Optimal Trajectory Design Under Uncertainty

by

Benjamin R. Saunders
B.S. Mechanical Engineering
United States Air Force Academy, 2010

Submitted to the Department of Aeronautics and Astronautics
in partial fulfillment of the requirements for the degree of
Master of Science in Aeronautics and Astronautics
at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
June 2012

Author .....
Department of Aeronautics and Astronautics
May 24, 2012

Certified by .....
Steven R. Hall, Ph.D.
Professor of Aeronautics and Astronautics
MacVicar Faculty Fellow
Thesis Supervisor

Certified by .....
David Benson, Ph.D.
Senior Member of the Technical Staff
The Charles Stark Draper Laboratory, Inc.
Thesis Supervisor

Accepted by .....
Eytan H. Modiano
Professor of Aeronautics and Astronautics
Chair, Graduate Program Committee

Optimal Trajectory Design Under Uncertainty

by

Benjamin R. Saunders

Submitted to the Department of Aeronautics and Astronautics
on May 24, 2012, in partial fulfillment of the
requirements for the degree of
Master of Science in Aeronautics and Astronautics

Abstract

Reference trajectory design for atmospheric reentry vehicles can be accomplished through trajectory optimization using optimal control techniques. However, this method generally focuses on nominal vehicle performance and does not include robustness considerations during trajectory design. This thesis explores the use of linear covariance analysis to directly include trajectory robustness in the design process. The covariance matrix can be propagated along a trajectory to provide the expected errors about the nominal trajectory in the presence of uncertainties. During the optimization process, the covariance matrix is used as a performance metric to be minimized, directly penalizing expected errors so that the trajectory is shaped to reduce its sensitivity to uncertainties. This technique can penalize the open-loop covariance of the trajectory or the closed-loop covariance with the inclusion of a feedback guidance law.

This covariance shaping technique is applied to reference trajectory design for a generic small reentry vehicle. A baseline trajectory is generated without any robustness considerations, along with an open-loop covariance shaped trajectory and a closed-loop covariance shaped trajectory, which uses a feedback guidance law based on a linear quadratic regulator scheme. Uncertainties in initial conditions, atmospheric density, aerodynamic coefficients, and unmodeled dynamics are applied to each trajectory and performance is analyzed using linear covariance analysis and Monte Carlo simulations. The results show that when the vehicle is flown closed-loop with feedback, shaping using the open-loop covariance produces a trajectory that is less robust than the baseline trajectory, while shaping using the closed-loop covariance generates a trajectory with reduced sensitivity to uncertainty for more robust performance.

Thesis Supervisor: Steven R. Hall, Ph.D.
Title: Professor of Aeronautics and Astronautics
MacVicar Faculty Fellow

Thesis Supervisor: David Benson, Ph.D.
Title: Senior Member of the Technical Staff
The Charles Stark Draper Laboratory, Inc.

Acknowledgements

I would like to begin by first thanking my two advisors at the Draper Laboratory, Ron Proulx and Dave Benson, as well as my advisor at MIT, Prof. Steven Hall. Ron and Dave, your previous work laid the foundation of my research and your guidance and support allowed me to continue the development of that work. Prof. Hall, your insight has been invaluable in shaping my research and thesis into its final form. I would also like to thank the U.S. Air Force, Draper Laboratory, and MIT for providing the research opportunities I have had over the last two years. Finally, I would like to thank my friends and family for your support throughout this process; your encouragement has always provided motivation. It is due to the effort and encouragement of all of these people that I was able to complete this thesis and I will always be grateful.

THIS PAGE INTENTIONALLY LEFT BLANK

The views expressed in this article are those of the author and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the U.S. Government

THIS PAGE INTENTIONALLY LEFT BLANK

Contents

THIS PAGE INTENTIONALLY LEFT BLANK

List of Figures

List of Tables

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 1

Introduction

The use of vehicles in space is more prevalent today than ever before, and their use can be expected to continue growing. Sending a vehicle beyond the atmosphere is a challenge, but returning it safely back to Earth is an entirely different problem. This is known as the atmospheric reentry problem and has been the subject of study since the beginning of space exploration. Traditionally, the mission planning process for reentry is conducted in two independent process; one for reference trajectory design and another for guidance algorithm design. Reference trajectory design is concerned primarily with generating the nominal path that the reentry vehicle takes as it moves from atmospheric entry to its terminal conditions, while the guidance algorithm is left to correct for errors from the nominal path due to disturbances and uncertainties. Uncertainties are not generally considered during reference trajectory design, but failure to do so can result in a trajectory that the guidance algorithm might not be able to successfully fly. To ensure robustness during reentry, techniques are needed that plan for uncertainty during the entire mission planning process.

1.1 Maneuverable Reentry Vehicles

Recent research concerning reentry has focused on maneuverable reentry vehicles that can be guided accurately to terminal conditions, even in the presence of disturbances. A large portion of this research has focused on reusable reentry vehicles such as the

X-33 considered in [1][2][3]. The X-33 was intended to be an unmanned technology demonstrator designed for launch to suborbital altitudes before gliding and landing autonomously on a runway. The X-34 was also designed as a test vehicle that was launched to suborbital altitudes to simulate reentry conditions [4]. The Crew Exploration Vehicle (CEV) is a low lift-to-drag (L/D) capsule type vehicle designed for manned spaceflight that has the capability to enter into the atmosphere then "skip" up again to increase its range [5]. Apart from these more modern reentry vehicles, the Space Shuttle is a proven reusable reentry vehicle, which has demonstrated the ability to return crew and cargo precisely to a landing location [6].

Considerable research has also examined the use of much smaller reentry vehicles. These vehicles are generally used for limited cargo transport, scientific payloads, and weapons deployment. Clarke [7] examined the Common Aero Vehicle (CAV) concept, which could provide the capability of rapid global delivery of small cargo payloads or weapons up to 2400 lbs. Small, high lift vehicles less than 500 lbs were considered by Abrahamson [8] for munitions deployment and Small [9] for carrying scientific payloads. These smaller vehicles require much smaller boost requirements to reach reentry conditions and as such are cheaper to use.

1.2 Reentry Mission Planning

For any reentry vehicle, proper mission planning is required to safely and precisely return to Earth. As noted, mission planning for the reentry problem is usually broken into two distinct phases; first, a nominal reference trajectory is generated to define the flight path the vehicle will take upon entering the atmosphere to its terminal conditions. The reference trajectory must take into account the nonlinear dynamics of the vehicle as well as any constraints that must be placed on the vehicle, such as heating or loading limitations, as well as mission constraints. The second phase involves the design of a guidance algorithm that will allow the vehicle to follow the reference trajectory while minimizing any deviations caused by perturbations along the way.

Reference trajectory design can be accomplished using optimal control techniques to develop a nominal flight path that meets constraints while minimizing a performance metric. Clarke [7] demonstrated the viability of using direct methods of solving optimal control problems by using the Legendre Pseudospectral Method to generate optimal reentry trajectories. Undurti [10] used similar techniques to generate optimal trajectories and construct footprints demonstrating the terminal capabilities of a small reentry vehicle. Abrahamson [8] furthered this research by developing optimal trajectories for a full boost through reentry mission profile. These studies all focused on exploring the capabilities of reentry vehicles under nominal conditions.

Traditionally, the responsibility for performance of reentry vehicles in off-nominal conditions is given to the vehicle's guidance algorithm. With a nominal reference trajectory designed, it is left to the guidance algorithm to minimize errors due to disturbances or uncertainties. Dukeman [2] and Lu [3] both demonstrated examples of reentry guidance for the X-33, while Tracy [4] developed an integrated guidance and control algorithm for the X-34. All three guidance techniques are based on the linear quadratic regulator (LQR). Bollino [1] used pseudospectral trajectory optimization techniques to develop a guidance algorithm using onboard trajectory generation. As the vehicle reenters the atmosphere, new nominal reference trajectories are repeatedly generated based on the vehicle's current position, mitigating the effects of any errors.

Instead of relying on the guidance algorithm to ensure robust performance, an alternative approach to reentry mission planning is to design reference trajectories in such a way as to reduce the sensitivity of the trajectory itself to uncertainties. This should then reduce the size of expected dispersions about the nominal trajectory. Seywald and Kumar [11] and Seywald [12] showed that the sensitivity matrix of the vehicle state can be used to quantify the sensitivity of a trajectory to perturbations. The nominal trajectory optimization problem can then be augmented so that the sensitivity matrix is included in the performance metric to be minimized, shaping the trajectory to reduce sensitivity to uncertainty. Another technique for quantifying the effects of uncertainties on a trajectory is linear covariance analysis, demonstrated by Geller [13] and Zanetti et al. [14]. This analysis is conducted by defining perturbations

that act on the vehicle and computing the covariance of the linearized system along the reference trajectory, which provides expected dispersions along the trajectory. This is in contrast to Monte Carlo simulations, which use random perturbations over a large number of simulated flights to determine the statistical effect of these disturbances. Linear covariance analysis offers a huge computational advantage over Monte Carlo simulations, requiring one simulated flight versus many. In the case of a system with nonlinear dynamics, however, Monte Carlo simulations will capture effects that may be missed due to the linearization inherent in the linear covariance analysis.

In many circumstances, linear covariance analysis is used to test robustness after a trajectory is designed. However, it was also used as early as 1968 by Vander Stoep [15] in conjunction with trajectory optimization to reduce the sensitivity of a trajectory to uncertainty. The optimal control problem is augmented so that the covariance matrix is included in the performance metric and minimized during the trajectory design. Zimmer et al. [16] used this covariance shaping technique to shape trajectories for continuous thrust spacecraft maneuvers. Small [9] analyzed both the sensitivity technique as well as the covariance shaping technique and demonstrated that both actually provide the same result during trajectory optimization, but that the formulation for utilizing covariance analysis was simpler to implement. The covariance analysis formulation was then applied to a reentry problem to demonstrate its capability for hypersonic atmospheric reentry. It is important to note that while linear covariance analysis is capable of taking into account a closed-loop guidance algorithm, as demonstrated in [13] and [14], the use of linear covariance during trajectory optimization in [9], [15], and [16] all only consider the open-loop covariance of the trajectory.

1.3 Covariance Shaping Techniques

The purpose of this thesis is not to develop and understand the capabilities of a particular reentry vehicle or reentry mission. Instead, the purpose is to refine and improve the mission planning process for atmospheric reentry by implementing a

covariance shaping technique during reference trajectory design, where the covariance is used within trajectory optimization to generate trajectories with reduced sensitivity to uncertainty. Having the capability to generate more robust trajectories directly within the mission planning process can reduce the need for iteration in the trajectory design process by removing the need to generate trajectories, test for robustness, and then repeating to improve performance.

In order to accomplish this in a manner that can be applied to a wide range of reentry missions and vehicles, this study will be kept as general as possible. The vehicle used is a hypothetical 250 lb reentry vehicle with a L/D ratio of 2.5. This L/D ratio provides moderate maneuverability and gliding capabilities within the atmosphere, where the vehicle is controlled using bank-to-turn (BTT) steering, which uses aerodynamic control surfaces such as flaps to rotate the vehicle and change angle-of-attack. The vehicle uses an onboard guidance algorithm based on LQR theory to track a nominal reference trajectory. The model used assumes three degree-of-freedom dynamics, where rotational dynamics are ignored and the vehicle is treated as a point mass. The reentry profile flown begins with the vehicle at an insertion point at 150,000 ft above the surface of the Earth located at 0 deg longitude, 0 deg latitude. The final condition is a specified location and altitude that simulates a terminal interface where another guidance algorithm would be used to accomplish final mission goals, such as landing or payload deployment.

The mission planning process used in this thesis involves the development of a guidance algorithm, followed by reference trajectory design. Linear covariance analysis is included in the trajectory optimization so that the trajectory design can be shaped to minimize the sensitivity to uncertainty. Both open-loop covariance and closed-loop covariance are used to quantify a trajectory's sensitivity to uncertainty to determine if one is more capable of generating more robust trajectories. The designed guidance algorithm is used in the closed-loop covariance analysis so that the performance of the guidance law is incorporated during the trajectory optimization process.

1.4 Thesis Overview

To develop this mission planning process, Chapter 2 provides a discussion of the theoretical framework for mission planning including reference trajectory planning, the LQR guidance algorithm design, and linear covariance analysis. Chapter 3 describes the system modeling used in this analysis including the vehicle model, environmental model, 3 DOF dynamics, and guidance law formulation. In Chapter 4, the optimal control problem used for trajectory optimization is defined, along with the framework used to model uncertainties in the analysis. This includes the linear covariance analysis, along with a Monte Carlo simulation, which is used to validate results. Three reference trajectories are designed using different mission planning techniques and, in Chapter 5, their performance is analyzed, with each trajectory being subjected to a variety of uncertainties and disturbances so that covariance analysis and Monte Carlo simulation expected dispersion results can be reviewed. Finally, Chapter 6 provides concluding remarks on the findings and areas where future study is warranted.

Chapter 2

Mission Planning Framework

Incorporating covariance shaping techniques into the mission planning process has the potential to reduce the sensitivity of reference trajectories to various uncertainties that are encountered in any real world operating environment that a hypersonic reentry vehicle might face. In order to implement these techniques, a general formulation for reference trajectory design is constructed, which will then be applied to the reentry problem. The benefit of having a general formulation is that it can also be applied to various other problems that require the development of nominal reference trajectories.

In this chapter, the reference trajectory design process is developed, which is based on trajectory optimization using optimal control techniques. The output of this process is a reference trajectory that defines the nominal path and control commands used to follow that path. Next, a guidance algorithm is designed using an LQR scheme that can be used to minimize deviations from the reference trajectory. Finally, an overview of linear covariance analysis is provided, including the computation of linear covariance along with methods for inclusion of different types of errors into the covariance model. These include initial state errors at the beginning of the trajectory, process noise that generates errors along the trajectory, and uncertainties in system parameters. The linear covariance is then included into the reference trajectory design using covariance shaping techniques to design trajectories with reduced sensitivity to uncertainty.

2.1 Reference Trajectory Design

The reference trajectory design process seeks to develop a trajectory that meets all design requirements and is optimized for a desired performance metric. The trajectory optimization problem can be formulated as an optimal control problem, which minimizes a cost function subject to a set of constraints. Consider a system of interest that is described by a state vector

$$\mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} \tag{2.1}$$

and a control vector

$$\mathbf{u} = \begin{bmatrix} u_1 \\ \vdots \\ u_m \end{bmatrix} . \tag{2.2}$$

The optimal control problem is to determine the time history of the controls, $\mathbf{u}(t)$, such that the cost function

$$J = E(\mathbf{x}(t_0), \mathbf{x}(t_f), t_0, t_f) + \int_{t_0}^{t_f} F(\mathbf{x}(t), \mathbf{u}(t), t)dt \tag{2.3}$$

is minimized subject to the dynamic constraint defined by plant dynamics

$$\dot{\mathbf{x}} = \mathbf{g}(\mathbf{x}(t), \mathbf{u}(t), t) . \tag{2.4}$$

The set of controls and associated states that optimizes the cost function is the nominal reference trajectory. The cost function consists of an endpoint cost, E, which is a function of the initial and terminal states as well as the initial and terminal times. It also includes an integral cost, F, which is a function of the states and controls integrated along the trajectory. The cost function defines the performance metric of interest that the optimal control problem seeks to minimize.

The trajectory optimization problem can also be subject to additional constraints in the form of event and path constraints as well as state and control bounds. Event constraints or endpoint constraints

$$\mathbf{e}^L \leq \mathbf{e}(\mathbf{x}(t_0), \mathbf{x}(t_f), t_0, t_f) \leq \mathbf{e}^U \tag{2.5}$$

require that a function of the endpoints of the trajectory remain between the lower and upper event bounds, $\mathbf{e}^L$ and $\mathbf{e}^U$. This can be used to constrain boundary conditions on the states at the terminal points of the trajectory. Path constraints

$$\mathbf{h}^L \leq \mathbf{h}(\mathbf{x}(t), \mathbf{u}(t), t) \leq \mathbf{h}^U \tag{2.6}$$

serve a similar purpose as event constraints, but require that any function of the state and control vectors along the trajectory remain between the lower and upper path bounds, $\mathbf{h}^L$ and $\mathbf{h}^U$. A specific type of path constraint that is commonly used is bounds directly on the values of the states and controls. State bounds

$$\mathbf{x}^L \leq \mathbf{x}(t) \leq \mathbf{x}^U \tag{2.7}$$

are constraints on the state variables themselves that require the states remain between the lower and upper state bounds, $\mathbf{x}^L$ and $\mathbf{x}^U$, at all times. Control bounds

$$\mathbf{u}^L \leq \mathbf{u}(t) \leq \mathbf{u}^U \tag{2.8}$$

constrain the control variables to remain between the lower and upper control bounds, $\mathbf{u}^L$ and $\mathbf{u}^U$, at all times. Time can also be constrained in a trajectory optimization problem; often times the initial time, t_0, will be fixed, but the final time, t_f, will be allowed to vary.

Incorporating all of these constraints, the optimal control problem can be formally defined as

$$\min\ J = E(\mathbf{x}(t_0), \mathbf{x}(t_f), t_0, t_f) + \int_{t_0}^{t_f} F(\mathbf{x}(t), \mathbf{u}(t), t)dt$$

subject to the dynamic constraint

$$\dot{\mathbf{x}} = \mathbf{g}(\mathbf{x}(t), \mathbf{u}(t), t)$$

as well as event and path constraints

$$\mathbf{e}^L \leq \mathbf{e}(\mathbf{x}(t_0), \mathbf{x}(t_f), t_0, t_f) \leq \mathbf{e}^U$$

$$\mathbf{h}^L \leq \mathbf{h}(\mathbf{x}(t), \mathbf{u}(t), t) \leq \mathbf{h}^U$$

$$\mathbf{x}^L \leq \mathbf{x}(t) \leq \mathbf{x}^U$$

$$\mathbf{u}^L \leq \mathbf{u}(t) \leq \mathbf{u}^U .$$

The solution to this problem are time histories of the optimal controls, $\mathbf{u}^*$, and states, $\mathbf{x}^*$, that minimize the cost, J.

Attempting to solve this optimal control problem analytically requires the introduction of costates and the derivation of the necessary conditions and boundary conditions. However, the complexity of many systems makes an analytical solution impossible, as is the case here. Instead, a variety of numerical methods can be employed to solve the problem, which can generally be classified as indirect or direct methods [17]. In indirect methods, the necessary and boundary conditions are used to define a boundary value problem, for which the solution is well understood. Unfortunately, solving problems using indirect methods has many drawbacks. As mentioned previously, indirect methods require the derivation of the necessary conditions, which requires costate expressions for each dynamics constraint as well as Lagrange multipliers for each event and path constraint [9]. These conditions must be analytically re-derived any time the optimal control problem changes, such as an adjustment to a constraint or the cost function. Additionally, in indirect methods, boundary value solvers require an initial guess, including a guess of the costates, which are not necessarily intuitive. Failure to generate a sufficiently accurate initial guess can prevent convergence to a solution, even if a solution exists. Finally, as the complexity of the

system increases in terms of the number of states, the number of controls, constraints, and dynamic complexity, the problem becomes computationaly more intensive, making solving problems of this nature less feasible [10].

In direct methods, instead of solving a boundary value problem based on the necessary conditions for optimality, algorithms seek to minimize the cost function directly by converting the optimal control problem into a nonlinear programming (NLP) problem. The particular algorithm that is used in this study is the Legendre pseudospectral (PS) method, implemented by the DIDO software [18]. This method works by evaluating the dynamics of the optimal control problem at discrete nodes and approximating the states and controls with Legendre polynomials. The problem then becomes a NLP, which can be solved by NLP software such as SNOPT. Using a software package such as DIDO provides several benefits: first, because PS methods do not require the derivation of necessary conditions, large changes, such as new cost functions, can be made to the problem formulation with very little effort. Also, PS methods generally have a much larger convergence radii than indirect methods, requiring a less accurate initial guess. Finally, it is also not necessary to provide an initial guess for costate values, a requirement for indirect methods. For these reasons, DIDO is chosen as the software tool to solve the trajectory optimization problems in this study.

2.2 Guidance Algorithm Design

Once a reference trajectory has been generated, a guidance algorithm must be formulated that can track the nominal reference trajectory in a realistic environment, which may include environmental or modeling uncertainties, disturbances, and unmodeled dynamics [11]. While there are numerous guidance algorithms available, optimal linear regulator theory provides one that is simple yet still effective. A number of studies [2][3][19] demonstrate the performance capabilities of a linear quadratic regulator (LQR) guidance scheme as well as the simplicity required for its implementation. The nominal reference trajectory described in Section 2.1 provides a nominal

state vector, $\mathbf{x}_N$, and an associated nominal control vector, $\mathbf{u}_N$. Using optimal linear regulator theory, a guidance algorithm can be developed for tracking the nominal reference trajectory in a real world environment. A simple representation of such a scheme is shown in Figure 2-1.

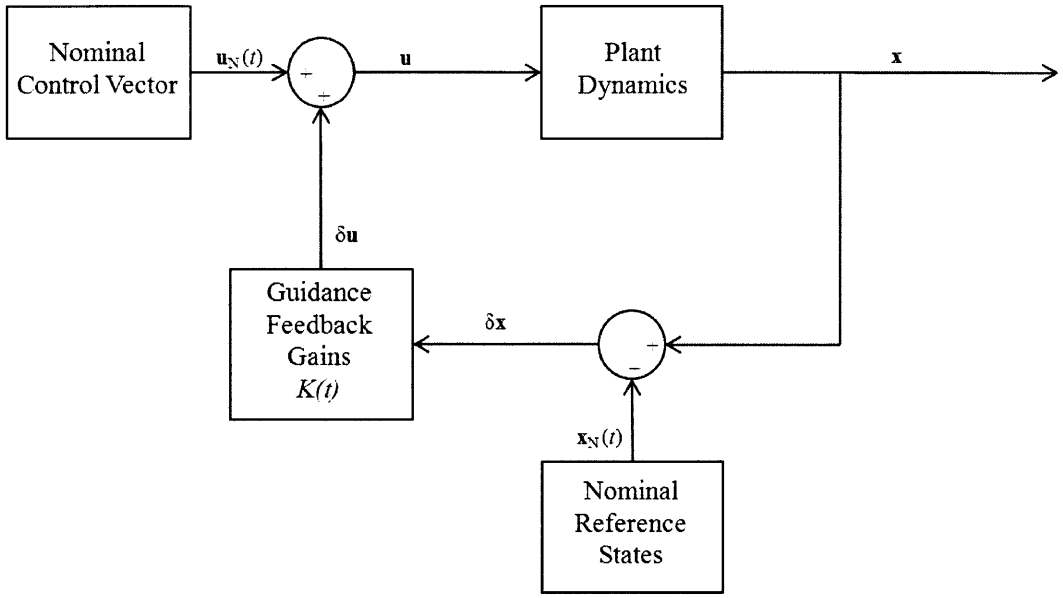

Figure 2-1: LQR Guidance Block Diagram

For the LQR guidance scheme, a cost function that aims to provide good tracking performance while minimizing control effort is

$$J = \frac{1}{2}\delta\mathbf{x}^T(t_f)S_f\delta\mathbf{x}(t_f) + \frac{1}{2}\int_{t_0}^{t_f}[\delta\mathbf{x}^T(t)Q\delta\mathbf{x}(t) + \delta\mathbf{u}^T(t)R\delta\mathbf{u}(t)]dt\,, \tag{2.9}$$

where

$$\delta\mathbf{x} = \mathbf{x} - \mathbf{x}_N \tag{2.10}$$

$$\mathbf{u} = \mathbf{u}_N + \delta\mathbf{u}\,. \tag{2.11}$$

In this notation, $\mathbf{x}$ and $\mathbf{u}$ are the vectors which represent the true state vector and the actual commanded control vector, $\delta\mathbf{x}$ is the state deviation vector and $\delta\mathbf{u}$ is the

correction to the nominal control vector. The weighting matrices, Q, R, and S_f, must be specified by the designer in order to balance the desired accuracy of tracking versus the amount of control effort used. The weighting matrix, R, penalizes control effort and must be symmetric and positive definite ($R > 0$). The weighting matrix, Q, penalizes state deviations and must be symmetric and positive semidefinite ($Q \geq 0$). Finally, S_f is the weighting matrix that penalizes state deviations at the final time and must also be symmetric and positive semidefinite ($S_f \geq 0$).

