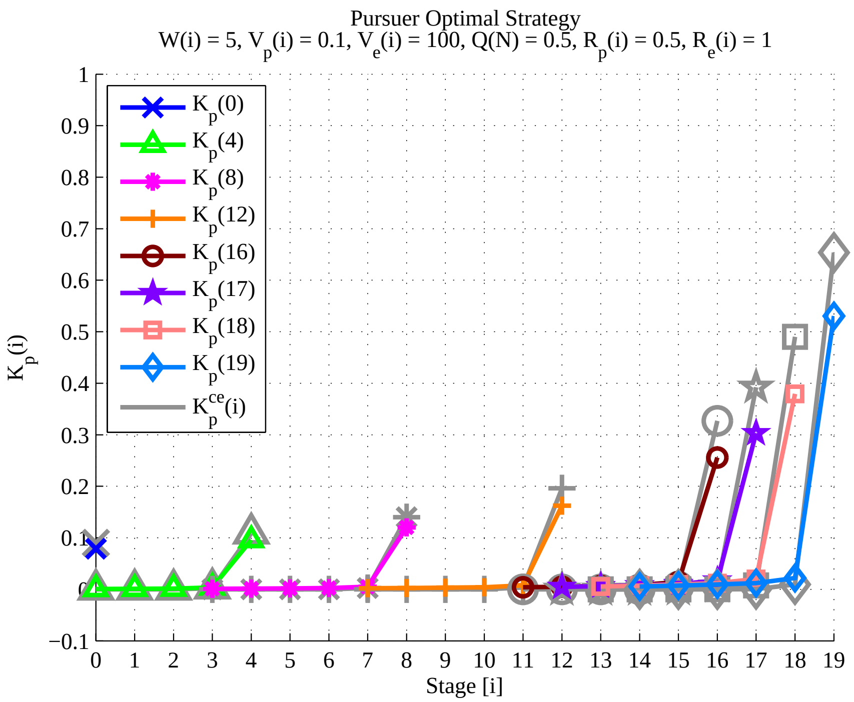

Figure 5.8: Pursuer's optimal strategy for $W(i) = 5$, $V_p(i) = 0.1$, and $V_e(i) = 100$.

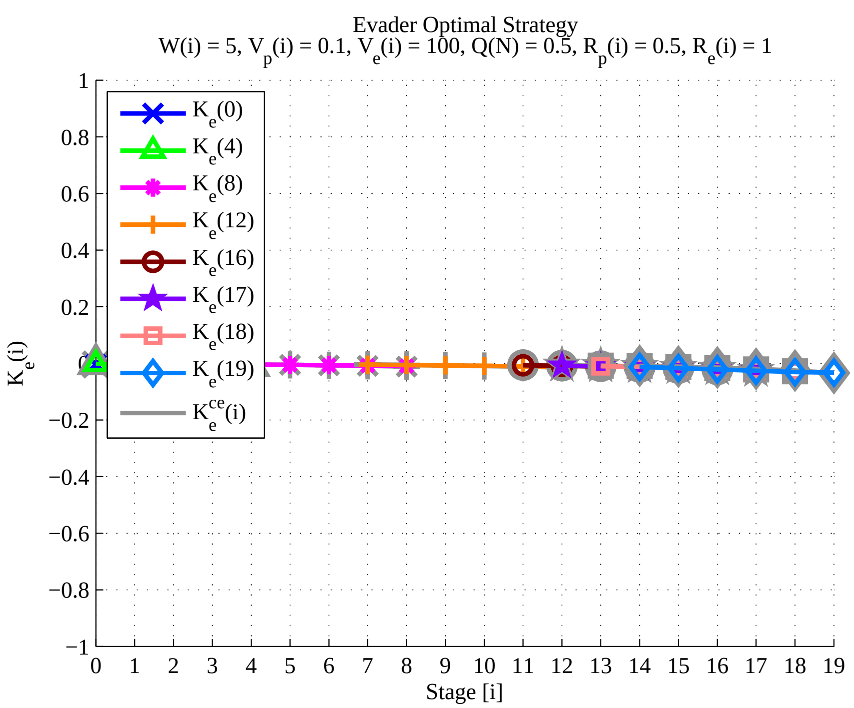

Figure 5.9: Evader's optimal strategy for $W(i) = 5$, $V_p(i) = 0.1$, and $V_e(i) = 100$.

In Fig. 5.10 we can see that, as expected, increasing the pursuer's measurement noise variance increases the optimal performance index value (in favor of the evader), and increasing the evader's measurement noise variance decreases the optimal performance index value (in favor of the pursuer). Note that as the evader's measurement noise variance increases the optimal performance index value appears to be approaching some limit. This seems to corroborate our previous finding that the evader control becomes open-loop as his measurement noise variance increases.

Since $\bar{x}(0) = 0$ ($x(0) = 0$ for the *corresponding deterministic game*) we can see from (3.18) that $J^{d^o} = 0$. Therefore, the non-zero J^o in Fig. 5.10 when the measurement noise variances are both approximately zero is due solely to process noise variance ($W(i) = 5$).

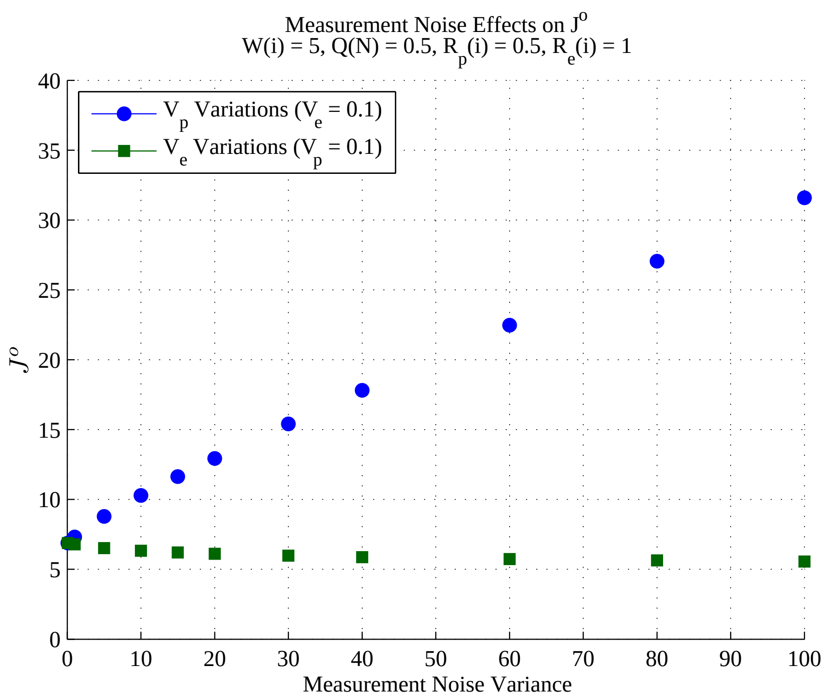

Figure 5.10: Measurement noise effects on J^o with $W(i) = 5$ and $R_p(i) = 0.5$.

Next, we look at a case where process noise variance is decreased from $W(i) = 5$ to $W(i) = 1$. Comparing Fig. 5.11 ($W(i) = 1$) with Fig. 5.5 ($W(i) = 5$) we can see the dramatic effect that decreased process noise variance has on the solution. For $W(i) = 1$ the stochastic system gain, $\bar{\sigma}(S_2(i))$, increases significantly, especially at the latter stages of the game, with increasing pursuer measurement noise variance. Comparing Fig. 5.12 ($W(i) = 1$) with Fig. 5.10 ($W(i) = 5$) we can see that decreasing the process noise variance results in a decreased optimal performance index value across the board (in favor of the pursuer). Since decreased process noise variance benefits the pursuer, what is happening with the stochastic system gain as reflected in $\bar{\sigma}(S_2(i))$ with increasing pursuer measurement noise variance? To further explore this question we will decrease process noise variance even further.

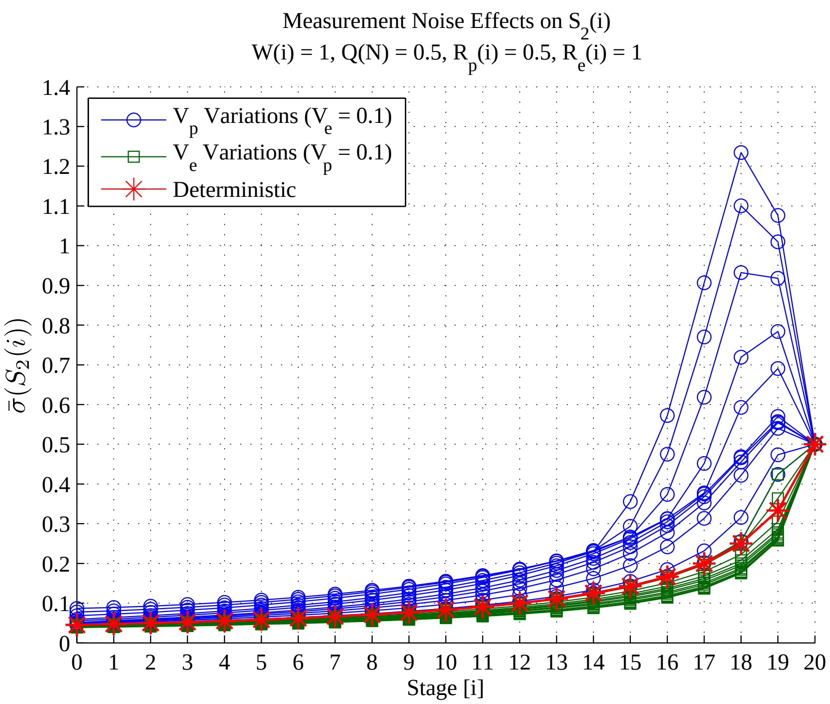

Figure 5.11: Measurement noise effects on $S_2(i)$ with $W(i) = 1$ and $R_p(i) = 0.5$.

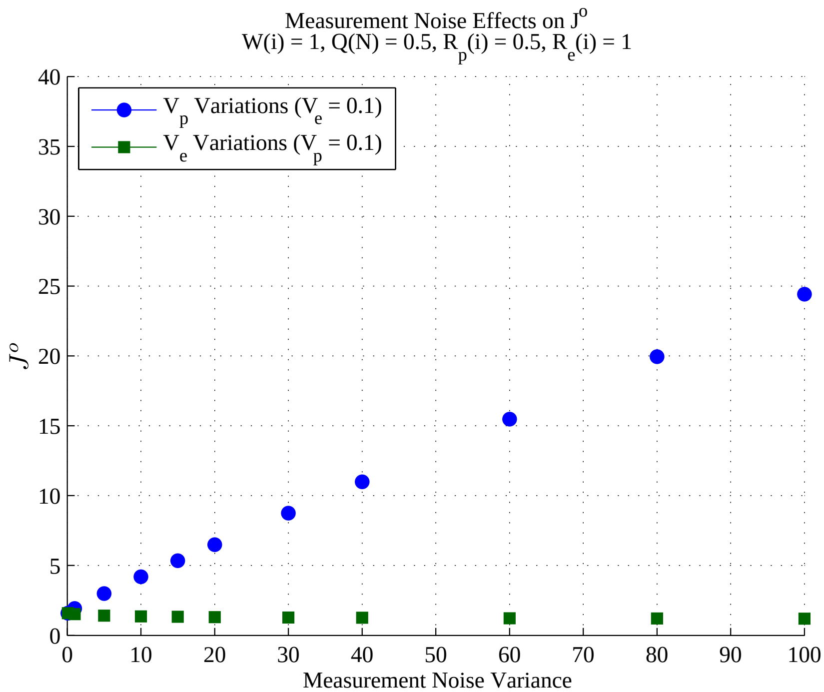

Figure 5.12: Measurement noise effects on J^o with $W(i) = 1$ and $R_p(i) = 0.5$.

We now take a closer look at the strategies when process noise variance is decreased to $W(i) = 0.1$ with $V_p(i) = 100$ and $V_e(i) = 0.1$. From Fig. 5.13 it is interesting to note that the pursuer's *certainty equivalent sub-optimal centralized solution kernel* is essentially zero due to the large measurement noise variance and small process noise variance (Kalman filter gain is approximately zero). However, note that the pursuer's stochastic optimal solution kernel is non-zero, and, furthermore, in Fig 5.14 we see that the pursuer is utilizing his smoothed state estimate in his control strategy. In particular, we see that at the latter stages of the game there is an oscillatory behavior that develops.

As the process noise variance approaches zero the evader, whose measurement noise variance is approximately zero, has almost perfect knowledge of the state with one-frame delay. That is, at stage i the evader is able to calculate the pursuer's control/measurement from the previous stage, $i - 1$. Therefore, effectively the only private information that the pursuer has is his measurement at the current stage; he can use that measurement to come up with a smoothed state estimate across previous stages. The apparent reason for the oscillatory pursuer control gains in Fig 5.14 is his desire to inject the noise from his current measurement into the system at some time constant from the end of the game in order to confuse the evader's estimator, while at the same time averaging through the noise. The effect of this behavior is reflected in Fig. 5.15 where we see that the evader's stochastic optimal solution kernel (which includes his enlarged Kalman filter) is oscillatory, whereas the evader's optimal control gains in Fig. 5.16 are well-behaved. This seems to suggest that the pursuer's oscillatory control gains are having much more impact on the evader's estimator gains than on the evader's control gains. We will see the consequences of the pursuer's optimal strategy using a Monte Carlo analysis in Section 5.4.

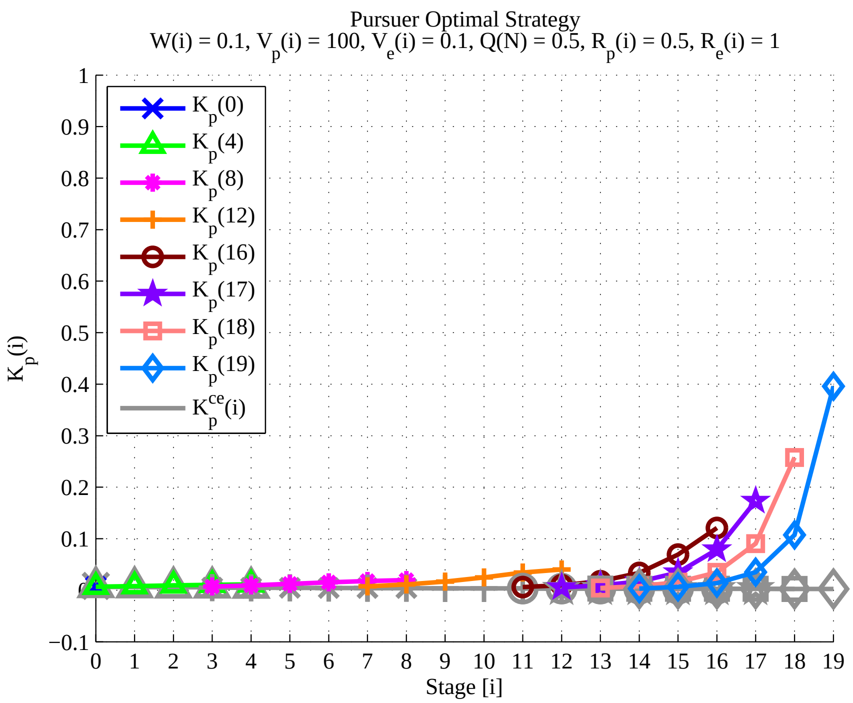

Figure 5.13: Pursuer's optimal strategy for $W(i) = 0.1$, $V_p(i) = 100$, and $V_e(i) = 0.1$.

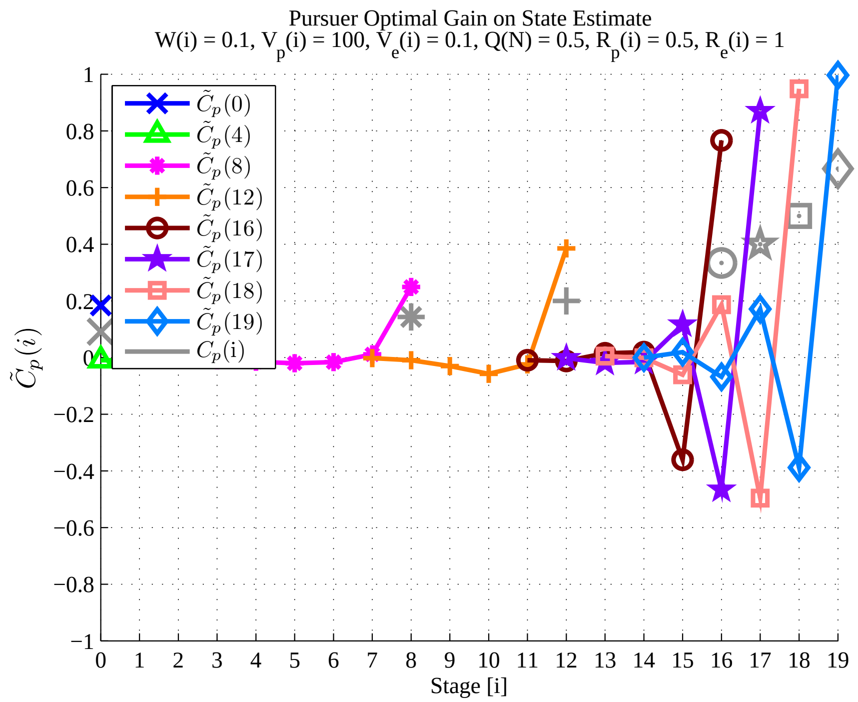

Figure 5.14: Pursuer's optimal gain for $W(i) = 0.1$, $V_p(i) = 100$, and $V_e(i) = 0.1$.

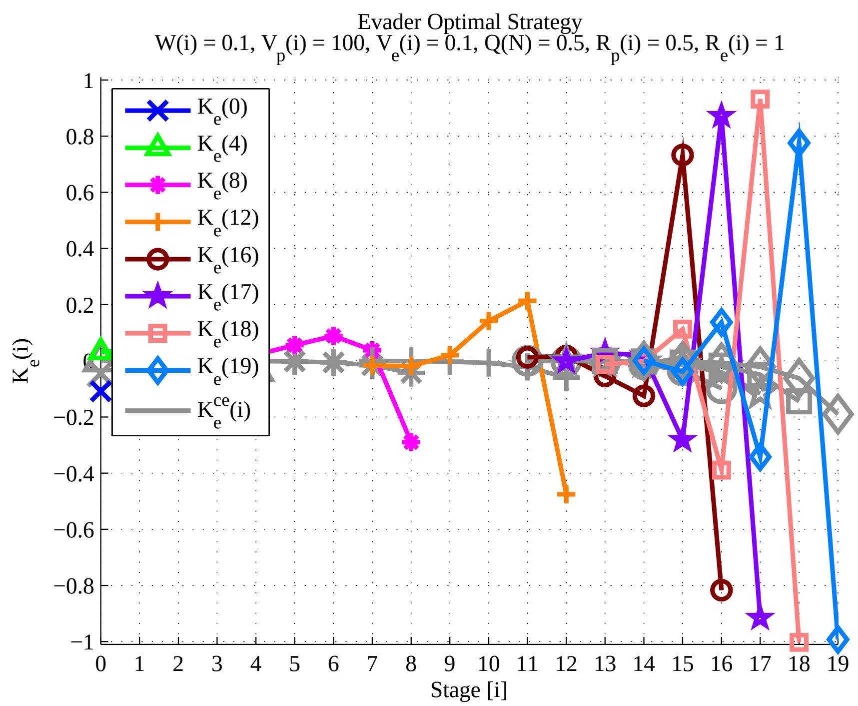

Figure 5.15: Evader's optimal strategy for $W(i) = 0.1$, $V_p(i) = 100$, and $V_e(i) = 0.1$.

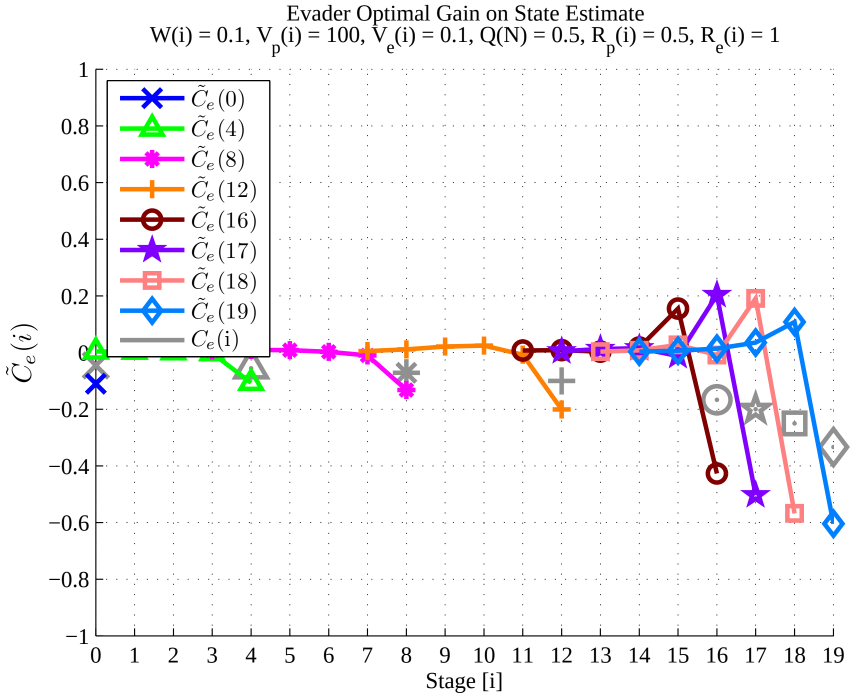

Figure 5.16: Evader's optimal gain for $W(i) = 0.1$, $V_p(i) = 100$, and $V_e(i) = 0.1$.

Lastly, we look at the case where the pursuer's control energy is more heavily penalized (i.e. $R_p(i)$ increases). We choose $R_p(i) = 0.95$, such that we still satisfy the deterministic *no conjugate point* condition (3.17). Fig. 5.17 and Fig. 5.18 show the stochastic system gain and optimal performance index value, respectively, for $W(i) = 5$. And Fig. 5.19 and Fig. 5.20 show the stochastic system gain and optimal performance index value, respectively, for $W(i) = 1$. Note, in particular, the dramatically different stochastic system gain due to the increased penalty on the pursuer's control when comparing Fig. 5.17 with Fig. 5.5 for $W(i) = 5$, and Fig. 5.19 with Fig. 5.11 for $W(i) = 1$.

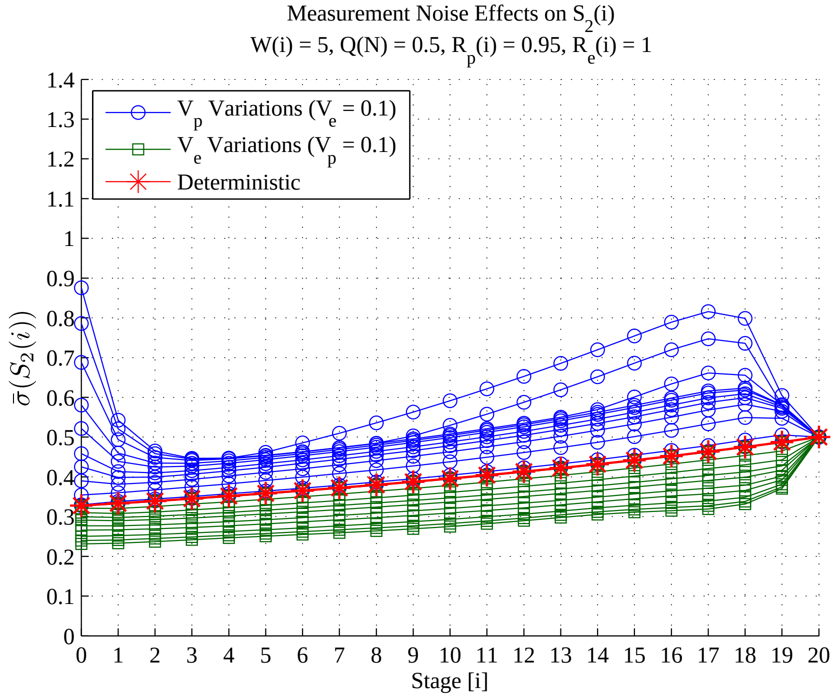

Figure 5.17: Measurement noise effects on $S_2(i)$ with $W(i) = 5$ and $R_p(i) = 0.95$.

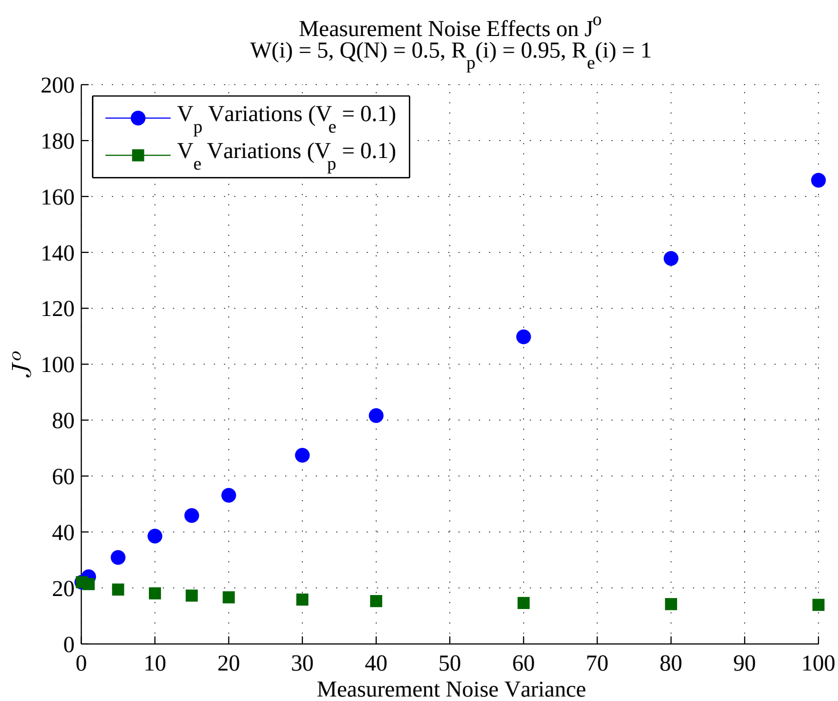

Figure 5.18: Measurement noise effects on J^o with $W(i) = 5$ and $R_p(i) = 0.95$.