The state deviations are governed by the perturbation dynamics

$$\delta\dot{\mathbf{x}} = A\delta\mathbf{x} + B\delta\mathbf{u}\,, \tag{2.12}$$

which is the linearization of the original nonlinear equations of motion given in Eq. (2.4), where A and B are defined as

$$A = \frac{\partial \mathbf{g}(\mathbf{x}, \mathbf{u}, t)}{\partial \mathbf{x}} \tag{2.13}$$

and

$$B = \frac{\partial \mathbf{g}(\mathbf{x}, \mathbf{u}, t)}{\partial \mathbf{u}}\,. \tag{2.14}$$

The cost function in Eq. (2.9) is minimized by a feedback law of the form

$$\delta\mathbf{u} = -K\delta\mathbf{x}\,. \tag{2.15}$$

In order to compute the gain matrix, K, a solution to the differential Riccati equation

$$-\dot{S}(t) = A^T S(t) + P(t)A - S(t)BR^{-1}B^T S(t) + Q \tag{2.16}$$

is required, which is accomplished by propagating $S(t)$ backwards in time from the terminal condition $S(t_f) = S_f$. The gain matrix, K, can then be computed by the equation

$$K = R^{-1}B^T S(t)\,, \tag{2.17}$$

providing the gain matrix for the feedback law that can be used in a guidance algorithm.

The formulation for the feedback law as described is for finite-horizon problems and requires solving the differential Riccati given in Eq. (2.16), which can be difficult. However, in many trajectory optimization problems, $S(t)$ settles down to a steady state value, S_{SS}. If $S(t) \rightarrow S_{SS}$ before the final time and it converges significantly faster than the linearized dynamics given by Eqs. (2.13) and (2.14) change, a quasi-steady state assumption can be made. In this case, the LQR cost function can then be written as a steady-state regulator performance cost

$$J = \frac{1}{2}\int_{t_0}^{\infty}[\delta\mathbf{x}^T(t)Q\delta\mathbf{x}(t) + \delta\mathbf{u}^T(t)R\delta\mathbf{u}(t)]dt\,. \tag{2.18}$$

This cost function is still minimized by the feedback law given in Eq. (2.15) and K is still defined by Eq. (2.17). However, S is now obtained from the easier to solve algebraic Riccati equation

$$0 = A^TS + SA - SBR^{-1}B^TS + Q\,. \tag{2.19}$$

In order to reduce the number of design parameters in a problem, the weighting matrices, Q and R, are generally chosen to be diagonal [19]. The individual diagonal elements of Q correspond to the various state variables of $\mathbf{x}$. The greater the value of a Q matrix element, the larger the penalty on deviations in that particular state variable. Similarly, the diagonal elements of R correspond to the control variables of $\mathbf{u}$. The greater the value of an R matrix element, the larger the penalty is for use of that control variable.

A simple method for choosing the various elements of the Q and R matrices is known as Bryson's rule [20]. According to this method, the diagonal elements of the Q and R matrices should be the inverse of the square of the maximum desired deviations for each state and control variable while all off-diagonal elements are zero

so that

$$Q = \begin{bmatrix} \frac{1}{\delta x_{1_{\max}}^2} & 0 & 0 & 0 \\ 0 & \frac{1}{\delta x_{2_{\max}}^2} & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & \frac{1}{\delta x_{n_{\max}}^2} \end{bmatrix} \tag{2.20}$$

and

$$R = \begin{bmatrix} \frac{1}{\delta u_{1_{\max}}^2} & 0 & 0 & 0 \\ 0 & \frac{1}{\delta u_{2_{\max}}^2} & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & \frac{1}{\delta u_{m_{\max}}^2} \end{bmatrix}. \tag{2.21}$$

Bryson's rule may not provide exactly the results desired as the weighting matrices do not enforce any strict requirements on the outcome. However, it generally does provide good tracking performance and can serve as an initial baseline for evaluating the guidance design. With the feedback law formulated, the performance of the guidance scheme implemented along the nominal reference trajectory can be evaluated. The guidance scheme can be tested under nominal conditions and with disturbances and errors added to test its performance under a variety of conditions. However, as mentioned previously, the guidance scheme and the nominal reference trajectory were designed independently. Section 2.3 will discuss a method to incorporate the structure of the guidance scheme into the reference trajectory design process. This changes how reference trajectories are planned, but also it improves performance, particularly when uncertainties and disturbances are considered.

2.3 Linear Covariance Analysis

A properly designed nominal reference trajectory will ensure that nominal design requirements are met while minimizing the desired performance metric. The guidance scheme designed in Section 2.2 provides good tracking performance in off-nominal conditions, but the traditional reference trajectory design process does not make any effort to adjust the design to reduce the sensitivity of the nominal reference trajectory

and the guidance law to uncertainties. One method for quantifying the sensitivity to uncertainty of a particular trajectory is through simulation. Given a variety of initial state errors, modeling errors, and environmental uncertainties, a Monte Carlo analysis can demonstrate state dispersions from the nominal trajectory. However, Monte Carlo simulations are expensive in terms of time and computing power, which may be limited. It is possible, however, to produce similar results as the Monte Carlo simulation analytically by using linear covariance techniques [13]. Given initial uncertainties and noise models, the linear covariance of a system can be calculated and propagated along the trajectory to show expected errors. Assuming that the effect of nonlinearities in the system dynamics is small, linear covariance analysis can provide an approximation of a Monte Carlo simulation in only one trajectory run versus a large number of runs in the Monte Carlo simulation. It is through this framework that the guidance algorithm can be incorporated into the reference trajectory design process, and the sensitivity of the reference trajectory to uncertainties can be reduced. This analysis will first explore the open-loop covariance of the nominal reference trajectory, but will then be expanded to include the guidance law in order to compute the closed-loop covariance of the system.

2.3.1 Linear Covariance Dynamics

The nominal dynamics of a system are governed by Eq. (2.4), but when uncertainties are considered to act on the system, the stochastic system dynamics can be modeled as

$$\dot{\mathbf{x}} = \mathbf{g}(\mathbf{x}(t), \mathbf{u}(t), t) + \mathbf{w}\,, \tag{2.22}$$

where $\mathbf{w}$ is vector of zero mean, Gaussian white noise with covariance given by

$$E[\mathbf{w}(t)\mathbf{w}^T(\tau)] = R_{ww}(t)\delta(t-\tau)\,. \tag{2.23}$$

The power spectral density of the white noise is given by $R_{ww}(t)$.

In a general stochastic system, the covariance of the state variables can be de-

scribed by

$$P = E[(\mathbf{x} - E[\mathbf{x}])(\mathbf{x} - E[\mathbf{x}])^T] \,, \tag{2.24}$$

where $\mathbf{x}$ now describes the stochastic system. However, linear covariance analysis is interested in the covariance of the linearized system. If the system operates near the reference state about which the system is linearized, the expected value of the stochastic dynamics is approximately equal to the nominal dynamics

$$E(\mathbf{x}) = \mathbf{x}_N \,. \tag{2.25}$$

The state deviations, $\delta\mathbf{x}$, defined by Eq. (2.10), can be used to allow the covariance of the stochastic system to be written as

$$P = E[(\mathbf{x} - \mathbf{x}_N)(\mathbf{x} - \mathbf{x}_N)^T] = E(\delta\mathbf{x}\delta\mathbf{x}^T) \,. \tag{2.26}$$

Using these assumptions, the covariance matrix, P, for a system with n states is written as

$$P = \begin{bmatrix} E(\delta x_1 \delta x_1) & E(\delta x_1 \delta x_2) & \cdots & E(\delta x_1 \delta x_n) \\ E(\delta x_2 \delta x_1) & E(\delta x_2 \delta x_2) & \cdots & E(\delta x_2 \delta x_n) \\ \vdots & \vdots & \ddots & \vdots \\ E(\delta x_n \delta x_1) & E(\delta x_n \delta x_2) & \cdots & E(\delta x_n \delta x_n) \end{bmatrix} . \tag{2.27}$$

The diagonal elements of P are the variances of each state variable, which quantify the deviations of the states from their nominal values, while the off-diagonal elements of P represent the correlations between each pair of state variables.

Given a continuous nominal reference trajectory, the open-loop dynamics of the linear state covariance are given by the Lyaponov equation

$$\dot{P} = AP + PA^T + R_{ww} \,, \tag{2.28}$$

where

$$P(t_0) = E[\delta\mathbf{x}(t_0)\delta\mathbf{x}(t_0)^T] \tag{2.29}$$

is the initial covariance of the states. For the open-loop covariance of the nominal reference trajectory when no feedback law is considered, A is simply the linearization of the nominal dynamics with respect to the state vector computed along the nominal trajectory given by Eq. (2.13). It is now possible, given a nominal reference trajectory, an initial uncertainty estimate, and a process noise model, to determine the expected state errors at any point by propagating Eq. (2.28) along the trajectory.

The guidance law designed in Section 2.2 can also be included into the calculation of the covariance. This is accomplished by linearizing the nominal dynamics with respect to both the states as well as the controls. From Eq. (2.15), $\delta\mathbf{u} = -K\delta\mathbf{x}$, so the augmented closed-loop linearization can be written as

$$G = A - BK = \frac{\partial \mathbf{g}}{\partial \mathbf{x}} - \frac{\partial \mathbf{g}}{\partial \mathbf{u}} K \,. \tag{2.30}$$

Using this linearization, Eq. (2.28) can be rewritten to model the dynamics of the closed-loop covariance as

$$\dot{P} = GP + GP^T + R_{ww} \,. \tag{2.31}$$

The closed-loop covariance can then be calculated to determine the expected state errors in the presence of the guidance law designed previously, allowing for performance of the guidance law and the reference trajectory to be evaluated simultaneously.

2.3.2 Covariance Shaped Trajectory Design

This process of determining the covariance of the system is useful for gaining an understanding of how well a trajectory can be tracked in off-nominal conditions. However, this evaluation happens only after the trajectory has been designed. Small [9] and Zimmer et al. [16] both demonstrate methods of utilizing the covariance calculations within the trajectory design process. The covariance of the trajectory can be calculated and directly penalized during the trajectory optimization by including it as a performance metric within the optimization cost function given in Eq. (2.3). Small [9] showed that the nominal state vector can be augmented to include the

elements of the covariance matrix, while the nominal equations of motion given in Eq. (2.4) are augmented with the covariance dynamics, governed by Eqs. (2.28) or (2.31). The covariance dynamics then serve as the differential constraints for the new covariance states during the optimization. Since the covariance matrix is symmetric, only the elements on or above the diagonal of the covariance matrix must be added as additional states. However, for a system with n states, this requires that the nominal state vector be augmented with $\frac{n(n+1)}{2}$ additional covariance states. Although pseudospectral methods can easily solve nominal trajectory optimization problems, the addition of this many states and associated differential constraints often renders trajectory optimization problems formulated in this way unsolvable.

However, there are other means of obtaining the covariance during the optimization rather than augmenting the nominal state vector with the covariance states. Within the optimization, the nominal dynamics do not depend on the covariance. However, the covariance does depend on the nominal state and control values at each point in time, which are accessable during the optimization. Using the nominal states and controls, the covariance can be calculated and propagated along the trajectory using Eq. (2.28) or Eq. (2.31). This can be accomplished by interpolating the nominal state and control values to obtain a continuous time history of each, then propagating the covariance dynamics using an integrator. This provides the full covariance matrix at any point along the trajectory and can be used within constraints or the cost function just like any other function of the nominal states and controls during trajectory optimization. Using an integrator to propagate the covariance increases the complexity and computational requirements any time that the covariance is required in the optimization, but it does not require the augmentation of the state vector, which proves beneficial in obtaining optimal solutions.

How the covariance is handled within the optimization cost function can also be adjusted according to the requirements of the designer. The covariance could be included in the integral cost, F, of the cost function, so that the covariance along the entire trajectory is reduced. However, a small uncertainty at the terminal condition of the trajectory is generally much more desirable than smaller uncertainties along

the trajectory. In this case, only the terminal covariance could be included into the endpoint cost, E, so that the final covariance is reduced as much as possible. The uncertainty of various states can also be penalized individually, so that if it is desirable that some states have much smaller covariances than others, a weighting penalty can be associated with those states within the cost function.

Regardless of the method of penalizing the covariance within the optimization, the end result is a trajectory that has been shaped in order to reduce sensitivity to uncertainty. As with any optimization, there is a tradeoff that including uncertainty in the cost function will reduce the amount to which any other desired performance metrics can be optimized. For a given performance metric, this means that a covariance shaped reference trajectory may perform worse than a nominal reference trajectory under nominal test conditions. However, a decrease in performance may be acceptable if a significant decrease in the sensitivity to uncertainty of the trajectory can also be obtained, especially in off-nominal conditions that more closely resemble a real world environment.

2.3.3 Parametric Uncertainties

The covariance formulation described in this section is useful because it provides a way to mathematically describe uncertainties in state variables, either due to initial uncertainties or process noise along the trajectory. Unfortunately, many other types of uncertainties exist beyond those directly involving the states, and it would be useful to understand and model how these uncertainties affect state deviations along the trajectory. These uncertainties often affect system parameters, which in atmospheric reentry problems can include parameters such as mass or aerodynamics properties, neither of which are state variables. Uncertainties in environmental parameters, such as density, are also often encountered. As these properties are merely parameters of the system, however, they are not included in the covariance calculations. It is possible to create indirect representations of some of these parametric uncertainties as state uncertainties, especially as process noise, but Seywald and Kumar [11] provide a more direct method for including parametric uncertainties in the covariance calculations.

Given a parameter of interest, p, with a nominal value of p_0, the state vector can be augmented such that

$$\mathbf{x}_a = \begin{bmatrix} \mathbf{x} \\ p \end{bmatrix} \tag{2.32}$$

so that the parameter, p, is now a state. In the nominal system, this parameter remains constant, so its dynamics are represented by

$$\dot{p} = 0\,, \tag{2.33}$$

and the nominal system dynamics become

$$\dot{\mathbf{x}}_a = \mathbf{g}_a(\mathbf{x}, \mathbf{u}, p, t) = \begin{bmatrix} \mathbf{g}(\mathbf{x}, \mathbf{u}, t) \\ \dot{p} \end{bmatrix}. \tag{2.34}$$

Because the parameter value does not change in the nominal system, it does not affect any of the nominal dynamics and it is therefore unnecessary to use the augmented state directly during the trajectory optimization. Instead, the augmented system only needs to be considered while calculating the covariance. The linearization of the system then becomes

$$G = A_a - B_a K = \frac{\partial \mathbf{g}_a}{\partial \mathbf{x}_a} - \frac{\partial \mathbf{g}_a}{\partial \mathbf{u}} K\,. \tag{2.35}$$

However, some parameters are not constant along the trajectory, which means that the parameter augmentation of the state vector with trivial dynamics does not work directly, as the dynamics are no longer trivial. Instead, the uncertainty of the parameter can be written as

$$p_{\text{act}} = (1 + C_p \sigma_p(\mathbf{x}))p\,, \tag{2.36}$$

where p_{act} is the actual parameter after random processes are accounted for, C_P is a constant parameter, σ_p is the standard deviation of the uncertainty of the parameter, and p is the nominal parameter value. Instead of directly including p in the state vector, the constant C_p is now included in the stochastic system. This allows p to vary

along the trajectory while the uncertainty of p is defined by its standard deviation, which can be constant or a function of the state vector. The stochastic variable, C_p, is then used to capture the uncertainty of the parameter by assuming that C_p is distributed normally with mean

$$E[C_p] = 0 \tag{2.37}$$

and variance

$$E[C_p^2] = 1\,. \tag{2.38}$$

By adding C_p to the state vector as a new state with trivial dynamics, the original uncertainty of the parameter is modeled within the covariance dynamics, where it can affect other state errors.

This method now provides a formulation that allows for uncertainties in state variables as well as model parameters to be considered in determining expected state deviations along a planned trajectory. This process takes into account a designed guidance law so that expected deviations for closed-loop simulation in off-nominal conditions can be computed. Finally, this method is incorporated into the trajectory design process, allowing for covariance shaped reference trajectories to be generated that reduce the sensitivity of the of the designed trajectories to a variety of state and parametric uncertainties.

Chapter 3

Atmospheric Reentry Modeling

In order to implement and test the mission planning techniques introduced in Chapter 2, a modeling framework for the atmospheric reentry problem is needed to describe the system of interest. This includes a vehicle model, an environmental model, the vehicle equations of motion, and the guidance algorithm used onboard the vehicle. To simplify the dynamics, a three degree-of-freedom (3 DOF) model is used, where the state of the vehicle is defined only by its position and velocity. This allows the vehicle to be effectively modeled as a point-mass, where the orientation and rotation dynamics of the vehicle are not governed by 3 DOF equations of motion. However, the vehicle's orientation is used as the means of controlling the vehicle.

3.1 Coordinate Frames

In the 3 DOF dynamics, the state vector of the vehicle is made up of three position states and three velocity states. These states are presented in spherical coordinates, which is more intuitive for atmospheric reentry where the vehicle can move large distances around the Earth. Control of the vehicle is accomplished by commanding the orientation of the vehicle relative to the velocity vector. This is modeled by the control vector, which contains three control variables that define the various components of the vehicle orientation. The state and control variables can be understood through the use of various reference frames. The coordinate frames introduced are adapted

from [4].

3.1.1 Earth Centered Inertial Frame

The Earth Centered Inertial (ECI) frame provides the inertial reference frame in which the motion of a body over the Earth is modeled. The frame is located at the center of the Earth with the z_I axis pointing along the Earth's rotation axis toward the Geographic North Pole. The x_I axis is oriented such that on the Vernal Equinox, it points toward the position of the Sun, while the final axis, $y_I = x_I \times z_I$, completes the right-handed orthogonal coordinate system, aligning the $x_I - y_I$ plane with the Earth's equator. The ECI frame translates through space as the Earth moves around the Sun, but the axes remains fixed in relation to Earth's rotation. It is assumed that the effects of rotating around the Sun are negligible.

3.1.2 Earth Centered Earth Fixed Frame

The Earth Centered Earth Fixed (ECEF) frame is located at the center of the Earth with the z_E axis pointing along the Earth's rotation axis toward the Geographic North Pole. The ECEF frame rotates along with the Earth, meaning that the x_E axis always points toward the same location on the Earth's surface. This point is defined as 0 deg latitude, 0 deg longitude. The remaining axis, $y_E = x_E \times z_E$, completes the right-handed orthogonal coordinate system. By definition the $x_E - y_E$ plane is aligned with the Earth's equator.

Transformations between any two coordinate frames can be accomplished by a series of rotations about relevant axes defined by transformation matrices. As the ECEF frame rotates with the Earth, its orientation relative to the ECI frame is dependent on the time of interest, t, and the angular rotation rate of the Earth, Ω_E. The z-axes of the two frames are aligned at all times and the x-axes are aligned once during each 24-hour period. Assuming that the frames were aligned at $t_0 = 0$, the transformation is accomplished by a positive rotation of the ECI frame about the z_I

axis by $\Omega_E t$. The transformation matrix defining this rotation from ECI to ECEF is

$$T_I^E = \begin{bmatrix} \cos(\Omega_E t) & \sin(\Omega_E t) & 0 \\ -\sin(\Omega_E t) & \cos(\Omega_E t) & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

3.1.3 Up East North Frame

The position vector of the vehicle can be used to define a local horizontal reference frame. This frame is a vehicle carried frame with the origin located at the vehicle center of mass. For simplicity, a Up-East-North (UEN) reference frame is used to fix the orientation of the axes. The x_{UEN} axis points Up, the y_{UEN} axis points East, and the z_{UEN} axis points North, establishing the $y_{\text{UEN}} - z_{\text{UEN}}$ plane as the local horizontal. The location of the origin of the UEN frame in relation to the ECEF frame is defined by the vehicle position vector. In spherical coordinates, this is represented by the radial distance, r, the longitude, μ, and the geocentric latitude, λ. The radial distance, r, is the distance between the origin of the ECEF frame, which is the center of the Earth, and the vehicle center of mass. The longitude, μ, is the angular position of the vehicle in the $x_E - y_E$ plane, measured counter-clockwise from the x_E axis. The geocentric latitude is the angular position of the vehicle measured from the $x_E - y_E$ plane. The transformation from the ECEF frame and the UEN frame is described by a sequence of two rotations:

1. Positive rotation about the z_E axis by the longitude, μ
2. Negative rotation about the y_{UEN} axis by the geocentric latitude, λ

The transformation matrix for these rotations is defined by

$$T_E^{\text{UEN}} = \begin{bmatrix} \cos\lambda & 0 & \sin\lambda \\ 0 & 1 & 0 \\ -\sin\lambda & 0 & \cos\lambda \end{bmatrix} \begin{bmatrix} \cos\mu & \sin\mu & 0 \\ -\sin\mu & \cos\mu & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} \cos\lambda\cos\mu & \cos\lambda\sin\mu & \sin\lambda \\ -\sin\mu & \cos\mu & 0 \\ -\sin\lambda\cos\mu & -\sin\lambda\sin\mu & \cos\lambda \end{bmatrix}.$$

3.1.4 Velocity Frame

The velocity vector of the vehicle can be used to define the Velocity (V) frame relative to the UEN frame. This frame is a vehicle carried frame with the origin located at the vehicle center of mass and aligned with the vehicle velocity vector in the UEN frame, $\vec{\mathbf{V}}_{\text{UEN}}$. In spherical coordinates, the velocity vector is made up of the speed, v, or magnitude of $\vec{\mathbf{V}}_{\text{UEN}}$, the flight path angle, γ, and the heading angle, ψ. The flight path angle is the inclination of $\vec{\mathbf{V}}_{\text{UEN}}$ from the local horizontal plane, $y_{\text{UEN}} - z_{\text{UEN}}$, measured positively above the horizontal. The heading angle is the orientation of $\vec{\mathbf{V}}_{\text{UEN}}$ in the $y_{\text{UEN}} - z_{\text{UEN}}$ plane, measured positively counter-clockwise from the East. Figure 3-1 shows $\vec{\mathbf{V}}_{\text{UEN}}$ located in the UEN frame.