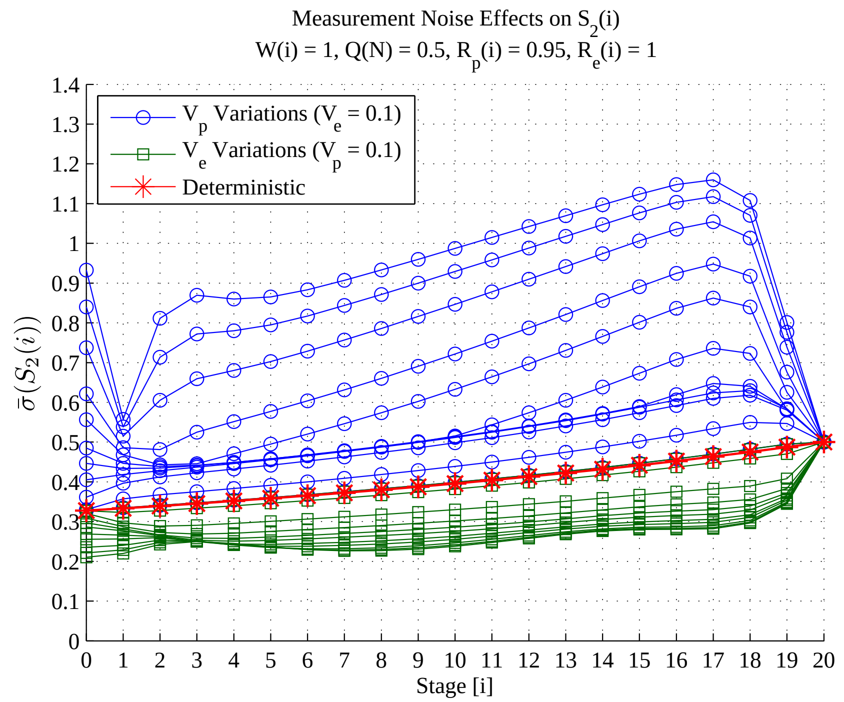

Figure 5.19: Measurement noise effects on $S_2(i)$ with $W(i) = 1$ and $R_p(i) = 0.95$.

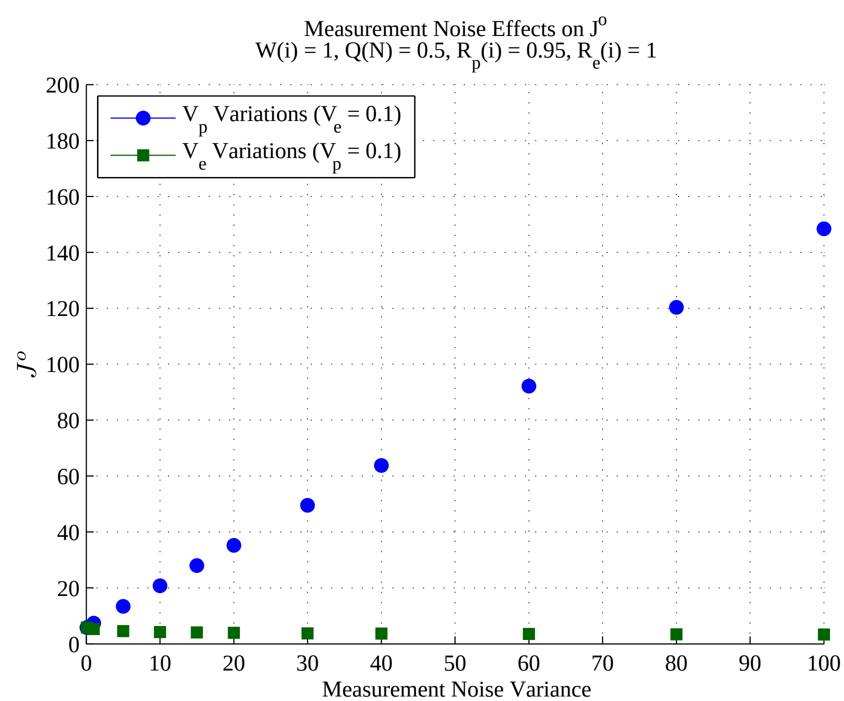

Figure 5.20: Measurement noise effects on J^o with $W(i) = 1$ and $R_p(i) = 0.95$.

As we saw with the increased stochastic system gains previously (specifically with $W(i) = 0.1$ and $R_p(i) = 0.5$), it appears from Fig. 5.19 that due to the high cost of control effort the pursuer is starting to inject noise at much earlier stages of the game as his measurement noise variance increases.[2] This behavior is reflected in Fig. 5.21 and Fig. 5.22 with the oscillatory behavior that we observe in the evader's stochastic optimal solution kernel. Again, we will see the consequences of the pursuer's optimal strategy using a Monte Carlo analysis in Section 5.4.

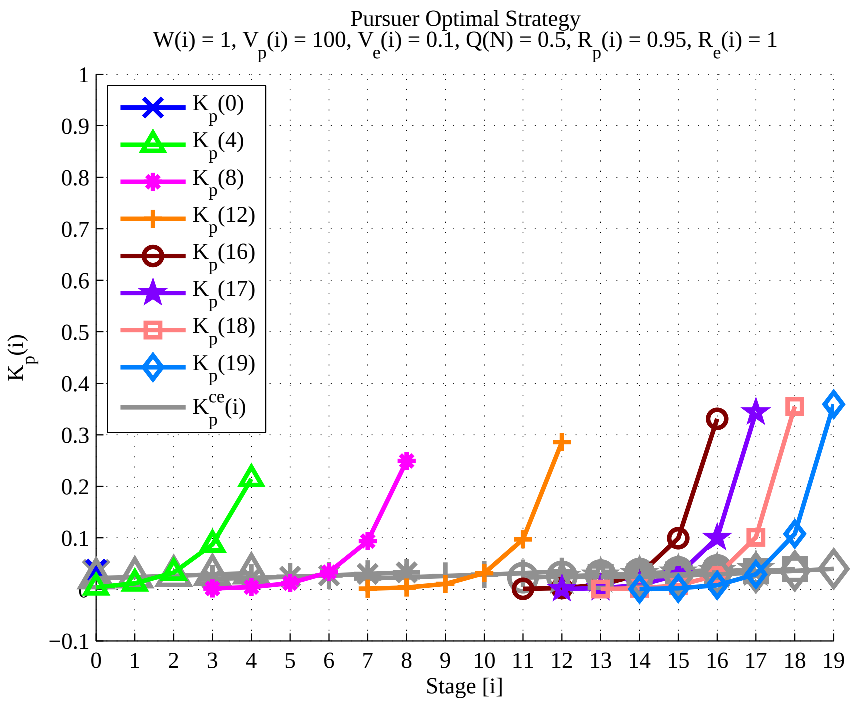

Figure 5.21: Pursuer's optimal strategy for $R_p(i) = 0.95$ with $W(i) = 1$, $V_p(i) = 100$, and $V_e(i) = 0.1$.

[2]In Fig. 5.19, the inflection point at stage 1 as the pursuer's measurement noise variance is increased is presumably due to the evader's ability to estimate the pursuer's control/measurement used at stage 0 given the measurement at stage 1. At this early stage of the game the pursuer has not had time to affect the evader's estimate, hence the inflection point.

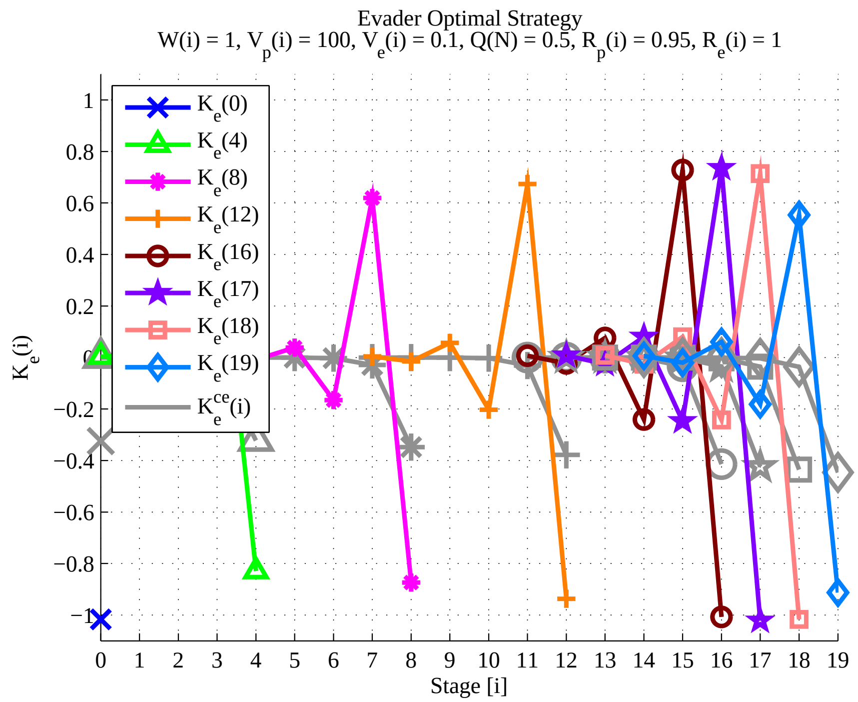

Figure 5.22: Evader's optimal strategy for $R_p(i) = 0.95$ with $W(i) = 1$, $V_p(i) = 100$, and $V_e(i) = 0.1$.

As an aside, we have experimented with providing the evader no-delay access to the pursuer's measurement.[3] That is, we provide the evader two measurements: his own measurement and the pursuer's measurement. In this case, the oscillatory behavior as observed in Fig. 5.22 ceases. The evader's stochastic optimal solution kernel operating on his first measurement is well-behaved, and the evader's stochastic optimal solution kernel operating on his second measurement is well-behaved. This appears to corroborate our claim that the oscillatory behavior observed in the figures above is due to the one-frame delay in the evader's ability to estimate the pursuer's control/measurement, and the pursuer's resulting desire to inject noise into the system.

In summary, we have identified the following trends through this simple numerical study:

[3]This is similar to the special information pattern of [BH68], where the pursuer was the perfect-information player and was assumed to have immediate access to the evader's estimate (no process noise was included).

- As the pursuer's measurement noise variance increases relative to the process noise variance and evader's measurement noise variance, the pursuer appears to inject noise into the system by increasing the gain on his smoothed state estimate. This behavior is exacerbated when the pursuer is more heavily penalized for control energy (i.e. as $R_p(i)$ increases). When the pursuer injects noise he appears to do so in a way such that his control averages through it, which is why we see the oscillatory behavior in the strategies.

- As the evader's measurement noise variance increases his control appears to become open-loop and he uses his deterministic gain operating on the *corresponding deterministic game* state trajectory. This is contrary to the pursuer's behavior when his measurement noise variance increases as previously mentioned. The reason for this difference in behavior is due to the asymmetry of the performance index and the fact that it is not a concave functional with respect to $x(N)$ and $u_e(i)$.

- A decrease in process noise variance benefits the pursuer, an increase in process noise variance benefits the evader (i.e. an increase in noise that affects both players benefits the evader).

5.4 Monte Carlo Analysis

In this section we use a Monte Carlo analysis to look at the resulting performance of the strategies that we discussed in Section 5.3. We use the same problem parameters as outlined in that section, except that we will use $\bar{x}(0) = 100$ to give the problem some realism. There are no units attached to these problem parameters (we will look at a more realistic set of problem parameters in Chapter 6), however, we define a miss distance, $d \triangleq \|x(N)\|$, at the final stage in order to present Monte Carlo statistics. The mean, standard deviation, minimum, and maximum values presented in the following tables are calculated with respect to this miss distance.

We use Definitions 5.2 and 5.3 above in order to compare the performance of the stochastic optimal strategies with the performance of the *certainty equivalent sub-optimal centralized strategies* for several different noise combinations. Specifically, we compare statistics, including the experimental performance index value, for three different strategy combinations: LQG2 (both players use their stochastic optimal strategy), P-LQG2 vs. E-LQG1 (evader is sub-optimal), and P-LQG1 vs. E-LQG2 (pursuer is sub-optimal).

To begin with we look at the no measurement noise case from Section 5.2. As we saw in that section, the LQG1 and LQG2 strategies are equivalent since the players have access to the exact same information. This is confirmed with the Monte Carlo analysis results presented in Table 5.1. As shown in the table, all three strategy combinations produce the exact same statistics. In addition, as confirmation that we have executed a sufficient number of analysis runs we can see in the table that for the LQG2 game the analytical and experimental performance index values are the same. The convergence of the experimental performance index value over the 10k analysis runs is shown in Fig. 5.23.

Table 5.1: Monte Carlo analysis (10k runs) with $W(i) = 5$, $V_p(i) = 0$, $V_e(i) = 0$, and $R_p(i) = 0.5$.

	LQG2	**P-LQG2/ E-LQG1**	**P-LQG1/ E-LQG2**
Mean	9.1	9.1	9.1
Std. Dev.	3.4	3.4	3.4
Min.	0.0	0.0	0.0
Max.	23.0	23.0	23.0
$E[J]$ (anal.)	234.1	N/A	N/A
$E[J]$ (exp.)	234.1	234.1	234.1

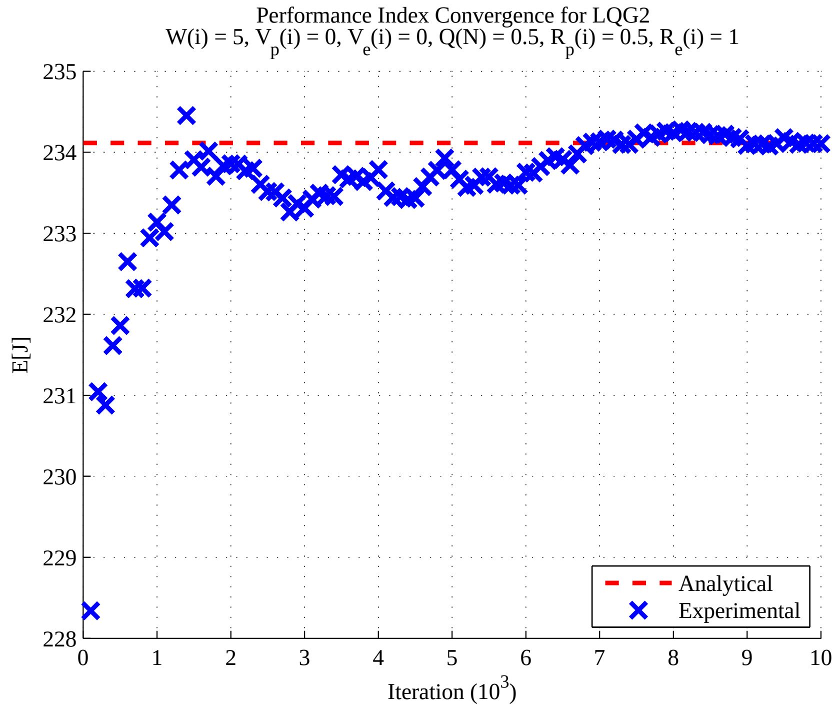

Figure 5.23: Experimental performance index convergence for Table 5.1 Monte Carlo analysis.

We now take a look at a summary of the results for several different noise combinations in Table 5.2. Note that for all cases the stochastic optimal solution (LQG2) satisfies the saddle point condition (2.5)

$$E\big[J(P\text{-}LQG2, E\text{-}LQG1)\big] \leq E\big[J(LQG2)\big] \leq E\big[J(P\text{-}LQG1, E\text{-}LQG2)\big]. \qquad (5.6)$$

Relatively speaking, it appears that the pursuer has much more to lose by playing a sub-optimal (P-LQG1) strategy as compared to the evader playing a sub-optimal (E-LQG1) strategy. For example, comparing cases #1 and #2 where one player has large measurement noise variance relative to the other player: In case #1 the performance index increases significantly for the $P\text{-}LQG1/E\text{-}LQG2$ strategy combination when the pursuer does not play his stochastic optimal strategy, whereas in case #2 the performance index decreases only slightly (beyond the first decimal place) for the $P\text{-}LQG2/E\text{-}LQG1$ strategy combination when the evader does not play his stochastic optimal strategy.

In fact, we saw in Fig. 5.5 that increases in the pursuer's measurement noise variance have a much more dramatic effect on the system gain that increases in the evader's measurement noise variance (which appear to reach a limit value). In Fig. 5.6 we saw that the pursuer's LQG2 strategy is significantly different than his LQG1 strategy (case #1), and in Fig. 5.9 we saw that the evader's LQG1 and LQG2 strategies are approximately the same (case #2). So, the Monte Carlo results re-enforce the preceding analysis that the pursuer's stochastic optimal strategy is non-traditional and provides him a significant advantage.

Comparing cases #1, #4, and #7 we can see the effect that process noise variance has on the pursuer's stochastic optimal strategy ($V_p(i) = 100$, $V_e(i) = 0.1$, and $R_p(i) = 0.5$). First, note that the performance index value for the LQG2 game decreases as process noise variance decreases, which means that a decrease in process noise variance favors the pursuer. Second, note that the penalty for the pursuer playing a sub-optimal LQG1 strategy increases as process noise variance decreases, which means that the evader has a significant advantage. So, a decrease in process noise variance is good for the pursuer *if* he plays his LQG2 strategy, otherwise, a decrease in process noise variance benefits the evader. For example, looking at case #7 the *P-LQG*1/*E-LQG*2 strategy combination results in a performance index value increase from 250.3 to 1496.4, which greatly favors the evader.

With large measurement noise variance and small process noise variance the pursuer's sub-optimal centralized Kalman filter becomes low-bandwidth, resulting in an open-loop LQG1 pursuer strategy. However, the pursuer's LQG2 strategy is not affected in the same way as we saw in Fig. 5.13 (case #7). As with case #7, we also observe a dramatic increase in the performance index value for cases #8 and #9 when the pursuer plays his sub-optimal LQG1 strategy. Therefore, we will take a closer look at a couple of these cases.

Table 5.2: Monte Carlo analysis (10k runs) for several different noise combinations.

					E[J]		
Case	$\mathbf{W(i)}$	$\mathbf{V_p(i)}$	$\mathbf{V_e(i)}$	$\mathbf{R_p(i)}$	**LQG2**	**P-LQG2/ E-LQG1**	**P-LQG1/ E-LQG2**
1	5	100	0.1	0.5	258.9	255.6	359.8
2	5	0.1	100	0.5	232.8	232.8	233.0
3	5	100	100	0.5	249.3	247.1	280.6
4	1	100	0.1	0.5	251.9	248.1	741.5
5	1	0.1	100	0.5	228.5	228.5	228.6
6	1	100	100	0.5	242.6	240.7	326.7
7	0.1	100	0.1	0.5	250.3	245.7	**1496.4**
8	5	100	0.1	0.95	1803.2	1789.4	**193238.5**
9	1	100	0.1	0.95	1786.0	1769.8	**8132545.7**

The Monte Carlo analysis results for case #7 from Table 5.2 are presented in more detail in Table 5.3. The convergence of the experimental performance index value over the 10k analysis runs is shown in Fig. 5.24.

Fig. 5.25 shows the three different strategy combinations for a single realization of the noise sequences. In other words, the initial state, process noise, pursuer measurement noise, and evader measurement noise sequences are exactly the same for these three simulation runs using the different strategy combinations. Note on the bottom plot (P-LQG1) that the pursuer is minimally responsive, whereas on the top two plots (P-LQG2), even though the pursuer has noisy measurements, he becomes much more active towards the latter stages of the game using his stochastic optimal strategy/control. Again, this behavior on the part of the pursuer is presumably to inject noise into the system in order to increase the evader's estimation error. The P-LQG1 and P-LQG2 controls are approximately the same up until stage 16. When the pursuer is using his P-LQG2 strategy the evader is unable to radically maneuver at the latter stages of the game (and drive up the performance index value), as he can against the P-LQG1 strategy.

Table 5.3: Monte Carlo (10k runs) with $W(i) = 0.1$, $V_p(i) = 100$, $V_e(i) = 0.1$, and $R_p(i) = 0.5$.

	LQG2	**P-LQG2/ E-LQG1**	**P-LQG1/ E-LQG2**
Mean	13.3	9.6	75.9
Std. Dev.	9.7	5.7	56.7
Min.	0.0	0.0	0.0
Max.	57.1	37.0	405.9
$E[J]$ (anal.)	250.1	N/A	N/A
$E[J]$ (exp.)	250.3	245.7	1496.4

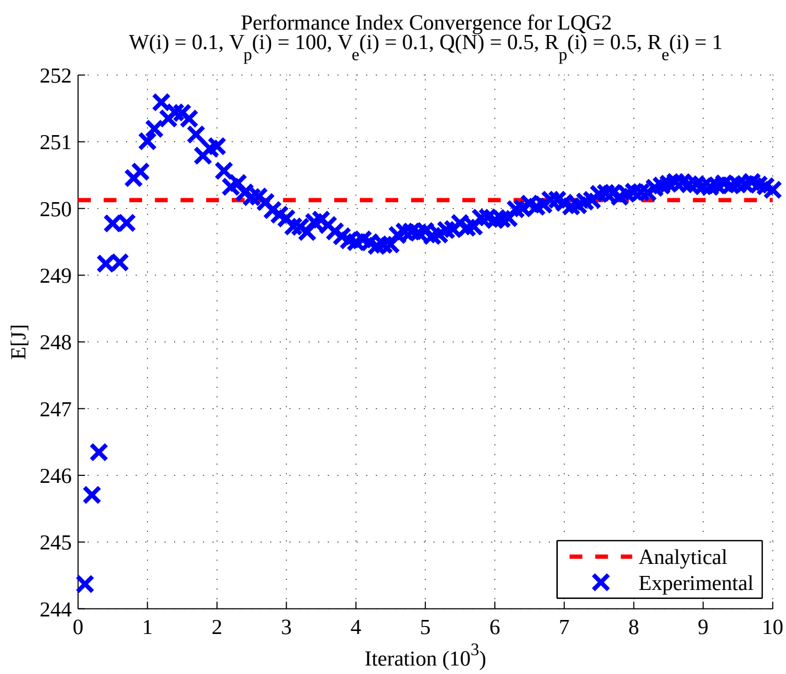

Figure 5.24: Experimental performance index convergence for Table 5.3 Monte Carlo analysis.

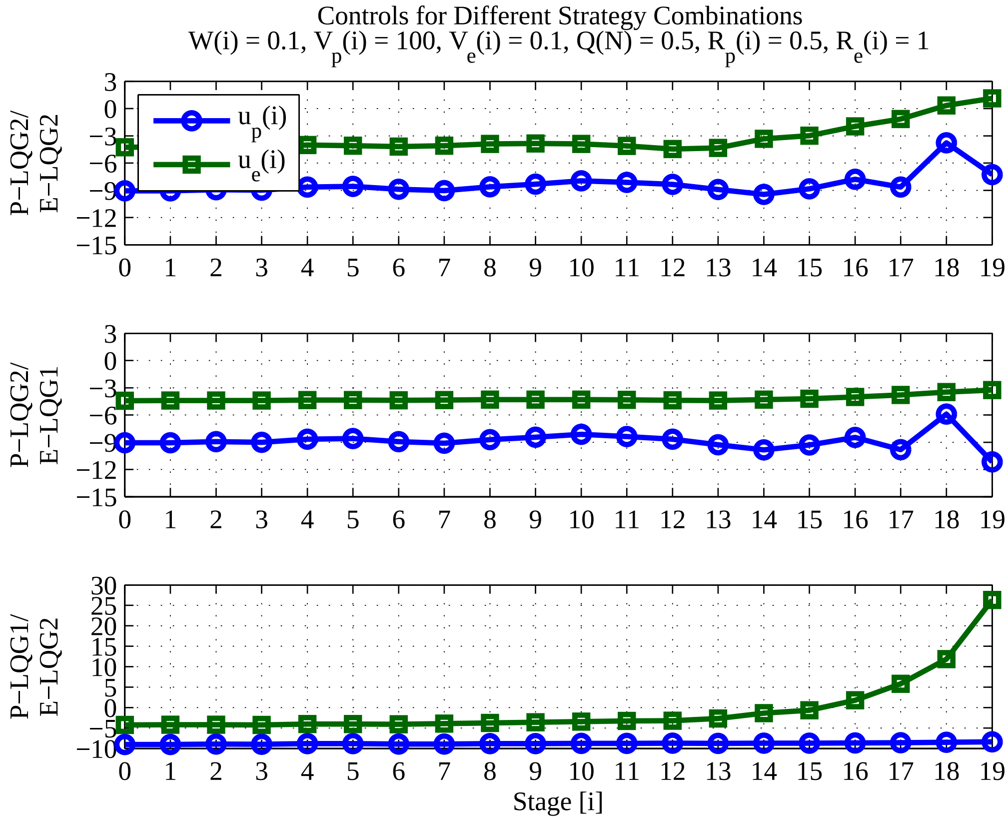

Figure 5.25: Control sequence for a single realization from Table 5.3 Monte Carlo analysis.

The Monte Carlo analysis results for case #9 from Table 5.2 are presented in more detail in Table 5.4. The convergence of the experimental performance index value over the 10k analysis runs is shown in Fig. 5.26.

As we did with case #7, Fig. 5.27 shows the three different strategy combinations for a single realization of the noise sequences for case #9. Again, note on the bottom plot that the evader is able to radically maneuver towards the latter stages of the game against the pursuer's P-LQG1 strategy. This greatly increases the miss distance and resulting performance index value as we can see in Table 5.4. Conversely, when the pursuer plays his stochastic optimal P-LQG2 strategy then his control sequence ($u_p(i)$) appears more erratic and, presumably, the evader therefore is unable to maneuver as aggressively.

Also, note that the pursuer's P-LQG2 control appears erratic throughout all stages of the game in Fig. 5.27, contrary to the P-LQG2 control in Fig. 5.25 (case #7). As we've been discussing, this is a consequence of the high stochastic system gain as shown in Fig. 5.19 (case #9).

Table 5.4: Monte Carlo (10k runs) with $W(i) = 1$, $V_p(i) = 100$, $V_e(i) = 0.1$, and $R_p(i) = 0.95$.