In the V frame, the x_V axis is aligned with $\vec{\mathbf{V}}_{\text{UEN}}$ and the y_V axis is located in the local horizontal plane. The z_V axis completes the orthogonal coordinate system and is located in the local vertical plane that includes the x_V axis. The transformation

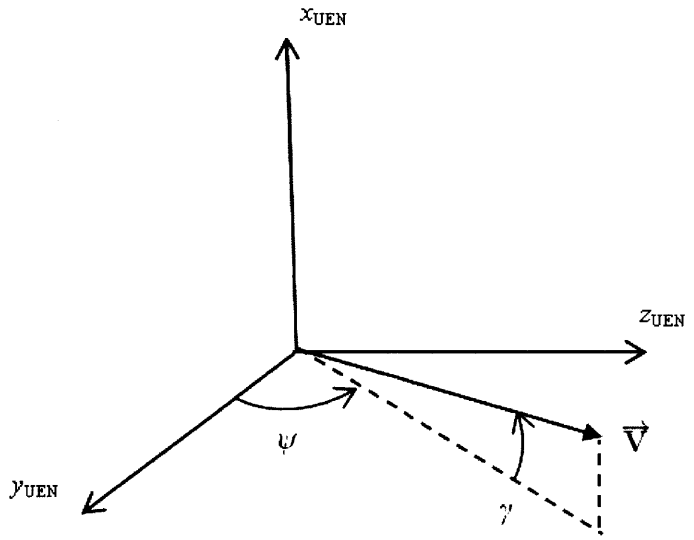

Figure 3-1: Definition of Velocity Vector

from the UEN frame to the V frame requires a sequence of four rotations:

1. Negative rotation about the y_{UEN} axis by $\frac{\pi}{2}$
2. Positive rotation about the temporary z axis by $\frac{\pi}{2}$
3. Negative rotation about the temporary z axis by the heading angle, ψ
4. Positive rotation about the y_V axis by the flight path angle, γ

The transformation matrix for these rotations is defined by

$$
\begin{aligned}
T_U^V &= \begin{bmatrix} \cos\gamma & 0 & -\sin\gamma \\ 0 & 1 & 0 \\ \sin\gamma & 0 & \cos\gamma \end{bmatrix} \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\frac{\pi}{2} & \sin\frac{\pi}{2} & 0 \\ -\sin\frac{\pi}{2} & \cos\frac{\pi}{2} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\frac{\pi}{2} & 0 & \sin\frac{\pi}{2} \\ 0 & 1 & 0 \\ -\sin\frac{\pi}{2} & 0 & \cos\frac{\pi}{2} \end{bmatrix} \\
&= \begin{bmatrix} \sin\gamma & \cos\gamma\cos\psi & \cos\gamma\sin\psi \\ 0 & \sin\psi & -\cos\psi \\ -\cos\gamma & \sin\gamma\cos\psi & \sin\gamma\sin\psi \end{bmatrix}.
\end{aligned}
$$

3.1.5 Wind Frame

The last three reference frames are defined by the orientation of the vehicle. The Wind (W) frame is a vehicle carried frame oriented such that the x_W axis is aligned with $\vec{\mathbf{V}}_{\text{UEN}}$, meaning x_W is also aligned with x_V. The Wind frame is obtained by rotating the Velocity frame negatively about the x_V axis by the vehicle bank angle, σ. Figure 3-2 shows the Wind frame relative to the Velocity frame. The transformation matrix for the rotation from the Velocity frame to the Wind frame is defined by

$$
T_V^W = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\sigma & -\sin\sigma \\ 0 & \sin\sigma & \cos\sigma \end{bmatrix}.
$$

3.1.6 Stability Frame

The Stability (S) frame is a vehicle carried frame that is obtained by rotating the Wind frame negatively about the z_W axis by the sideslip angle, β. This means that

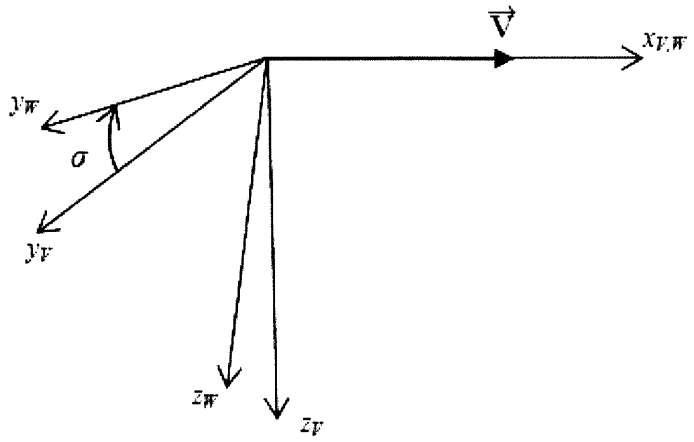

Figure 3-2: Definition of Wind Frame

the z_S and the z_W axes are aligned. The transformation matrix for this rotation is defined by

$$T_W^S = \begin{bmatrix} \cos\beta & -\sin\beta & 0 \\ \sin\beta & \cos\beta & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

3.1.7 Body Fixed Frame

The Body Fixed (B) reference frame is a vehicle carried frame that is obtained by rotating the Stability frame positively about the y_S axis by the angle-of-attack, α. This means that the y_B and y_S axes are aligned. The vehicle orientation is now completely defined such that the Body Fixed frame is oriented along the physical axes of the vehicle. The x_B axis points to the vehicle nose, the z_B axis points to the bottom of the vehicle, and the y_B axis points to the right of the vehicle. The relations between the Wind, Stability, and Body Fixed reference frames are shown in Figure 3-3. The transformation matrix for the rotation from the Stability frame to the Body

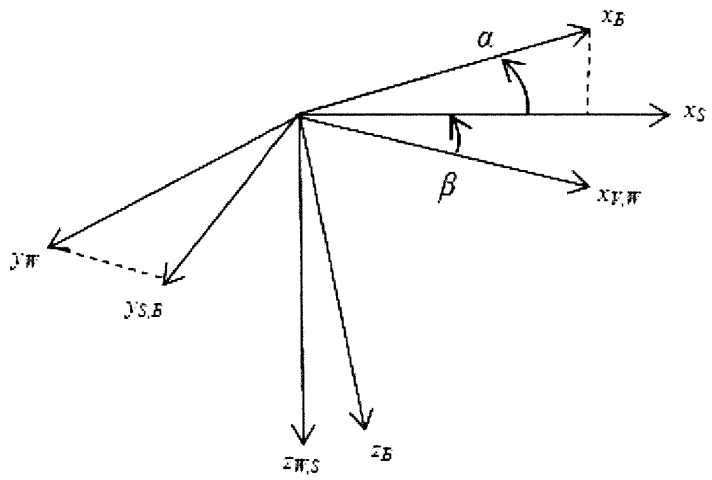

Figure 3-3: Wind, Stability, and Body Fixed Frames

Fixed frame is defined by

$$T_S^B = \begin{bmatrix} \cos\alpha & 0 & -\sin\alpha \\ 0 & 1 & 0 \\ \sin\alpha & 0 & \cos\alpha \end{bmatrix}.$$

3.2 Vehicle Model

The test vehicle under investigation is an unpowered, small reentry body that is manueverable using bank-to-turn steering as well as capable of long range glides. A model is needed to describe the properties and characteristics of this vehicle. These include aerodynamic properties, which define the forces acting on the vehicle during flight, and physical parameters, which impact the dynamics of the vehicle throughout atmospheric reentry.

3.2.1 Aerodynamic Coefficients

During reentry, the vehicle is subject to a variety of effects from interaction with the atmosphere. This interaction induces aerodynamic forces and torques on the body that then drive the dynamics of the vehicle and makes controllable flight possible. Because the 3 DOF dynamics do not govern the rotation of the vehicle, only the forces acting on the body are considered in the vehicle dynamics. These forces are defined according to the reference frame of choice. In the Stability frame, the aerodynamic forces are lift, L, which acts opposite the z_S axis, drag, D, which acts opposite the x_S axis, and the side force, Y, which acts along the y_S axis. These three forces are shown in Figure 3-4. These forces are defined as

$$L = qSC_L\,, \tag{3.1}$$

$$D = qSC_D\,, \tag{3.2}$$

and

$$Y = qSC_Y\,, \tag{3.3}$$

where S is the aerodynamic reference surface area of the vehicle. The dynamic pressure, q, ignoring wind, is given by

$$q = \frac{1}{2}\rho v^2 \tag{3.4}$$

and C_L, C_D, and C_Y are the aerodynamic coefficients for each of the forces. The local atmospheric density is given by ρ.

Aerodynamic forces can also be expressed in the Body Fixed frame. In this frame, the forces are the normal force, N, which acts opposite the z_B axis, the axial force, A, which acts opposite the x_B axis, and the side force, Y, which acts along the y_B axis. The definition of the side force is consistent between the Stability and Body Fixed frames because the y_B and y_S axes are aligned. The body forces are shown in

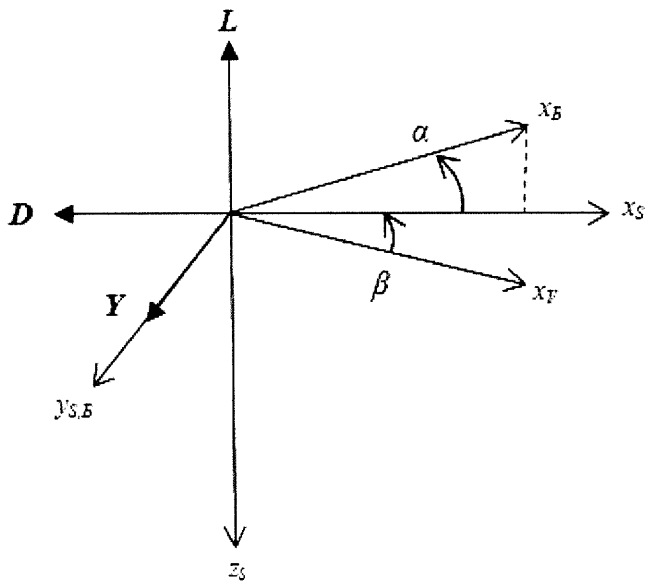

Figure 3-4: Lift and Drag in Stability Frame

Figure 3-5. The normal force and axial forces are defined as

$$N = qSC_N \tag{3.5}$$

and

$$A = qSC_A\,, \tag{3.6}$$

where C_N and C_A are the aerodynamic coefficients for those forces.

Typically, the aerodynamic forces are referred to in the Stability frame when considering vehicle dynamics and in the Body Fixed frame when discussing the aerodynamic properties of the vehicle, which are defined by the aerodynamic coefficients of the normal and axial forces. In a real world environment, these coefficients are a function of a variety of factors including the aerodynamic angles (α, β), their rates $(\dot{\alpha}, \dot{\beta})$, body angular rotation rates, flow characteristics such as Reynolds and Mach number (M, Re), flap deflection (δ), and other dependencies which are related to the geometry of the vehicle such as center of gravity, vehicle shape, size, etc. [1]. The aerodynamic coefficient models used for the test vehicle take into account the major

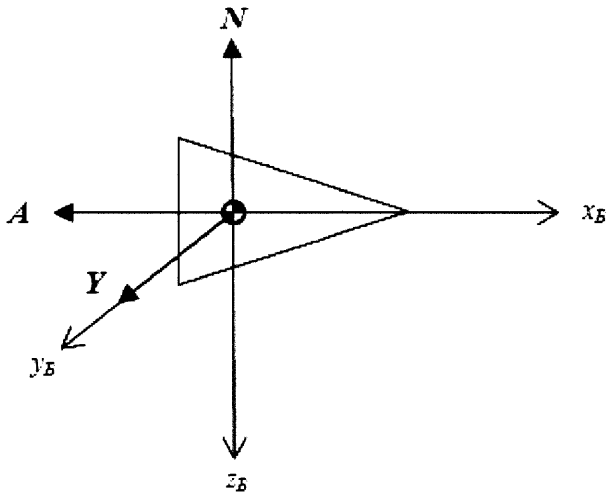

Figure 3-5: Aerodynamic Forces in Body Fixed Frame

dependencies of aerodynamic angles, Mach number, and flap deflection such that

$$C_N, C_A = f(\alpha, \beta, M, \delta) \tag{3.7}$$

and

$$C_Y = f(\beta, M, \delta)\,. \tag{3.8}$$

For vehicles in hypersonic atmospheric reentry, the sideslip angle β is usually approximately zero, meaning the vehicle makes coordinated turns. With the assumption that $\beta = 0$, the side force, Y, is also assumed to be zero. This assumption for atmospheric reentry vehicles will be used from this point on to reduce the complexity of the reentry problem.

As shown in Figure 3-6, the lift and drag forces are geometrically related to the normal and axial forces by the angle-of-attack, α. The aerodynamic coefficients for these forces follow the same relation, which can be expressed by

$$C_L = C_N \cos\alpha - C_A \sin\alpha \tag{3.9}$$

and

$$C_D = C_N \sin\alpha + C_A \cos\alpha \tag{3.10}$$

to allow lift and drag coefficients to be computed from the vehicle aerodynamic properties.

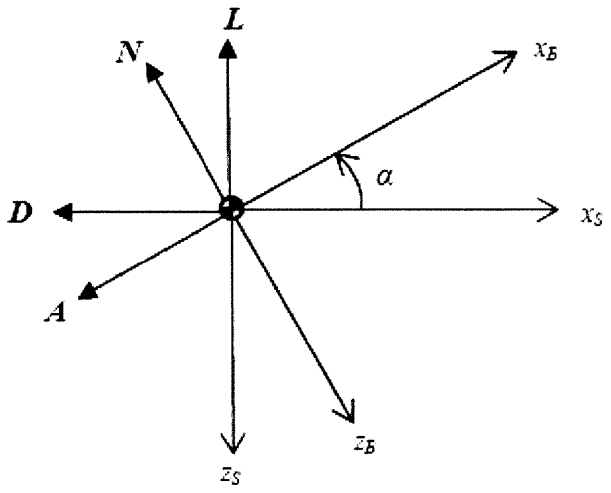

Figure 3-6: Geometric Relationship from Lift and Drag to Body Forces

The axial and normal force coefficients must take into account forces induced by both the body of the vehicle and by flap deflections. They can then be expressed as

$$C_A = C_{A_{\text{body}}} + C_{A_{\text{flap}}} \tag{3.11}$$

and

$$C_N = C_{N_{\text{body}}} + C_{N_{\text{flap}}}\,, \tag{3.12}$$

where $C_{A_{\text{body}}}$ and $C_{N_{\text{body}}}$ are the force coefficients due to body effects and $C_{A_{\text{flap}}}$ and $C_{N_{\text{flap}}}$ are the force coefficients from flap deflections. In determining the body force coefficients, it is possible for empirical testing to be used to gather aerodynamic force data for a variety of flight conditions. The data is then generally presented in a lookup table that uses angle-of-attack, Mach number, or other conditions as pa-

rameters. However, as the vehicle model is a hypothetical test vehicle, there is no empirical data from which to build a table of coefficients. Instead, an analytic approximation for asymmetric, hypersonic reentry vehicles is used [9]. These analytic representations are a function of angle-of-attack and Mach number, and allow for the aerodynamic properties to be adjusted to develop a vehicle with desired characteristics. The approximations for axial and normal force coefficients are

$$C_{A_{\text{body}}} = C_{A_a} e^{-C_{A_b}(M - C_{A_c})} + C_{A_d} + C_{A_k}\alpha^2 \tag{3.13}$$

and

$$C_{N_{\text{body}}} = C_{N0} + C_{N_\alpha}\alpha\,. \tag{3.14}$$

The fitting parameters, $C_{A_a}, C_{A_b}, C_{A_c}, C_{A_d}, C_{A_k}, C_{N0}, C_{N_\alpha}$, are all constants and are chosen to achieve the desired flight characteristics (L/D, trim angle-of-attack) of the vehicle.

The expressions in Eqs. (3.13) and (3.14) allow for the body aerodynamic coefficients to be calculated for a given angle-of-attack and Mach number, but forces due to flap deflection, δ, must also be considered. When the flaps of a vehicle deflect, the purpose is to create a force, which rotates the vehicle to a new orientation. As flap deflection increases, so does the induced force, which changes the net normal force acting on the vehicle. It is assumed that flap deflections to not induce an axial force such that for all time

$$C_{A_{\text{flap}}} = 0\,. \tag{3.15}$$

However, the 3 DOF dynamics only represent the motion of a point-mass, which means that the flaps and induced forces are not directly modeled in the equations of motion. To overcome this limitation, the 3 DOF model is assumed to have a trim angle-of-attack, α_T, where the net moment on the vehicle is zero. In this condition, the body normal force, N_{body}, still acts on the vehicle, but without inducing a moment and the flap normal force, N_{flap} is zero as shown in Figure 3-7. In order to deviate from trim conditions, a nonzero flap normal force, N_{flap}, is required to induce a rotational

moment as shown in Figure 3-8.

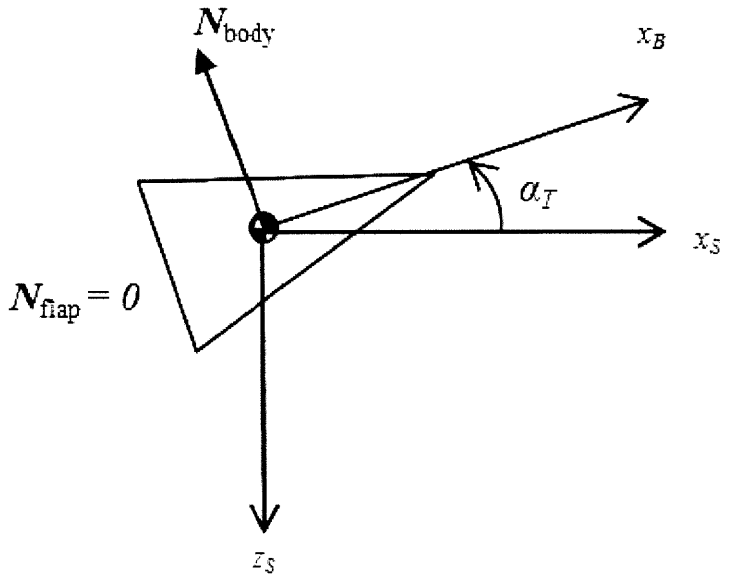

Figure 3-7: Vehicle at Trim Conditions

In order to determine the magnitude of the flap normal force and its contribution to the overall normal force coefficient, consideration must be given to the moments acting on the vehicle. While moments are not considered in the 3 DOF dynamics, they are important for determining the flap contribution to the normal force coefficient. Both the body and the flap induce a moment on the vehicle, which for a vehicle in steady state, the net moment is zero. The moment equations for the body and flap moments are

$$M_{\text{body}} = qSdC_{M_\alpha}(\alpha - \alpha_T) \tag{3.16}$$

and

$$M_{\text{flap}} = qSd(C_{M_\delta}\delta) \tag{3.17}$$

respectively, where q is the dynamic pressure, S is the reference surface area, and d is the diameter of the vehicle body. The nondimensional derivative C_{M_α} describes how the body moment coefficient changes with angle-of-attack and C_{M_δ} is the nondimensional derivative that describes how the flap moment coefficient changes with flap

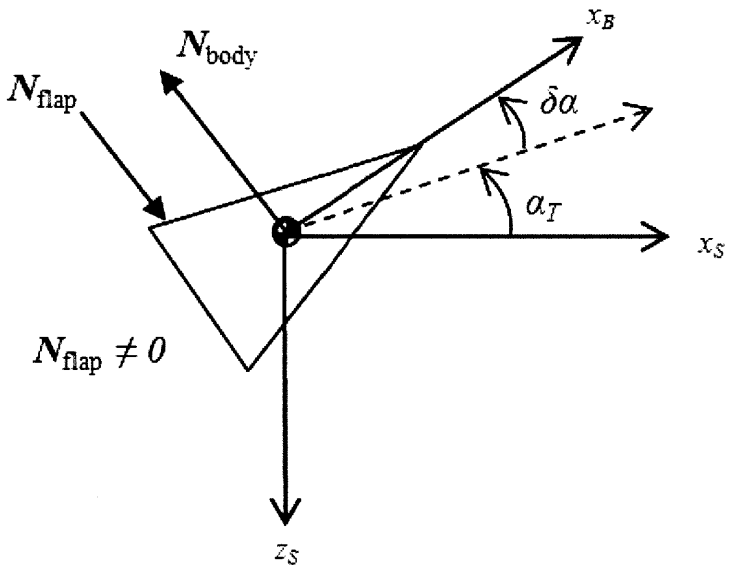

Figure 3-8: Vehicle in Off-Trim Conditions

deflection. Setting the sum of these equations equal to zero and solving for the flap deflection, δ, results in

$$\delta = \frac{-C_{M_\alpha}}{C_{M_\delta}}(\alpha - \alpha_T)\,. \tag{3.18}$$

The deviation from the trim angle-of-attack, α_T, is given by

$$\delta\alpha = \alpha - \alpha_T\,. \tag{3.19}$$

Once the flap deflection has been computed, the normal force coefficient for the flap is found by

$$C_{N_{\text{flap}}} = C_{N_\delta}\delta = -\frac{C_{N_\delta}C_{M_\alpha}}{C_{M_\delta}}(\alpha - \alpha_T)\,, \tag{3.20}$$

where C_{N_δ} is the derivative that describes how the flap normal force coefficient changes with flap deflection. All of the components of Eqs. (3.11) and (3.12) are now defined, allowing the overall normal and axial force coefficients to be calculated.

3.2.2 Vehicle Parameters

Apart from the aerodynamic properties of the vehicle, other physical characteristics must also be defined. In 3 DOF dynamics, where the vehicle is treated as a point-mass, the exact physical layout of the body is not important. The two parameters that are needed in 3 DOF dynamics are the mass, m, and the reference surface area, S. As the test vehicle is meant to be a small, maneuverable vehicle, the mass and reference surface area should also be small.

Additionally, the fact that 3 DOF dynamics do not govern rotation rates allows the vehicle to change orientations instantaneously, which is highly unrealistic. This can become an issue if a nominal reference trajectory calls for attitude changes that are not physically possible in a real world environment. In order to prevent this, rate limits are set on the rates of change for angle-of-attack and bank angle. Also, a limit is put on the minimum and maximum value of the angle-of-attack itself. This prevents the vehicle from preforming behavior that is not modeled by the current aerodynamic coefficient expressions, such as stalls or extreme attitudes during hypersonic flight.

To design a functional 3 DOF vehicle model, the aerodynamic parameters introduced previously as well as the physical parameters from this section must all be chosen to define the flight characteristics of the vehicle. The values chosen for this vehicle are given in Table 3.1. They result in a small, agile vehicle with a maximum lift-to-drag (L/D) ratio of 2.5 at a trim angle-of-attack of 8.083 deg.

3.3 Environmental Model

An environmental model for the reentry vehicle must be developed to consider elements such as the shape of the Earth, its rotation, gravity, and atmospheric effects. The fidelity of the models for these elements can vary widely, from a simplistic flat Earth model with constant gravity and atmospheric density to complex models, which take into account irregularities in the Earth's shape, variations in density, and other factors. There are a variety of tradeoffs that must be considered when determining how accurate an environmental model should be. In atmospheric reentry, vehicle

Table 3.1: Reentry Vehicle Properties and Parameters

Parameter	**Value**	**Units**
Weight, m	250	lbs
Reference Surface Area, S	1.310	ft^2
C_{A_a}	0.317	-
C_{A_b}	0.550	-
C_{A_c}	1.00	-
C_{A_d}	0.083	-
C_{A_k}	1.00	-
C_{N0}	0.0200	-
C_{N_α}	3.00	-
C_{N_δ}	-0.1670	-
C_{M_α}	-0.8709	-
C_{M_δ}	0.1745	-
$\alpha_{\min}$	0	deg
$\alpha_{\max}$	15	deg
$\dot{\alpha}_{\max}$	± 10	deg/sec
$\dot{\sigma}_{\max}$	± 30	deg/sec

flight profiles cover a large range of altitudes as well as substantial distances over the Earth's surface. This requires that atmospheric density variations be taken into account as well as effects due to the Earth's rotation. However, increased accuracy in modeling also leads to increased complexity that can cause significant increases in computational costs in cases where numerical optimization methods are used. The costs of increasing accuracy must be weighed against the necessity for that accuracy. The modeling techniques and associated assumptions used will now be discussed.