	LQG2	**P-LQG2/ E-LQG1**	**P-LQG1/ E-LQG2**
Mean	67.8	66.1	6978.8
Std. Dev.	36.0	18.8	5196.4
Min.	0.0	1.7	0.3
Max.	211.5	137.4	34777.2
$E[J]$ (anal.)	1786.4	N/A	N/A
$E[J]$ (exp.)	1786.0	1769.8	8132545.7

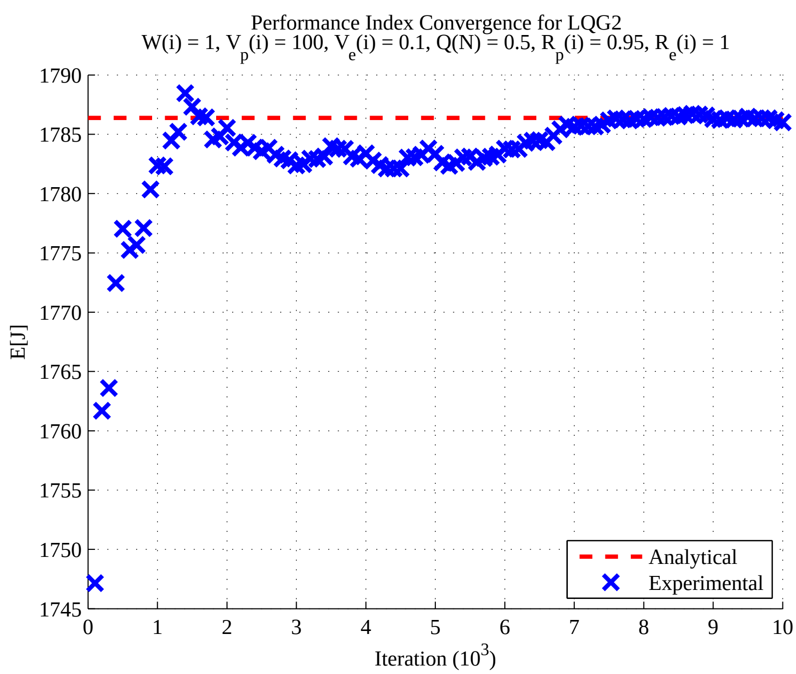

Figure 5.26: Experimental performance index convergence for Table 5.4 Monte Carlo analysis.

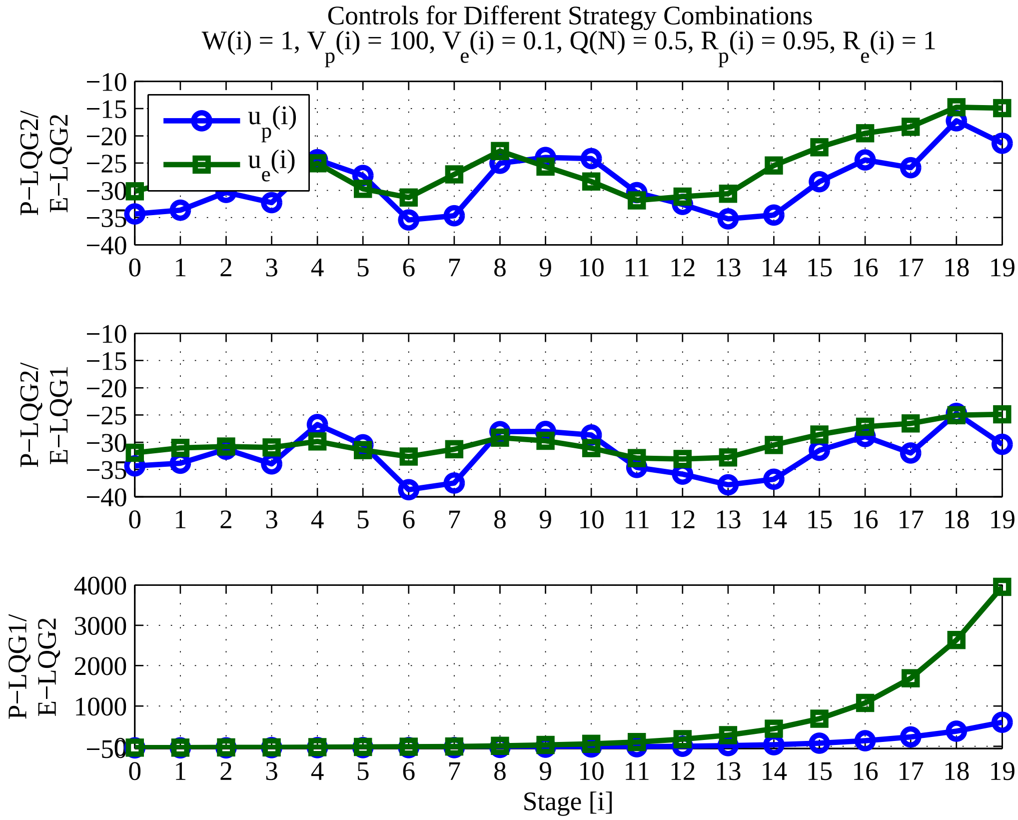

Figure 5.27: Control sequence for a single realization from Table 5.4 Monte Carlo analysis.

5.5 Summary

In summary, our Monte Carlo analysis in Section 5.4 confirmed the stochastic optimal strategy performance that we discussed in Section 5.3. This analysis provided additional insight into the controls, specifically, we observed that the pursuer's stochastic optimal control provides him a significant advantage over his *certainty equivalent sub-optimal centralized control* when he has noisy measurements relative to the evader or when he is not as maneuverable relative to the evader. Next, in Chapter 6 we will apply our stochastic optimal solutions to a missile guidance problem.

CHAPTER 6

A Pursuit-Evasion Game: Missile Guidance

Our goal in this chapter is to apply the LQG multistage optimal solution to solve the stochastic homing missile guidance endgame problem from a game-theoretic standpoint.[1] The guidance algorithm that we develop here would takeover once the pursuing missile (interceptor) is closing in on the evading missile (incoming threat) and the sensors on each missile have locked on the opposing missile. A control law would have to convert the guidance acceleration commands that we generate into actuator (e.g. attitude control motor) commands.

6.1 Background Information

Missile guidance algorithms have been of interest since the Cold War era. In fact, a lot of the early work in game theory/dynamic games taking place in the 1950s and 1960s was with application to military operations, such as Isaacs' work at The RAND Corporation [Isa55].

Fig. 6.1 shows the three different phases of flight for an incoming threat (boost, midcourse, and terminal) and the different interceptors available to the United State Missile Defense Agency depending on phase of flight. Interceptors are typically classified as either "blast fragmentation" or "hit-to-kill". With blast fragmentation the interceptor uses a warhead that detonates in close proximity to the incoming threat. With hit-to-kill the interceptor physically impacts the incoming threat, destroying it with kinetic energy alone. Hit-to-kill requires more precise guidance and control as compared to blast fragmentation, however,

[1]Note that [BH68, Section VI] solved a similar problem in continuous-time for their special information pattern.

if the interceptor can physically impact the incoming threat then there is a high chance of truly neutralizing the threat.

With either type of interceptor, the hard problem to solve is how to detect and neutralize an incoming threat that is moving several times the speed of sound. Obviously, the detection part of the problem involves imperfect information; ground support/fire control equipment and/or onboard sensors are only able to estimate the position/velocity of the incoming threat. In the presence of jamming, which leads to increased estimation error, or when the incoming threat is highly maneuverable the problem becomes even more difficult. As a result, the question we ask here is: What is the best short-duration endgame strategy (less than ten seconds to impact), from both the interceptor's viewpoint and from the incoming threat's viewpoint, given the relative measurement noises and relative maneuverability?

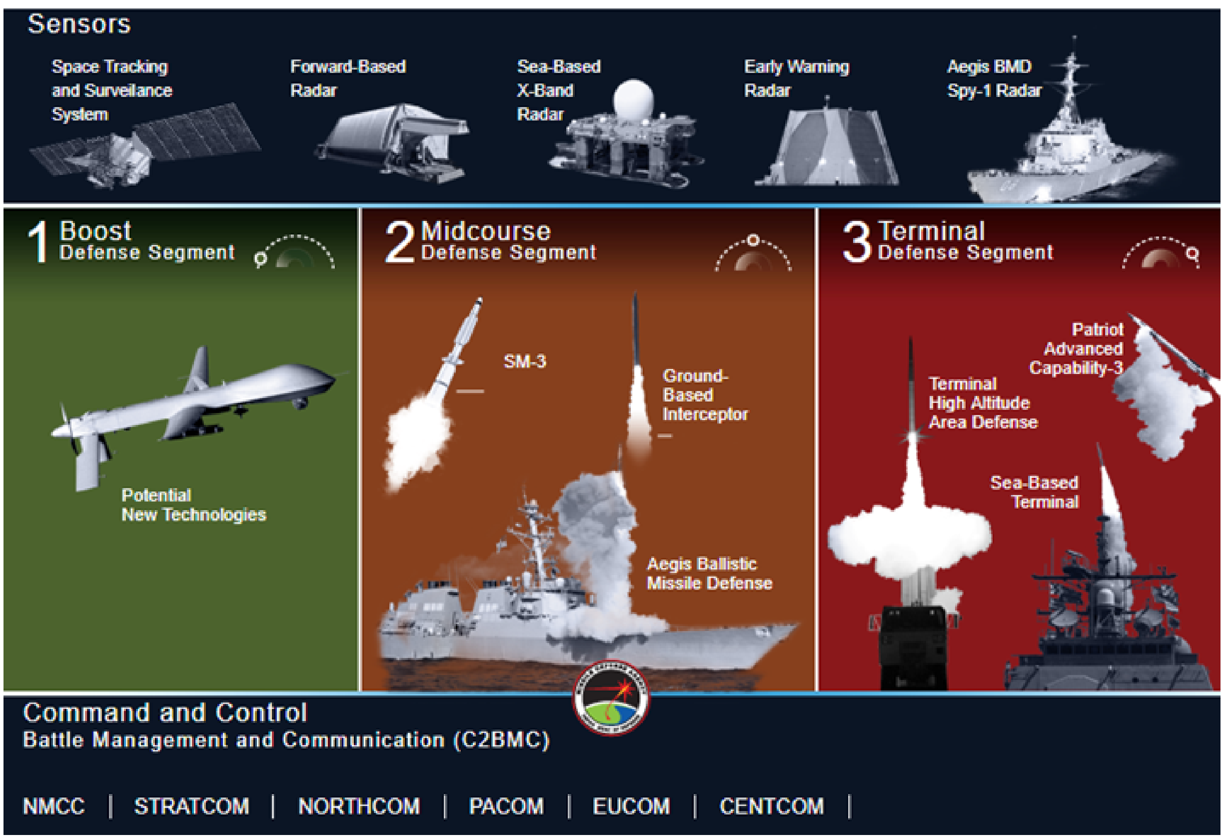

Figure 6.1: Types of intercept (pursuer) missiles by flight phase (graphic courtesy of www.mda.mil).

6.2 Problem Definition

We start out by defining the 2-D homing missile guidance problem in terms of its deterministic continuous-time representation and the dynamics associated with each player. As mentioned in Section 2.2, we define a projected-relative state-space using the state transition matrix. We then discretize the dynamics and add the applicable noise parameters in order to define the problem in terms of (2.1), (2.2), and (2.3).

Our goal here is to assess the stochastic optimal solutions as applied to a realistic problem. However, we also desire to keep the problem defined within a simplified framework so that we can easily assess the stochastic optimal solution performance. As such, we make following assumptions/restrictions:

1. Gravitational differences between the two players are negligible and can be ignored.
2. The pursuer and evader are modeled as point masses.[2]
3. The pursuer and evader can linearize around their trajectories during the short-duration endgame.
4. The pursuer and evader continuous-time matrices are time-invariant.
5. There is no actuator model included which means that the pursuer and evader are able to immediately achieve their commanded accelerations.[3]

These assumptions are in-line with accepted assumptions as listed in [BS17] and [SOT09].

The players' 2-D state dynamics include inertial x/y position, r^x and r^y, and inertial x/y velocity, v^x and v^y. This means that the state dimension is $n = 4$. Each player's control input affects their acceleration in inertial space which means that $m = l = 2$.[4] As a further

[2] As a future improvement we could include missile (autopilot) dynamics.

[3] As a future improvement we could, for example, include the time response of attitude control motors used for terminal agility.

[4] As a future improvement we could restrict control to the lateral body-axis direction (i.e. perpendicular to thrust).

indication that we are working in continuous-time we will include the time parameter, t, in parenthesis with each parameter, even if the parameter is time-invariant. Using this notation, the pursuer's deterministic 2-D state dynamics are represented as

$$\begin{bmatrix} \dot{r}_p^x(t) \\ \dot{r}_p^y(t) \\ \dot{v}_p^x(t) \\ \dot{v}_p^y(t) \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}}_{\Phi_p(t)} \underbrace{\begin{bmatrix} r_p^x(t) \\ r_p^y(t) \\ v_p^x(t) \\ v_p^y(t) \end{bmatrix}}_{x_p(t)} + \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}}_{\Gamma_p(t)} u_p(t) \tag{6.1}$$

and the evader's deterministic 2-D state dynamics are represented as

$$\begin{bmatrix} \dot{r}_e^x(t) \\ \dot{r}_e^y(t) \\ \dot{v}_e^x(t) \\ \dot{v}_e^y(t) \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}}_{\Phi_e(t)} \underbrace{\begin{bmatrix} r_e^x(t) \\ r_e^y(t) \\ v_e^x(t) \\ v_e^y(t) \end{bmatrix}}_{x_e(t)} + \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}}_{\Gamma_e(t)} u_e(t). \tag{6.2}$$

We know that for a time-invariant system matrix, Φ, we can write the state transition matrix, Ψ, from time, τ, to time, $t \geq \tau$, as

$$\Psi(t, \tau) = \Psi(t - \tau) = e^{\Phi(t-\tau)} = \sum_{k=0}^{\infty} \frac{\Phi^k}{k!}(t - \tau)^k \tag{6.3}$$

where we have written the matrix exponential in terms of its power series representation. We now define new states such that

$$\begin{aligned} \tilde{x}_p(t) &\triangleq \Psi_p(t_f, t)x_p(t) \\ \tilde{x}_e(t) &\triangleq \Psi_e(t_f, t)x_e(t) \end{aligned} \tag{6.4}$$

where Ψ_p and Ψ_e are the pursuer and evader state transition matrix, respectively, from current time, t, to final time, $t_f \geq t$. Since $\Phi_p(t) = \Phi_e(t)$ we know that $\Psi_p(t_f, t) = \Psi_e(t_f, t)$. Using (6.3), along with (6.1) and (6.2), we find the expression for the state transition matrix as [5]

$$\Psi(t_f, t) = \begin{bmatrix} 1 & 0 & (t_f - t) & 0 \\ 0 & 1 & 0 & (t_f - t) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{6.5}$$

where

$$\Psi_p(t_f, t) = \Psi_e(t_f, t) = \Psi(t_f, t).$$

Since we are ultimately interested in the relative position/velocity between the two players, we difference these new states (6.4) to define a projected-relative state-space

$$\begin{aligned} \tilde{x}(t) &\triangleq \tilde{x}_p(t) - \tilde{x}_e(t) \\ &= \Psi(t_f, t)\big(x_p(t) - x_e(t)\big). \end{aligned} \tag{6.6}$$

We can derive the differential equation governing this projected-relative state-space using (6.1) and (6.2) and the fact that

$$\frac{d}{dt}\Psi(t_f, t) = -\Psi(t_f, t)\Phi \tag{6.7}$$

as

$$\begin{aligned} \dot{\tilde{x}}(t) &= \frac{d}{dt}\Psi(t_f, t)(x_p(t)) - x_e(t)) \\ &= -\Psi(t_f, t)\Phi_p(t)x_p(t) + \Psi(t_f, t)\dot{x}_p(t) \\ &\quad + \Psi(t_f, t)\Phi_e(t)x_e(t) - \Psi(t_f, t)\dot{x}_e(t) \\ &= \Psi(t_f, t)\Gamma_p(t)u_p(t) - \Psi(t_f, t)\Gamma_e(t)u_e(t). \end{aligned} \tag{6.8}$$

Remark 6.1. *Reference [BH75, Appendix A4 and Section 9.4] for further description of the development used in this section.*

[5]Note that $(\Phi_p(t))^k$ and $(\Phi_e(t))^k$ are nilpotent for $k \geq 2$.

We can easily write the solution to (6.8) as

$$\tilde{x}(t) = \tilde{x}(t_0) + \int_{t_0}^{t} \big(\Psi(t_f,\tau)\Gamma_p(\tau)u_p(\tau) - \Psi(t_f,\tau)\Gamma_e(\tau)u_e(\tau)\big)d\tau. \tag{6.9}$$

We now wish to discretize over one sample period, Δ. That is, we want to find the state transition equation from time, $t_1 = \Delta i$, to time, $t_2 = \Delta(i+1)$. As discussed in [FPW98, Section 4.3.3] we can perform this discretization by assuming a zero-order hold on the control input over the sample period. We plug in the assumed problem parameters (i.e. $\Gamma_p(t) = \Gamma_e(t)$) to find

$$\Psi(t_f,t)\Gamma_p(t) = \Psi(t_f,t)\Gamma_e(t) = \begin{bmatrix} (t_f - t) & 0 \\ 0 & (t_f - t) \\ 1 & 0 \\ 0 & 1 \end{bmatrix}. \tag{6.10}$$

We now pull the controls outside of the integral since they are assumed constant over the sample period. This results in the following state transition equation from time, $t_1 = \Delta i$, to time, $t_2 = \Delta(i+1)$,

$$\tilde{x}(\Delta(i+1)) = \tilde{x}(\Delta i) + \int_{\Delta i}^{\Delta(i+1)} \begin{bmatrix} (t_f - \tau) & 0 \\ 0 & (t_f - \tau) \\ 1 & 0 \\ 0 & 1 \end{bmatrix} d\tau \big(u_p(\Delta i) - u_e(\Delta i)\big). \tag{6.11}$$

Define $\eta \triangleq t_f - \tau$ so that the integration now appears as

$$\tilde{x}(\Delta(i+1)) = \tilde{x}(\Delta i) + \int_{t_f-\Delta(i+1)}^{t_f-\Delta i} \begin{bmatrix} \eta & 0 \\ 0 & \eta \\ 1 & 0 \\ 0 & 1 \end{bmatrix} d\eta \big(u_p(\Delta i) - u_e(\Delta i)\big) \tag{6.12}$$

which has the solution

$$\tilde{x}(\Delta(i+1)) = \tilde{x}(\Delta i) + \underbrace{\begin{bmatrix} \frac{(t_f-\Delta i)^2-(t_f-\Delta(i+1))^2}{2} & 0 \\ 0 & \frac{(t_f-\Delta i)^2-(t_f-\Delta(i+1))^2}{2} \\ \Delta & 0 \\ 0 & \Delta \end{bmatrix}}_{\tilde{\Gamma}(\Delta i)} \big(u_p(\Delta i) - u_e(\Delta i)\big). \tag{6.13}$$

Dropping the sample period, Δ, we can now write the multistage form of projected-relative state dynamics (6.8) as

$$\tilde{x}(i+1) = \tilde{x}(i) + \tilde{\Gamma}(i)\big(u_p(i) - u_e(i)\big). \tag{6.14}$$

This state transition equation is still written in its deterministic form (without process noise). In the following section we will introduce the process and measurement noises for our particular problem.

6.3 Analysis Setup

Typically it is assumed that the forward speed is constant over the short-duration endgame and each player applies an acceleration in the lateral body-axis direction, perpendicular to thrust. Note that our governing equations (6.1) and (6.2) allow the players to apply an acceleration in the inertial x/y direction without regard for body-axis acceleration direction. In order to simplify our analysis and negate the need for a rotation from body-axis to inertial-axis, we initialize the problem in such a way as to approximately enforce the lateral body-axis acceleration direction.

Fig. 6.2 shows the initial condition geometry for our analysis. Each player has an initial velocity that is along the inertial x-axis, and the players are offset some distance apart along the inertial y-axis. In a sense, we have defined the problem in 2-D, which allows for future analysis capabilities, but we have forced the results to be approximately 1-D based on the initial conditions. This means that although our controls are 2-D, we should expect that only the inertial y-axis control will be non-zero (and, indeed, this is the case in the results that follow).

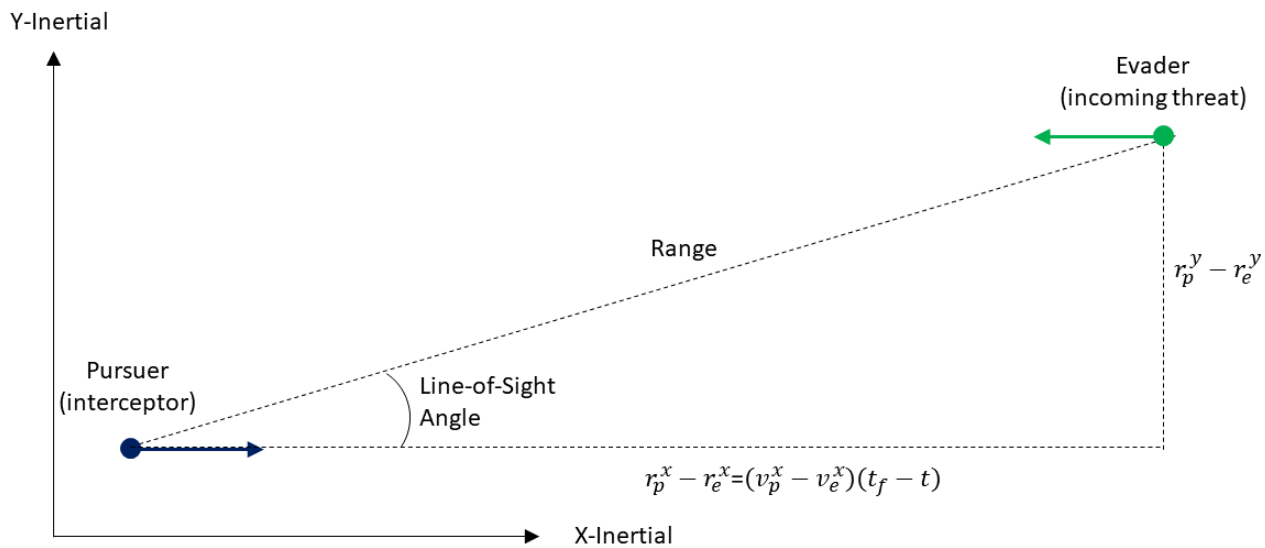

Figure 6.2: Missile guidance endgame problem, initial condition geometry.

We now define the nonclassical information pattern involved. Each player makes an independent, noise-corrupted, measurement of the relative inertial y-position and relative inertial y-velocity.[6] The measurement dimensions are therefore $p = q = 2$. We use the inverse state transition matrix (6.5) as part of the measurement matrices in order to transform from our projected-relative state-space back into physical-relative states that can be sensed/measured.

[6]As a future improvement we could allow each player only a line-of-sight angle (or relative inertial y-position) measurement (no rate measurement).

The independent measurements are therefore expressed as

$$z_p(i) = \underbrace{\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \Psi^{-1}(t_f, \Delta i)}_{\tilde{\Theta}_p(i)} \tilde{x}(i) + v_p(i)$$
$$= \tilde{\Theta}_p(i)\tilde{x}(i) + v_p(i) \tag{6.15}$$

$$z_e(i) = \underbrace{\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \Psi^{-1}(t_f, \Delta i)}_{\tilde{\Theta}_e(i)} \tilde{x}(i) + v_e(i)$$
$$= \tilde{\Theta}_e(i)\tilde{x}(i) + v_e(i) \tag{6.16}$$

which appear exactly as (2.2) and (2.3). Lastly, we add process noise to the deterministic dynamics (6.14) to get

$$\tilde{x}(i+1) = \tilde{x}(i) + \tilde{\Gamma}(i)u_p(i) - \tilde{\Gamma}(i)u_e(i) + w(i) \tag{6.17}$$

which appears exactly as (2.1). Note that the measurement noises are in terms of the physical-relative states, but the process noise is in terms of the projected-relative states.

We now consider a missile guidance endgame problem with $t_f = 5$ sec and a sample time of $\Delta = 0.1$ sec. This leads to a multistage game with $N = 50$ stages. The problem parameters are as follows (all units are in meters and seconds): [7]

$$E[x_p(0)] = \begin{bmatrix} \bar{r}_p^x(0) \\ \bar{r}_p^y(0) \\ \bar{v}_p^x(0) \\ \bar{v}_p^y(0) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1000 \\ 0 \end{bmatrix}$$

[7] As a future improvement we could more accurately model the *a priori* state variance, $E[\tilde{x}(0)\tilde{x}^T(0)]$, and process noise variance, $W(i)$, for the projected-relative state-space. For the sake of simplicity in our analysis we assume these are diagonal matrices.

$$E[x_e(0)] = \begin{bmatrix} \bar{r}_e^x(0) \\ \bar{r}_e^y(0) \\ \bar{v}_e^x(0) \\ \bar{v}_e^y(0) \end{bmatrix} = \begin{bmatrix} 10000 \\ 1000 \\ -1000 \\ 0 \end{bmatrix}$$

$$E[\tilde{x}(0)] = \Psi(5,0)\big(E[x_p(0)] - E[x_e(0)]\big)$$

$$= \begin{bmatrix} 0 \\ -1000 \\ 2000 \\ 0 \end{bmatrix}$$

$$E[\tilde{x}(0)\tilde{x}^T(0)] = \begin{bmatrix} 0.01 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 \\ 0 & 0 & 0.01 & 0 \\ 0 & 0 & 0 & 10 \end{bmatrix}$$

$$W(i) = \begin{bmatrix} 0.01 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0.01 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$Q(N) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$R_p(i) = \begin{bmatrix} 0.25 & 0 \\ 0 & 0.25 \end{bmatrix}$$

$$R_e(i) = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}.$$

Note that $Q(N)$ only has a weighting on the final relative y-position. For measurement noise, we use the angle measurement sensor model as specified in [SKT90, Eqn. 55], converted to distance as follows

$$V_{dist}(i) = \Big(0.25 + 5.625 * 10^{-7}\big((\bar{v}_p^x(0) - \bar{v}_e^x(0))(t_f - \Delta i)\big)^2\Big)/\Delta. \tag{6.18}$$

Therefore, each player's measurement noise variance (units of m^2 and $(m/s)^2$) appears as [8]

$$V_p(i) = k_{V_p}\begin{bmatrix}\Big(0.25 + 5.625 * 10^{-7}\big((\bar{v}_p^x(0) - \bar{v}_e^x(0))(t_f - \Delta i)\big)^2\Big)/\Delta & 0 \\ 0 & 1\end{bmatrix} \tag{6.19}$$

$$V_e(i) = k_{V_e}\begin{bmatrix}\Big(0.25 + 5.625 * 10^{-7}\big((\bar{v}_p^x(0) - \bar{v}_e^x(0))(t_f - \Delta i)\big)^2\Big)/\Delta & 0 \\ 0 & 1\end{bmatrix} \tag{6.20}$$

where, nominally, $k_{V_p} = k_{V_e} = 1$. We use these factors to adjust one player's measurement noise variance relative to the other player.