3.3.1 Earth Shape and Gravity Model

The shape of the Earth must be defined because in order to conduct proper mission planning, particularly in the terminal phase of flight, how the vehicle moves in relation to the surface of the Earth must be known. In simplest terms, the Earth can be modeled as a sphere with a constant radius defined as the mean equatorial radius, R_e. In this model, the altitude above the surface of the spherical Earth, a, is given by

$$a = r - R_e \,, \tag{3.21}$$

where at any point around the spherical Earth, the altitude is the same for a given radial distance, r. The spherical model also provides a simple means for approximating the gravity field around Earth. Using Newton's law of universal gravitation, the attractive force acting between two objects separated by a distance can be expressed by

$$F = G\frac{Mm}{r^2} \,, \tag{3.22}$$

where G is the universal gravitation constant, M is the mass of the Earth, m is the mass of the vehicle, and r is the radial distance between them. The acceleration due to gravity can then be solved by dividing by the mass of the vehicle and combining the universal gravitation constant and the mass of the Earth to get the Earth's gravitational constant, G_m. This results in the inverse gravity law

$$g = \frac{GM}{r^2} = \frac{G_m}{r^2} \,. \tag{3.23}$$

In order to increase the fidelity of the model, a more accurate model of the Earth based on an ellipsoid approximation can be used. This is an appropriate approximation because in rotating bodies such as the Earth, the rotation causes a bulge to form around the equator. In this model, the radius of the Earth at the equator is greater than the radius at the poles, which means that away from the equatorial plane, the altitude above the spherical Earth, a, is not equal to the altitude above the ellipsoidal Earth. Instead, the altitude above the surface of the ellipsoidal Earth, or geodetic altitude, z, must be calculated, which can be approximated using the radial distance, r, and the geocentric latitude, λ, by

$$z = r + R_e \left(\frac{1}{2} f(1 - \cos(2\lambda)) + f^2 \left(\frac{R_e}{4r} - \frac{1}{16} \right) (1 - \cos(4\lambda)) - 1 \right) , \tag{3.24}$$

where f is the flattening coefficient of the Earth, based on the semi-major and semi-minor axes of the ellipsoidal model. There is also a difference between the spherical model and the ellipsoidal model in describing latitude. The geocentric latitude is the angular position measured about the center of the Earth, while the geodetic latitude is the angle measured between the normal of the ellipsoid and the equatorial plane. The geodetic latitude can be approximated using the radial distance and the geocentric latitude by

$$\lambda_{\text{geodetic}} = \lambda + f \frac{R_e}{r} \sin(2\lambda) + f^2 \frac{R_e}{r} \sin(4\lambda) \left(\frac{R_e}{r} - \frac{1}{4} \right) . \tag{3.25}$$

It is important to model the ellipsoidal Earth when discussing the physical shape of the surface because mission requirements often depend on it. Mission height requirements are usually specified by the geodetic altitude rather than radial distances from the center of the Earth, just as coordinates on the surface of the Earth use geodetic latitude. The ellipsoidal nature of the Earth also affects the shape of the gravity field around the it. This is important in orbital mechanics because a non-spherical gravity field causes a body's orbital plane to precess around the Earth's rotation axis. However, reentry flights are short enough that these effects will not be considered and the inverse gravity law given in Eq. (3.23) is used.

3.3.2 Atmospheric Model

A model for the Earth's atmosphere is also necessary because the atmosphere has a direct impact on the aerodynamic forces acting on the vehicle. There are many atmospheric models available of varying degrees of fidelity, but here the 1976 Standard Atmosphere model is used [21]. It provides data on temperature, pressure, density, and speed of sound for altitudes well beyond the trajectories designed. Instead of using tabular data from the model directly, an analytic curve-fit developed from [9] is used in order to decrease the amount of computational time required to retrieve required data. Using a 9th order polynomial fit to the logarithm of the density data and a 9th order polynomial fit to the speed of sound data, the analytic models are

$$\rho = e^{a_0 + a_1 h_p + a_2 h_p^2 + \cdots a_9 h_p^9} \tag{3.26}$$

and

$$c = b_0 + b_1 h_p + b_2 h_p^2 + \cdots b_9 h_p^9 \,, \tag{3.27}$$

where ρ is the density, c is the speed of sound, and h_p is the geopotential altitude. The fitting coefficients, a_n and b_n, are given in Table 3.2. The geopotential altitude serves to account for the mass attraction of the Earth and rotational effects. It is related to the geometric altitude by

$$h_p = \frac{z R_0}{z + R_0} \,, \tag{3.28}$$

where R_0 is the radius of the Earth at 45 degrees latitude and z is the geodetic altitude given by Eq. (3.24).

The Earth and atmospheric model presented in this section define the environment that the reentry vehicle operates in and how that environment affects the motion of the vehicle. While there are other atmospheric effects, such as wind, they will not be considered in this analysis. The constants introduced in this section to model the Earth are given in Table 3.3.

Table 3.2: Density and Speed of Sound Fitting Coefficients

Parameter Order	**Density ρ (lb/ft^3)**	**Speed of Sound c (ft/sec)**
0	-2.60	1135.26
1	-2.18×10^{-5}	-8.34×10^{-3}
2	-4.81×10^{-10}	2.15×10^{-7}
3	5.61×10^{-15}	-4.21×10^{-12}
4	-6.16×10^{-20}	5.57×10^{-17}
5	5.34×10^{-25}	-4.33×10^{-22}
6	-2.79×10^{-30}	1.94×10^{-27}
7	8.10×10^{-36}	-4.98×10^{-33}
8	-1.22×10^{-41}	6.80×10^{-39}
9	7.41×10^{-48}	-3.83×10^{-45}

Table 3.3: Environmental Parameters

Property	**Value**	**Units**
R_0	20855530	ft
R_e	20925650	ft
f	0.003352811	–
G_m	1.307745×10^{15}	ft^3/sec^2
Ω_E	4.178074×10^{-3}	deg/sec

3.4 Dynamics Model

The full order dynamics that completely govern the translational and rotational movement of a body are derived by applying Newton's second law for a rigid body. Derivations of these six degree-of-freedom (6 DOF) equations of motion can be seen in [1][4][8], where the full state of the vehicle is described by twelve position, velocity, attitude, and rotation rate states.

The assumptions used to simplify the equations of motion are that attitude and rotation rate states are ignored, creating a 3 DOF model and that the sideslip angle, β, is assumed to be zero at all times. The 3 DOF equations of motion with $\beta = 0$ were adapted from [9]. The state vector for the 3 DOF vehicle is

$$\mathbf{x} = \begin{bmatrix} r \\ \mu \\ \lambda \\ v \\ \gamma \\ \psi \end{bmatrix}, \tag{3.29}$$

where $[r, \mu, \lambda]$ is the position vector in spherical coordinates in the ECEF frame and $[v, \gamma, \psi]$ is the velocity vector in spherical coordinates in the UEN frame. Control of a vehicle is generally accomplished through flap deflections used to rotate the body. In the point-mass equations of the 3 DOF dynamics, the attitude of the vehicle is used as a proxy for control. Assuming $\beta = 0$, the control vector is

$$\mathbf{u} = \begin{bmatrix} \alpha \\ \sigma \end{bmatrix}, \tag{3.30}$$

where α is the angle-of-attack and σ is the bank angle. The dynamics of the system can then be written as the set of nonlinear equations that describe the rates of change of each of the state variables as functions of the states, $\mathbf{x}$, the controls, $\mathbf{u}$, and the

time, t, such that

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{r} \\ \dot{\mu} \\ \dot{\lambda} \\ \dot{v} \\ \dot{\gamma} \\ \dot{\psi} \end{bmatrix} = \mathbf{g}(\mathbf{x}, \mathbf{u}, t) \,. \tag{3.31}$$

The rate of change for each of the states is

$$\begin{aligned}
\dot{r} &= v \sin\gamma \\
\dot{\mu} &= \frac{v \cos\gamma \cos\psi}{r \cos\lambda} \\
\dot{\lambda} &= \frac{v \cos\gamma \sin\psi}{r} \\
\dot{v} &= -\frac{D}{m} - g \sin\gamma + \Omega_E^2 r \cos\lambda (\sin\gamma \cos\lambda - \cos\gamma \sin\lambda \sin\psi) \\
\dot{\gamma} &= \frac{L \cos\sigma}{mv} + (\frac{v}{r} + \frac{g}{v}) \cos\gamma + 2\Omega_E \cos\lambda \cos\psi \\
&+ \frac{\Omega_E^2 r}{v} \cos\lambda (\cos\gamma \cos\lambda + \sin\gamma \sin\lambda \sin\psi) \\
\dot{\psi} &= \frac{L \sin\sigma}{mv \cos\gamma} - \frac{v}{r} \cos\gamma \cos\psi \tan\lambda \\
&+ 2\Omega_E (\tan\gamma \cos\lambda \sin\psi - \sin\lambda) \\
&- \frac{\Omega_E^2 r}{v \cos\gamma} \sin\lambda \cos\lambda \cos\psi
\end{aligned}$$

where m is the vehicle mass, g is the acceleration due to gravity, D is the drag, L is the lift, and Ω_E is the rate of angular rotation of the Earth. It is important to note that for the control variables, the bank angle, σ, is expressed explicitly in the rate equations, but the angle-of-attack, α, is expressed indirectly through the aerodynamic forces.

In these equations, the position rates $(\dot{r}, \dot{\mu}, \dot{\lambda})$ are solely functions of the position and velocity states. However, the velocity rates $(\dot{v}, \dot{\gamma}, \dot{\psi})$ are functions of the position and velocity states as well as several other accelerations. The effects of gravitational acceleration are contained in the speed and flight path angle rate equations, while

Coriolis acceleration terms and centrifugal accelerations terms are contained in all three velocity component rate equations. The aerodynamic forces are represented by lift in the flight path angle and heading angle rate equations, and by drag in the speed rate equation.

3.5 Guidance Algorithm

Apart from the physical properties affecting the reentry problem, the guidance algorithm must also be defined. As defined in Chapter 2, the guidance algorithm is a feedback law of the form

$$\delta \mathbf{u} = -K\delta \mathbf{x}\,, \tag{3.32}$$

where the feedback gain matrix, K, is determined by LQR techniques. The computation of K requires the linearization of the nonlinear system dynamics, given in Section 3.4, about a nominal trajectory. This linearization can be accomplished numerically, but for computational speed, the partial derivatives required for the linearization are expressed analytically. While analytically deriving the partial derivatives can be a time consuming process initially, the computational benefits become apparent when using the linearization within an optimization where a large number of functions calls occurs. The equations of motion used in this study are the same as used by Small, so the analytic linearization is the same; the complete derivation of this linearization can be seen in [9]. Using the analytic linearization is beneficial because not only is it used in the guidance algorithm, it is also required during covariance calculations, providing additional computational efficiency during trajectory optimization. To maintain the same linearization for both the guidance algorithm and the covariance dynamics, time is used as the independent variable.

In order to solve for K, the weighting matrices, Q and R, must also be defined. The weighting matrices, explained in detail in Section 2.2, are used by the designer to trade between penalties on state errors and on control usage. Bryson's rule was introduced as a means of selecting the elements of the weighting matrices based on the maximum desired deviations for each of the state and control variables. This

method is used to choose initial values for the weighting matrices.

Due to the limited energy available, it is not possible for the system reach all feasible states, so the system is not fully controllable. However, it is possible to obtain good tracking along reference trajectories by setting the elements of Q associated with the altitude and speed to zero, so that altitude and speed errors are not directly penalized. Errors in longitude, latitude, flight path angle, and heading angle are still penalized so that the directional components of the position and velocity vectors are tracked, which results in generally good overall tracking of reference trajectories. Using Bryson's rule, the four elements of the Q matrix corresponding to the directional components of the positions and velocity vectors as well as the two elements of the R matrix are computed from designed maximum allowable deviations to give

$$Q = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{(0.5 \text{ deg})^2} & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{(0.5 \text{ deg})^2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{(4 \text{ deg})^2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{(2 \text{ deg})^2} \end{bmatrix} \tag{3.33}$$

and

$$R = \begin{bmatrix} \frac{1}{(1.5 \text{ deg})^2} & 0 \\ 0 & \frac{1}{(20 \text{ deg})^2} \end{bmatrix}. \tag{3.34}$$

With the linearization analytically derived and the weighting matrices defined, the feedback gain matrix can be computed at any point along a nominal trajectory. During simulation, the gain matrix is computed offline along the nominal trajectory, then interpolated onboard at the current state during each call to the guidance algorithm. During trajectory optimization, whenever the guidance law is required during closed-loop covariance propagation, the gains are calculated for the nominal trajectory before the propagation, then interpolated as needed as the covariance is computed.

Chapter 4

Mission Planning for Atmospheric Reentry

The basic framework for the mission planning process outlined in Chapter 2 provides the method needed to implement covariance shaping during atmospheric reentry reference trajectory design. Using those tools and the model developed in Chapter 3, the optimal control problem used to generate reference trajectories is defined, along with the uncertainty model used to drive the covariance shaping used during the reference trajectory design process. The uncertainty model includes the definition of the uncertainties themselves as well as their implementation with linear covariance analysis and Monte Carlo simulations. Both of these tools allow the sensitivity of a trajectory to be quantified and will be used to measure the performance of the mission planning process.

4.1 Optimal Control Problem Formulation

Reference trajectory design is accomplished using trajectory optimization where an optimal control problem is formulated using the vehicle, environmental, and 3 DOF dynamics model from Chapter 3 to define the system of interest. The constraints used to complete the definition of the optimal control problem are introduced in this section.

First, two modifications are made to the system in order to facilitate the formulation of the optimal control problem. In the vehicle model, limits were set on the rates of change for the angle-of-attack and the bank angle to prevent instantaneous attitude changes. Also, bounds were set on the angle-of-attack to prevent large attitudes, which are not valid with the current aerodynamic model. With the angle-of-attack acting as a control variable, its value can be limited during optimization, but there is no way to directly limit the rates of α and σ within the optimal control problem framework.

To overcome this limitation, the state vector is augmented so that the angle-of-attack and the bank angle become additional state variables, while the rates of the angle-of-attack and bank angle become the new control variables. This allows for bounds to be set on both the values of the angles as well as their rates, so that immediate changes in the angles do not occur. The rates for the angle-of-attack and bank angle states in the equations of motion are simply the rates given by the new control variables.

Additionally, for simplicity, the radial distance from the center of the Earth to the vehicle, r, is replaced with the altitude of the vehicle over the equatorial radius, a, to provide a more intuitive value in the state vector. The radial distance and altitude, a, are related by

$$r = R_e + a\,; \tag{4.1}$$

taking the time derivative of this relationship results in

$$\dot{r} = \dot{a}\,. \tag{4.2}$$

As the dynamics are equivalent, the variables can be interchanged freely without affecting the equations of motion. Incorporating these modifications, the state vector

from Eq. (3.29) becomes

$$\mathbf{x} = \begin{bmatrix} a \\ \mu \\ \lambda \\ v \\ \gamma \\ \psi \\ \alpha \\ \sigma \end{bmatrix}, \tag{4.3}$$

while the control vector from Eq. (3.30) becomes

$$\mathbf{u} = \begin{bmatrix} \dot{\alpha} \\ \dot{\sigma} \end{bmatrix}. \tag{4.4}$$

These modifications are implemented solely for use in the optimization and will not be utilized elsewhere.

4.1.1 State and Control Bounds

For the optimal control problem formulation, bounds must be placed on the states and controls for several reasons. First, there are singularities present in the 3 DOF equations of motion at

$$\begin{aligned} a &= -R_e \\ \lambda &= \pm 90 \text{ deg} \\ v &= 0 \\ \gamma &= \pm 90 \text{ deg} \end{aligned}$$

that must be avoided so that the optimization algorithm does not attempt to evaluate an undefined equation. Also, bounds can be used to prevent the optimization from allowing the undesirable vehicle attitudes and attitude rates previously discussed.

This includes the minimum and maximum allowable angle-of-attack as well as the limits on the angle-of-attack and bank angle rates given in Table 3.1. In addition, bounds on the altitude, a, can be used to prevent infeasible behavior such as flying underground. A lower bound can be placed on lowest allowable value of a, which occurs at the poles of the Earth where $a = -70,160$ ft. Finally, large but finite bounds can be placed on the remainder of the states, providing a finite search area for the optimization algorithm to operate in, while still leaving these states effectively unconstrained. The complete bounds on the states and controls used are

$$\mathbf{x}^L = \begin{bmatrix} -70,160 \text{ ft} \\ -360 \text{ deg} \\ -89 \text{ deg} \\ 1 \text{ ft/sec} \\ -89 \text{ deg} \\ -360 \text{ deg} \\ 0 \text{ deg} \\ -360 \text{ deg} \end{bmatrix} \leq \mathbf{x}(t) \leq \begin{bmatrix} 10,000,000 \text{ ft} \\ 360 \text{ deg} \\ 89 \text{ deg} \\ 100,000 \text{ ft/sec} \\ 89 \text{ deg} \\ 360 \text{ deg} \\ 15 \text{ deg} \\ 360 \text{ deg} \end{bmatrix} = \mathbf{x}^U \tag{4.5}$$

and

$$\mathbf{u}^L = \begin{bmatrix} -10 \text{ deg/sec} \\ -30 \text{ deg/sec} \end{bmatrix} \leq \mathbf{u}(t) \leq \begin{bmatrix} 10 \text{ deg/sec} \\ 30 \text{ deg/sec} \end{bmatrix} = \mathbf{u}^U . \tag{4.6}$$

4.1.2 Initial, Terminal and Event Constraints

While the state and control bounds limit general vehicle behavior, constraints on the initial and terminal state values must also be chosen to specify the desired trajectory. The trajectory chosen to test the reference trajectory design techniques is the entry phase of atmospheric reentry, which begins when the vehicle reaches an insertion point in the atmosphere sufficiently dense enough for aerodynamic control. The entry phase ends at a terminal interface where a terminal guidance algorithm is used to achieve final mission requirements. Trajectory characteristics outside of the entry phase can vary according to mission requirements and will not be considered.

The insertion point chosen for the entry phase is at 0 deg longitude, 0 deg latitude with an altitude of 150,000 ft. The velocity is 13,000 ft/sec traveling horizontally due East. It is also assumed that the vehicle has zero angle-of-attack and bank angle. The initial constraints are

$$\mathbf{x}_0 = \begin{bmatrix} 150,000 \text{ ft} \\ 0 \text{ deg} \\ 0 \text{ deg} \\ 13,000 \text{ ft/sec} \\ 0 \text{ deg} \\ 0 \text{ deg} \\ 0 \text{ deg} \\ 0 \text{ deg} \end{bmatrix}. \tag{4.7}$$

The terminal interface is chosen so that the vehicle reaches a fixed target location at a desired speed, flight path angle, and angle-of-attack. In order to allow flexibility in the trajectory optimization, the terminal heading and bank angle are allowed to vary. The terminal target location is defined in geodetic coordinates, which prevents directly constraining the terminal state values for altitude and latitude. Instead, events for the target geodetic altitude, z, and latitude, $\lambda_{\text{geodetic}}$, can be written as

$$e_z = z(a_f, \lambda_f) \tag{4.8}$$

and

$$e_{\lambda_{\text{geodetic}}} = \lambda_{\text{geodetic}}(a_f, \lambda_f) \tag{4.9}$$

using the conversions given in Eqs. (3.24) and (3.25). By setting the upper and lower event constraints to the same value, the event functions can be forced to a particular value. An event constraint is not required for the longitude as there is no difference between geocentric and geodetic longitude; instead a terminal constraint can be placed directly on μ_f. The same is true for the terminal constraints on speed, flight path angle, and angle-of-attack, which are constant regardless of the final target location.

The terminal target position chosen is 9 deg longitude, 1.5 deg geodetic latitude at a geodetic altitude of 45,000 ft, well within the range of the test vehicle. At the terminal interface, the required velocity is 7,200 ft/sec, the flight path angle is −45 deg, and the angle-of-attack is 0 deg. The event constraints can be written as

$$e_z^L = e_z^U = 45,000 \text{ ft} \tag{4.10}$$

and

$$e^L_{\lambda_{\text{geodetic}}} = e^U_{\lambda_{\text{geodetic}}} = 1.5 \text{ deg}, \tag{4.11}$$

while the remainder of the terminal constraints are

$$\mu_f = 9 \text{ deg}, \tag{4.12}$$

$$v_f = 7,200 \text{ ft/sec}, \tag{4.13}$$

$$\gamma_f = -45 \text{ deg}, \tag{4.14}$$

and

$$\alpha_f = 0 \text{ deg}. \tag{4.15}$$

4.1.3 Path Constraints

The final set of constraints that must be considered are path constraints along the trajectory beyond the state and control bounds. The constraints that are applied are limits on the dynamic pressure and on the geodetic altitude. Setting a minimum path constraint on dynamic pressure ensures that the aerodynamic forces acting on the vehicle are large enough that the vehicle remains controllable. A minimum path constraint on the geodetic altitude is necessary to ensure that the optimization does not design a trajectory that goes beneath the surface of the Earth. Upper path constraints are chosen to be large enough to leave the path functions effectively un-

constrained. The path functions are

$$h_q = q(t) \tag{4.16}$$

for the dynamic pressure, where $q(t)$ is given by Eq. (3.4), and

$$h_z = z(t) \tag{4.17}$$

for the geodetic altitude, where $z(t)$ is given by Eq. (3.24). The path constraints can then be written

$$\mathbf{h}^L = \begin{bmatrix} 1.450 \text{ psi} \\ 0 \text{ ft} \end{bmatrix} \tag{4.18}$$

and

$$\mathbf{h}^U = \begin{bmatrix} 1,450 \text{ psi} \\ 10,000,000 \text{ ft} \end{bmatrix}. \tag{4.19}$$

4.2 Trajectory Shaping Techniques

The primary mechanism for shaping the design of a trajectory during the optimization is through the formulation of the cost function, which defines the performance metric that is to be minimized. In order to reduce the sensitivity of the trajectory to uncertainty, the linear covariance of the trajectory can be used as the performance metric to quantify how the trajectory responds to uncertainty. The covariance can then be penalized within the cost function to decrease expected state errors and therefore reduce the sensitivity of the trajectory to uncertainties. This is known as the covariance shaping technique and will be implemented in several ways for the reentry problem.

4.2.1 Minimum Effort

In order to validate covariance shaping as a viable method for designing trajectories, there must first be a baseline trajectory design process that covariance shaped tra-

jectories can be judged against. To accomplish this, a performance metric is used that measures the control effort along the trajectory. This minimum effort metric is purely a function of the nominal trajectory and does not incorporate any measure of uncertainty. A cost function penalizing this metric is used to design a baseline minimum effort trajectory to which covariance shaped trajectories can be compared.

The minimum effort cost function seeks to minimize the amount of control usage over the course of the trajectory. In this context, control usage means changes in angle-of-attack and bank angle. This is a commonly used performance metric for reentry vehicles as it can minimize the use of control actuators such as flaps. Increased use of control actuators often leads to issues such as actuator failure, ablation or other effects which negatively affect vehicle control. Within the current problem formulation, accumulated control effort can be measured by integrating the angular rate commands over the course of the trajectory. The angular rates can be normalized by their limits and squared so that a cost function can be written as

$$J_{\text{minEff}} = \int_{t_0}^{t_f} \left\{ \left(\frac{\dot{\alpha}}{\dot{\alpha}^U} \right)^2 + \left(\frac{\dot{\sigma}}{\dot{\sigma}^U} \right)^2 \right\} dt \,. \tag{4.20}$$

Normalizing each control variable as a ratio of their limit value allows for an equal weighting between penalizing effort in angle-of-attack and bank angle.