6.4 Monte Carlo Analysis

We now use the problem setup of the previous section to perform a Monte Carlo analysis with variations in measurement noise variance and evader maneuverability. As we did in Section 5.4, we use Definitions 5.2 and 5.3 in order to compare the performance of the stochastic optimal strategies with the performance of the *certainty equivalent sub-optimal centralized strategies*. We compare statistics for four different strategy combinations: LQG1 (both players use their *certainty equivalent sub-optimal centralized strategy*), LQG2 (both players use their stochastic optimal strategy), P-LQG2 vs. E-LQG1 (evader is sub-optimal), and P-LQG1 vs. E-LQG2 (pursuer is sub-optimal).

The mean, standard deviation, minimum, and maximum values presented in the following tables are calculated with respect to the relative inertial y-position at closest approach. That is, we are looking at the inertial y miss distance.

[8] In reality, the rate measurement noise variance and position noise variance would be correlated. For the sake of simplicity in our analysis we assume the measurement noise variance matrices are diagonal.

6.4.1 Same Measurement Noise Variance, Evader Low Maneuverability

We start off by analyzing the stochastic missile guidance problem using the default parameters outlined above.[9] Fig. 6.3 shows the LQD state trajectories (i.e. the *corresponding deterministic game* state trajectories). The initial line-of-sight (ILOS) is 5.7 deg, and at the final stage of the game the y miss distance is 0.7 m (x miss distance is 0 m). Note in the lower-right subplot of Fig. 6.3 that, as expected due to our initial conditions, the players use only inertial y-axis acceleration/control (first element of the control vectors is zero, second element of the control vectors is non-zero).

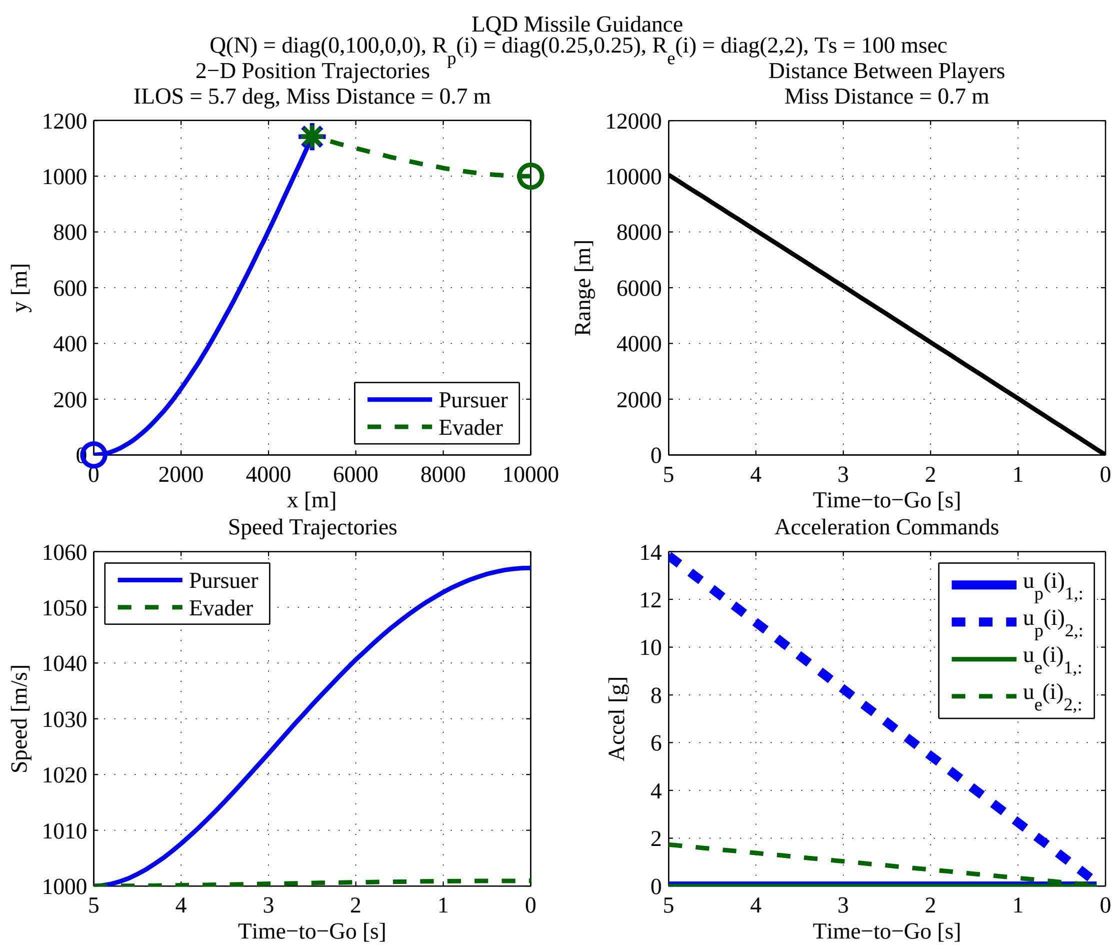

Figure 6.3: Missile guidance deterministic state trajectory for $R_e/R_p = 8$.

[9] With a slight abuse of notation we will indicate this relative maneuverability in our figures/tables as "$R_e/R_p = 8$".

Table 6.1 shows the miss distance statistics from a 10k run Monte Carlo analysis using the four different strategy combinations. Note that the LQG1 strategy combination is *not* a saddle point solution; the pursuer can play his P-LQG2 strategy and decrease the miss distance standard deviation from 3.4 m to 2.5 m, or the evader can play his E-LQG2 strategy and increase the miss distance standard deviation from 3.4 m to 4.9 m. Conversely, note that the LQG2 strategy *is* a saddle point solution - if either player deviates from his LQG2 strategy then the miss distance changes in favor of the other player. The magnitude of this change is asymmetric; the pursuer has more to lose by not playing his stochastic optimal strategy. For the P-LQG1/E-LQG2 strategy combination the miss distance standard deviation almost doubles from its LQG2 value of 2.6 m to 4.9 m.

Table 6.1: Missile guidance Monte Carlo (10k runs) with $R_e/R_p = 8$ and $k_{V_p} = 1$.

	LQG1	**LQG2**	**P-LQG2/ E-LQG1**	**P-LQG1/ E-LQG2**
Mean (m)	-0.7	-0.7	-0.7	-0.7
Std. Dev. (m)	3.4	2.6	2.5	4.9
Min. (m)	0.0	0.0	0.0	0.0
Max. (m)	12.8	10.8	10.4	19.3

The scatter plot in Fig. 6.4 graphically shows the saddle point property of the stochastic optimal solutions. In particular, notice the right subplot; when the pursuer plays his suboptimal P-LQG1 strategy the evader is able to double the miss distance standard deviation.

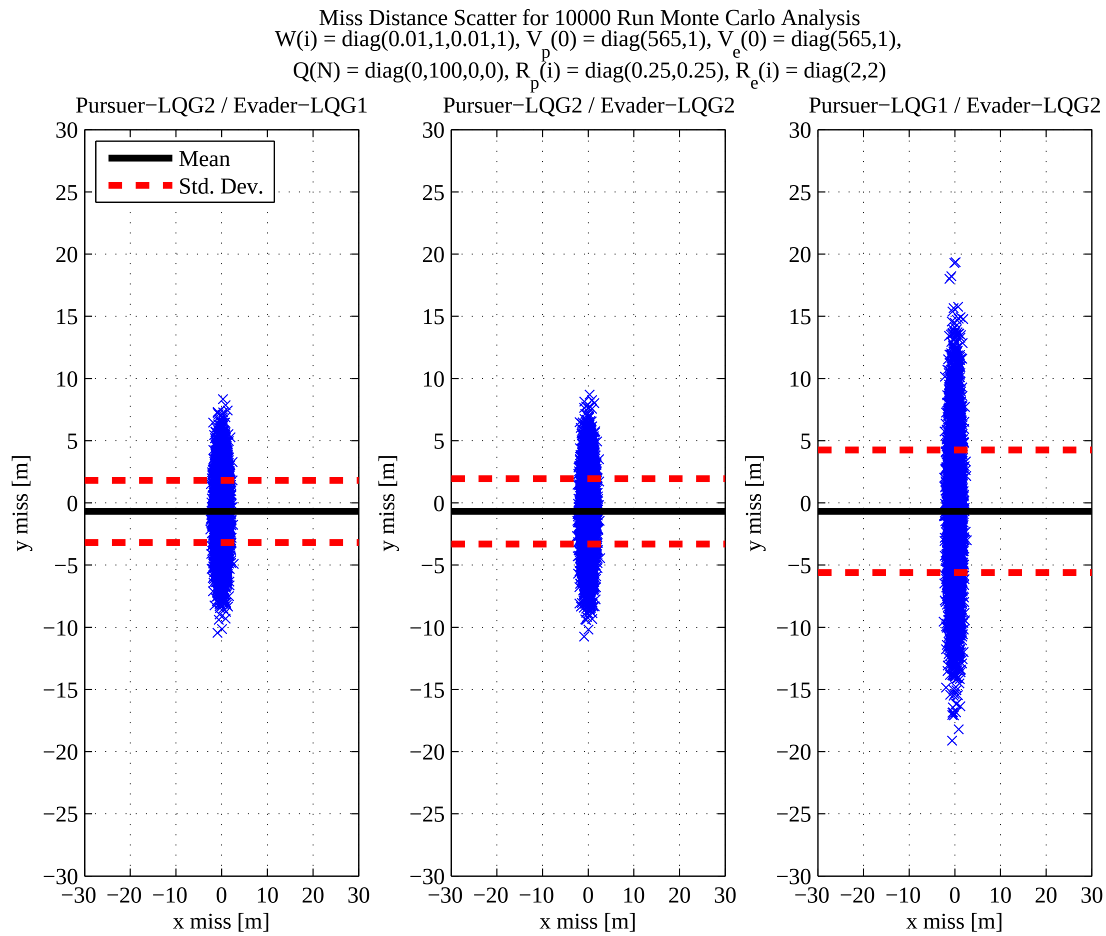

Figure 6.4: Miss distance scatter plot for Table 6.1 Monte Carlo analysis.

Fig 6.5 shows the control sequences for a single realization. Note that when the pursuer is playing his stochastic optimal P-LQG2 strategy that he appears to maneuver within approximately 1 sec time-to-go (decreasing towards 0 g's then abruptly increasing towards 3 g's), whereas when he plays his sub-optimal P-LQG1 strategy (bottom subplot) this maneuvering is not as defined. For this particular realization, when the pursuer plays his P-LQG1 strategy the y miss distance magnitude is 3.7 m, whereas when he plays his P-LQG2 strategy the y miss distance magnitude is <2.3 m. So, the abrupt maneuvering on the part of the pursuer when he uses his stochastic optimal strategy appears to help in minimizing the miss distance.

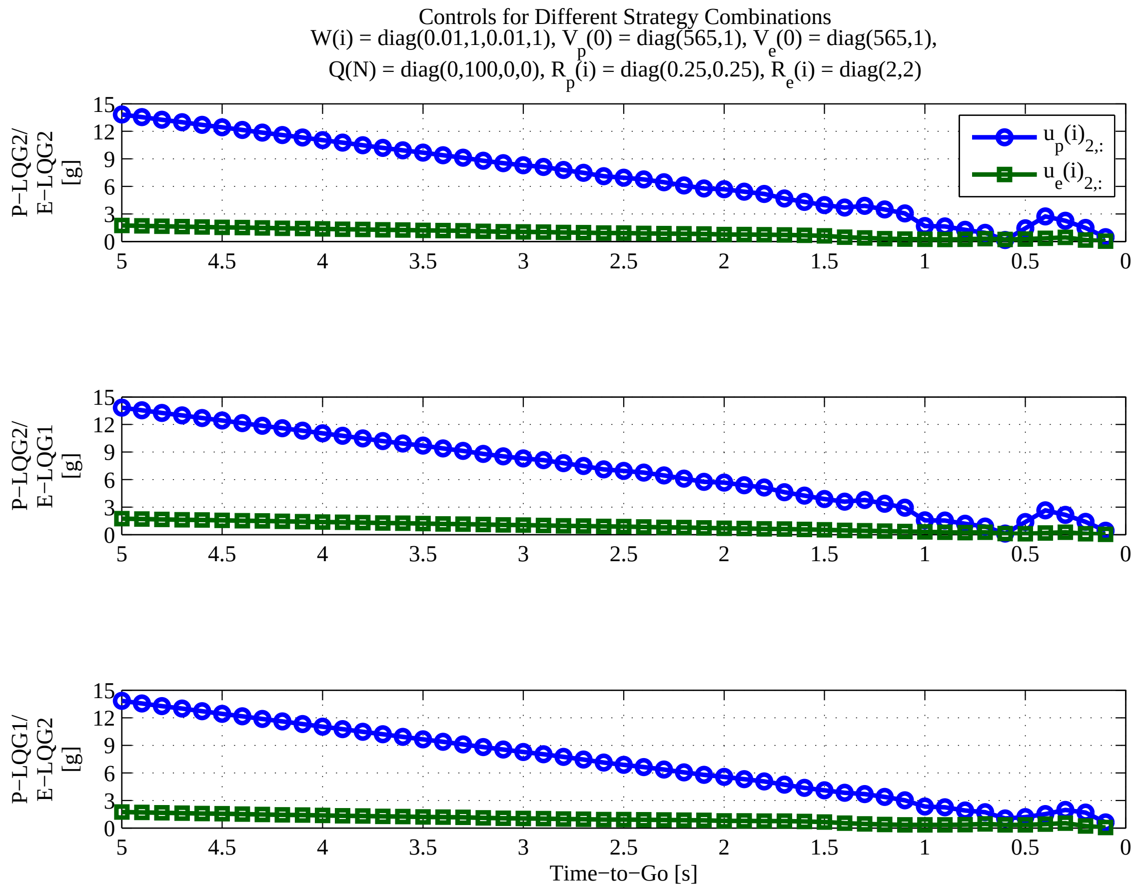

Figure 6.5: Control sequence for a single realization from Table 6.1 Monte Carlo analysis.

6.4.2 Pursuer Increased Measurement Noise Variance, Evader Low Maneuverability

We now look at the case where the pursuer's measurement noise variance is twice as large as the evader's ($k_{V_p} = 2$). Situations such as this could be representative of jamming which increases the pursuer's measurement uncertainty, and, hence, estimation error.

Table 6.2 shows the miss distance statistics from a 10k run Monte Carlo analysis using the four different strategy combinations. Note that when the pursuer plays his sub-optimal P-LQG1 strategy against the evader's optimal E-LQG2 strategy the miss distance standard deviation increases greater than threefold relative to the LQG2 strategy combination.

Table 6.2: Missile guidance Monte Carlo (10k runs) with $R_e/R_p = 8$ and $k_{V_p} = 2$.

	LQG1	**LQG2**	**P-LQG2/ E-LQG1**	**P-LQG1/ E-LQG2**
Mean (m)	-0.7	-0.7	-0.7	-0.7
Std. Dev. (m)	4.7	3.0	2.8	10.3
Min. (m)	0.0	0.0	0.0	0.0
Max. (m)	18.4	12.6	12.0	41.7

Fig. 6.6 shows the miss distance scatter plots for the different strategy combinations, which reflects the statistical results in Table 6.2. As with the previous example, in Fig. 6.7 we see that when the pursuer is playing his stochastic optimal P-LQG2 strategy that he appears to abruptly maneuver within approximately 1 sec time-to-go. For this particular realization, when the pursuer plays his P-LQG1 strategy the y miss distance magnitude is 5.5 m, whereas when he plays his P-LQG2 strategy the y miss distance magnitude is <2.1 m. Again, the abrupt maneuvering on the part of the pursuer when he uses his stochastic optimal strategy appears to help in minimizing the miss distance.

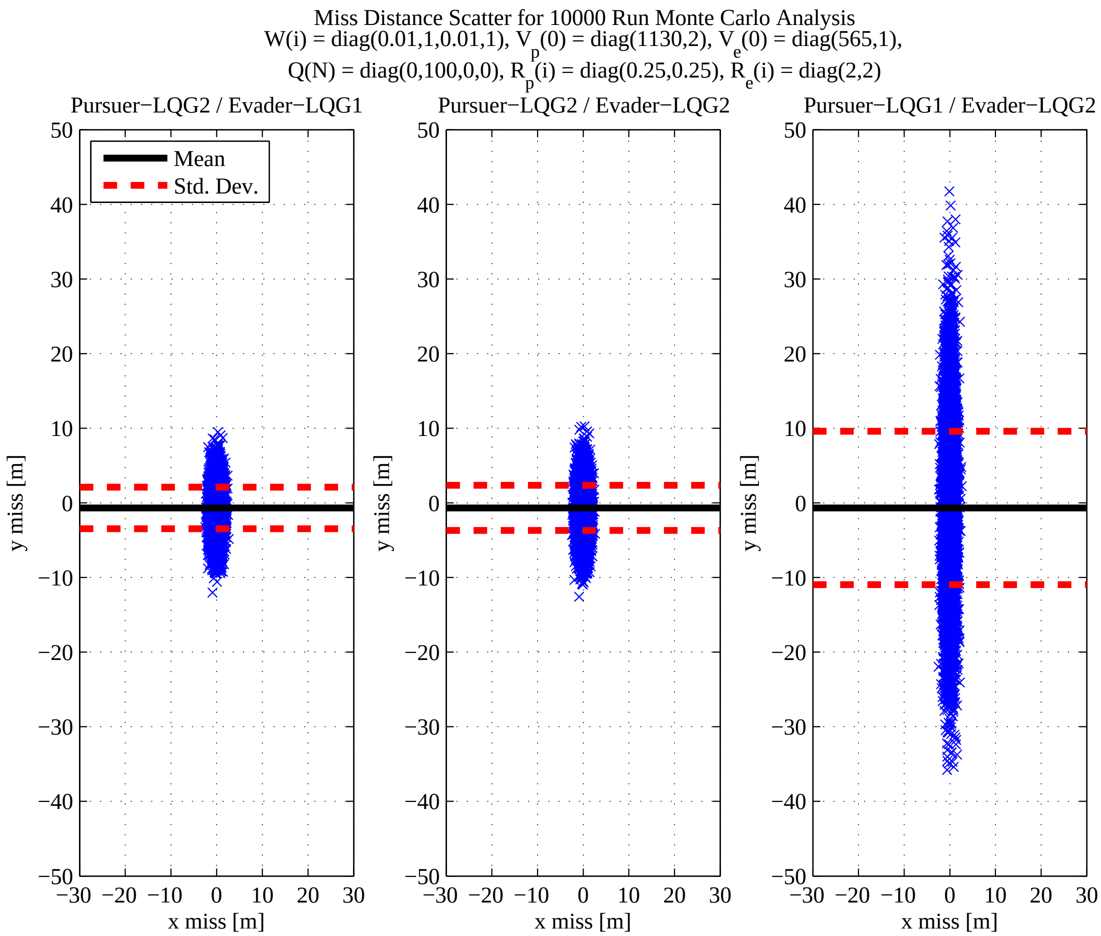

Figure 6.6: Miss distance scatter plot for Table 6.2 Monte Carlo analysis.

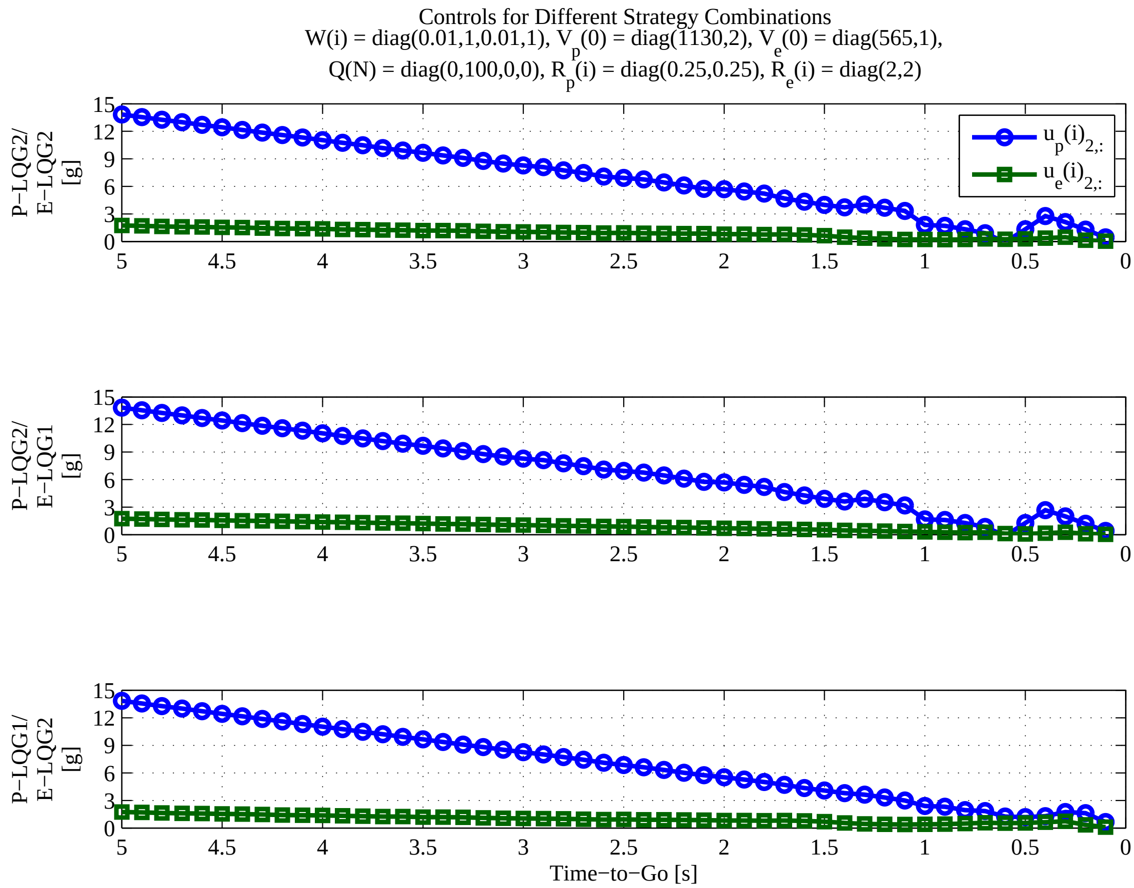

Figure 6.7: Control sequence for a single realization from Table 6.2 Monte Carlo analysis.

6.4.3 Same Measurement Noise Variance, Evader Increased Maneuverability

Finally, we look at the case where evader is not as heavily penalized in terms of his performance index control weighting. Here we will investigate the case where the pursuer is only twice as maneuverable as the evader. We keep $R_p(i)$ the same, but we change the evader's control weighting to be [10]

$$R_e(i) = \begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix}.$$

[10] With a slight abuse of notation we will indicate this relative maneuverability in our figures/tables as "$R_e/R_p = 2$".

This allows the evader to increase his acceleration commands and possibly increase the miss distance.

As discussed in [SOT09], this scenario is representative of trying to intercept a highly maneuverable tactical ballistic missile (TBM). TBMs are difficult to intercept due to insufficient maneuverability advantage on the part of the pursuer (interceptor), in addition to the inherent imperfect information/estimation error.

Fig. 6.8 shows the LQD state trajectories for this missile guidance problem. Note in the lower-right subplot that, as expected, the evader is able to pull half as many g's as the pursuer. Comparing with our previous results in Fig. 6.3 we can also see that the pursuer maneuvers almost twice as much (compared to when the evader had a much higher control weighting).

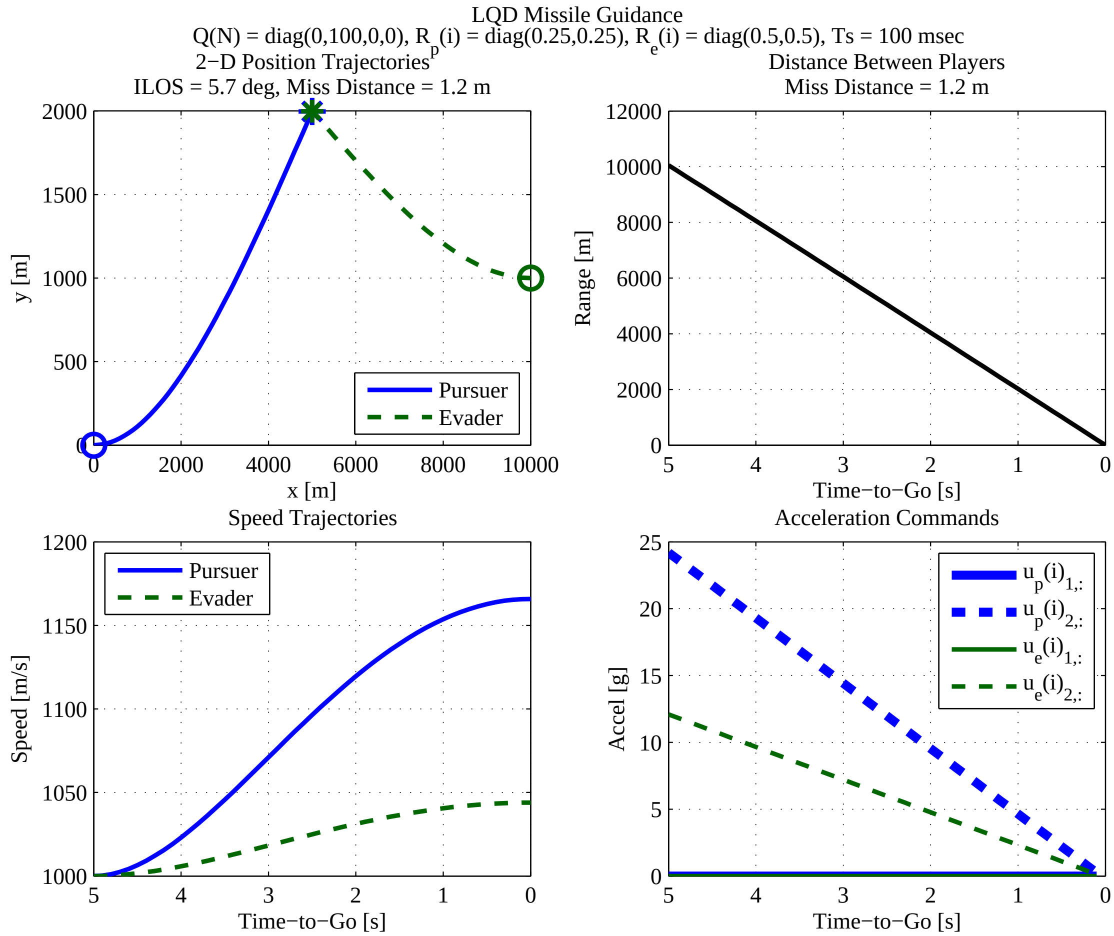

Figure 6.8: Missile guidance deterministic state trajectory for $R_e/R_p = 2$.