This cost function not only minimizes the rotation of the body, but also helps smooth the control variables by reducing abrupt changes in attitude, which can help prevent the generation of infeasible trajectories. The feasibility of a trajectory can be validated by comparing the discretized trajectory from the optimization results with a propagated trajectory. Because the optimization results are only defined at discrete nodes, to generate a continuous reference trajectory, the control variables must first be interpolated at the node points then the differential equations defining the system dynamics can then be propagated from the initial conditions as an initial value problem. This provides a continuous, propagated reference trajectory for comparison with the discretized trajectory from the optimization. A feasible trajectory would have very little error between the two trajectories.

However, when a discretized trajectory has control variables with large changes between node points, the interpolation of the controls will not be as accurate and can result in trajectories that when propagated, are shown to not match the optimization results and are therefore infeasible. One benefit of the minimum effort cost function is that the smoothness that it induces in the control variables can prevent this situation from happening, reducing the chances of generating an infeasible trajectory.

4.2.2 Open-Loop Covariance Shaping

The first covariance shaping technique that is implemented for designing trajectories utilizes the open-loop covariance. The open-loop covariance is solely a function of the nominal states and controls of the trajectory and does not take into account a feedback guidance algorithm. The benefit of using an open-loop covariance shaping technique is that it does not require that a guidance law be defined before the trajectory is designed. This can allow for the rapid development of new trajectories without worrying about the specifics of a guidance law. However, the open-loop covariance does not necessarily provide insights into the properties of a trajectory when a guidance law is introduced.

While the goal for this reentry problem is to reduce the sensitivity of a trajectory to uncertainty, this is particularly important at the end of the flight, because arriving at the terminal conditions with minimal error can be critical and failure to do so can lead to mission failure. Given an initial covariance matrix and process noise model, the open-loop covariance can be propagated along the trajectory to the terminal conditions according to Eq. (2.28). A cost function can then be written as a function of the terminal open-loop covariance of the states. For this study, reducing terminal position error was assumed to be an important mission goal, so the cost function penalizes the terminal values of the covariance for longitude and latitude. However, errors in other states will be examined later. In addition, in order to obtain similar smoothing effects as the minimum effort cost function, a component of the

cost function will also penalize control effort. This results in the cost function

$$J_{\mathrm{OL}} = w(\sigma^2_{\mu_{\mathrm{OL}}} + \sigma^2_{\lambda_{\mathrm{OL}}}) + \int_{t_0}^{t_f} \left\{ \left(\frac{\dot{\alpha}}{\dot{\alpha}^U} \right)^2 + \left(\frac{\dot{\sigma}}{\dot{\sigma}^U} \right)^2 \right\} dt \,, \tag{4.21}$$

where w is a weighting factor used to adjust how much the contributions of the covariance cost and the control effort cost are penalized.

4.2.3 Closed-Loop Covariance Shaping

The final trajectory shaping method considered is the closed-loop covariance shaping technique. The cost function for this technique is similar to the open-loop shaping cost function, but instead makes use of the closed-loop covariance. This incorporates the feedback guidance algorithm designed in Chapter 3 and is governed by the dynamics given in Eq. (2.31). The closed-loop covariance provides expected state errors under the guidance law, such that it directly quantifies the sensitivity to uncertainty under the guidance algorithm and therefore allows for real world predictions to be made about the performance characteristics of the trajectory.

However, incorporating the closed-loop covariance into the cost function does have drawbacks. Because the guidance law is designed using LQR techniques, feedback gains are required every time that the closed-loop covariance is calculated. Generating these gains during the optimization process causes two issues: first, calculating the gains during each covariance calculation drastically increases the computational time of the optimization. Additionally, as the optimization algorithm seeks a solution, it will often generate and evaluate trajectories that break dynamic constraints, which is generally not an issue as a long as the final optimal solution is feasible. However, when the feedback guidance law is included, the gains calculated using LQR techniques for extremely infeasible trajectories are nonsensical and in some cases, the gains cannot be calculated at all, which stops the optimization process. To avoid the issues associated with generating feedback gains during the optimization process, the gains are generated offline for a sample trajectory, which is then used as the a

priori guess trajectory for generating the closed-loop covariance shaped trajectory. The optimization is then carried out using the gains from the guess trajectory so that new gains do not have to be generated during each covariance calculation. Once an optimal solution is found, guidance gains are computed offline for the new trajectory, which then becomes the *a priori* guess trajectory for the next optimization. This iterative process continues until the feedback gains calculated from the guess trajectory match the feedback gains from the optimal output trajectory, providing the final closed-loop covariance shaped trajectory.

Like the open-loop covariance shaped trajectory, the closed-loop covariance cost function will seek to minimize longitudinal and latitudinal errors while minimizing control effort. The closed-loop covariance cost function is

$$J_{\mathrm{CL}} = w(\sigma^2_{\mu_{\mathrm{CL}}} + \sigma^2_{\lambda_{\mathrm{CL}}}) + \int_{t_0}^{t_f} \left\{ \left(\frac{\dot{\alpha}}{\dot{\alpha}^U} \right)^2 + \left(\frac{\dot{\sigma}}{\dot{\sigma}^U} \right)^2 \right\} dt \,, \tag{4.22}$$

where w is again the weighting factor to balance the two components of the cost.

4.3 Uncertainty Modeling

Uncertainty in the reentry problem is captured in two distinct ways. First, linear covariance analysis is used within the mission planning process to design trajectories that minimize the sensitivity to uncertainty. Second, the effects of uncertainties on the nonlinear 3 DOF dynamics are explored through the use of Monte Carlo simulations. These simulations will be compared to expected errors from the linear covariance analysis and used to validate the trajectory-shaping process.

4.3.1 Uncertainty Parameters

There are many types of uncertainties that affect reentry vehicles that can be modeled a variety of different ways, but the following uncertainties provide a reasonable representation of what might be encountered and are used during mission planning and testing. It is assumed that all uncertainties used are normally distributed with

Table 4.1: Insertion Error Parameters

State	3σ **Error**	**Units**
Altitude	0	ft
Longitude	0.045	deg
Latitude	0.045	deg
Speed	50	ft/sec
Flight Path Angle	1	deg
Heading Angle	1	deg

zero mean and defined standard deviations.

The first type of uncertainty included is insertion error, where the vehicle does not arrive at the entry phase with exactly the intended initial conditions. This includes errors in both the initial position and velocity states and it is assumed that the initial errors are uncorrelated. The insertion errors are distributed according to the values in Table 4.1.

Errors can also occur due to uncertainty in the system dynamics. These uncertainties take the form of random disturbances due to unmodeled forces acting on the vehicle during reentry, which can be attributed to factors such as wind or local density variations. These perturbations are modeled as process noise acting on the velocity state rate equations for speed, flight path angle, and heading angle by including additive white Gaussian noise in the nominal nonlinear dynamics to give the actual rates

$$\dot{v}_{\text{act}} = \dot{v}_N + \eta_v \,, \tag{4.23}$$

$$\dot{\gamma}_{\text{act}} = \dot{\gamma}_N + \eta_\gamma \,, \tag{4.24}$$

and

$$\dot{\psi}_{\text{act}} = \dot{\psi}_N + \eta_\psi \,, \tag{4.25}$$

where the subscript N denotes the nominal equations of motion and the subscript act

Table 4.2: System Dynamics Error Parameters

State	Symbol	Spectral Density	Units
Speed	η_v	2.778	ft^2/sec^3
Flight Path Angle	η_γ	4.444×10^{-5}	deg^2/sec
Heading Angle	η_ψ	4.444×10^{-5}	deg^2/sec

denotes the actual stochastic equations of motion. The random variables η_v, η_γ, and η_ψ represent the white noise perturbations with spectral densities given in Table 4.2.

Finally, there are several parameters in the plant model whose uncertainty is modeled. These include the aerodynamic coefficients, C_N and C_A, and the atmospheric density. The uncertainty in these parameters is modeled as constant rather than as random noise along the trajectory. For the aerodynamic coefficients the uncertainty is characterized as zero mean, with 3σ confidence values of 5% of the nominal coefficient values such that

$$\sigma_{C_N}(\mathbf{x}) = 0.0167 C_{N_N}(\mathbf{x}) \tag{4.26}$$

and

$$\sigma_{C_A}(\mathbf{x}) = 0.0167 C_{A_N}(\mathbf{x}) \,, \tag{4.27}$$

where the N subscript denotes the nominal parameter value. To characterize uncertainty in the density model NASA's Earth Global Reference Atmospheric Model (GRAM) is used [22]. This model provides statistical distributions about the nominal density as a function of the geodetic altitude. The GRAM data shows that 1σ error values for density increase approximately exponentially from 0 ft to 333,000 ft geodetic altitude, going from 0.3517% to 15.3659%. This altitude range includes all of the altitudes encountered in this study. In order to increase computational speed, an exponential fit is applied to the model to provide an analytic expression for the

density uncertainty as a function of geodetic altitude given by

$$\sigma_\rho(z) = 0.003517 \exp\left(\frac{z}{87368 \text{ ft}}\right) . \tag{4.28}$$

These parameter uncertainties are incorporated by using

$$\rho = (1 + C_\rho \sigma_\rho(\mathbf{x}))\rho_N , \tag{4.29}$$

$$C_N = (1 + C_{C_N} \sigma_{C_N}(\mathbf{x}))C_{N_N} , \tag{4.30}$$

and

$$C_A = (1 + C_{C_A} \sigma_{C_A}(\mathbf{x}))C_{A_N} , \tag{4.31}$$

where the σ terms represent the standard deviation of the parameter errors. The stochasticity of the system is captured by the constants C_ρ, C_{C_N}, and C_{C_A}, which are normally distributed with zero mean and a standard deviation of 1.

4.3.2 Linear Covariance Propagation

For use during the mission planning process, these uncertainties need to be structured within the linear covariance framework. While linear covariance analysis does provide insight into the stochastic system, it uses a linearized model of the dynamics. Any effects due to nonlinearities in the actual plant dynamics will not be captured with the covariance. This means that while linear covariance analysis is a useful tool for quantifying the effects of uncertainty on a system, it will not provide a completely accurate estimate of expected errors.

The uncertainty models introduced are incorporated in three ways. Including the parameter uncertainties for the aerodynamic coefficients and density requires the augmentation of the state vector as described in Subsection 2.3.3. This is accomplished by including the constant parameters C_ρ, C_{C_N}, and C_{C_A} introduced in Eqs. (4.29), (4.30), and (4.31) as additional states during covariance propagration. The state

vector for covariance calculations can be written as

$$\mathbf{x} = \begin{bmatrix} a \\ \mu \\ \lambda \\ v \\ \gamma \\ \psi \\ C_\rho \\ C_{C_N} \\ C_{C_A} \end{bmatrix} \tag{4.32}$$

and the equations of motion are updated to include trivial dynamics for the parameter states as

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{a} \\ \dot{\mu} \\ \dot{\lambda} \\ \dot{v} \\ \dot{\gamma} \\ \dot{\psi} \\ 0 \\ 0 \\ 0 \end{bmatrix} . \tag{4.33}$$

To incorporate the insertion errors, initial variance values are computed for each of the state errors by squaring the standard deviation of the uncertainty for each state. Assuming no correlation between the insertion errors, these initial variances make up the diagonal elements of the initial covariance matrix while the rest of the

initial matrix elements are set to zero giving

$$\mathrm{diag}(P_0) = \begin{bmatrix} 0 \\ 0.015^2 \text{ deg}^2 \\ 0.015^2 \text{ deg}^2 \\ 16.67^2 \text{ ft}^2/\text{s}^2 \\ 0.33^2 \text{ deg}^2 \\ 0.33^2 \text{ deg}^2 \\ 1 \\ 1 \\ 1 \end{bmatrix} . \tag{4.34}$$

With trivial dynamics and initial values of 1, the uncertainty of the added parameter states remains constant over the trajectory. Process noise acting on the system dynamics is implemented in the linear covariance analysis by populating the diagonal of the spectral density matrix for the process noise

$$\mathrm{diag}(R_{ww}) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 2.778 \text{ ft}^2/\text{s}^3 \\ 4.444 \times 10^{-5} \text{ deg}^2/\text{s} \\ 4.444 \times 10^{-5} \text{ deg}^2/\text{s} \\ 0 \\ 0 \\ 0 \end{bmatrix} . \tag{4.35}$$

.

With P_0 and R_{ww} defined, the covariance of the system can be calculated by propagating the covariance dynamics given in Chapter 2 by Eqs. (2.28) or (2.31). The former allows for the calculation of the open-loop covariance while the latter takes into account the feedback guidance algorithm for the computation of the closed-loop

covariance. During reference trajectory design, the full set of uncertainties detailed is used to provide a worst-case scenario so that sensitivity to a wide variety of uncertainties is reduced. However, after the trajectory design is complete, uncertainties can be applied to trajectories individually or in combination to analyze the effects of specific types of errors.

Regardless of whether the covariance is calculated during reference trajectory design or post-optimization analysis, the computation can be accomplished using the same algorithm. Using the initial covariance, P_0, and the covariance dynamics provided by either Eq. (2.28) or (2.31), the covariance is found by propagating the covariance matrix along the trajectory using any ordinary differential equation (ODE) solver. As noted in [9], the computational time of the covariance propagation must be very small during the trajectory optimization to reduce overall computation expense. In order to achieve this, a fixed-step Runge-Kutta 4th order algorithm is used to solve the covariance initial value problem.

The fixed-step Runge-Kutta method allows for accuracy in the covariance calculation to be traded with the run time of the propagation by adjusting the number of steps, n. Because of the number of function calls that are required within the optimization, it is imperative that the run time of the covariance propagation be as small as possible, while still retaining sufficient accuracy. Table 4.3 shows a comparison of the fixed-step Runge-Kutta method with several step sizes versus the Matlab `ode45` solver [23] for the covariance from a sample trajectory, demonstrating the computational efficiency advantage of the Runge-Kutta method while obtaining reasonably accurate covariance values. For use during trajectory optimization, the smaller number of steps, $n = 100$, is used to reduce computation time, while $n = 500$ steps is used for post-optimization linear covariance analysis.

4.3.3 Monte Carlo Simulations

The other method of analyzing the effects of uncertainty on a trajectory are with Monte Carlo simulations. Monte Carlo simulations are conducted by inserting random errors into the full nonlinear dynamics and propagating the dynamics to the terminal

Table 4.3: 3σ Terminal Errors from Covariance Propagation

3σ **Error State**	`ode45`	**RK4** $n = 100$	**RK4** $n = 500$	**Units**
$3\sigma_a$	7029.4	7119.6	7044.7	ft
$3\sigma_\mu$	0.0455	0.0450	0.0454	deg
$3\sigma_\lambda$	0.0643	0.0646	0.0643	deg
$3\sigma_v$	57.51	31.68	57.70	ft/sec
$3\sigma_\gamma$	3.17	1.78	3.12	deg
$3\sigma_\psi$	16.94	12.03	16.79	deg
Run Time	10.479	0.513	2.508	sec

conditions. This process is repeated a large number of times and the resulting errors are used to draw statistical conclusions about the effects of uncertainty on the system. Because the nonlinear dynamics are used, Monte Carlo simulations provide a more accurate expected errors than the linear covariance analysis. The process, however, is much more computationally expensive because it requires many propagations of the trajectory, versus the single propagation required to calculate the covariance.

The simulation uses the original state vector, control vector, and dynamics given by Eqs. (3.29), (3.30), and (3.31). Like the covariance calculations, the trajectory can be propagated using the nonlinear 3 DOF dynamics using any ODE solver; in this case, a fixed-step Runge-Kutta 4th order method will also be used. The discretized control history generated during trajectory optimization is interpolated to provide a continuous set of commands that is then propagated with the 4th order Runge-Kutta algorithm.

The simplest method for propagating the trajectory is to ignore the guidance law and propagate the trajectory open-loop, using only the nominal control history. With no perturbations, propagating open-loop should provide zero terminal error. However, due to the discrete nature of the designed trajectory obtained from the optimization, when the controls are interpolated at the discrete node points and propagated, there

Table 4.4: Trajectory Propagation Errors

State	Designed Trajectory	Propagated Trajectory	Error	Units
a	45000	44953.21	46.79	ft
μ	9	8.99968	3.15×10^{-4}	deg
λ	1.5	1.49990	7.72×10^{-5}	deg
v	7200	7197.05	2.95	ft/sec
γ	-45	-45.046	0.046	deg
ψ	41.563	41.585	-0.022	deg

will be some error between the designed trajectory and the propagated trajectory. Table 4.4 shows terminal errors seen between the designed and propagated trajectories for a sample trajectory designed using the minimum effort cost function.

As long as the errors between the designed and propagated trajectories are small, the trajectory is considered feasible and the Runge-Kutta algorithm provides a suitable means for propagating the trajectory. The guidance law may then be included to obtain the closed-loop propagation of the trajectory. In the case where state errors occur, the guidance law will attempt to minimize deviations from the nominal trajectory. The guidance algorithm requires knowledge of the nominal trajectory and controls, which must both be interpolated at the time steps of the propagation. In addition, the guidance algorithm also relies on the feedback gain matrix K, which is computed from the linearized dynamics equations. For the simulation, K is calculated at points along the nominal trajectory as a function of time, and is interpolated during the simulation for use in the guidance algorithm. From here, the closed-loop trajectory may then be propagated from the initial conditions using the fixed-step Runge-Kutta algorithm.

With the nominal simulation framework constructed, perturbations may now be added to model the uncertainties described in Subsection 4.3.1. These errors can be added individually or in any combination to see the response of the nonlinear

system. Monte Carlo simulations provide the statistical response full plant model and trajectory to uncertainties, capturing the full effects of the nonlinear dynamics, and can be used to validate the accuracy of the linear covariance analysis. The different types of uncertainties are included in several different ways. Insertion errors are added by shifting the initial conditions by a random variable that is normal distributed with zero mean and the appropriate standard deviation for each state's error. The parametric uncertainties for density and aerodynamic coefficients are implemented by shifting those parameters using equations (4.29) through (4.31). Finally, the process noise affecting the dynamics of the system must be included. However, the inclusion of stochastic variables in the differential equations of the vehicle dynamics requires some changes to be made when using numerical integration [24]. To properly implement the process noise into the dynamics for propagation, the traditional Runge-Kutta algorithm must be modified to include the stochastic variable $\mathbf{w}$, which is the white noise acting on the system. This stochastic variable is computed once during each time step and is based on the spectral density of the process noise affecting the system, R_{ww}, and the step size of the integration, Δt. It is a normally distributed with zero mean and a variance defined by

$$\mathrm{Var}(\mathbf{w}) = \frac{R_{ww}}{\Delta t}\,. \tag{4.36}$$

This allows $\mathbf{w}$ to be calculated by

$$\mathbf{w} = \sqrt{\frac{R_{ww}}{\Delta t}}Z\,, \tag{4.37}$$

where Z is a normally distributed random number with a variance of one. Using this stochastic variable, the final formulation of the augmented Runge-Kutta algorithm is

$$y_{n+1} = y_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)\,, \tag{4.38}$$

where

$$k_1 = f(y_n, t_n)\Delta t + \mathbf{w}\Delta t\,,$$

$$k_2 = f(y_n + \frac{1}{2}k_1, t_n + \frac{1}{2}\Delta t)\Delta t + \mathbf{w}\Delta t\,,$$

$$k_3 = f(y_n + \frac{1}{2}k_2, t_n + \frac{1}{2}\Delta t)\Delta t + \mathbf{w}\Delta t\,,$$

and

$$k_4 = f(y_n + k_3, t_n + \Delta t)\Delta t + \mathbf{w}\Delta t\,.$$

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 5

Covariance Shaping Technique Results

Implementing the mission planning process discussed in Chapter 4, nominal reference trajectories are designed using the described shaping techniques. Linear covariance analysis can be used to generate the expected errors for each of the trajectories, which can then be compared with Monte Carlo simulation data to validate the results from the covariance analysis. Using these tools, the reference trajectories designed can be compared to determine the capability of the covariance shaping techniques to reduce the sensitivity of the reference trajectories to uncertainty.

5.1 Nominal Reference Trajectories

Using the three trajectory shaping techniques discussed in Section 4.2, nominal reference trajectories are generated. These three trajectories all have the same initial and terminal conditions as well as path and event constraints, but are shaped by the minimum effort, open-loop covariance, and closed-loop covariance shaping techniques. Figures 5-1 through 5-8 show the resulting flight paths.

Due to the nature of the vehicle, the initial conditions, and the terminal conditions, all three trajectories make use of a "skip" manuever, where the vehicle dives lower into the atmosphere to gain speed before climbing higher into less dense air for reduced

drag flight. However, the different shaping techniques cause each of the trajectories to take different paths to reach the terminal conditions. The minimum effort trajectory is driven by its cost function that penalizes nominal control changes. This can be seen in the smoothness of the angle-of-attack and bank angle in Figures 5-7 and 5-8 and results in the vehicle taking the widest turn to reach the target. There are also no rapid changes seen in any of the other state variables. The minimum effort trajectory has small skip around 100 sec into the flight followed by a more substantial skip just after 200 sec into the flight.

The open-loop covariance shaped trajectory was designed by penalizing open-loop covariance in the terminal position states to reduce the sensitivity of the trajectory to perturbations. It is hoped that penalizing open-loop covariance will also result in improved closed-loop performance. The open-loop covariance shaped trajectory must also balance the smoothing component of its cost function with the penalty on covariance. This results in a trajectory that has more active nominal control use than the minimum effort trajectory. It also has a much more substantial skip manuever early in the flight, seen in Figure 5-3, where it flies at almost a constant altitude for an extended period of time before diving into a smaller skip then to the target. The extended length of the first skip at high altitudes allows the vehicle to fly with less drag, resulting in higher speeds throughout the majority of the trajectory. This cause the open-loop trajectory to finish prior to the minimum effort trajectory.

Finally, there is the closed-loop covariance shaped trajectory, which is shaped by penalizing the closed-loop covariance of the trajectory generated under the influence of a feedback guidance law. The shaping technique also seeks to reduce closed-loop trajectory sensitivity to uncertainty, but tries to do so in a more direct manner. The closed-loop covariance shaped trajectory spends significantly more time lower in the atmosphere than the other two trajectories, which slows it down. In order to reach the terminal conditions, it must take a more direct route to the target, as seen in Figures 5-1 and 5-2. This trajectory also used more nominal control movement than the minimum effort trajectory, holding the minimum angle-of-attack for around 30 sec before quickly increasing the angle. It also spends close to 40 sec with a 0 deg

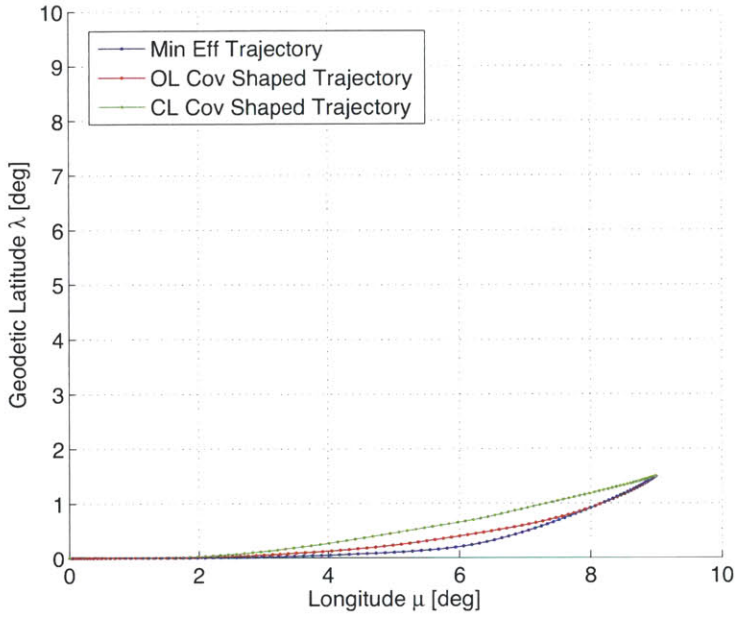

Figure 5-1: Reference Trajectory Ground Tracks

bank angle as it moves directly to the terminal conditions.

5.2 Linear Covariance Analysis

With the nominal path for each of the reference trajectories designed, the linear covariance of the reference trajectories can now be analyzed. The purpose of both the open-loop and closed-loop covariance shaping techniques is to minimize the longitude and latitude error at the final conditions, but the two methods go about this in different ways. Minimizing the covariance for the vehicle flying open-loop is simple to implement because a guidance algorithm do not have to be designed and it could potentially serve as a proxy for reducing the covariance when the vehicle is flown closed-loop. Minimizing the closed-loop covariance seeks to more directly improve performance, but does require the prior design of a guidance algorithm. To better understand these shaping methods, the covariance and how it changes are examined over the course of the trajectory. This is conducted first with the open-loop covariance to verify that the open-loop covariance shaping technique does reduce expected

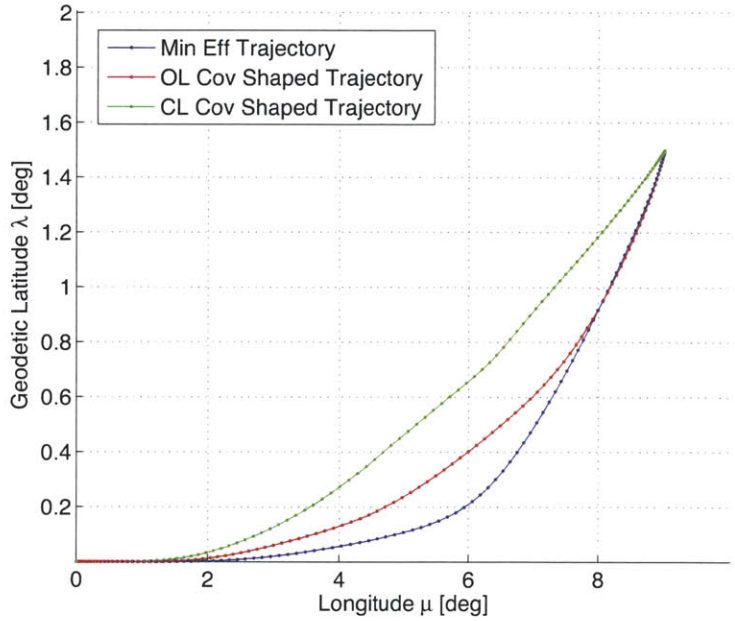

Figure 5-2: Reference Trajectory Ground Tracks - Expanded View

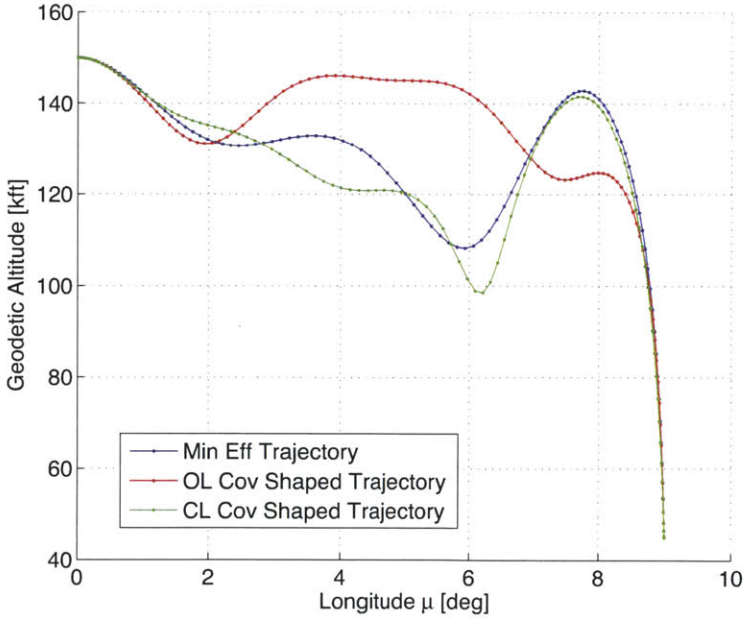

Figure 5-3: Reference Trajectory Altitude Profiles

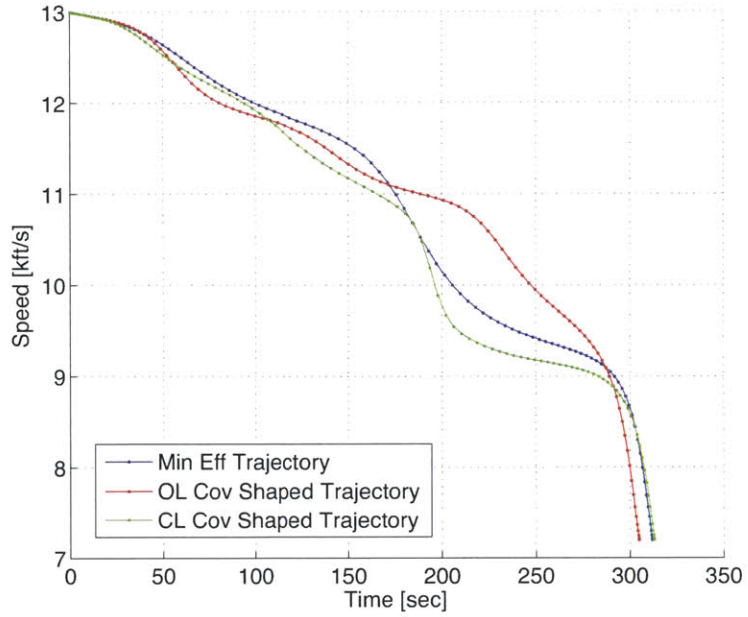

Figure 5-4: Reference Trajectory Speed Profiles

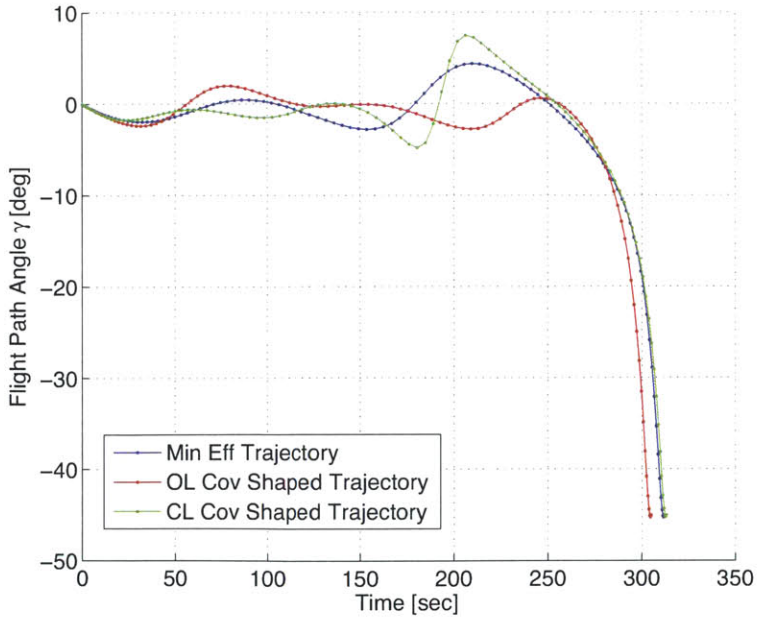

Figure 5-5: Reference Trajectory Flight Path Angle Profiles

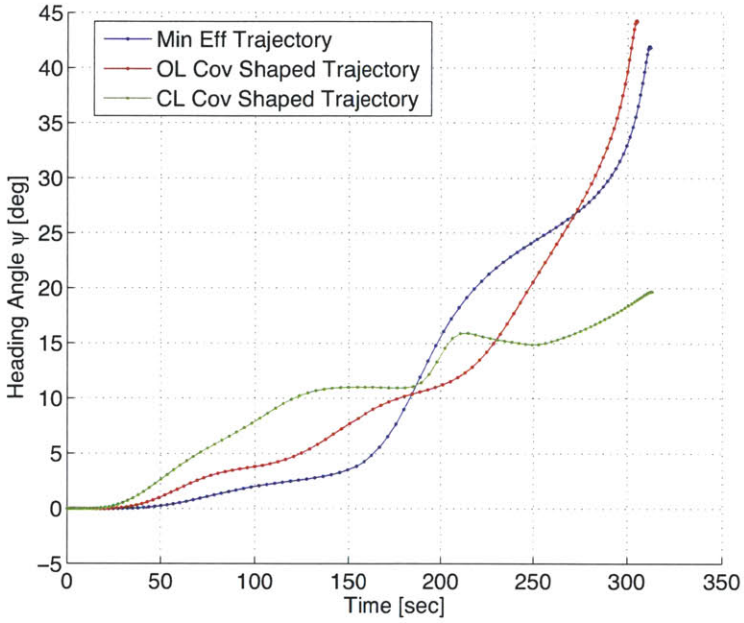

Figure 5-6: Reference Trajectory Heading Angle Profiles

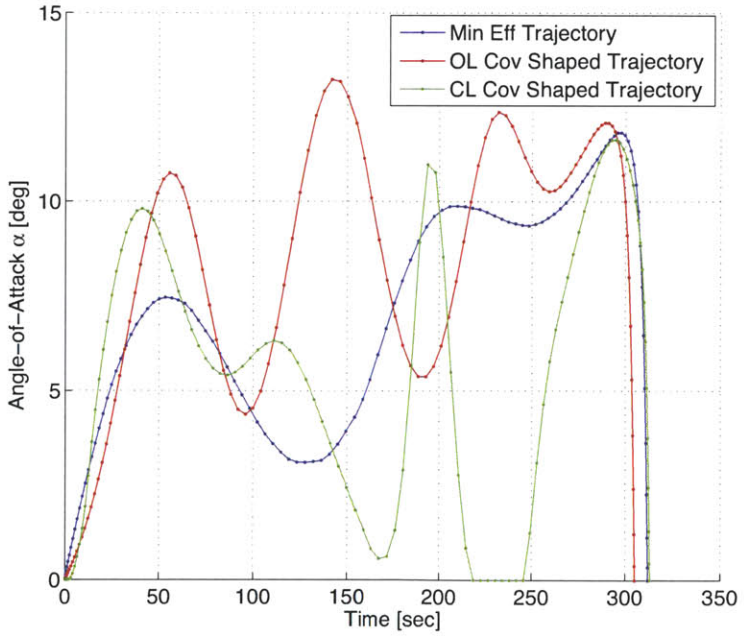

Figure 5-7: Reference Trajectory Angle-of-Attack Histories

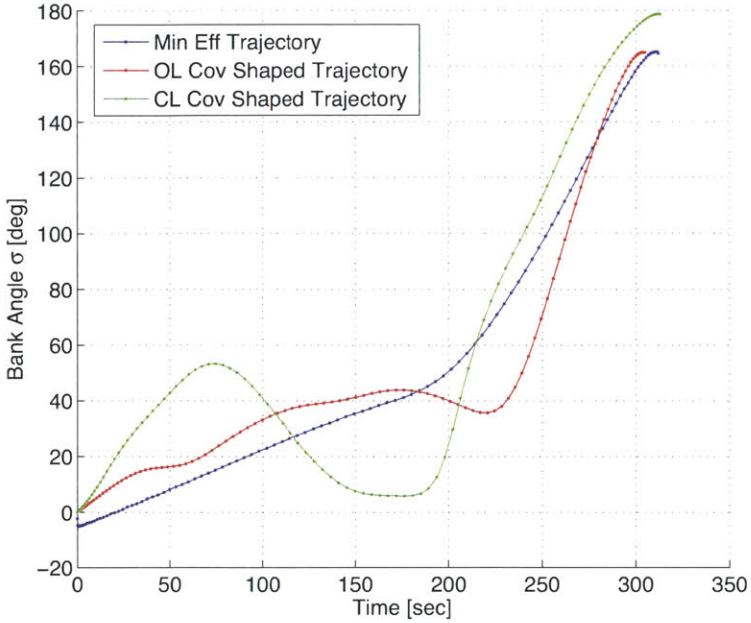

Figure 5-8: Reference Trajectory Bank Angle Histories

errors when no guidance algorithm is used to correct for deviations from the reference trajectory. Next, the closed-loop covariance is examined to how the expected errors change when using the guidance algorithm as well as ensure that the closed-loop covariance shaping technique does improve closed-loop performance. Terminal error ellipses are also generated to provide 3σ uncertainty regions where the vehicle can be expected to finish the trajectory. The linear covariance analysis is conducted using the same set of uncertainties developed in Chapter 4 that are used during the reference trajectory design and it validates that the open-loop and closed-loop covariance shaping techniques function as intended.

5.2.1 Open-Loop Covariance

Figures 5-9 through 5-12 show the open-loop linear covariance along each trajectory, indicating the expected performance of the vehicle when flown using only the nominal controls from the reference trajectory and with no control corrections are made to correct deviations from the reference. This provides a view of how the different shap-

ing techniques influence the expected open-loop errors. Figures 5-9 and 5-10 show the 3σ expected error in the longitude and geodetic latitude as functions of time. As expected, the open-loop covariance shaping technique provides smaller open-loop expected terminal errors in longitude than the minimum effort technique, showing that in open-loop conditions, shaping the trajectory should reduce the terminal error. However, in geodetic latitude as seen in Figure 5-10, the error for the open-loop covariance shaped trajectory was about the same as the minimum effort trajectory. This results from the fact that the shaping technique penalizes the sum of the variances for longitude and latitude, which means that the error in both states will not necessarily be reduced so long as the error in one state is significantly smaller. It can also be seen the expected errors for the open-loop covariance shaped trajectory are larger than errors from the minimum effort trajectory for portions of the flight. For the shaping techniques used here, where only the terminal error is penalized, errors along the trajectory can become quite large. Depending on mission requirements, the shaping technique could be amended to penalize error along the entire trajectory if that was an important consideration.

The closed-loop covariance shaping technique did not generate a trajectory that reduces expected open-loop errors. In fact, in both longitude and latitude error, the closed-loop covariance shaped trajectory has substantially worse open-loop performance than both the minimum effort trajectory and the open-loop covariance shaped trajectory. What this indicates is that in order to reduce closed-loop covariance, the closed-loop covariance shaping technique relies heavily on the performance of the guidance algorithm. For this trajectory in a situation where the guidance system on the vehicle fails, simply flying the remainder of the trajectory open-loop using the nominal controls would provide poor performance. This indicates that understanding the reliance on guidance capabilities by a closed-loop covariance shaped trajectory should be an important consideration in developing contingency plans for guidance failures.

Figure 5-11 provides a representation of the nominal ground track of each trajectory along with the associated terminal 3σ open-loop error ellipses. Figure 5-12

provides a closer comparison of the relative sizes of each of the ellipses, showing that the open-loop covariance shaped trajectory should have the smallest position error spread in open-loop conditions, while the closed-loop covariance shaped trajectory has the largest position error ellipse.

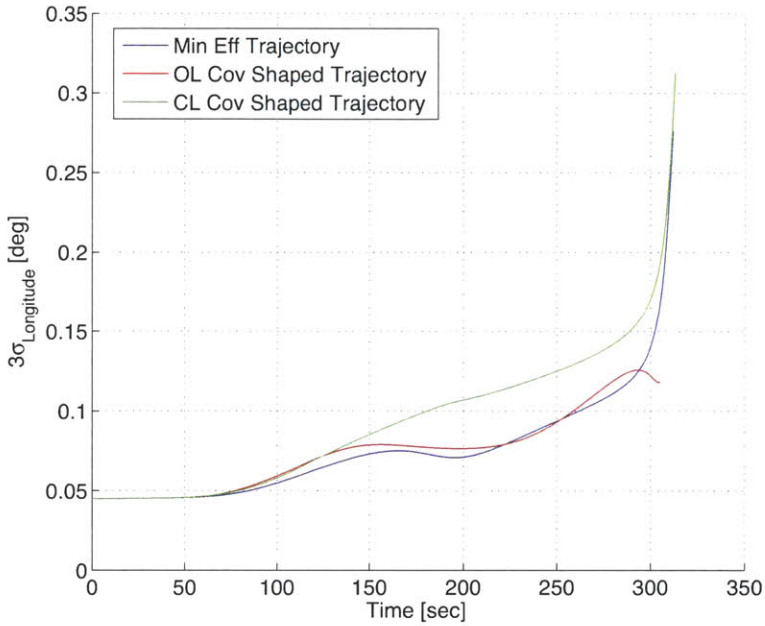

Figure 5-9: Open-loop Covariance - 3σ Longitude Error

5.2.2 Closed-loop Covariance

The closed-loop covariance, shown in Figures 5-13 through 5-16, shows significant reductions in error from the open-loop covariance for all three trajectories, as is to be expected in the presence of a feedback guidance law. While the performance of the guidance law is important, first it must be determined whether covariance shaping techniques are able to provide improved closed-loop performance over the minimum effort trajectory. The 3σ expected closed-loop error in longitude seen in Figure 5-13 shows that the minimum effort trajectory has better performance than both the open-loop covariance and the closed-loop covariance shaped trajectories. However, looking at the expected error in latitude in Figure 5-14, the closed-loop

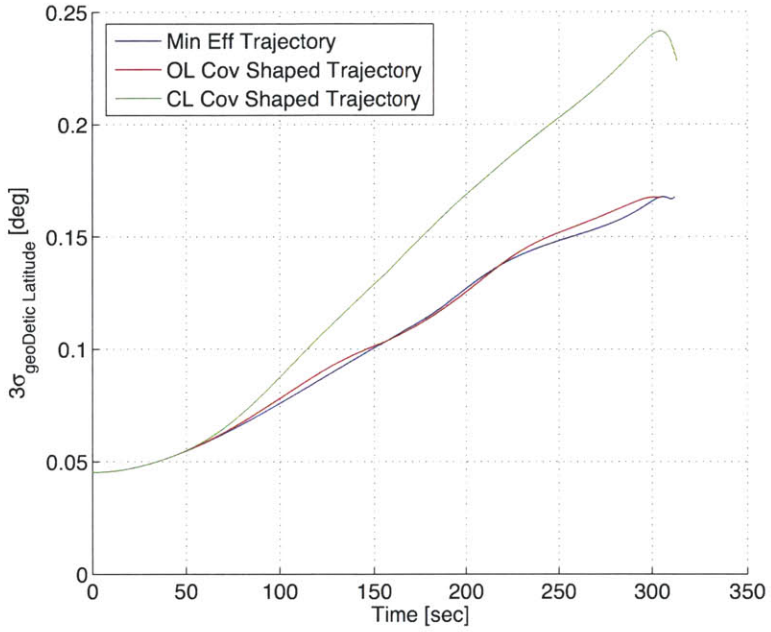

Figure 5-10: Open-loop Covariance - 3σ Geodetic Latitude Error

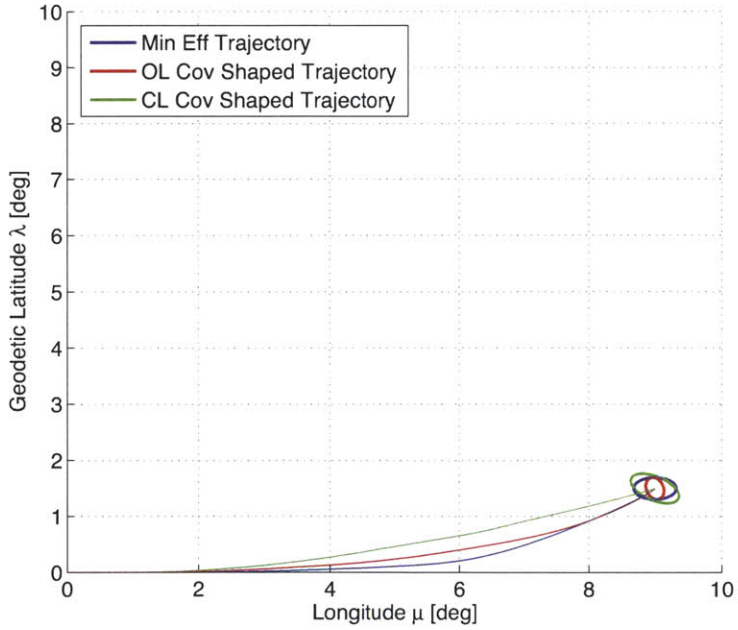

Figure 5-11: Open-loop Covariance - 3σ Position Error Ellipses - Full Trajectory

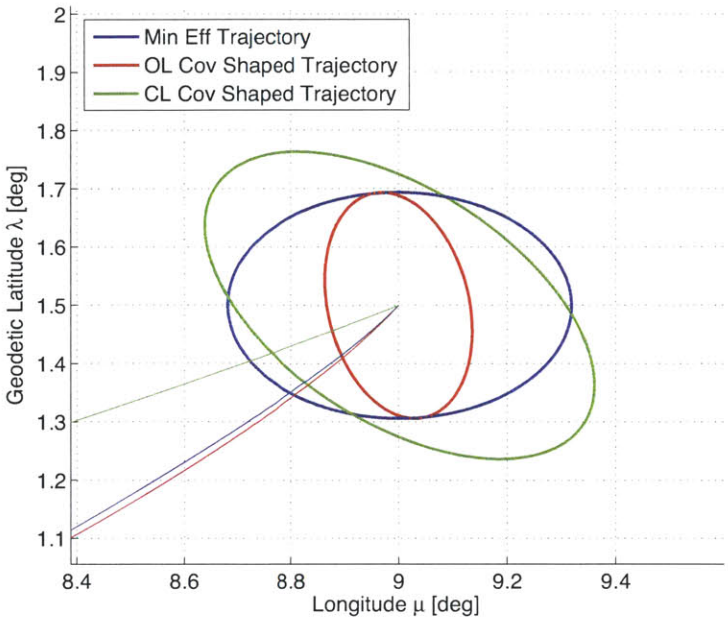

Figure 5-12: Open-loop Covariance - 3σ Position Error Ellipses At Target

covariance shaped trajectory has a significantly smaller error than both of the other trajectories. Comparing the actual values, the minimum effort trajectory had a 3σ longitude error of 0.0450 deg while the closed-loop covariance shaped trajectory had a 3σ error of 0.0540 deg, for 0.009 deg or 19.93% worse performance. In geodetic latitude, the minimum effort trajectory had a 3σ error of 0.0529 deg while the closed-loop covariance shaped trajectory had a 3σ error of 0.0364 deg, for 0.017 deg or 31.31% better performance. Even though the longitude error does increase, by the definition of the cost function used, the linear covariance analysis indicates that there is an expected net improvement in performance for the closed-loop covariance shaped trajectory over the minimum effort trajectory.

This is not true for the open-loop covariance shaped trajectory. Figures 5-13 and 5-14 demonstrate larger closed-loop errors in both longitude and geodetic latitude for this trajectory over the minimum effort trajectory. This demonstrates that when using a feedback guidance law, the open-loop covariance is not necessarily a good proxy metric for measuring sensitivity to uncertainty, and using it as such during

trajectory optimization can lead to worse expected results when the guidance law is applied. Including the guidance law in the covariance calculations to more directly quantify how uncertainties are affecting the trajectory is therefore important when using covariance shaping techniques. Figures 5-15 and 5-16 show the 3σ closed-loop error ellipses along with the nominal ground tracks of the three trajectories, demonstrating the reduced size of the error ellipse for the closed-loop covariance shaped trajectory.