Table 6.3 shows the miss distance statistics from a 10k run Monte Carlo analysis using the four different strategy combinations. Note that when the pursuer plays his sub-optimal P-LQG1 strategy against the evader's optimal E-LQG2 strategy the miss distance standard deviation increases two-orders of magnitude relative to the LQG2 strategy combination. This is the most stark contrast we've seen in these missile guidance examples between P-LQG1 and P-LQG2 strategy performance. When the pursuer plays his P-LQG2 strategy he is guaranteed a miss distance standard deviation <3.3 m, whereas when he plays his P-LQG1 strategy the miss distance standard deviation could increase to 173.9 m (assuming that the evader plays his optimal E-LQG2 strategy).

Table 6.3: Missile guidance Monte Carlo (10k runs) with $R_e/R_p = 2$ and $k_{V_p} = 1$.

	LQG1	**LQG2**	**P-LQG2/ E-LQG1**	**P-LQG1/ E-LQG2**
Mean (m)	-1.2	-1.2	-1.2	-3.5
Std. Dev. (m)	4.0	3.3	2.7	173.9
Min. (m)	0.0	0.0	0.0	0.0
Max. (m)	15.2	14.4	12.1	601.8

Fig. 6.9 shows the miss distance scatter plots for the different strategy combinations, which reflects the statistical results in Table 6.3. The asymmetric y miss distance for the P-LQG1/E-LQG2 strategy combination (right subplot) is due to the fact that the x miss distance is non-zero mean; there are some cases outside the plot bounds at x = -200 m. This is a consequence of the evader being able to aggressively maneuver and hence influence not only y miss distance, but x miss distance as well. The reason the additional cases are centered around -200 m is due to our sample time of $\Delta = 0.1$ sec and the relative velocity of 2000 m/s. So, over one sample period the players move 200 m in the inertial x direction.

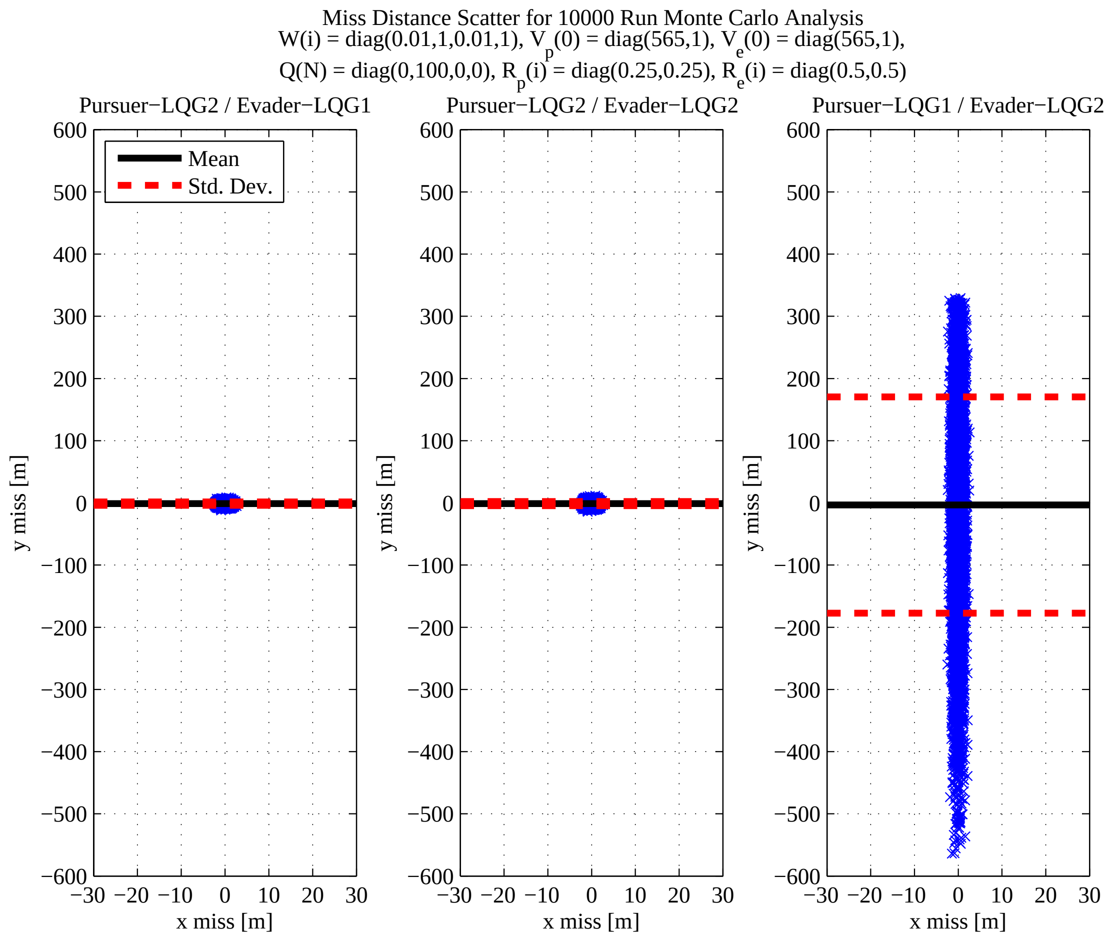

Figure 6.9: Miss distance scatter plot for Table 6.3 Monte Carlo analysis.

In Fig. 6.10 we see that when the pursuer is playing his sub-optimal P-LQG1 strategy (bottom subplot) the evader is able to aggressively maneuver within approximately 1 sec time-to-go and greatly increase the miss distance. The evader's maneuver appears to lead the pursuer's response in this case. Conversely, when the pursuer is playing his stochastic optimal P-LQG2 strategy he appears to abruptly maneuver, with relatively little response from the evader, within approximately 1 sec time-to-go. For this particular realization, when the pursuer plays his P-LQG1 strategy the y miss distance magnitude is 29 m, whereas when he plays his P-LQG2 strategy the y miss distance magnitude is <3.2 m.

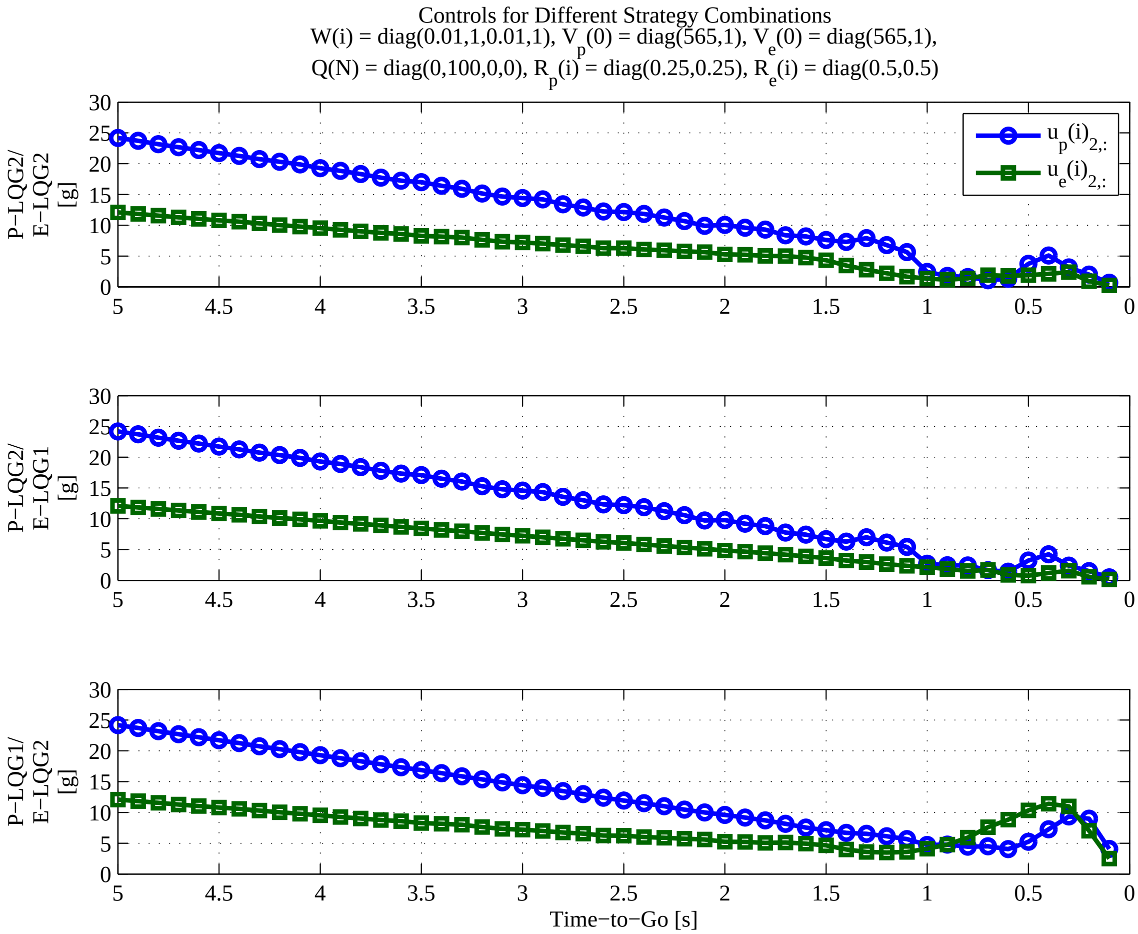

Figure 6.10: Control sequence for a single realization from Table 6.3 Monte Carlo analysis.

6.5 Summary

In summary, we have demonstrated the dramatic performance improvement for the pursuer (interceptor) when he uses his stochastic optimal missile guidance strategy (P-LQG2) relative to his *certainty equivalent sub-optimal centralized missile guidance strategy* (P-LQG1). As we also saw in our Section 5.4 study, this increase in performance is even more apparent when the pursuer's measurement noise is increased relative to the evader, and especially when the pursuer is only twice as maneuverable as the evader. Therefore, our stochastic optimal missile guidance strategy shows great promise for environments where the pursuer's measurements are corrupted and the estimation error is high, as well as, environments where the incoming threat is highly maneuverable (such as trying to intercept TBMs).

As we've noted throughout this chapter, there are several areas within our analysis that could be further developed in order to add more realism to the missile guidance results. In particular, we could allow only lateral body-axis accelerations, we could more accurately model the *a priori* state variance and process noise variance for the projected-relative state-space, and we could use only line-of-sight angle (or relative inertial y-position) measurement (no rate measurement). These are all candidate enhancements for future research.

CHAPTER 7

Concluding Remarks

In this dissertation we have shown that by forming an enlarged state-space the LQG multistage game with nonclassical information pattern may be solved as a deterministic two-sided optimization problem (Section 4.2). The advantage of using this direct solution method is that it leads to a natural decomposition into deterministic and stochastic terms in the optimal control strategies (Section 4.4). We proved that the derived affine strategies are indeed optimal since they form a saddle point solution (Section 4.6). In contrast to Willman's formal solution method, we also derived expressions for the optimal performance index value and optimal performance index differential, both of which provide insight into the stochastic nature of the problem (Section 4.7).

We then used the optimal control strategies to study the impacts of noise and relative maneuverability on strategy performance. Specifically, we saw in Chapter 5 that the pursuer's stochastic optimal control strategy provides him a significant advantage over his *certainty equivalent sub-optimal centralized strategy*. We looked at this in terms of the Lagrange multiplier sequence ($S_2(i)$) behavior, as well as, the pursuer and evader optimal control kernels ($K_p(i)$ and $K_e(i)$, respectively). As the pursuer's measurement noise variance increases, or as the pursuer becomes less maneuverable relative to the evader, he appears to inject noise into the system so as to increase the evader's estimation error. We demonstrated the effects of this stochastic optimal strategy by way of several different Monte Carlo examples.

Finally, we applied the stochastic optimal control strategies to three different missile guidance problems in Chapter 6. Although we made several simplifying assumptions along the way, the results of our Monte Carlo analysis show significant performance advantage for the pursuer when using his stochastic optimal strategy. In fact, our results show great promise for intercept guidance laws in the presence of large measurement uncertainty facing highly maneuverable incoming threats.

In summary, the contribution of this dissertation includes expressions for the LQG multistage game saddle point strategies and optimal performance index value, as well as, numerical results and analysis for stochastic systems with nonclassical information patterns that demonstrate the optimality of these strategies.

7.1 Areas for Continued Research

As with any research endeavor, while investigating and learning more about the problem at hand, we naturally ask additional questions and see potential for future research directions. Accordingly, throughout the development of this work we've noted several areas for continued research that we shall outline here for the interested reader.

1. Starting with the optimal strategies in Section 4.2 and our analysis in Section 4.4, investigate alternate forms of the optimal strategies that provide further insight/intuition into each player's decision making.

2. The optimal strategies in Section 4.2 become unwieldy with increasing number of stages. Consider scheduling sample time as a function of time-to-go so that the number of stages (i.e. storage/memory required) is minimized for a particular discrete-time application. That is, decrease the sample time as time-to-go decreases and a higher-fidelity solution is required towards the critical final time.

3. In addition, the enlarged matrices in Section 4.2 are in general sparse and/or symmetric. For example, the symmetric $S_2(i)$ and $\mathbf{P}(i)$ matrices start out sparse at stage 0 and become more dense as the stages progress. Consider memory/storage optimization techniques that could take advantage of this sparse and symmetric structure.

4. Reduced-dimension, sub-optimal strategies could be obtained by defining an appropri-

ate time constant used to discard smoothing terms in the optimal strategies. These reduced-dimension, sub-optimal strategies might still perform significantly better than the *certainty equivalent sub-optimal centralized strategies*, while at the same time requiring significantly less memory/storage than the optimal strategies.

5. As mentioned in Section 4.3, a more robust and efficient method is warranted to solve the two-point boundary value problem of Section 4.2. This could include quasilinearization, or possibly starting with a smaller-stage game and using that solution to initialize the next larger-stage game, building up to the complete N-stage game.

6. Continue to investigate the solution existence criteria of Section 4.5. Specifically, investigate the effect of process and measurement noise variances on the convexity and concavity conditions.

7. Apply dynamic programing to further validate our claim of global optimality in Section 4.6.

8. Use the methodology outlined in Section 4.8 to show the optimal strategy differences for the multistage version of the Behn and Ho game [BH68] when the perfect-information pursuer has a one-stage delay vs. no-delay information pattern. Take the discrete-to-continuous limit for these two different information patterns and compare with the Behn and Ho continuous-time solution.[1]

9. Enhance the application of our missile guidance strategies by implementing the suggested improvements detailed throughout Chapter 6.

10. Consider applying our enlarged state-space methodology to multi-player games. That is, consider the case where one group of cooperative players is competing against another group of cooperative players (e.g. wartime strategies).

[1]Reference the conclusions drawn in [Beh68, Appendix III-A].

11. Consider applying our enlarged state-space methodology to the cooperative control problem. That is, consider the case where the sign on $u_e(i)$ in the performance index is positive.

12. Consider applying our enlarged state-space methodology to nonlinear applications, such as aircraft evasive maneuvering and/or dog-fight maneuvering.

APPENDIX A

An Alternate Form of the Kalman Filter: The Enlarged Kalman Filter

A.1 Traditional Form of the Kalman Filter

Consider the multistage system

$$\begin{aligned} x(i+1) &= \Phi(i)x(i) + \Gamma(i)u(i) + w(i) \\ z(i) &= \Theta(i)x(i) + v(i) \end{aligned} \tag{A.1}$$

for $i = 0, 1, ..., N-1$. The parameters that appear in this system are defined in Table A.1. All random variables are modeled as delta-correlated Gaussian noises; all noises are

Table A.1: Parameter definitions for Kalman filter example.

Parameter	**Description**
$x(i) \in \mathbb{R}^n$	State
$\Phi(i) \in \mathbb{R}^{n \times n}$	System Matrix
$\Gamma(i) \in \mathbb{R}^{n \times m}$	Control Matrix
$u(i) \in \mathbb{R}^m$	Control
$w(i) \in \mathbb{R}^n$	Process Noise
$z(i) \in \mathbb{R}^p$	Measurement
$\Theta(i) \in \mathbb{R}^{p \times n}$	Measurement Matrix
$v(i) \in \mathbb{R}^p$	Measurement Noise

uncorrelated with the initial state:

$$\begin{aligned} x(0) &\sim N(\bar{x}(0), M(0)) \\ w(i) &\sim N(0, W(i)) \\ v(i) &\sim N(0, V(i)). \end{aligned}$$

The *a priori* (pre-measurement update) Kalman filter error, $\bar{e}(i)$, and *a posteriori* (post-measurement update) Kalman filter error, $\hat{e}(i)$, are defined as

$$\begin{aligned} \bar{e}(i) &\triangleq x(i) - \bar{x}(i) \\ \hat{e}(i) &\triangleq x(i) - \hat{x}(i) \end{aligned}$$

where $\bar{x}(i)$ and $\hat{x}(i)$ will be defined shortly.

Using the above definitions, the Kalman filter equations are found by taking the following expectations, conditioned on the complete measurement history up to stage i, denoted as $Z(i)$,

$$\begin{aligned} \bar{x}(i+1) &\triangleq E[x(i+1)|Z(i)] \\ &= \Phi(i)\hat{x}(i) + \Gamma(i)u(i) \\ M(i+1) &\triangleq E[\bar{e}(i+1)\bar{e}^T(i+1)|Z(i)] \\ &= \Phi(i)P(i)\Phi^T(i) + W(i) \\ P(i) &\triangleq E[\hat{e}(i)\hat{e}^T(i)|Z(i)] \\ &= M(i) - M(i)\Theta^T(i)\big(\Theta(i)M(i)\Theta^T(i) + V(i)\big)^{-1}\Theta(i)M(i) \\ \hat{x}(i) &\triangleq E[x(i)|Z(i)] \\ &= \bar{x}(i) + M(i)\Theta^T(i)\big(\Theta(i)M(i)\Theta^T(i) + V(i)\big)^{-1}\big(z(i) - \Theta(i)\bar{x}(i)\big). \end{aligned} \tag{A.2}$$

For further information on the development of these equations refer to [SC08].

A.2 Alternate Form of the Kalman Filter

Now, consider rewriting the usual system equations (A.1) as follows. Define a new enlarged state vector, $X(i)$,

$$X(i) \triangleq \begin{bmatrix} x(i) \\ v(i) \end{bmatrix} \in \mathbb{R}^{(n+p)}$$

and a new enlarged process noise vector, $Y(i)$,

$$Y(i) \triangleq \begin{bmatrix} w(i) \\ v(i+1) \end{bmatrix} \in \mathbb{R}^{(n+p)}$$

with statistics

$$\mathbf{Y}(i) \triangleq E[Y(i)Y^T(i)].$$

Note that the enlarged state vector and enlarged process noise vector are uncorrelated $\forall i \leq j$

$$\begin{aligned} E[(X(i) - E[X(i)])(Y(j) - E[Y(j)])^T] &= \begin{bmatrix} E[(x(i) - \bar{x}(i))w^T(j)] & E[(x(i) - \bar{x}(i))v^T(j+1)] \\ E[v(i)w^T(j)] & E[v(i)v^T(j+1)]. \end{bmatrix} \\ &= \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}. \end{aligned}$$

The system equations may now be written as

$$\begin{aligned} \begin{bmatrix} x(i+1) \\ v(i+1) \end{bmatrix} &= \underbrace{\begin{bmatrix} \Phi(i) & 0_{n,p} \\ 0_{p,n} & 0_{p,p} \end{bmatrix}}_{F(i)} \begin{bmatrix} x(i) \\ v(i) \end{bmatrix} + \underbrace{\begin{bmatrix} \Gamma(i) \\ 0_{p,m} \end{bmatrix}}_{G(i)} u(i) + \begin{bmatrix} I_n & 0_{n,p} \\ 0_{p,n} & I_p \end{bmatrix} \begin{bmatrix} w(i) \\ v(i+1) \end{bmatrix} \\ z(i) &= \underbrace{\begin{bmatrix} \Theta(i) & I_p \end{bmatrix}}_{H(i)} \begin{bmatrix} x(i) \\ v(i) \end{bmatrix} \end{aligned} \tag{A.3}$$

or, in compact notation as

$$\begin{aligned} X(i+1) &= F(i)X(i) + G(i)u(i) + Y(i) \\ z(i) &= H(i)X(i). \end{aligned} \tag{A.4}$$

The system equations appear with process noise only since the measurement noise is now included in the enlarged state vector.

The enlarged Kalman filter errors are defined as

$$\bar{\mathcal{E}}(i) \triangleq X(i) - \bar{X}(i)$$
$$\hat{\mathcal{E}}(i) \triangleq X(i) - \hat{X}(i)$$

where $\bar{X}(i)$ and $\hat{X}(i)$ will be defined shortly.

An alternate form of the Kalman filter may now be written as

$$\begin{aligned}
\bar{X}(i+1) &\triangleq E[X(i+1)|Z(i)] \\
&= F(i)\hat{X}(i) + G(i)u(i) \\
\mathcal{M}(i+1) &\triangleq E[\bar{\mathcal{E}}(i+1)\bar{\mathcal{E}}^T(i+1)|Z(i)] \\
&= F(i)\mathcal{P}(i)F^T(i) + \mathbf{Y}(i) \\
\mathcal{P}(i) &\triangleq E[\hat{\mathcal{E}}(i)\hat{\mathcal{E}}^T(i)|Z(i)] \\
&= \mathcal{M}(i) - \mathcal{M}(i)H^T(i)\big(H(i)\mathcal{M}(i)H^T(i)\big)^{-1}H(i)\mathcal{M}(i) \\
L(i) &\triangleq \mathcal{M}(i)H^T(i)\big(H(i)\mathcal{M}(i)H^T(i)\big)^{-1} \\
\hat{X}(i) &\triangleq E[X(i)|Z(i)] \\
&= \bar{X}(i) + L(i)\big(z(i) - H(i)\bar{X}(i)\big)
\end{aligned} \tag{A.5}$$

where

$$\bar{X}(0) = \begin{bmatrix} \bar{x}(0) \\ 0 \end{bmatrix}$$
$$\mathcal{M}(0) = \begin{bmatrix} M(0) & 0 \\ 0 & V(0) \end{bmatrix}.$$

Let's take a moment to look at the $L(i)$ equation. Note that a necessary condition for the matrix inverse to exist is that $H(i)$ is full row rank and $\mathcal{M}(i)H^T(i)$ is full column rank. A sufficient condition is that $H(i)$ is full row rank and $\mathcal{M}(i)$ is full column rank (invertible). As long as there are no repeated measurements then $H(i)$ will be full row rank, and due to the process noise variance in $\mathcal{M}(i+1)$ we know that $\mathcal{M}(i)$ is full column rank (invertible). Therefore, $\big(H(i)\mathcal{M}(i)H^T(i)\big)^{-1}$ exists at all stages.