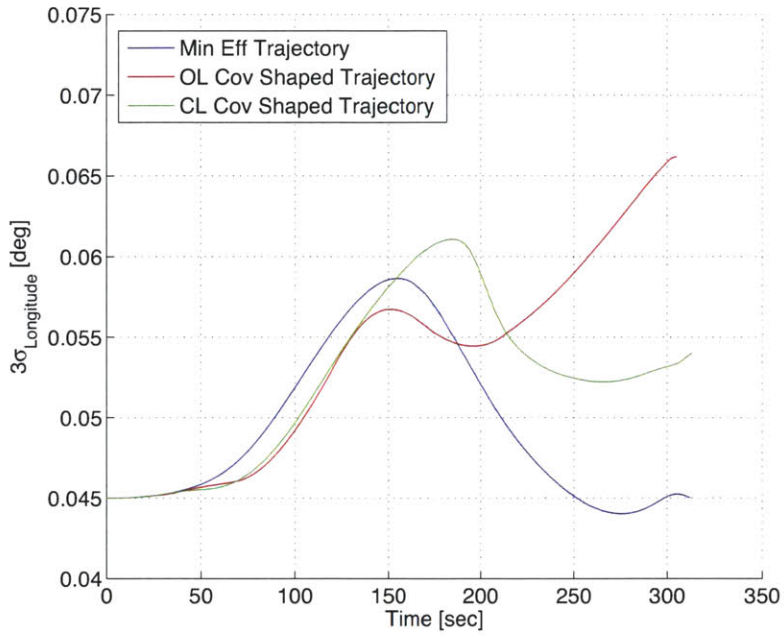

Figure 5-13: Closed-loop Covariance - 3σ Longitude Error

To provide a better understanding of how the guidance law impacts expected errors, Figures 5-17 and 5-18 show a comparison of the open-loop and closed-loop covariances for the closed-loop shaped trajectory. The errors in longitude and geodetic latitude grow quickly in the open-loop case, while the guidance law is able to prevent large increases in error when the vehicle is flown closed-loop. Figure 5-19 shows the difference in magnitude of the terminal 3σ error ellipse for both cases.

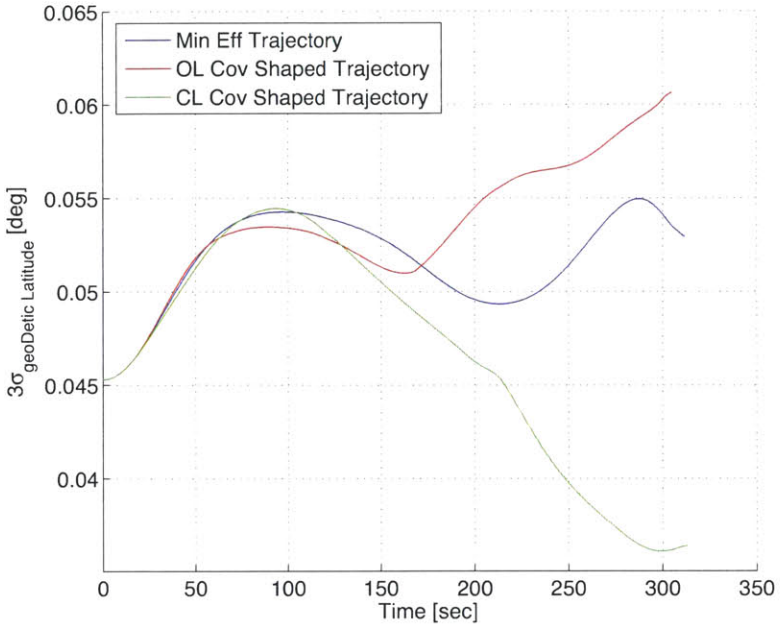

Figure 5-14: Closed-loop Covariance - 3σ Geodetic Latitude Error

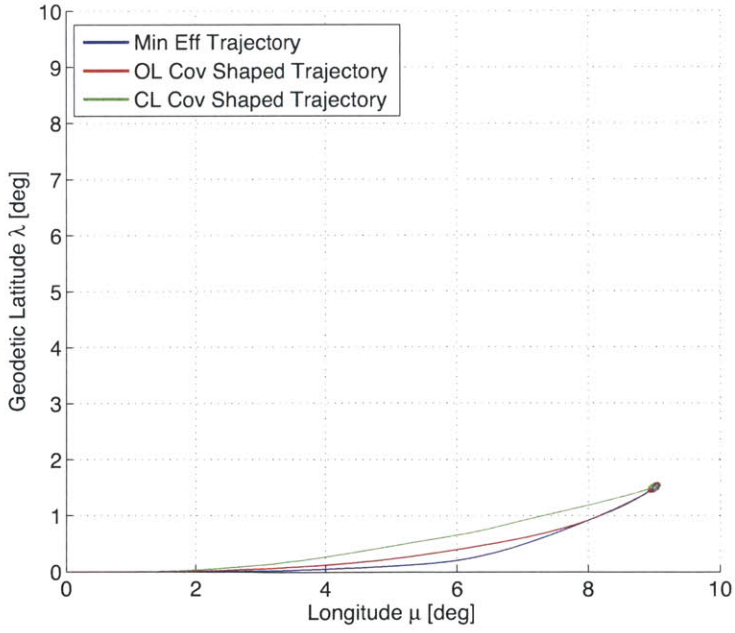

Figure 5-15: Closed-loop Covariance - 3σ Position Error Ellipses - Full Trajectory

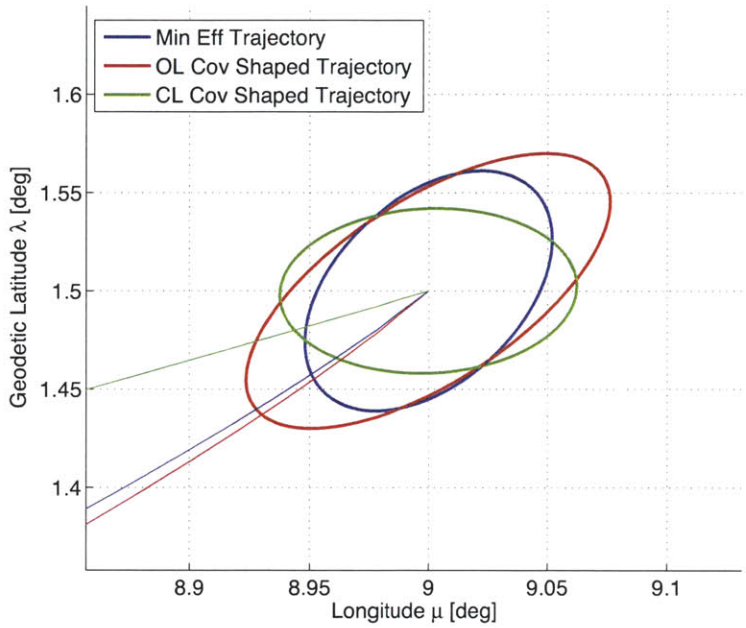

Figure 5-16: Closed-loop Covariance - 3σ Position Error Ellipses At Target

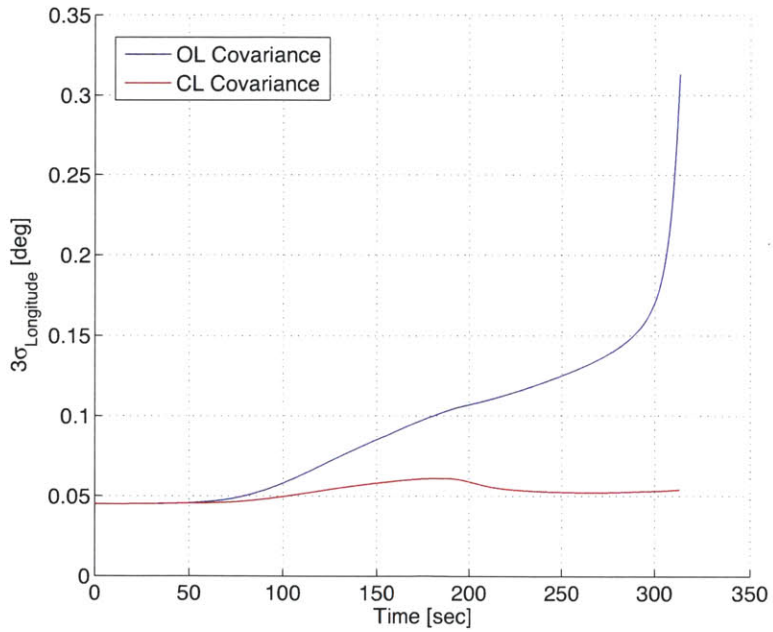

Figure 5-17: Open-loop vs. Closed-loop Covariance - 3σ Longitude Error

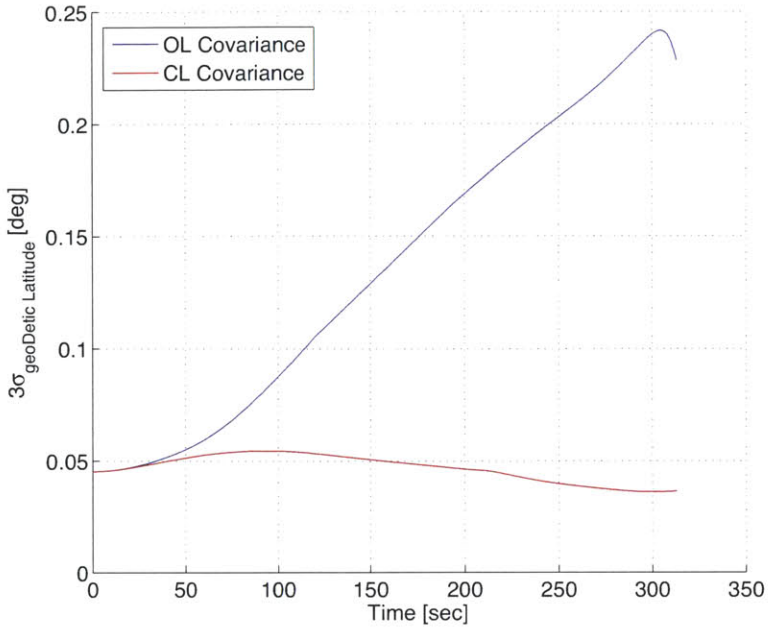

Figure 5-18: Open-loop vs. Closed-loop Covariance - 3σ Geodetic Latitude Error

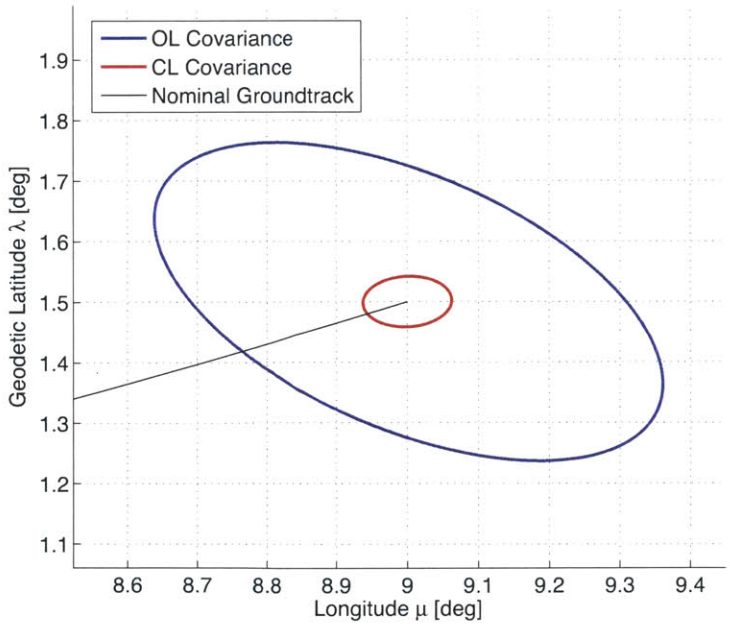

Figure 5-19: Open-loop vs. Closed-loop Covariance - 3σ Position Error Ellipses At Target

5.3 Monte Carlo Simulations

While the linear covariance analysis has shown that shaping techniques can be used to reduce the expected error for a given trajectory, this is only demonstrated using linearized dynamics. When the nonlinear dynamics of the system are linearized, effects of those nonlinearities can be masked during the covariance analysis. To validate that the linear covariance analysis does provide an acceptable representation of expected errors, Monte Carlo (MC) simulations are used with the nonlinear dynamics to generate expected errors from the nonlinear system model. These simulations are conducted using the error models developed in Chapter 4, which fall into the categories of insertion errors, process noise, density uncertainty and aerodynamic coefficient uncertainty. Each of these error types is tested individually with Monte Carlo simulations and compared with linear covariance analysis results for validation as well as to understand the contribution of each type of uncertainty to the errors in the system. Then all of the errors are applied simultaneously and tested with a Monte Carlo simulation to provide a final validation for using linear covariance analysis during reference trajectory design.

For validation purposes, the Monte Carlo simulations in the sections are conducted using the closed-loop covariance shaped trajectory. For each type of uncertainty being tested, a simulation consisting of 1000 cases is conducted where in each case, perturbations are applied to the nominal system and the vehicle equations of motion are propagated to the terminal conditions. Deviations from the reference trajectory are computed for each case and the 3σ error distributions for all of the case are computed. These can then be compared with the 3σ error results from the linear covariance analysis for validation as well as show expected errors for the full nonlinear vehicle model.

5.3.1 Insertion Errors

The insertion errors used included longitude and latitude errors, along with errors in the velocity magnitude and direction. These errors only act at the initial condition,

so the guidance algorithm is not required to compensate for any additional source of error over the course of the trajectory. Figures 5-20 and 5-21 show the results of the Monte Carlo simulation for longitude and geodetic latitude errors. The 3σ errors from both the Monte Carlo simulation and the linear covariance analysis are shown for comparison. Figure 5-22 shows the terminal positions from the Monte Carlo simulation as well as the 3σ error ellipses from both the Monte Carlo and covariance analysis, with all three figures show that the expected errors from the covariance analysis provide a good approximation of the error from the Monte Carlo simulation.

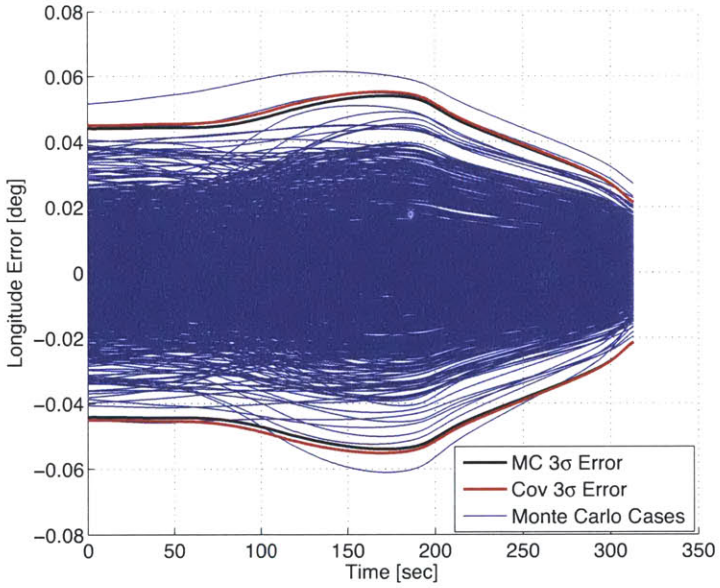

Figure 5-20: Insertion Error MC Simulation - Longitude Dispersions

5.3.2 Process Noise

The process noise model used applies unmodeled dynamics perturbations to the rate equations for each of the velocity states of speed, flight path angle and heading angle. These unmodeled dynamics are applied along the entire trajectory, so the guidance law is required to constantly compensate for new errors. Figures 5-23 and 5-24 show the results of the Monte Carlo simulation for longitude and geodetic latitude errors

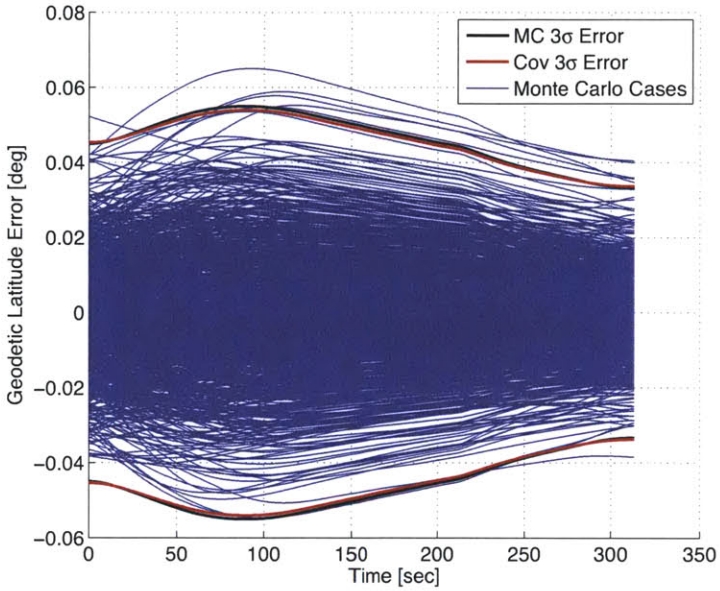

Figure 5-21: Insertion Error MC Simulation - Geodetic Latitude Dispersions

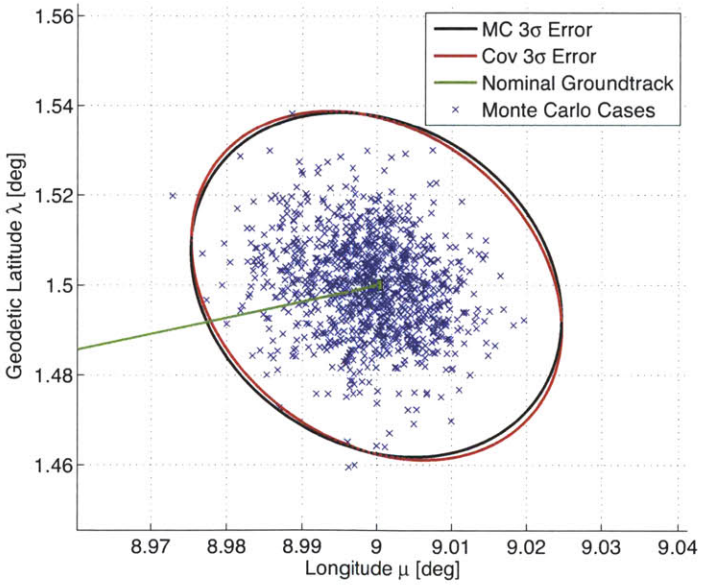

Figure 5-22: Insertion Error MC Simulation - Terminal Position Dispersions

and Figure 5-25 shows the terminal results.

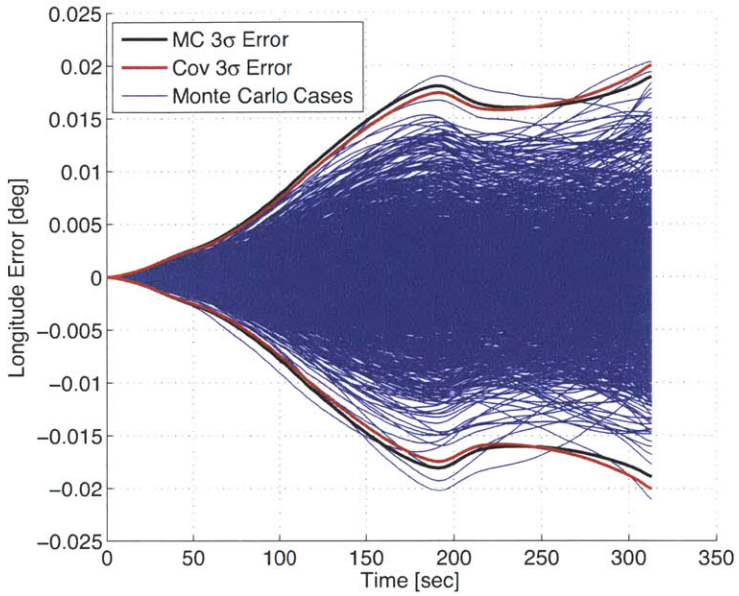

Figure 5-23: Process Noise MC Simulation - Longitude Dispersions

5.3.3 Density Uncertainty

Density uncertainty, described in Subsection 4.3.1, is applied to the model as a parametric uncertainty where the uncertainty profile of density as a function of altitude is known. The Monte Carlo results are shown in Figures 5-26 through 5-28, along with the linear covariance results. The density error is a constant bias for each case, where the nominal density value is perturbed according to the uncertainty profile, which is scaled by a constant random value throughout the trajectory. The main effect of density uncertainty is that it causes differences in the aerodynamic forces of lift and drag, which scale linearly with density. In the reference trajectory, a commanded angle-of-attack has associated lift and drag forces, but if the actual density varies from the nominal value, the lift and drag forces will be different than expected, which results in errors in velocity and position. Figure 5-28 shows the terminal spread from the Monte Carlo simulation lies on a line, such that the minor axis of the error ellipse is zero.