It is easy to see that the alternate form of the Kalman filter (A.5) is equivalent to the traditional form of the Kalman filter (A.2) by writing out the individual enlarged matrix elements

$$
\begin{aligned}
\bar{X}(i+1) &= \begin{bmatrix} \bar{x}(i+1) \\ \bar{v}(i+1) \end{bmatrix} = \begin{bmatrix} \Phi(i)\hat{x}(i) + \Gamma(i)u(i) \\ 0 \end{bmatrix} \\
\mathcal{M}(i+1) &= \begin{bmatrix} \Phi(i)\mathcal{P}_1(i)\Phi^T(i) + W(i) & 0 \\ 0 & V(i+1) \end{bmatrix} \triangleq \begin{bmatrix} \mathcal{M}_1(i+1) & \mathcal{M}_2(i+1) \\ \mathcal{M}_2^T(i+1) & \mathcal{M}_3(i+1) \end{bmatrix} \\
\mathcal{P}(i) &= \mathcal{M}(i) - \begin{bmatrix} \mathcal{M}_1(i)\Theta^T(i) \\ \mathcal{M}_3(i) \end{bmatrix} \big(\Theta(i)\mathcal{M}_1(i)\Theta^T(i) + \mathcal{M}_3(i)\big)^{-1} \begin{bmatrix} \Theta(i)\mathcal{M}_1(i) & \mathcal{M}_3(i) \end{bmatrix} \\
&= \left[\begin{matrix} \mathcal{M}_1(i) - \mathcal{M}_1(i)\Theta^T(i)\big(\Theta(i)\mathcal{M}_1(i)\Theta^T(i) + \mathcal{M}_3(i)\big)^{-1}\Theta(i)\mathcal{M}_1(i) & \dots \\ -\mathcal{M}_3(i)\big(\Theta(i)\mathcal{M}_1(i)\Theta^T(i) + \mathcal{M}_3(i)\big)^{-1}\Theta(i)\mathcal{M}_1(i) & \dots \end{matrix} \right. \\
&\qquad \left. \begin{matrix} \dots & -\mathcal{M}_1(i)\Theta^T(i)\big(\Theta(i)\mathcal{M}_1(i)\Theta^T(i) + \mathcal{M}_3(i)\big)^{-1}\mathcal{M}_3(i) \\ \dots & \mathcal{M}_3(i) - \mathcal{M}_3(i)\big(\Theta(i)\mathcal{M}_1(i)\Theta^T(i) + \mathcal{M}_3(i)\big)^{-1}\mathcal{M}_3(i) \end{matrix} \right] \\
&\triangleq \begin{bmatrix} \mathcal{P}_1(i) & \mathcal{P}_2(i) \\ \mathcal{P}_2^T(i) & \mathcal{P}_3(i) \end{bmatrix} \\
L(i) &= \begin{bmatrix} \mathcal{M}_1(i)\Theta^T(i)\big(\Theta(i)\mathcal{M}_1(i)\Theta^T(i) + \mathcal{M}_3(i)\big)^{-1} \\ \mathcal{M}_3(i)\big(\Theta(i)\mathcal{M}_1(i)\Theta^T(i) + \mathcal{M}_3(i)\big)^{-1} \end{bmatrix} \triangleq \begin{bmatrix} L_1(i) \\ L_2(i) \end{bmatrix} \\
\hat{X}(i) &= \begin{bmatrix} \hat{x}(i) \\ \hat{v}(i) \end{bmatrix} = \begin{bmatrix} \bar{x}(i) + L_1(i)\big(z(i) - \Theta(i)\bar{x}(i)\big) \\ \bar{v}(i) + L_2(i)\big(z(i) - \Theta(i)\bar{x}(i)\big) \end{bmatrix}.
\end{aligned}
$$

Discarding the superfluous $\mathcal{P}(i)$ elements (only $\mathcal{P}_1(i)$ affects the estimate), and noting that

$\mathcal{M}_3(i) = V(i)$, we can simplify these equations as

$$\begin{aligned}
\bar{x}(i+1) &= \Phi(i)\hat{x}(i) + \Gamma(i)u(i) \\
\mathcal{M}_1(i+1) &= \Phi(i)\mathcal{P}_1(i)\Phi^T(i) + W(i) \\
\mathcal{P}_1(i) &= \mathcal{M}_1(i) - \mathcal{M}_1(i)\Theta^T(i)\big(\Theta(i)\mathcal{M}_1(i)\Theta^T(i) + V(i)\big)^{-1}\Theta(i)\mathcal{M}_1(i) \\
\hat{x}(i) &= \bar{x}(i) + \mathcal{M}_1(i)\Theta^T(i)\big(\Theta(i)\mathcal{M}_1(i)\Theta^T(i) + V(i)\big)^{-1}\big(z(i) - \Theta(i)\bar{x}(i)\big).
\end{aligned} \tag{A.6}$$

Since $\mathcal{M}_1(i) = M(i), \forall i$, the alternate form of the Kalman filter (A.5) is equivalent to the traditional form (A.2).

Furthermore, due to the enlarged state-space, $\hat{X}(i)$ also contains an explicit estimate of the measurement noise at the current stage, $\hat{v}(i)$. We can find a simplified expression for $\hat{v}(i)$ by adding and subtracting a term as follows

$$\begin{aligned}
\hat{v}(i) &= V(i)\big(\Theta(i)\mathcal{M}_1(i)\Theta^T(i) + V(i)\big)^{-1}\big(z(i) - \Theta(i)\bar{x}(i)\big) \\
&= \big(V(i) + \Theta(i)\mathcal{M}_1(i)\Theta^T(i) - \Theta(i)\mathcal{M}_1(i)\Theta^T(i)\big)\big(\Theta(i)\mathcal{M}_1(i)\Theta^T(i) + V(i)\big)^{-1}\big(z(i) - \Theta(i)\bar{x}(i)\big) \\
&= z(i) - \Theta(i)\bar{x}(i) - \Theta(i)\mathcal{M}_1(i)\Theta^T(i)\big(\Theta(i)\mathcal{M}_1(i)\Theta^T(i) + V(i)\big)^{-1}\big(z(i) - \Theta(i)\bar{x}(i)\big) \\
&= z(i) - \Theta(i)\hat{x}(i)
\end{aligned} \tag{A.7}$$

which is as expected.

A.3 Alternate Form of the Kalman Filter with State History: The Enlarged Kalman Filter

Now, let's consider an *N-stage* problem. Define an enlarged state vector, $X(i) \in \mathbb{R}^{(N+1)(n+p)}$, comprised of state and measurement noise histories up to and including stage i. The $X(i)$ sub-partitions are

$$X(i)_j = \begin{cases} x(j), & j \leq i \\ 0, & i < j \leq N \\ v(j-N-1), & N+1 \leq j \leq N+1+i \\ 0, & N+1+i < j \leq 2N+1 \end{cases}$$

for $i = 0, 1, ..., N$, $j = 0, 1, ..., 2N + 1$. In general, here's how $X(i)$ appears at stage i:

$$X(i) = \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ \vdots \\ x(i) \\ 0 \\ \vdots \\ 0 \\ v(0) \\ v(1) \\ v(2) \\ \vdots \\ v(i) \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

Define an enlarged process noise vector, $Y(i) \in \mathbb{R}^{n+p}$, comprised of the state process noise at the current stage (i) and measurement noise at the next stage ($i + 1$). The $Y(i)$ sub-partitions are

$$Y(i)_j = \begin{cases} w(i), & j = 0 \\ v(i+1), & j = 1 \end{cases}$$

for $i = 0, 1, ..., N - 1$. Note that the enlarged process noise vector is still zero-mean and delta-correlated.

Define enlarged system, control, and measurement matrices with the following sub-partitions

$$F(i)_{j,k} = \begin{cases} I_n, & j = k \leq i \\ \Phi(i), & j = i+1, k = i \\ I_p, & N+1 \leq j = k \leq N+1+i \\ 0, & otherwise \end{cases}$$

$$F(i) \in \mathbb{R}^{(N+1)(n+p)\times(N+1)(n+p)},$$

$$G(i)_j = \begin{cases} \Gamma(i), & j = i+1 \\ 0, & otherwise \end{cases}$$

$$G(i) \in \mathbb{R}^{(N+1)(n+p)\times m},$$

$$G_y(i)_{j,k} = \begin{cases} I_n, & j = i+1, k = 0 \\ I_p, & j = (N+1)+(i+1), k = 1 \\ 0, & otherwise \end{cases}$$

$$G_y(i) \in \mathbb{R}^{(N+1)(n+p)\times(n+p)},$$

$$H(i)_{j,k} = \begin{cases} \Theta(j), & j = k \leq i \\ I_p, & j \leq i, k = N+1+j \\ 0, & otherwise \end{cases}$$

$$H(i) \in \mathbb{R}^{(i+1)p\times(N+1)(n+p)},$$

for $i = 0, 1, ..., N-1$.

Then, with appropriately sized system, control, and measurement matrices we may write the enlarged state-space as

$$\begin{aligned} X(i+1) &= F(i)X(i) + G(i)u(i) + G_y(i)Y(i) \\ Z(i) &= H(i)X(i) \end{aligned} \tag{A.8}$$

for $i = 0, 1, ..., N-1$, where $Z(i) \in \mathbb{R}^{(i+1)p}$ is the measurement history up to and including stage i. This state-space is functionally identically to (A.1).

We may also define the enlarged mean state, mean-square state (second moment), covariance, and process noise variance as follows

$$\bar{\mathbf{X}}(i) \triangleq E[X(i)]$$
$$\mathbf{X}(i) \triangleq E[X(i)X^T(i)]$$
$$\mathbf{P}(i) \triangleq \mathbf{X}(i) - \bar{\mathbf{X}}(i)\bar{\mathbf{X}}^T(i)$$
$$\mathbf{Y}(i) \triangleq E[Y(i)Y^T(i)].$$

Taking the above unconditional expectations of (A.8) results in the following statistical dynamic equations. The mean state sequence is

$$\begin{aligned} \bar{\mathbf{X}}(i+1) &= F(i)\bar{\mathbf{X}}(i) + G(i)E[u(i)] \\ \bar{\mathbf{X}}(0)_j &= \begin{cases} \bar{x}(0), & j = 0 \\ 0, & otherwise \end{cases} \end{aligned} \tag{A.9}$$

where $\bar{\mathbf{X}}(i) \in \mathbb{R}^{(N+1)(n+p)}$. The mean-square state sequence is

$$\begin{aligned} \mathbf{X}(i+1) = {} & F(i)\mathbf{X}(i)F^T(i) + F(i)E[X(i)u^T(i)]G^T(i) + G(i)E[u(i)X^T(i)]F^T(i) \\ & + G(i)E[u(i)u^T(i)]G^T(i) + G_y(i)\mathbf{Y}(i)G_y^T(i) \\ \mathbf{X}(0)_{j,k} = {} & \begin{cases} M(0) + \bar{x}(0)\bar{x}^T(0), & j = k = 0 \\ V(0), & j = k = N+1 \\ 0, & otherwise \end{cases} \end{aligned} \tag{A.10}$$

where $\mathbf{X}(i) \in \mathbb{R}^{(N+1)(n+p)\times(N+1)(n+p)}$. The covariance sequence is

$$
\begin{aligned}
\mathbf{P}(i+1) = {} & F(i)\mathbf{X}(i)F^T(i) + F(i)E[X(i)u^T(i)]G^T(i) + G(i)E[u(i)X^T(i)]F^T(i) \\
& + G(i)E[u(i)u^T(i)]G^T(i) + G_y(i)\mathbf{Y}(i)G_y^T(i) - F(i)\bar{\mathbf{X}}(i)\bar{\mathbf{X}}^T(i)F^T(i) \\
& - F(i)\bar{\mathbf{X}}(i)E[u^T(i)]G^T(i) - G(i)E[u(i)]\bar{\mathbf{X}}^T(i)F^T(i) - G(i)E[u(i)]E[u^T(i)]G^T(i) \\
= {} & F(i)\mathbf{P}(i)F^T(i) + F(i)\big(E[X(i)u^T(i)] - \bar{\mathbf{X}}(i)E[u^T(i)]\big)G^T(i) \\
& + G(i)\big(E[u(i)X^T(i)] - E[u(i)]\bar{\mathbf{X}}^T(i)\big)F^T(i) \\
& + G(i)\big(E[u(i)u^T(i)] - E[u(i)]E[u^T(i)]\big)G^T(i) + G_y(i)\mathbf{Y}(i)G_y^T(i) \\
\mathbf{P}(0)_{j,k} = {} & \begin{cases} M(0), & j = k = 0 \\ V(0), & j = k = N+1 \\ 0, & otherwise \end{cases}
\end{aligned}
\tag{A.11}
$$

where $\mathbf{P}(i) \in \mathbb{R}^{(N+1)(n+p)\times(N+1)(n+p)}$.

The estimation problem is now a least squares optimization as follows: Given $Z(i)$, find $\hat{X}(i)$ such that the performance index, J, is minimized

$$
J = \min_{\hat{X}(Z(i))} E\left\{ \frac{1}{2} \sum_{i=0}^{N-1} \|X(i) - \hat{X}(i)\|^2 \right\} \tag{A.12}
$$

subject to (A.8).

We know *a priori* that the conditional mean is the optimal estimate to minimize the unconditional error variance. Given the linear dynamics and Gaussian noises, we may write the admissible form of the optimal conditional mean estimator as an affine function of $Z(i)$

$$
\begin{aligned}
\hat{X}(i) &\triangleq E[X(i)|Z(i)] \\
&\triangleq \ell(i) + L(i)Z(i) \\
&= \ell(i) + L(i)H(i)X(i)
\end{aligned}
\tag{A.13}
$$

where $\ell(i) \in \mathbb{R}^{(N+1)(n+p)}$ is a vector and non-zero only when $\bar{x}(0) \neq 0$, and $L(i) \in \mathbb{R}^{(N+1)(n+p)\times(i+1)p}$ is a matrix.

Since the performance index is a scalar quantity, we can use the *trace* (Tr) operator without affecting the result. We can then use the cyclic property to rearrange the order of matrix multiplication, which then allows us to take the unconditional expectation of the performance index using the previously defined enlarged statistical parameters. Using (A.13) in (A.12) this procedure yields

$$\begin{aligned}
J &= \min_{\ell(i),L(i)} E\left\{\frac{1}{2}\sum_{i=0}^{N-1}\|X(i)-\ell(i)-L(i)H(i)X(i)\|^2\right\}\\
&= \min_{\ell(i),L(i)} E\left\{\frac{1}{2}\sum_{i=0}^{N-1}\big(X(i)-\ell(i)-L(i)H(i)X(i)\big)^T\big(X(i)-\ell(i)-L(i)H(i)X(i)\big)\right\}\\
&= \min_{\ell(i),L(i)} E\Bigg\{\frac{1}{2}\sum_{i=0}^{N-1}Tr\Big(\big(X(i)-L(i)H(i)X(i)\big)^T\big(X(i)-L(i)H(i)X(i)\big)\\
&\qquad + \ell^T(i)\ell(i)-\ell^T(i)\big(X(i)-L(i)H(i)X(i)\big)-\big(X(i)-L(i)H(i)X(i)\big)^T\ell(i)\Big)\Bigg\}\\
&= \min_{\ell(i),L(i)} \frac{1}{2}\sum_{i=0}^{N-1}Tr\Big(E\left[X(i)X^T(i)\right]\big(I-L(i)H(i)\big)^T\big(I-L(i)H(i)\big)\\
&\qquad + \ell^T(i)\ell(i)-2\ell^T(i)\big(E\left[X(i)\right]-L(i)H(i)E\left[X(i)\right]\big)\Big)\\
&= \min_{\ell(i),L(i)} \frac{1}{2}\sum_{i=0}^{N-1}Tr\Big(\mathbf{X}(i)\big(I-L(i)H(i)\big)^T\big(I-L(i)H(i)\big)\\
&\qquad + \ell^T(i)\ell(i)-2\ell^T(i)\big(I-L(i)H(i)\big)\bar{\mathbf{X}}(i)\Big).
\end{aligned}$$

Taking the partial derivatives associated with the necessary conditions for optimality

$$\frac{\partial J}{\partial \ell(i)} = 0$$
$$\frac{\partial J}{\partial L(i)} = 0$$

and performing some algebraic manipulation results in [1]

$$\begin{aligned}
\ell(i) &= \big(I-L(i)H(i)\big)\bar{\mathbf{X}}(i)\\
L(i) &= \big(\mathbf{X}(i)-\bar{\mathbf{X}}(i)\bar{\mathbf{X}}^T(i)\big)H^T(i)\Big(H(i)\big(\mathbf{X}(i)-\bar{\mathbf{X}}(i)\bar{\mathbf{X}}^T(i)\big)H^T(i)\Big)^{-1}\\
&= \mathbf{P}(i)H^T(i)\big(H(i)\mathbf{P}(i)H^T(i)\big)^{-1}.
\end{aligned} \tag{A.14}$$

[1]We use the *trace* differential properties detailed in Appendix B.2.2 to find these partial derivatives.

Note that the $L(i)$ equation requires that $\big(H(i)\mathbf{P}(i)H^T(i)\big)^{-1}$ exist.[2] A necessary condition for this matrix inverse to exist is that $H(i)$ is full row rank and $\mathbf{P}(i)H^T(i)$ is full column rank. A sufficient condition is that $H(i)$ is full row rank and $\mathbf{P}(i)$ is full column rank (invertible). As long as there are no repeated measurements then $H(i)$ will be full row rank. The $\mathbf{P}(i)$ matrix (A.11) is initially a sparse matrix that becomes more dense as i increases. This means that $\mathbf{P}(i)$ is not invertible, but as it turns out from the structure of $H(i)$ and $\mathbf{P}(i)$, $\mathbf{P}(i)H^T(i)$ is full column rank at all stages which means $\big(H(i)\mathbf{P}(i)H^T(i)\big)^{-1}$ exists at all stages. We will show this structure using a simple numerical example below.

The optimal estimate can now be written as

$$\begin{aligned}\hat{X}(i) &= \ell(i) + L(i)Z(i) \\ &= \bar{\mathbf{X}}(i) + L(i)\big(Z(i) - H(i)\bar{\mathbf{X}}(i)\big) \\ &= \bar{\mathbf{X}}(i) + \mathbf{P}(i)H^T(i)\big(H(i)\mathbf{P}(i)H^T(i)\big)^{-1}\big(Z(i) - H(i)\bar{\mathbf{X}}(i)\big).\end{aligned} \tag{A.15}$$

An interesting property of this enlarged state Kalman filter is the following: Using the definition of $\hat{X}(i)$ in (A.13) as the conditional mean, we can take the conditional expectation of (A.15)

$$\begin{aligned}E[\hat{X}(i)|Z(i)] = \hat{X}(i) &= E[\bar{\mathbf{X}}(i) + L(i)\big(H(i)X(i) - H(i)\bar{\mathbf{X}}(i)\big)|Z(i)] \\ &= \bar{\mathbf{X}}(i) + L(i)\big(H(i)\hat{X}(i) - H(i)\bar{\mathbf{X}}(i)\big) \\ \big(I - L(i)H(i)\big)\hat{X}(i) &= \big(I - L(i)H(i)\big)\bar{\mathbf{X}}(i) \\ \big(I - L(i)H(i)\big)\big(\hat{X}(i) - \bar{\mathbf{X}}(i)\big) &= 0.\end{aligned} \tag{A.16}$$

That is, the error between the conditional mean, $\hat{X}(i)$, and unconditional mean, $\bar{\mathbf{X}}(i)$, lies in the null space of $\big(I - L(i)H(i)\big)$.

Theorem A.1. *As further confirmation that we have found the optimal estimate, the conditional mean estimate in* (A.15) *is optimal in terms of the Orthogonal Projection Lemma. That is, the estimation error and estimate are orthogonal,*

$$E\left[\big(X(i) - \hat{X}(i)\big)^T\hat{X}(i)\right] = 0.$$

[2]By definition and construction $\mathbf{P}(i) \geq 0$ and is symmetric. Therefore, we could also define the symmetric square root $\mathbf{P}^{1/2}(i)$ and write the inverse as $\big(H(i)\mathbf{P}^{1/2}(i)\big(H(i)\mathbf{P}^{1/2}(i)\big)^T\big)^{-1}$.

Proof. Using the optimal estimator parameters as derived in (A.14) and, once again, taking advantage of the *trace* operator we find

$$
\begin{aligned}
E\left[\left(X(i)-\hat{X}(i)\right)^T\hat{X}(i)\right] &= E\left[\left(X(i)-\ell(i)-L(i)H(i)X(i)\right)^T\left(\ell(i)+L(i)H(i)X(i)\right)\right]\\
&= Tr\Big(E\Big[\ell^T(i)X(i)+X(i)X^T(i)L(i)H(i)-\ell^T(i)\ell(i)\\
&\quad -2\ell^T(i)L(i)H(i)X(i)-X(i)X^T(i)H^T(i)L^T(i)L(i)H(i)\Big]\Big)\\
&= Tr\Big(\ell^T(i)\big(-\ell(i)+\bar{\mathbf{X}}(i)-2L(i)H(i)\bar{\mathbf{X}}(i)\big)\\
&\quad +\mathbf{X}(i)\big(I-L(i)H(i)\big)^T L(i)H(i)\Big)\\
&= Tr\Big(\ell^T(i)\big(-\big(I-L(i)H(i)\big)\bar{\mathbf{X}}(i)+\bar{\mathbf{X}}(i)-2L(i)H(i)\bar{\mathbf{X}}(i)\big)\\
&\quad +\mathbf{X}(i)\big(I-L(i)H(i)\big)^T L(i)H(i)\Big)\\
&= Tr\Big(-\ell^T(i)L(i)H(i)\bar{\mathbf{X}}(i)+\mathbf{X}(i)\big(I-L(i)H(i)\big)^T L(i)H(i)\Big)\\
&= Tr\Big(\big(\mathbf{X}(i)-\bar{\mathbf{X}}(i)\bar{\mathbf{X}}^T(i)\big)\big(I-L(i)H(i)\big)^T L(i)H(i)\Big)\\
&= Tr\Big(\mathbf{P}(i)\big(I-L(i)H(i)\big)^T L(i)H(i)\Big)\\
&= Tr\Big(L(i)H(i)\mathbf{P}(i)-L(i)H(i)\mathbf{P}(i)H^T(i)L^T(i)\Big)\\
&= Tr\Big(L(i)H(i)\mathbf{P}(i)-L(i)H(i)\mathbf{P}(i)H^T(i)\big(H(i)\mathbf{P}(i)H^T(i)\big)^{-1}H(i)\mathbf{P}(i)\Big)\\
&= Tr\Big(L(i)H(i)\mathbf{P}(i)-L(i)H(i)\mathbf{P}(i)\Big)\\
&= 0.
\end{aligned}
$$

■

As a result of the enlarged state-space, the optimal $\hat{X}(i)$ (A.15) not only produces a current state estimate, but also a smoothed estimate of all past states. In general, here's how $\hat{X}(i)$ appears at stage i:

$$\hat{X}(i) = \begin{bmatrix} E[x(0)|Z(i)] \\ E[x(1)|Z(i)] \\ E[x(2)|Z(i)] \\ \vdots \\ E[x(i)|Z(i)] \\ 0 \\ \vdots \\ 0 \\ E[v(0)|Z(i)] \\ E[v(1)|Z(i)] \\ E[v(2)|Z(i)] \\ \vdots \\ E[v(i)|Z(i)] \\ 0 \\ \vdots \\ 0 \end{bmatrix} \triangleq \begin{bmatrix} \hat{x}(0|i) \\ \hat{x}(1|i) \\ \hat{x}(2|i) \\ \vdots \\ \hat{x}(i|i) = \hat{x}(i) \\ 0 \\ \vdots \\ 0 \\ \hat{v}(0|i) \\ \hat{v}(1|i) \\ \hat{v}(2|i) \\ \vdots \\ \hat{v}(i|i) = \hat{v}(i) \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

Enlarged Kalman Filter Example

Due to the extensive matrix algebra involved, instead of showing analytical equivalency to the traditional Kalman filter and smoother we will demonstrate the aspects of this enlarged estimator by means of a simple numerical comparison for a three-stage ($N = 3$), scalar problem. The time-invariant parameters are

$$\Phi(i) = 1, \quad \Gamma(i) = 0, \quad \Theta(i) = 2$$
$$\bar{x}(0) = 100, \quad M(0) = 20, \quad W(i) = 5, \quad V(i) = 3.$$

Although these parameters are time-invariant, we will carry indexes on the non-zero and non-identity parameters while writing out the estimator equations in order to keep the results more general. We use the following realization

$$x(0) = 105$$

$$w(0) = 2, \quad w(1) = -2, \quad w(2) = 0.5$$

$$v(0) = -1, \quad v(1) = 1, \quad v(2) = -0.5$$

which, referring to (A.1), leads to the following states and measurements

$$\begin{aligned}
x(1) &= x(0) + w(0) = 107 \\
x(2) &= x(1) + w(1) = 105 \\
x(3) &= x(2) + w(2) = 105.5 \\
z(0) &= \Theta(0)x(0) + v(0) = 209 \\
z(1) &= \Theta(1)x(1) + v(1) = 215 \\
z(2) &= \Theta(2)x(2) + v(2) = 209.5.
\end{aligned} \tag{A.17}$$

In terms of the enlarged state-space (A.8) we have the following state and measurement histories.

Stage 0 Enlarged State-Space: $(i = 0)$

$$\underbrace{\begin{bmatrix} x(0) \\ x(1) \\ 0 \\ 0 \\ v(0) \\ v(1) \\ 0 \\ 0 \end{bmatrix}}_{X(1)} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}}_{F(0)} \underbrace{\begin{bmatrix} x(0) \\ 0 \\ 0 \\ 0 \\ v(0) \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{X(0)} + \underbrace{\begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}}_{G_y(0)} \underbrace{\begin{bmatrix} w(0) \\ v(1) \end{bmatrix}}_{Y(0)} = \begin{bmatrix} x(0) \\ x(0) + w(0) \\ 0 \\ 0 \\ v(0) \\ v(1) \\ 0 \\ 0 \end{bmatrix} \tag{A.18}$$

$$\underbrace{\begin{bmatrix} z(0) \end{bmatrix}}_{Z(0)} = \underbrace{\begin{bmatrix} \Theta(0) & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}}_{H(0)} X(0) = \begin{bmatrix} \Theta(0)x(0) + v(0) \end{bmatrix}$$

Stage 1 Enlarged State-Space: $(i = 1)$

$$\underbrace{\begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ 0 \\ v(0) \\ v(1) \\ v(2) \\ 0 \end{bmatrix}}_{X(2)} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}}_{F(1)} \underbrace{\begin{bmatrix} x(0) \\ x(1) \\ 0 \\ 0 \\ v(0) \\ v(1) \\ 0 \\ 0 \end{bmatrix}}_{X(1)} + \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}}_{G_y(1)} \underbrace{\begin{bmatrix} w(1) \\ v(2) \end{bmatrix}}_{Y(1)} = \begin{bmatrix} x(0) \\ x(1) \\ x(1) + w(1) \\ 0 \\ v(0) \\ v(1) \\ v(2) \\ 0 \end{bmatrix} \tag{A.19}$$

$$\underbrace{\begin{bmatrix} z(0) \\ z(1) \end{bmatrix}}_{Z(1)} = \underbrace{\begin{bmatrix} \Theta(0) & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & \Theta(1) & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}}_{H(1)} X(1) = \begin{bmatrix} \Theta(0)x(0) + v(0) \\ \Theta(1)x(1) + v(1) \end{bmatrix}$$

Stage 2 Enlarged State-Space: $(i = 2)$

$$\underbrace{\begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \\ v(0) \\ v(1) \\ v(2) \\ v(3) \end{bmatrix}}_{X(3)} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}}_{F(2)} \underbrace{\begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ 0 \\ v(0) \\ v(1) \\ v(2) \\ 0 \end{bmatrix}}_{X(2)} + \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}}_{G_y(2)} \underbrace{\begin{bmatrix} w(2) \\ v(3) \end{bmatrix}}_{Y(2)} = \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(2) + w(2) \\ v(0) \\ v(1) \\ v(2) \\ v(3) \end{bmatrix} \tag{A.20}$$

$$\underbrace{\begin{bmatrix} z(0) \\ z(1) \\ z(2) \end{bmatrix}}_{Z(2)} = \underbrace{\begin{bmatrix} \Theta(0) & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & \Theta(1) & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & \Theta(2) & 0 & 0 & 0 & 1 & 0 \end{bmatrix}}_{H(2)} X(2) = \begin{bmatrix} \Theta(0)x(0) + v(0) \\ \Theta(1)x(1) + v(1) \\ \Theta(2)x(2) + v(2) \end{bmatrix}$$

As can be seen, the enlarged state-space in (A.18), (A.19), and (A.20) is functionally identical to (A.17) – it produces the exact same state and measurement histories as the traditional state-space.[3]

We now construct the filters and smoothers at each stage. At each stage we will first calculate the current state estimate using our two methods: the enlarged Kalman filter (A.15) and the traditional Kalman filter (A.2). We will then calculate the smoothed state estimate. We get this as a by-product using the enlarged Kalman filter equations. We will calculate the smoothed state estimate for the traditional Kalman filter using [BH75, Section 13.2].[4] For a measurement at stage $i \geq k$ we have the following recursion to calculate the

[3]Note in stage 2 (A.20) that $v(3)$ (which is $v(N)$ in general) is superfluous; we write it for consistency with previous stages even though it is not used in any measurement.

[4]The author believes that equations 13.2.16 and 13.2.17 in [BH75, Section 13.2] contain some inaccuracies. As such, (A.21) was derived using 13.2.7 and 13.2.9 in [BH75, Section 13.2] and modifying for fixed stage k with increasing i.

smoothed state estimate at stage k given measurements up to stage i, $\hat{x}(k|i)$,

$$\begin{aligned} P(k|i) &= P(k|i-1)\Phi^T(i-1)\big(I - P(i)\Theta^T(i)V^{-1}(i)\Theta(i)\big)^T, \quad P(i|i) = P(i) \\ \hat{x}(k|i) &= \hat{x}(k|i-1) + P(k|i)\Theta^T(i)V^{-1}(i)\big(z(i) - \Theta(i)\bar{x}(i)\big), \quad \hat{x}(i|i) = \hat{x}(i). \end{aligned} \tag{A.21}$$

Stage 0 Filter: $(i = 0)$

At stage 0 there is no past measurement history, therefore, we end up with only the filter equations at this stage. Due to the simplicity of the equations at stage 0 it is easy to see analytical equivalency of the enlarged Kalman filter and traditional Kalman filter. As such, we will write out these equations explicitly/parametrically.

Enlarged Kalman Filter:

From (A.15) we find the following expression

$$\hat{X}(0) = \bar{\mathbf{X}}(0) + \mathbf{P}(0)H^T(0)\big(H(0)\mathbf{P}(0)H^T(0)\big)^{-1}\big(Z(0) - H(0)\bar{\mathbf{X}}(0)\big),$$

$$\underbrace{\begin{bmatrix} \hat{x}(0) \\ 0 \\ 0 \\ 0 \\ \hat{v}(0) \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{\hat{X}(0)} = \underbrace{\begin{bmatrix} \bar{x}(0) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{\bar{\mathbf{X}}(0)} + \underbrace{\begin{bmatrix} M(0) & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & V(0) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}}_{\mathbf{P}(0)} \underbrace{\begin{bmatrix} \Theta^T(0) \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{H^T(0)}$$

$$
* \left(\underbrace{\begin{bmatrix} \Theta(0) & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}}_{H(0)} \underbrace{\begin{bmatrix} M(0) & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & V(0) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}}_{\mathbf{P}(0)} \underbrace{\begin{bmatrix} \Theta^T(0) \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{H^T(0)} \right)^{-1}
$$

$$
* \left(\underbrace{\begin{bmatrix} \Theta(0)x(0) + v(0) \end{bmatrix}}_{Z(0)} - \underbrace{\begin{bmatrix} \Theta(0) & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}}_{H(0)} \underbrace{\begin{bmatrix} \bar{x}(0) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{\bar{\mathbf{X}}(0)} \right)
$$

$$
= \begin{bmatrix} \bar{x}(0) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} M(0)\Theta^T(0) \\ 0 \\ 0 \\ 0 \\ V(0) \\ 0 \\ 0 \\ 0 \end{bmatrix} \left(\Theta(0)M(0)\Theta^T(0) + V(0)\right)^{-1} \left(\Theta(0)x(0) + v(0) - \Theta(0)\bar{x}(0)\right)
$$

$$= \begin{bmatrix} \bar{x}(0) + M(0)\Theta^T(0)\left(\Theta(0)M(0)\Theta^T(0) + V(0)\right)^{-1}\left(\Theta(0)x(0) + v(0) - \Theta(0)\bar{x}(0)\right) \\ 0 \\ 0 \\ 0 \\ V(0)\left(\Theta(0)M(0)\Theta^T(0) + V(0)\right)^{-1}\left(\Theta(0)x(0) + v(0) - \Theta(0)\bar{x}(0)\right) \\ 0 \\ 0 \\ 0 \end{bmatrix}.$$

Note that the $\hat{x}(0)$ expression (first row) is identical to (A.2), and the $\hat{v}(0)$ expression (fifth row) is equivalent to $\hat{v}(0) = z(0) - \Theta(0)\hat{x}(0)$ as we showed in (A.7). Plugging in the parameter values for this example we calculate

$$\hat{X}(0) = \begin{bmatrix} 104.34 \\ 0 \\ 0 \\ 0 \\ 0.33 \\ 0 \\ 0 \\ 0 \end{bmatrix}. \tag{A.22}$$

Traditional Kalman Filter:

We know due to analytical equivalency (as previously mentioned) that (A.2) will produce the same results

$$\begin{aligned} \hat{x}(0) &= \bar{x}(0) + M(0)\Theta^T(0)\left(\Theta(0)M(0)\Theta^T(0) + V(0)\right)^{-1}\left(\Theta(0)x(0) + v(0) - \Theta(0)\bar{x}(0)\right) \\ &= 104.34 \\ \hat{v}(0) &= z(0) - \Theta(0)\hat{x}(0) \\ &= 0.33. \end{aligned} \tag{A.23}$$

Therefore, at stage 0 the enlarged Kalman filter (A.22) and traditional Kalman filter (A.23) yield identical results.

Stage 1 Filter and Smoother: $(i = 1)$

At stage 1 we will show the parametric structure of the estimators up to the point that matrix inverses are required, at which point we will plug in the parameter values for this example.

Enlarged Kalman Filter:

From (A.9) we find the following expression

$$\bar{\mathbf{X}}(1) = F(0)\bar{\mathbf{X}}(0)$$
$$= \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}}_{F(0)} \underbrace{\begin{bmatrix} \bar{x}(0) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{\bar{\mathbf{X}}(0)} = \begin{bmatrix} \bar{x}(0) \\ \bar{x}(0) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}.$$

From (A.11) we find the following expression

$$\mathbf{P}(1) = F(0)\mathbf{P}(0)F^T(0) + G_y(0)\mathbf{Y}(0)G_y^T(0)$$

$$
= \underbrace{\begin{bmatrix} 1&0&0&0&0&0&0&0\\ 1&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&1&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0 \end{bmatrix}}_{F(0)} \underbrace{\begin{bmatrix} M(0)&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&V(0)&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0 \end{bmatrix}}_{\mathbf{P}(0)} \underbrace{\begin{bmatrix} 1&1&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&1&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0 \end{bmatrix}}_{F^T(0)}
$$

$$
+ \underbrace{\begin{bmatrix} 0&0\\ 1&0\\ 0&0\\ 0&0\\ 0&0\\ 0&1\\ 0&0\\ 0&0 \end{bmatrix}}_{G_y(0)} \underbrace{\begin{bmatrix} W(0)&0\\ 0&V(1) \end{bmatrix}}_{\mathbf{Y}(0)} \underbrace{\begin{bmatrix} 0&1&0&0&0&0&0&0\\ 0&0&0&0&0&1&0&0 \end{bmatrix}}_{G_y^T(0)}
$$

$$
= \begin{bmatrix} M(0)&M(0)&0&0&0&0&0&0\\ M(0)&M(0)+W(0)&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&V(0)&0&0&0\\ 0&0&0&0&0&V(1)&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0 \end{bmatrix}.
$$

Using the above $\bar{\mathbf{X}}(1)$ and $\mathbf{P}(1)$ in (A.15) we find the following expression

$$\hat{X}(1) = \bar{\mathbf{X}}(1) + \mathbf{P}(1)H^T(1)\big(H(1)\mathbf{P}(1)H^T(1)\big)^{-1}\big(Z(1) - H(1)\bar{\mathbf{X}}(1)\big),$$

$$\underbrace{\begin{bmatrix} \hat{x}(0|1) \\ \hat{x}(1) \\ 0 \\ 0 \\ \hat{v}(0|1) \\ \hat{v}(1) \\ 0 \\ 0 \end{bmatrix}}_{\hat{X}(1)} = \underbrace{\begin{bmatrix} \bar{x}(0) \\ \bar{x}(0) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{\bar{\mathbf{X}}(1)} + \underbrace{\begin{bmatrix} M(0) & M(0) & 0 & 0 & 0 & 0 & 0 & 0 \\ M(0) & M(0)+W(0) & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & V(0) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & V(1) & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}}_{\mathbf{P}(1)} \underbrace{\begin{bmatrix} \Theta^T(0) & 0 \\ 0 & \Theta^T(1) \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}}_{H^T(1)}$$

$$* \left(\underbrace{\begin{bmatrix} \Theta(0) & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & \Theta(1) & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}}_{H(1)} \mathbf{P}(1)H^T(1) \right)^{-1}$$

$$* \left(\underbrace{\begin{bmatrix} \Theta(0)x(0) + v(0) \\ \Theta(1)x(1) + v(1) \end{bmatrix}}_{Z(1)} - \underbrace{\begin{bmatrix} \Theta(0) & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & \Theta(1) & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}}_{H(1)} \underbrace{\begin{bmatrix} \bar{x}(0) \\ \bar{x}(0) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{\bar{\mathbf{X}}(1)} \right)$$

$$= \begin{bmatrix} \bar{x}(0) \\ \bar{x}(0) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} M(0)\Theta^T(0) & M(0)\Theta^T(1) \\ M(0)\Theta^T(0) & \left(M(0)+W(0)\right)\Theta^T(1) \\ 0 & 0 \\ 0 & 0 \\ V(0) & 0 \\ 0 & V(1) \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

$$* \left(\begin{bmatrix} \Theta(0) & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & \Theta(1) & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} M(0)\Theta^T(0) & M(0)\Theta^T(1) \\ M(0)\Theta^T(0) & \left(M(0)+W(0)\right)\Theta^T(1) \\ 0 & 0 \\ 0 & 0 \\ V(0) & 0 \\ 0 & V(1) \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \right)^{-1}$$

$$* \begin{bmatrix} \Theta(0)x(0) + v(0) - \Theta(0)\bar{x}(0) \\ \Theta(1)x(1) + v(1) - \Theta(1)\bar{x}(0) \end{bmatrix}$$

$$= \begin{bmatrix} \bar{x}(0) \\ \bar{x}(0) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} M(0)\Theta^T(0) & M(0)\Theta^T(1) \\ M(0)\Theta^T(0) & \left(M(0)+W(0)\right)\Theta^T(1) \\ 0 & 0 \\ 0 & 0 \\ V(0) & 0 \\ 0 & V(1) \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

$$
* \begin{bmatrix} \Theta(0)M(0)\Theta^T(0) + V(0) & \Theta(0)M(0)\Theta^T(1) \\ \Theta(1)M(0)\Theta^T(0) & \Theta(1)\left(M(0)+W(0)\right)\Theta^T(1) + V(1) \end{bmatrix}^{-1}
$$

$$
* \begin{bmatrix} \Theta(0)x(0) + v(0) - \Theta(0)\bar{x}(0) \\ \Theta(1)x(1) + v(1) - \Theta(1)\bar{x}(0) \end{bmatrix}.
$$

Plugging in the parameter values for this example we calculate

$$
\hat{X}(1) = \begin{bmatrix} \hat{x}(0|1) \\ \hat{x}(1) \\ 0 \\ 0 \\ \hat{v}(0|1) \\ \hat{v}(1) \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 104.69 \\ 107.13 \\ 0 \\ 0 \\ -0.38 \\ 0.73 \\ 0 \\ 0 \end{bmatrix}. \tag{A.24}
$$

Traditional Kalman Filter:

From (A.2) we find the following expressions

$$
\begin{aligned}
\bar{x}(1) &= \hat{x}(0) \\
P(0) &= M(0) - M(0)\Theta^T(0)\big(\Theta(0)M(0)\Theta^T(0) + V(0)\big)^{-1}\Theta(0)M(0) \\
M(1) &= P(0) + W(0) \\
\hat{x}(1) &= \bar{x}(1) + M(1)\Theta^T(1)\big(\Theta(1)M(1)\Theta^T(1) + V(1)\big)^{-1}\big(z(1) - \Theta(1)\bar{x}(1)\big).
\end{aligned}
$$

Plugging in the parameter values for this example we calculate

$$
\begin{aligned}
\bar{x}(1) &= 104.34 \\
P(0) &= 0.7229 \\
M(1) &= 5.72 \\
\hat{x}(1) &= 107.13 \\
\hat{v}(1) &= z(1) - \Theta(1)\hat{x}(1) = 0.73.
\end{aligned} \tag{A.25}
$$

These values for $\hat{x}(1)$ and $\hat{v}(1)$ (A.25) are identical to the enlarged Kalman filter current state estimate in $\hat{X}(1)$ (A.24).

Now, for the smoothing at $k = 0$ given measurements up to $i = 1$ we use (A.21) to find the following expressions

$$P(0|1) = P(0)\big(I - P(1)\Theta^T(1)V^{-1}(1)\Theta(1)\big)^T$$
$$\hat{x}(0|1) = \hat{x}(0) + P(0|1)\Theta^T(1)V^{-1}(1)\big(z(1) - \Theta(1)\bar{x}(1)\big)$$

where

$$P(1) = M(1) - M(1)\Theta^T(1)\big(\Theta(1)M(1)\Theta^T(1) + V(1)\big)^{-1}\Theta(1)M(1).$$

Plugging in the parameter values for this example we calculate

$$\begin{aligned} P(1) &= 0.6631 \\ P(0|1) &= 0.0838 \\ \hat{x}(0|1) &= 104.69 \\ \hat{v}(0|1) &= z(0) - \Theta(0)\hat{x}(0|1) = -0.38. \end{aligned} \tag{A.26}$$

These values for $\hat{x}(0|1)$ and $\hat{v}(0|1)$ (A.26) are identical to the enlarged Kalman filter smoothed state estimate in $\hat{X}(1)$ (A.24).

Stage 2 Filter and Smoother: $(i = 2)$

At stage 2 we will show the parametric structure of the estimators up to the point that matrix inverses are required, at which point we will plug in the parameter values for this example.

Enlarged Kalman Filter:

From (A.9) we find the following expression

$$\bar{\mathbf{X}}(2) = F(1)\bar{\mathbf{X}}(1)$$

$$
= \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}}_{F(1)} \underbrace{\begin{bmatrix} \bar{x}(0) \\ \bar{x}(0) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{\bar{\mathbf{X}}(1)} = \begin{bmatrix} \bar{x}(0) \\ \bar{x}(0) \\ \bar{x}(0) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}.
$$

From (A.11) we find the following expression

$$
\mathbf{P}(2) = F(1)\mathbf{P}(1)F^T(1) + G_y(1)\mathbf{Y}(1)G_y^T(1)
$$

$$
= \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}}_{F(1)} \underbrace{\begin{bmatrix} M(0) & M(0) & 0 & 0 & 0 & 0 & 0 & 0 \\ M(0) & M(0)+W(0) & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & V(0) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & V(1) & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}}_{\mathbf{P}(1)}
$$

$$
* \underbrace{\begin{bmatrix} 1&0&0&0&0&0&0&0\\ 0&1&1&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&1&0&0&0\\ 0&0&0&0&0&1&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0 \end{bmatrix}}_{F^T(1)} + \underbrace{\begin{bmatrix} 0&0\\ 0&0\\ 1&0\\ 0&0\\ 0&0\\ 0&0\\ 0&1\\ 0&0 \end{bmatrix}}_{G_y(1)} \underbrace{\begin{bmatrix} W(1) & 0\\ 0 & V(2) \end{bmatrix}}_{\mathbf{Y}(1)} \underbrace{\begin{bmatrix} 0&0&1&0&0&0&0&0\\ 0&0&0&0&0&0&1&0 \end{bmatrix}}_{G_y^T(1)}
$$

$$
= \begin{bmatrix} M(0) & M(0) & M(0) & 0 & 0 & 0 & 0 & 0\\ M(0) & M(0)+W(0) & M(0)+W(0) & 0 & 0 & 0 & 0 & 0\\ M(0) & M(0)+W(0) & M(0)+W(0)+W(1) & 0 & 0 & 0 & 0 & 0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&V(0)&0&0&0\\ 0&0&0&0&0&V(1)&0&0\\ 0&0&0&0&0&0&V(2)&0\\ 0&0&0&0&0&0&0&0 \end{bmatrix}.
$$

Using the above $\bar{\mathbf{X}}(2)$ and $\mathbf{P}(2)$ in (A.15) we find the following expression

$$
\hat{X}(2) = \bar{\mathbf{X}}(2) + \mathbf{P}(2)H^T(2)\big(H(2)\mathbf{P}(2)H^T(2)\big)^{-1}\big(Z(2) - H(2)\bar{\mathbf{X}}(2)\big),
$$

$$\underbrace{\begin{bmatrix} \hat{x}(0|2) \\ \hat{x}(1|2) \\ \hat{x}(2) \\ 0 \\ \hat{v}(0|2) \\ \hat{v}(1|2) \\ \hat{v}(2) \\ 0 \end{bmatrix}}_{\hat{X}(2)} = \underbrace{\begin{bmatrix} \bar{x}(0) \\ \bar{x}(0) \\ \bar{x}(0) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{\bar{\mathbf{X}}(2)} + \underbrace{\begin{bmatrix} M(0) & M(0) & M(0) & 0 & 0 & 0 & 0 & 0 \\ M(0) & M(0)+W(0) & M(0)+W(0) & 0 & 0 & 0 & 0 & 0 \\ M(0) & M(0)+W(0) & M(0)+W(0)+W(1) & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & V(0) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & V(1) & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & V(2) & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}}_{\mathbf{P}(2)}$$

$$* \underbrace{\begin{bmatrix} \Theta^T(0) & 0 & 0 \\ 0 & \Theta^T(1) & 0 \\ 0 & 0 & \Theta^T(2) \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}}_{H^T(2)} \left(\underbrace{\begin{bmatrix} \Theta(0) & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & \Theta(1) & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & \Theta(2) & 0 & 0 & 0 & 1 & 0 \end{bmatrix}}_{H(2)} \mathbf{P}(2) H^T(2) \right)^{-1}$$

$$* \left(\underbrace{\begin{bmatrix} \Theta(0)x(0)+v(0) \\ \Theta(1)x(1)+v(1) \\ \Theta(2)x(2)+v(2) \end{bmatrix}}_{Z(2)} - \underbrace{\begin{bmatrix} \Theta(0) & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & \Theta(1) & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & \Theta(2) & 0 & 0 & 0 & 1 & 0 \end{bmatrix}}_{H(2)} \underbrace{\begin{bmatrix} \bar{x}(0) \\ \bar{x}(0) \\ \bar{x}(0) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{\bar{\mathbf{X}}(2)} \right)$$

$$
= \begin{bmatrix} \bar{x}(0) \\ \bar{x}(0) \\ \bar{x}(0) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} M(0)\Theta^T(0) & M(0)\Theta^T(1) & M(0)\Theta^T(2) \\ M(0)\Theta^T(0) & (M(0)+W(0))\,\Theta^T(1) & (M(0)+W(0))\,\Theta^T(2) \\ M(0)\Theta^T(0) & (M(0)+W(0))\,\Theta^T(1) & (M(0)+W(0)+W(1))\,\Theta^T(2) \\ 0 & 0 & 0 \\ V(0) & 0 & 0 \\ 0 & V(1) & 0 \\ 0 & 0 & V(2) \\ 0 & 0 & 0 \end{bmatrix}
$$

$$
* \begin{bmatrix} \Theta(0)M(0)\Theta^T(0)+V(0) & \Theta(0)M(0)\Theta^T(1) & \dots \\ \Theta(1)M(0)\Theta^T(0) & \Theta(1)\,(M(0)+W(0))\,\Theta^T(1)+V(1) & \dots \\ \Theta(2)M(0)\Theta^T(0) & \Theta(2)\,(M(0)+W(0))\,\Theta^T(1) & \dots \end{bmatrix}
$$

$$
\begin{matrix} \dots & \Theta(0)M(0)\Theta^T(2) \\ \dots & \Theta(1)\,(M(0)+W(0))\,\Theta^T(2) \\ \dots & \Theta(2)\,(M(0)+W(0)+W(1))\,\Theta^T(2)+V(2) \end{matrix} \Bigg]^{-1} \begin{bmatrix} \Theta(0)x(0)+v(0)-\Theta(0)\bar{x}(0) \\ \Theta(1)x(1)+v(1)-\Theta(1)\bar{x}(0) \\ \Theta(2)x(2)+v(2)-\Theta(2)\bar{x}(0) \end{bmatrix}.
$$

Plugging in the parameter values for this example we calculate

$$
\hat{X}(2) = \begin{bmatrix} \hat{x}(0|2) \\ \hat{x}(1|2) \\ \hat{x}(2) \\ 0 \\ \hat{v}(0|2) \\ \hat{v}(1|2) \\ \hat{v}(2) \\ 0 \end{bmatrix} = \begin{bmatrix} 104.66 \\ 106.89 \\ 105.03 \\ 0 \\ -0.32 \\ 1.23 \\ -0.56 \\ 0 \end{bmatrix}. \tag{A.27}
$$

Traditional Kalman Filter:

From (A.2) we find the following expressions

$$\begin{aligned}
\bar{x}(2) &= \hat{x}(1) \\
P(1) &= M(1) - M(1)\Theta^T(1)\big(\Theta(1)M(1)\Theta^T(1) + V(1)\big)^{-1}\Theta(1)M(1) \\
M(2) &= P(1) + W(1) \\
\hat{x}(2) &= \bar{x}(2) + M(2)\Theta^T(2)\big(\Theta(2)M(2)\Theta^T(2) + V(2)\big)^{-1}\big(z(2) - \Theta(2)\bar{x}(2)\big).
\end{aligned}$$

Using the previous results from stage 1 and plugging in the parameter values for this example we calculate

$$\begin{aligned}
\bar{x}(2) &= 107.13 \\
P(1) &= 0.6631 \\
M(2) &= 5.66 \\
\hat{x}(2) &= 105.03 \\
\hat{v}(2) &= z(2) - \Theta(2)\hat{x}(2) = -0.56.
\end{aligned} \tag{A.28}$$

These values for $\hat{x}(2)$ and $\hat{v}(2)$ (A.28) are identical to the enlarged Kalman filter current state estimate in $\hat{X}(2)$ (A.27).

Now, for the smoothing at $k = 0$ and $k = 1$ given measurements up to $i = 2$ we use (A.21) to find the following expressions

$$\begin{aligned}
P(0|2) &= P(0|1)\big(I - P(2)\Theta^T(2)V^{-1}(2)\Theta(2)\big)^T \\
P(1|2) &= P(1)\big(I - P(2)\Theta^T(2)V^{-1}(2)\Theta(2)\big)^T \\
\hat{x}(0|2) &= \hat{x}(0|1) + P(0|2)\Theta^T(2)V^{-1}(2)\big(z(2) - \Theta(2)\bar{x}(2)\big) \\
\hat{x}(1|2) &= \hat{x}(1) + P(1|2)\Theta^T(2)V^{-1}(2)\big(z(2) - \Theta(2)\bar{x}(2)\big)
\end{aligned}$$

where

$$P(2) = M(2) - M(2)\Theta^T(2)\big(\Theta(2)M(2)\Theta^T(2) + V(2)\big)^{-1}\Theta(2)M(2).$$

Using the previous results from stage 1 and plugging in the parameter values for this example we calculate

$$
\begin{aligned}
P(2) &= 0.6623 \\
P(0|2) &= 0.0098 \\
P(1|2) &= 0.0775 \\
\hat{x}(0|2) &= 104.66 \\
\hat{x}(1|2) &= 106.89 \\
\hat{v}(0|2) &= z(0) - \Theta(0)\hat{x}(0|2) = -0.32 \\
\hat{v}(1|2) &= z(1) - \Theta(1)\hat{x}(1|2) = 1.23.
\end{aligned}
\tag{A.29}
$$

These values for $\hat{x}(0|2)$, $\hat{x}(1|2)$, $\hat{v}(0|2)$, and $\hat{v}(1|2)$ (A.29) are identical to the enlarged Kalman filter smoothed state estimate in $\hat{X}(2)$ (A.27).

In summary, we have shown by means of a simple three-stage, scalar example that $\hat{X}(i)$ produces a current state estimate, as well as a smoothed estimate of all past states, and that these estimates are identical to the traditional Kalman filter and smoother.

APPENDIX B

Mathematical Background

In the following sections we will consider matrices A, B and C, which are not functions of matrix X.

B.1 Linear Algebra

B.1.1 Matrix Inversion Lemma

From [BV04], we can derive the *matrix inversion lemma* as follows. Given an equation such as

$$(A + BC)x = b \tag{B.1}$$

where x and b are vectors and A is nonsingular, we can solve for x using

$$x = (A + BC)^{-1}b. \tag{B.2}$$

Likewise, we could introduce a variable $y \triangleq Cx$ and instead write (B.1) as

$$Ax + By = b. \tag{B.3}$$

Now, solving (B.3) for x we find

$$x = A^{-1}(b - By) \tag{B.4}$$

and substituting into our variable y definition we find

$$y = CA^{-1}(b - By)$$

$$= \big(I + CA^{-1}B\big)^{-1}CA^{-1}b. \tag{B.5}$$

Finally, substituting (B.5) into (B.4) we find

$$\begin{aligned} x &= A^{-1}\big(b - B\big(I + CA^{-1}B\big)^{-1}CA^{-1}b\big) \\ &= \big(A^{-1} - A^{-1}B\big(I + CA^{-1}B\big)^{-1}CA^{-1}\big)b. \end{aligned} \tag{B.6}$$

We now have two expressions for x as seen in (B.2) and (B.6), and by equating these expressions, since b is arbitrary, we can see that

$$\big(A + BC\big)^{-1} = A^{-1} - A^{-1}B\big(I + CA^{-1}B\big)^{-1}CA^{-1}. \tag{B.7}$$

This expression (B.7) is known as the *matrix inversion lemma*.

B.1.2 Singular Value Decomposition

Any matrix $A \in \mathbb{R}^{m \times n}$ with $rank(A) = r$ can be factored using

$$A = U\Sigma V^T \tag{B.8}$$

where $U \in \mathbb{R}^{m \times m}$ and $V \in \mathbb{R}^{n \times n}$ are orthonormal matrices and $\Sigma \in \mathbb{R}^{m \times n}$ is a sparse matrix of singular values. The upper-left block element of Σ is $r \times r$ in dimension with the r non-zero singular values along its diagonal

$$\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_r > 0. \tag{B.9}$$

We write the maximum (or first) singular value of matrix A as $\bar{\sigma}(A)$, and it is related to the ℓ_2 induced norm of the matrix as

$$\bar{\sigma}(A) = \sup_{x \neq 0} \frac{\|Ax\|_2}{\|x\|_2}. \tag{B.10}$$

That is, the maximum singular value may be thought of as the maximum gain of the matrix.

B.1.3 Discrete Sylvester Equation

From [Lau05], the general linear matrix equation

$$\sum_{i=1}^{k} A_i X B_i = C \tag{B.11}$$

may be written as

$$[B_1^T \otimes A_1 + B_2^T \otimes A_2 + \ldots + B_k^T \otimes A_k] vec(X) = vec(C) \tag{B.12}$$

with unique solution (when the matrix is nonsingular)

$$vec(X) = [B_1^T \otimes A_1 + B_2^T \otimes A_2 + \ldots + B_k^T \otimes A_k]^{-1} vec(C) \tag{B.13}$$

where $\otimes$ is the Kronecker product, and *vec* is the *vectorization* or column stacking operator.

B.1.4 Trace Operator

Some properties of the *trace* (Tr) operator are

$$Tr(AB) = Tr(BA) = Tr(B^T A^T) = Tr(A^T B^T) \tag{B.14}$$

$$Tr(A^T B) = vec(A)^T vec(B). \tag{B.15}$$

B.2 Matrix Calculus

B.2.1 Matrix Differential Properties

Some properties of the matrix differential are

$$d(X^T) = (dX)^T \tag{B.16}$$

$$dI = d(XX^{-1}) \tag{B.17}$$

$$0 = (dX)X^{-1} + X(dX^{-1}) \tag{B.18}$$

$$dX^{-1} = -X^{-1}(dX)X^{-1}. \tag{B.19}$$

B.2.2 Trace Differential Properties

From [Min00], to convert from *trace* differential form to derivative form

$$dy = Tr(AdX) \implies \frac{\partial y}{\partial X} = A. \tag{B.20}$$

Some properties of the *trace* differential are

$$dTr(AX) = Tr(A(dX)) \tag{B.21}$$

$$dTr(AX^T) = Tr(A(dX)^T) = Tr((dX)A^T) = Tr(A^T(dX)) \tag{B.22}$$

$$dTr(AXB) = Tr(A(dX)B) = Tr(BA(dX)) \tag{B.23}$$

$$\begin{aligned} dTr(X^TAX) &= Tr((dX)^TAX + X^TA(dX)) \\ &= Tr(X^TA^T(dX) + X^TA(dX)) = Tr(X^T(A + A^T)(dX)) \end{aligned} \tag{B.24}$$

$$\begin{aligned} dTr(AXBX^TC) &= Tr(A(dX)BX^TC + AXB(dX)^TC) \\ &= Tr((BX^TCA + B^TX^TA^TC^T)(dX)). \end{aligned} \tag{B.25}$$

APPENDIX C

MATLAB Code: Solving the LQG Multistage Game

The MATLAB code included below may be used to solve the system of equations in Section 4.2, including the two-point boundary value problem solution as outlined in Section 4.3.

```
%------------------------------------------------------------------%
% CLEMENS_MULTISTAGE.M
% Solve two-point boundary value problem associated with the LQG
% multistage game.
% Notes:
% 1. Accommodates non-zero xbar0, non-scalar problem.
% 2. The stage of the game is always the third dimension, unless the
%    second dimension (column) is singleton, in which case the
%    stage is the second dimension.
%------------------------------------------------------------------%

function [bp,Kp,Lp,Hp,Kp_det,be,Ke,Le,He,Ke_det,S1,S2,Jopt_cs] = ...
            clemens_multistage(Gammap,Gammae,Thetap,Thetae,Rp,Re,QN,N,n
               ,m,l,p,q,xbar0,M0,W,Vp,Ve,max_iter,varargin)

% up(i) = -bp(i) - Kp(i)Zp(i)
% ue(i) =  be(i) + Ke(i)Ze(i)

%--------------- TPBVP convergence criteria ------------------%

diff_tpbvp = 1e-10;
```

```
k1_tpbvp = 1.0;    % allow large changes in S2
k2_tpbvp = 0.5;    % allow small changes in S2
k3_tpbvp = 0.075; % allow small changes in S2
if max(max(max(Vp)))<20 && max(max(max(Ve)))<20
    k_tpbvp = k1_tpbvp;
elseif max(max(max(Vp)))<100 && max(max(max(Ve)))<100
    k_tpbvp = k2_tpbvp;
else
    k_tpbvp = k3_tpbvp;
end

%------------------ Initialize enlarged matrices ----------------------%

dimxl = N*(n+p+q); % 'xl' means 'extra-large', defines size of cell
   arrays, actual matrix dimension is N*(n+p+q)

% cell array initialization
ca_init = cell(3*N,3*N,N);
ca_init(1:N,        1:N,:) = {zeros(n,n)}; ca_init(1:N,        N+1:2*N,:) =
    {zeros(n,p)}; ca_init(1:N,        2*N+1:3*N,:) = {zeros(n,q)};
ca_init(N+1:2*N,    1:N,:) = {zeros(p,n)}; ca_init(N+1:2*N,    N+1:2*N,:) =
    {zeros(p,p)}; ca_init(N+1:2*N,    2*N+1:3*N,:) = {zeros(p,q)};
ca_init(2*N+1:3*N,1:N,:) = {zeros(q,n)}; ca_init(2*N+1:3*N,N+1:2*N,:) =
    {zeros(q,p)}; ca_init(2*N+1:3*N,2*N+1:3*N,:) = {zeros(q,q)};

S1          = ca_init;
S1{N,N,N}   = QN;

P                = ca_init;
P{1,1,1}         = M0;    % non-zero mean state
```

```
P{N+1,N+1,1}      = Vp(:,:,1);    % 0 mean noise
P{N+N+1,N+N+1,1}= Ve(:,:,1);    % 0 mean noise

Xbar            = zeros(dimxl,N);
Xbar(1:n,1)     = xbar0;

F      = ca_init(:,:,1:N-1);
Ftild  = F;

Gp                  = cell(3*N,N-1);
Gp(1:N,:)           = {zeros(n,m)};
Gp(N+1:2*N,:)       = {zeros(p,m)};
Gp(2*N+1:3*N,:)     = {zeros(q,m)};
Ge                  = cell(3*N,N-1);
Ge(1:N,:)           = {zeros(n,l)};
Ge(N+1:2*N,:)       = {zeros(p,l)};
Ge(2*N+1:3*N,:)     = {zeros(q,l)};

Hp = ca_init(N+1:2*N-1,:,1);    % N-1 measurements with dim p
He = ca_init(2*N+1:3*N-1,:,1); % N-1 measurements with dim q

Gy = cell(3*N,3,N-1);
Gy(1:N,         1,:) = {zeros(n,n)}; Gy(1:N,         2,:) = {zeros(n,p)}; Gy
    (1:N,         3,:) = {zeros(n,q)};
Gy(N+1:2*N,     1,:) = {zeros(p,n)}; Gy(N+1:2*N,     2,:) = {zeros(p,p)}; Gy(
    N+1:2*N,     3,:) = {zeros(p,q)};
Gy(2*N+1:3*N,1,:) = {zeros(q,n)}; Gy(2*N+1:3*N,2,:) = {zeros(q,p)}; Gy
    (2*N+1:3*N,3,:) = {zeros(q,q)};

bp = zeros(m,N-1);
```

```
be = zeros(l,N-1);

Kp      = zeros(m,p*(N-1),N-1);
Kp_det  = zeros(m,dimxl,N-1);
Ke      = zeros(l,q*(N-1),N-1);
Ke_det  = zeros(l,dimxl,N-1);

Lp = zeros(dimxl,p*(N-1),N-1);
Le = zeros(dimxl,q*(N-1),N-1);

for i = 1:N-1

    F(i+1,i,i) = {eye(n)};

    for j = 1:i
        F(j,j,i)          = {eye(n)};
        F(N+j,N+j,i)      = {eye(p)};
        F(2*N+j,2*N+j,i)= {eye(q)};
    end

    Gp(i+1,i) = {Gammap(:,:,i)};
    Ge(i+1,i) = {Gammae(:,:,i)};

    Hp(i,i)    = {Thetap(:,:,i)};
    Hp(i,N+i) = {eye(p)};

    He(i,i)       = {Thetae(:,:,i)};
    He(i,2*N+i) = {eye(q)};

    Gy(i+1,     1,i) = {eye(n)}; % W(i)
```

```
    Gy(N+1+i,   2,i) = {eye(p)}; % Vp(i+1)
    Gy(N+N+1+i,3,i) = {eye(q)}; % Ve(i+1)

end

% Convert unnecessary cell arrays to matrices
S1      = cell2mat(S1);
P       = cell2mat(P);
F       = cell2mat(F);
Ftild   = cell2mat(Ftild);
Gp      = cell2mat(Gp);
Ge      = cell2mat(Ge);
Hp      = cell2mat(Hp);
He      = cell2mat(He);
Gy      = cell2mat(Gy);

%———————— Solve for deterministic S1 sequence ————————————%

for i = N-1:-1:1
    S1(:,:,i) = F(:,:,i)'*(eye(dimxl)+S1(:,:,i+1)*Gp(:,(i-1)*m+1:i*m)*
        Rp(:,:,i)^-1*Gp(:,(i-1)*m+1:i*m)'-S1(:,:,i+1)*Ge(:,(i-1)*l+1:i*l
        )*Re(:,:,i)^-1*Ge(:,(i-1)*l+1:i*l)')^-1*S1(:,:,i+1)*F(:,:,i);
end

if isempty(varargin{1})
    % initialize S2_guess sequence (for stochastic game) using the S1
        sequence (for deterministic game)
    S2_guess = S1;
else
    S2_guess = varargin{1};
```

```
end

diff = inf;

%———————— Solve primary system of equations ————————————%

for iter = 1:max_iter

    disp(['Starting Iteration #',num2str(iter)])
    disp('     Starting Forward Propagation')

    % Solve for state covariance sequence and control strategies
    for i = 1:N-1

        Lp(:,1:i*p,i) = P(:,:,i)*Hp(1:i*p,:)'*(Hp(1:i*p,:)*P(:,:,i)*Hp
            (1:i*p,:)')^-1;
        Le(:,1:i*q,i) = P(:,:,i)*He(1:i*q,:)'*(He(1:i*q,:)*P(:,:,i)*He
            (1:i*q,:)')^-1;

        % Discrete Sylvester Equation:  -Kp(i) + A*Kp(i)*B + C = 0
        A = Rp(:,:,i)^-1*Gammap(:,:,i)'*(eye(n)+S2_guess(n*i+1:n*i+n,n*
            i+1:n*i+n,i+1)*Gammap(:,:,i)*Rp(:,:,i)^-1*Gammap(:,:,i)'-
            S2_guess(n*i+1:n*i+n,n*i+1:n*i+n,i+1)*Gammae(:,:,i)*Re(:,:,i
            )^-1*Gammae(:,:,i)')^-1*...
             S2_guess(n*i+1:n*i+n,n*i+1:n*i+n,i+1)*Gammae(:,:,i)*Re(:,:,
                 i)^-1*Gammae(:,:,i)'*S2_guess(n*i+1:n*i+n,:,i+1)*Gp(:,(i
                 -1)*m+1:i*m); % Note that this last multiplication could
                  be simplified to S2_guess(n*i+1:n*i+n,n*i+1:n*i+n,i+1)*
                 Gammap(:,:,i)
```

```
    B = Hp(1:i*p,:)*(eye(dimxl)-Le(:,1:i*q,i)*He(1:i*q,:))*Lp(:,1:i
        *p,i); % Note that Hp(1:i*p,:)*Lp(:,1:i*p,i) = I
    C = Rp(:,:,i)^-1*Gammap(:,:,i)'*(eye(n)+S2_guess(n*i+1:n*i+n,n*
        i+1:n*i+n,i+1)*Gammap(:,:,i)*Rp(:,:,i)^-1*Gammap(:,:,i)'-
        S2_guess(n*i+1:n*i+n,n*i+1:n*i+n,i+1)*Gammae(:,:,i)*Re(:,:,i
        )^-1*Gammae(:,:,i)')^-1*...
         (S2_guess(n*i+1:n*i+n,:,i+1)*F(:,:,i)*Lp(:,1:i*p,i) -
             S2_guess(n*i+1:n*i+n,n*i+1:n*i+n,i+1)*Gammae(:,:,i)*Re
             (:,:,i)^-1*Gammae(:,:,i)'*S2_guess(n*i+1:n*i+n,:,i+1)*F
             (:,:,i)*(eye(dimxl)-Le(:,1:i*q,i)*He(1:i*q,:))*Lp(:,1:i*
             p,i));
    Kp(:,1:i*p,i) = dlyap(A,B,C);

    % Substitute Kp(:,1:i*p,i) and solve for Ke(:,1:i*q,i)
    Ke(:,1:i*q,i) = -Re(:,:,i)^-1*Gammae(:,:,i)'*(eye(n)-S2_guess(n
        *i+1:n*i+n,n*i+1:n*i+n,i+1)*Gammae(:,:,i)*Re(:,:,i)^-1*
        Gammae(:,:,i)')^-1*S2_guess(n*i+1:n*i+n,:,i+1)*(F(:,:,i)-Gp
        (:,(i-1)*m+1:i*m)*Kp(:,1:i*p,i)*Hp(1:i*p,:))*Le(:,1:i*q,i);

    Ftild(:,:,i)= (F(:,:,i) - Gp(:,(i-1)*m+1:i*m)*Kp(:,1:i*p,i)*Hp
        (1:i*p,:) - Ge(:,(i-1)*l+1:i*l)*Ke(:,1:i*q,i)*He(1:i*q,:));

    P(:,:,i+1)  = Ftild(:,:,i)*P(:,:,i)*Ftild(:,:,i)' + Gy(:,:,i)*
        diag([diag(W(:,:,i));diag(Vp(:,:,i+1));diag(Ve(:,:,i+1))])*
        Gy(:,:,i)';

end % i loop

S2_guess_fwd = S2_guess;
```

```
    disp('     Starting Backward Propagation')

    % Backward propagate Lagrange multiplier sequence.
    for i = N-1:-1:1
        S2_guess(:,:,i) = Hp(1:i*p,:)'*Kp(:,1:i*p,i)'*Rp(:,:,i)*Kp(:,1:
           i*p,i)*Hp(1:i*p,:) - He(1:i*q,:)'*Ke(:,1:i*q,i)'*Re(:,:,i)*
           Ke(:,1:i*q,i)*He(1:i*q,:) + Ftild(:,:,i)'*S2_guess(:,:,i+1)*
           Ftild(:,:,i);
    end

    % Compare forward and backward propagations
    diff_last = diff;
    diff = max(max(max(abs(S2_guess-S2_guess_fwd))));
    disp(['     Max Change in Lagrange Multiplier Sequence:  ',num2str(
       diff),' (k_tpbvp = ',num2str(k_tpbvp),')'])
    if (diff < diff_tpbvp)
        disp('>>>>>>>>>>  Converged!  <<<<<<<<<<')
        break
    elseif diff > diff_last
        k_tpbvp = max(k_tpbvp*0.9,0.075);
    end
    S2_guess = S2_guess_fwd+k_tpbvp*(S2_guess-S2_guess_fwd);

end % iter loop

if iter == max_iter
    disp('>>>>>>>>>>  Solution did not converge!  <<<<<<<<<<')
    S2_guess = 9999*ones(size(S2_guess));
end
S2 = S2_guess;
```

```
% Check convexity/concavity of pursuer/evader solution
convexity_ok = 1;
concavity_ok = 1;
for i = 1:N-1
    if any(~(eig(Rp(:,:,i)+Gammap(:,:,i)'*S2(n*i+1:n*i+n,n*i+1:n*i+n,i
        +1)*Gammap(:,:,i)) > 0)) % if any eigenvalues are not greater
        than zero
         convexity_ok = 0;
         warning(['Pursuer Convexity Violated at Stage ',num2str(i)])
    end
    if any(~(eig(-Re(:,:,i)+Gammae(:,:,i)'*S2(n*i+1:n*i+n,n*i+1:n*i+n,i
        +1)*Gammae(:,:,i)) < 0)) % if any eigenvalues are not less than
        zero
         concavity_ok = 0;
         warning(['Evader Concavity Violated at Stage ',num2str(i)])
    end
end
if convexity_ok && concavity_ok
    disp('Solution is Valid!')
else
    warning('>>>>>  Convexity and/or Concavity Violated!  <<<<<')
    pause
end

%----------- Solve secondary system of equations ----------------%

% Already solved for S1 above in order to initialize S2
for i = 1:N-1
```

```
        Xbar(:,i+1)     = (eye(dimxl)+Gp(:,(i-1)*m+1:i*m)*Rp(:,:,i)^-1*Gp(:,(
            i-1)*m+1:i*m)'*S1(:,:,i+1)-Ge(:,(i-1)*l+1:i*l)*Re(:,:,i)^-1*Ge
            (:,(i-1)*l+1:i*l)'*S1(:,:,i+1))^-1*F(:,:,i)*Xbar(:,i);
        Kp_det(:,:,i)  = Rp(:,:,i)^-1*Gp(:,(i-1)*m+1:i*m)'*(eye(dimxl)+S1
            (:,:,i+1)*Gp(:,(i-1)*m+1:i*m)*Rp(:,:,i)^-1*Gp(:,(i-1)*m+1:i*m)'-
            S1(:,:,i+1)*Ge(:,(i-1)*l+1:i*l)*Re(:,:,i)^-1*Ge(:,(i-1)*l+1:i*l)
            ')^-1*S1(:,:,i+1)*F(:,:,i);
        bp(:,i)         = Kp_det(:,:,i)*Xbar(:,i)-Kp(:,1:i*p,i)*Hp(1:i*p,:)*
            Xbar(:,i);
        Ke_det(:,:,i)  = -Re(:,:,i)^-1*Ge(:,(i-1)*l+1:i*l)'*(eye(dimxl)+S1
            (:,:,i+1)*Gp(:,(i-1)*m+1:i*m)*Rp(:,:,i)^-1*Gp(:,(i-1)*m+1:i*m)'-
            S1(:,:,i+1)*Ge(:,(i-1)*l+1:i*l)*Re(:,:,i)^-1*Ge(:,(i-1)*l+1:i*l)
            ')^-1*S1(:,:,i+1)*F(:,:,i);
        be(:,i)         = Ke_det(:,:,i)*Xbar(:,i)-Ke(:,1:i*q,i)*He(1:i*q,:)*
            Xbar(:,i);
end

%---------- Calculate optimal performance index ----------------%

Jopt_cs = 0;
for i = 1:N-1
     Jopt_cs = Jopt_cs + 0.5*trace(S2(:,:,i+1)*Gy(:,:,i)*diag([diag(W
         (:,:,i));diag(Vp(:,:,i+1));diag(Ve(:,:,i+1))])*Gy(:,:,i)');
end
Jopt_cs = Jopt_cs + 0.5*trace(S1(:,:,1)*Xbar(:,1)*Xbar(:,1)') + 0.5*
    trace(S2(:,:,1)*P(:,:,1));

return;
```

References

[Beh68] Robert Dietrich Behn. *Linear Stochastic Differential Games*. PhD thesis, Harvard University, 1968.

[Ber60] L. D. Berkovitz. "A Variational Approach to Differential Games." Technical Report P-2205, The RAND Corporation, Santa Monica, CA, November 1960.

[BF55] L. D. Berkovitz and W. H. Fleming. "On Differential Games with Integral Payoff." Technical Report P-717, The RAND Corporation, Santa Monica, CA, August 1955.

[BH68] Robert D. Behn and Yu-Chi Ho. "On a Class of Linear Stochastic Differential Games." *IEEE Transactions on Automatic Control*, **AC-13**(3):227–240, June 1968.

[BH75] Arthur E. Bryson, Jr. and Yu-Chi Ho. *Applied Optimal Control*. Taylor & Francis, 1975.

[BO81] Arunabha Bagchi and Geert Jan Olsder. "Linear-Quadratic Stochastic Pursuit-Evasion Games." *Applied Mathematics and Optimization*, (7):95–123, 1981.

[BS17] Joseph Z. Ben-Asher and Jason L. Speyer. "Games In Aerospace: Homing Missile Guidance." In Tamer Basar and Georges Zaccour, editors, *Handbook of Dynamic Game Theory*. Springer International Publishing, 2017.

[BV04] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[FPW98] Gene F. Franklin, J. David Powell, and Michael Workman. *Digital Control of Dynamic Systems*. Addison Wesley Longman, Inc., 1998.

[GNL14] Abhishek Gupta, Ashutosh Nayyar, Cédric Langbort, and Tamer Basar. "Common Information Based Markov Perfect Equilibria for Linear-Gaussian Games with Asymmetric Information." *Society for Industrial and Applied Mathematics*, **52**(5):3228–3260, 2014.

[Gup14] Abhishek Gupta. *Dynamic Sequential Decision Problems with Asymmetric Information: Some Existence Results*. PhD thesis, University of Illinois at Urbana-Champaign, 2014.

[HBB65] Y. C. Ho, A. E. Bryson, Jr., and S. Baron. "Differential Games and Optimal Pursuit-Evasion Strategies." *IEEE Transactions on Automatic Control*, **AC-10**(4):385–389, October 1965.

[Isa55] Rufus Isaacs. "Differential Games I, II, III, IV." Research Memorandum 1391, 1399, 1411, 1486, The RAND Corporation, Santa Monica, CA, 1954-1955.

[Isa65] Rufus Isaacs. *Differential Games*. John Wiley and Sons, Inc., 1965.

[Lau05] Alan J. Laub. *Matrix Analysis for Scientists & Engineers*. Society for Industrial and Applied Mathematics, 2005.

[LL11] Laurent Lessard and Sanjay Lall. "A State-Space Solution to the Two-Player Decentralized Optimal Control Problem." In *Allerton Conference on Communication, Control, and Computing*, pp. 1559–1564, 2011.

[LL12] Laurent Lessard and Sanjay Lall. "Optimal Controller Synthesis for the Decentralized Two-Player Problem with Output Feedback." In *American Control Conference*, pp. 6314–6321, 2012.

[Mah08] Aditya Mahajan. *Sequential decomposition of sequential dynamic teams: applications to real-time communication and networked control system*. PhD thesis, University of Michigan, 2008.

[Min00] Thomas P. Minka. "Old and New Matrix Algebra Useful for Statistics.", December 2000. MIT Media Lab note.

[Nas50] John F. Nash, Jr. "Equilibrium Points in n-Person Games." *Proceedings of the National Academy of Sciences of the United States of America*, **36**(1):48–49, January 1950.

[NGL14] Ashutosh Nayyar, Abhishek Gupta, Cédric Langbort, and Tamer Basar. "Common Information Based Markov Perfect Equilibria for Stochastic Games with Asymmetric Information: Finite Games." *IEEE Transactions on Automatic Control*, **59**(3):555–570, March 2014.

[NM44] John von Neumann and Oskar Morgenstern. *Theory of Games and Economic Behavior*. Princeton University Press, 1944.

[NMT13] Ashutosh Nayyar, Aditya Mahajan, and Demosthenis Teneketzis. "Decentralized Stochastic Control with Partial History Sharing: A Common Information Approach." *IEEE Transactions on Automatic Control*, **58**(7):1644–1658, July 2013.

[Pac16] Meir Pachter. "Linear-Quadratic Gaussian Dynamic Games with a Control-Sharing Information Pattern.", 2016. Chapter 8 of unpublished book.

[PP10] M. Pachter and K. D. Pham. "Discrete-Time Linear-Quadratic Dynamic Games." *Journal of Optimization Theory and Applications*, **146**(1):151–179, July 2010.

[RL69] Ian B. Rhodes and David G. Luenberger. "Differential Games with Imperfect State Information." *IEEE Transactions on Automatic Control*, **AC-14**(1):29–38, February 1969.

[SC08] Jason L. Speyer and Walter H. Chung. *Stochastic Processes, Estimation, and Control*. Society for Industrial and Applied Mathematics, 2008.

[SH69] A. W. Starr and Y. C. Ho. "Nonzero-Sum Differential Games." *Journal of Optimization Theory and Applications*, **3**(3):184–206, 1969.

[Sha53] L. S. Shapely. "Stochastic Games." *Proceedings of the National Academy of Sciences of the United States of America*, **39**(10):1095–1100, October 1953.

[SKT90] Jason L. Speyer, Kevin D. Kim, and Minjea Tahk. "A Passive Homing Missile Guidance Law Based on New Target Maneuver Models." Technical Report AFATL-TP-90-05, Air Force Armament Laboratory, Eglin Air Force Base, FL, January 1990.

[SOT09] Josef Shinar, Yaakov Oshman, Vladimir Turetsky, and Johnny Evers. "Integrated Estimator/Guidance Law Design for Improved Ballistic Missile Defense.", March 2009. Modified from a presentation at AIAA Missile Sciences Conference in November 2004.

[Wil68] W. W. Willman. "Formal Solutions for a Class of Stochastic Pursuit-Evasion Games." Technical Report 575, Harvard University, Cambridge, MA, November 1968.

[Wil69] Warren W. Willman. "Formal Solutions for a Class of Stochastic Pursuit-Evasion Games." *IEEE Transactions on Automatic Control*, **AC-14**(5):504–509, October 1969.

[Wit68] H. S. Witsenhausen. "A Counterexample in Stochastic Optimum Control." *SIAM Journal on Control*, **6**(1), 1968.

[Wit73] H. S. Witsenhausen. "A Standard Form for Sequential Stochastic Control." *Mathematical Systems Theory*, **7**(1):5–11, March 1973.