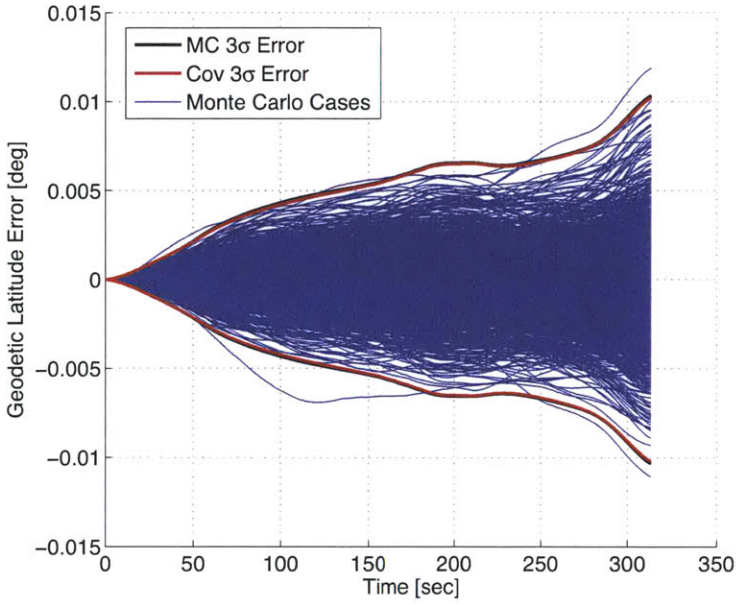

Figure 5-24: Process Noise MC Simulation - Geodetic Latitude Dispersions

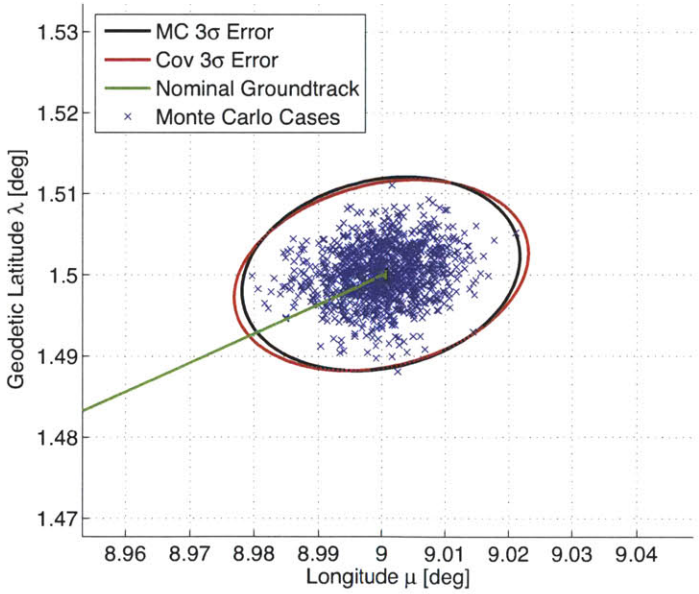

Figure 5-25: Process Noise MC Simulation - Terminal Position Dispersions

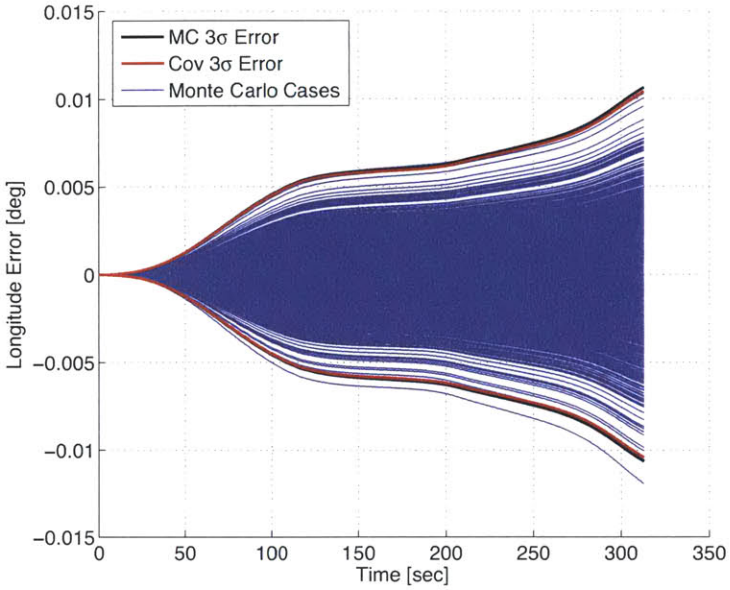

Figure 5-26: Density Uncertainty MC Simulation - Longitude Dispersions

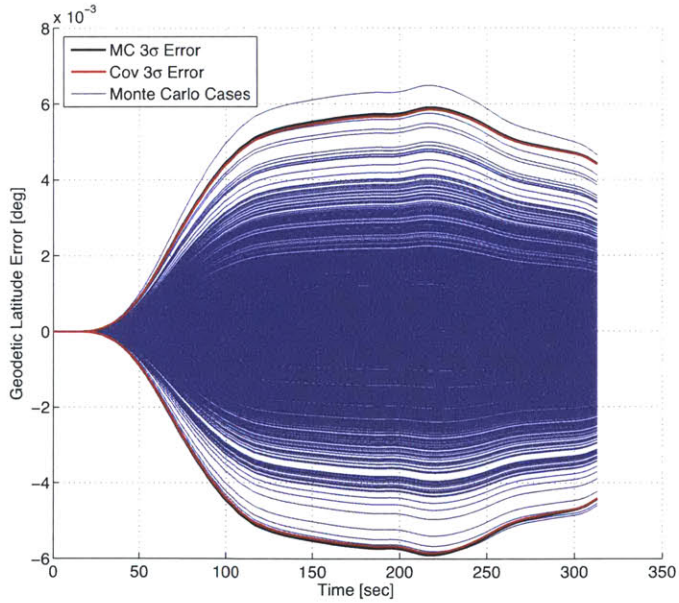

Figure 5-27: Density Uncertainty MC Simulation - Geodetic Latitude Dispersions

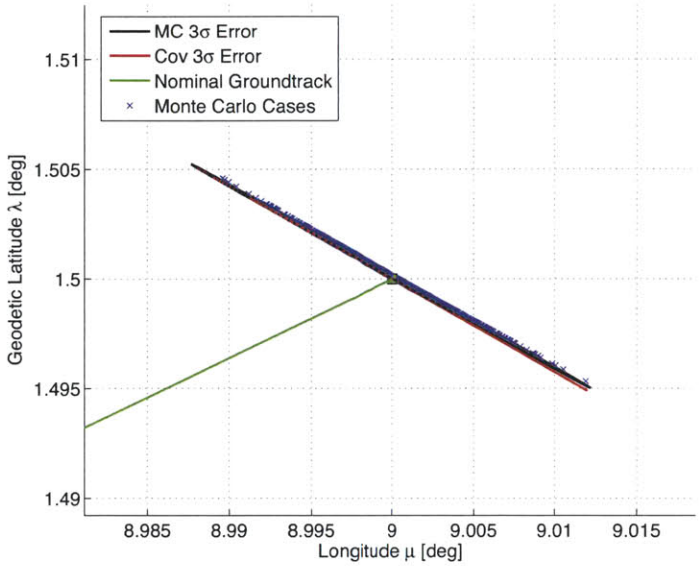

Figure 5-28: Density Uncertainty MC Simulation - Terminal Position Dispersions

5.3.4 Aerodynamic Coefficient Uncertainty

The final set of uncertainties applied is parametric uncertainties for the aerodynamic coefficients for the normal and axial forces, described in Subsection 4.3.1. These uncertainties were implemented similarly to the density uncertainty, but with a constant uncertainty along the trajectory that is scaled randomly for each case. Aerodynamic coefficient uncertainties have a similar effect as density uncertainty in that they cause differences in the actual versus expected aerodynamic forces. The guidance algorithm must then generate command corrections to compensate for errors induced by these uncertainties. Figures 5-29 through 5-31 show the effects of these uncertainties, along with the linear covariance analysis results.

5.3.5 Combined Uncertainties

Taking all of these error sources, a final Monte Carlo simulation is conducted to ensure that the linear covariance analysis provides a good approximation for expected errors from all uncertainty sources. The output from the combined error Monte Carlo simulation is shown in Figures 5-32 through 5-34, demonstrating that the covariance

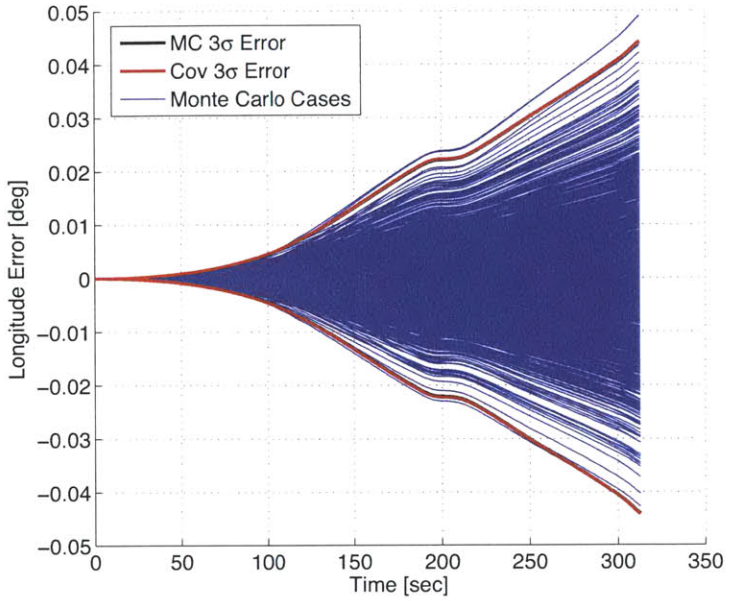

Figure 5-29: Aerodynamic Coefficient Uncertainty MC Simulation - Longitude Dispersions

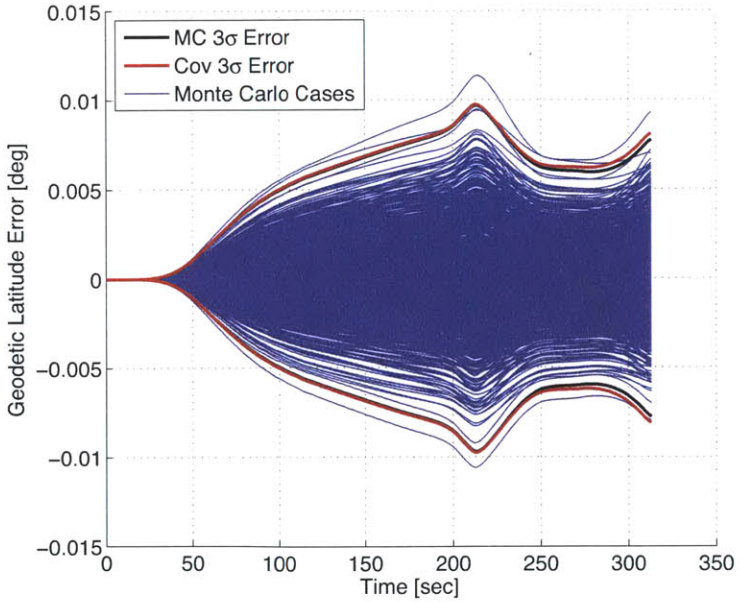

Figure 5-30: Aerodynamic Coefficient Uncertainty MC Simulation - Geodetic Latitude Dispersions

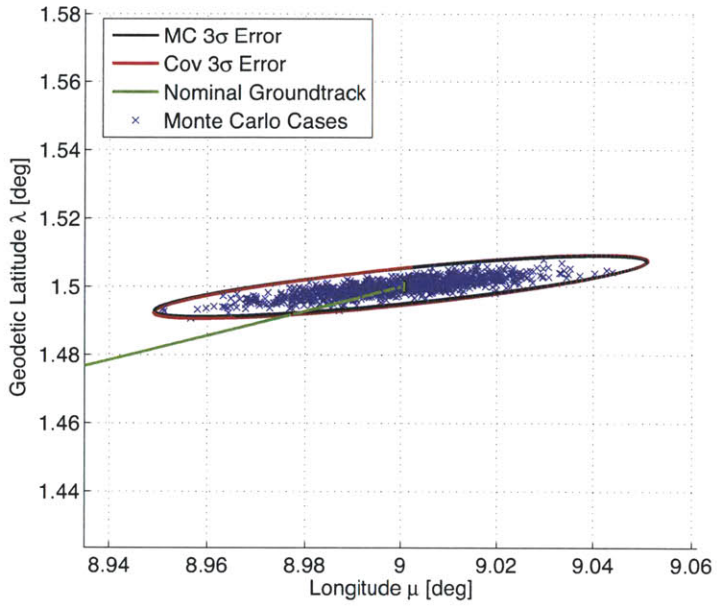

Figure 5-31: Aerodynamic Coefficient Uncertainty MC Simulation - Terminal Position Dispersions

results approximate the Monte Carlo results. The insertion errors account for all of the initial error in Figures 5-26 and 5-27, but the contributions of each type of uncertainty to errors along the trajectory should be considered. The shape of the trajectory itself also impacts how these uncertainties induce errors. The majority of the flight consists of the vehicle moving across lines of longitude rather than lines of latitude. Because of this, individually, density and aerodynamic coefficient uncertainty cause much smaller latitude errors than longitude errors. Both of these uncertainty types cause changes in the aerodynamic forces acting on the vehicle, which will cause error mostly in the direction the vehicle is traveling, which is longitudinally for most of the flight. However, insertion errors and process noise cause more comparable amounts of longitude and latitude error, which occurs because these uncertainties affect the directional components of the velocity vector. Even if the vehicle is only traveling longitudinally on the nominal trajectory, these perturbations can change the velocity direction to induce latitude errors as well.

It is also interesting that the sum of all the terminal errors for longitude and

latitude from each of the individual error sources is greater than the terminal errors for the combined error tests. This means that the effects of the various types of uncertainty do not necessarily constructively cause greater errors. It is also possible that the combined effects of some of the uncertainties cancels out some of the error induced when examined individually. The combined uncertainties serve to model a variety of typical disturbances encountered during reentry and these Monte Carlo simulations demonstrate that the guidance law used is capable of reducing dispersions and that the linear covariance analysis is an appropriate metric for measuring a trajectories response to uncertainty for use during trajectory optimization.

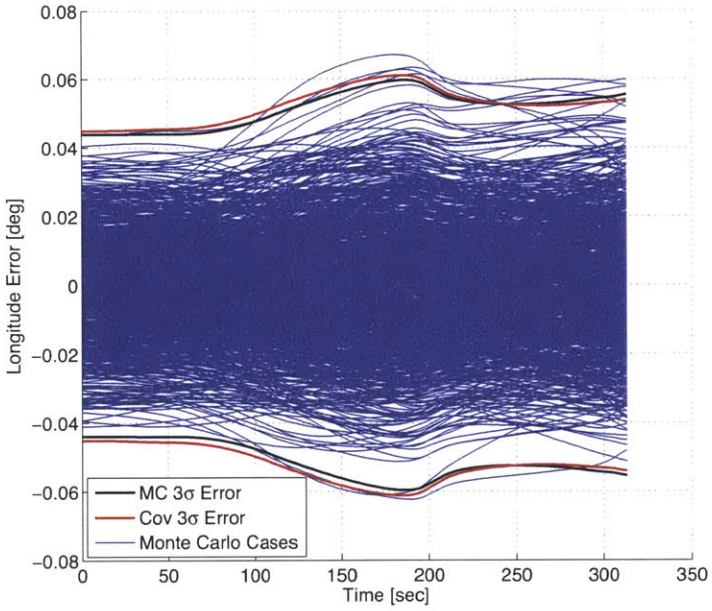

Figure 5-32: Combined Uncertainties MC Simulation - Longitude Dispersions

5.4 Mission Planning Performance

Monte Carlo simulations were also conducted using the combined uncertainty sources for the minimum effort and open-loop covariance shaped trajectories. This data allows for a comparison of the expected performance improvements predicted by linear covariance analysis with the actual results from the Monte Carlo simulations. The

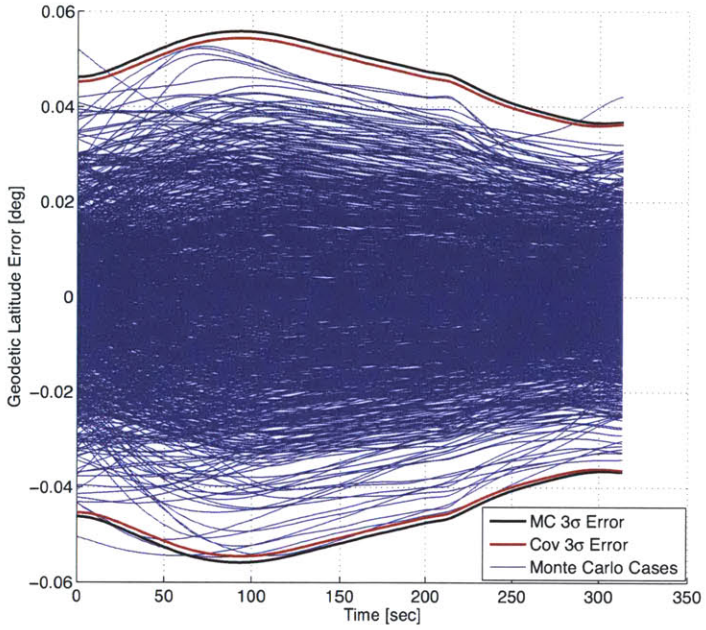

Figure 5-33: Combined Uncertainties MC Simulation - Geodetic Latitude Dispersions

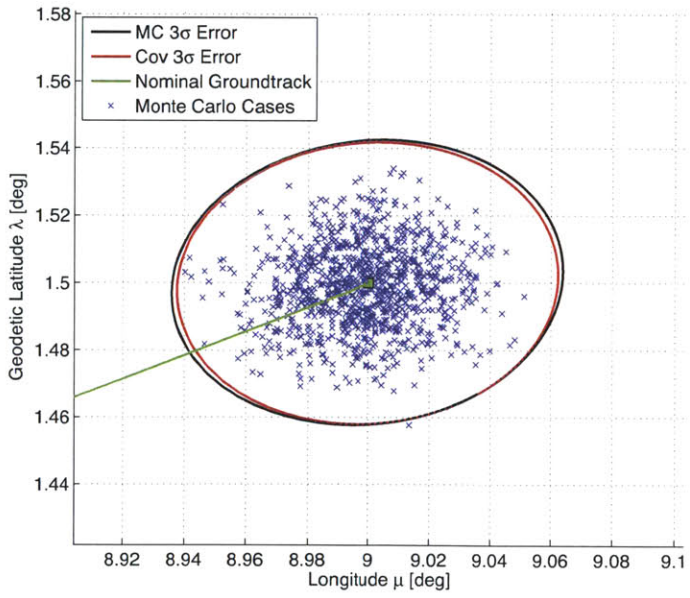

Figure 5-34: Combined Uncertainties MC Simulation - Terminal Position Dispersions

Table 5.1: MC and Covariance 3σ Error - Longitude (deg)

Minimum Effort			Open-Loop Shaped			Closed-Loop Shaped		
Cov	MC	% Diff	Cov	MC	% Diff	Cov	MC	% Diff
0.0450	0.0467	3.49	0.0662	0.0676	2.15	0.0540	0.0555	2.62
% Improvement from minEff			-46.95	-44.98		-19.93	-18.89	

design goals in this study focused on reducing error in longitude and latitude to minimize the overall position error through the use of covariance shaping techniques. The linear covariance analysis provided initial indications that shaping using the closed-loop covariance as a metric for quantifying a trajectories sensitivity to uncertainty was a valid method for reducing expected terminal position errors. With the Monte Carlo simulations, this claim can be validated and the performance of the various shaping techniques can be compared.

5.4.1 Position Errors

The results from the closed-loop linear covariance analysis given in Section 5.2 showed the expected performance of each of the three trajectory shaping methods. Monte Carlo simulations provide a verification of that performance so that the shaping techniques can be compared. The minimum effort trajectory serves as a baseline for comparison to determine if open-loop and closed-loop covariance shaping are viable means for reducing sensitivity to uncertainty. Table 5.1 provides results from the linear covariance analysis and Monte Carlo simulations for each trajectory for longitude error. The percent differences between the covariance and Monte Carlo results is provided along with the percent improvement of the covariance shaping methods over the minimum effort trajectory. Both the closed-loop and open-loop covariance shaped trajectories performed worse in longitude error than the minimum effort trajectory, with the open-loop covariance shaped trajectory having a 44.98% larger 3σ error in the Monte Carlo simulation.

Table 5.2 shows the same information for geodetic latitude errors. While the open-loop covariance shaped trajectory still performs worse than the minimum effort trajectory, the closed-loop covariance shaped trajectory shows substantial improvement, with a 31.22% smaller 3σ error than the minimum effort trajectory. This improvement offsets the increase in longitude error that the closed-loop covariance shaped trajectory saw and shows that closed-loop covariance shaping is effective at reducing the sensitivity of a trajectory to uncertainty. The linear covariance analysis and Monte Carlo simulation results for longitude and latitude errors agree with small percent differences across all of the trajectories, providing further validation of the closed-loop covariance shaping technique. These results also further support the fact that open-loop covariance is not only a poor proxy for quantifying trajectory sensitivity to uncertainty, using it as such a metric with a feedback guidance law can produce a trajectory that is more sensitive to uncertainty. This shows that the addition of a guidance law changes the behavior of the vehicle sufficiently that it must be considered to obtain better performance results by covariance shaping.

While these initial results does not show drastic improvements in performance, they prove that closed-loop covariance shaping is a viable method for reducing the sensitivity that a trajectory has to uncertainty. As uncertainty will always be a factor during mission planning and reference trajectory design, it is important that the impact of uncertainty can be mitigated. The simple shaping method used in this study serves as a baseline method that can be modified and tailored to improve performance in mission specific ways. Instead of just penalizing terminal position errors, deviations in all states could be minimized, or error along the trajectory could be reduced if proper path following is critical. Any function of the states errors can be formulated and used by the mission planner to generate reference trajectories that have reduced sensitivity to the effects of uncertainty.

5.4.2 Velocity Errors

One final aspect of these reference trajectories that has not yet been examined is the error in the velocity states of speed, flight path angle and heading angle. The shaping

Table 5.2: MC and Covariance 3σ Error - Geodetic Latitude (deg)

Minimum Effort			Open-Loop Shaped			Closed-Loop Shaped		
Cov	MC	% Diff	Cov	MC	% Diff	Cov	MC	% Diff
0.0529	0.0534	0.87	0.0606	0.0605	0.31	0.0364	0.0367	1.00
% Improvement from minEff			-14.57	-13.22		31.31	31.22	

techniques used did not actively seek to minimize errors in these states, but it is often important for mission requirements that terminal errors for velocity states be small. To demonstrate that the shaping techniques used in this study can be adapted to other design goals such as reducing velocity state errors, the linear covariance of speed, flight path angle, and heading angle are calculated and compared to errors in those states obtained from Monte Carlo simulations.

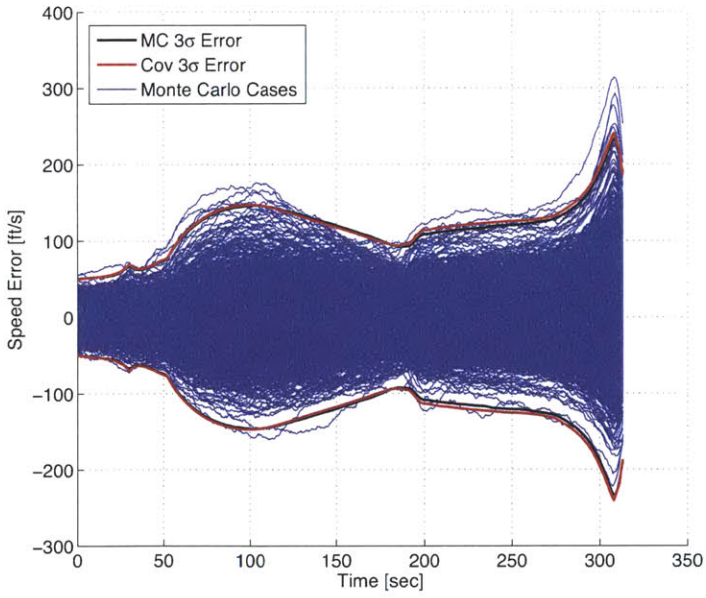

Figure 5-35: Combined Uncertainties MC Simulation - Speed Dispersions

Figures 5-35 through 5-37 show the Monte Carlo results and linear covariance analysis 3σ errors for speed, flight path angle and heading angle for the closed-loop covariance shaped trajectory. Tables 5.3 through 5.5 show the differences between

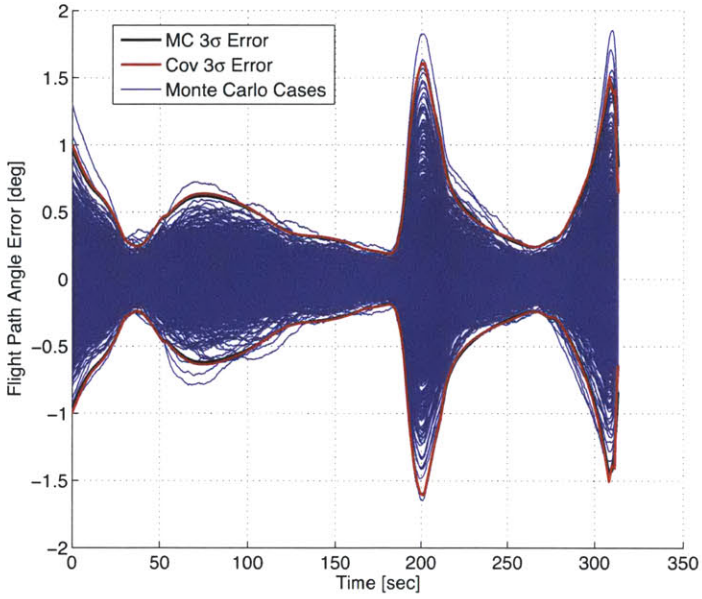

Figure 5-36: Combined Uncertainties MC Simulation - Flight Path Angle Dispersions

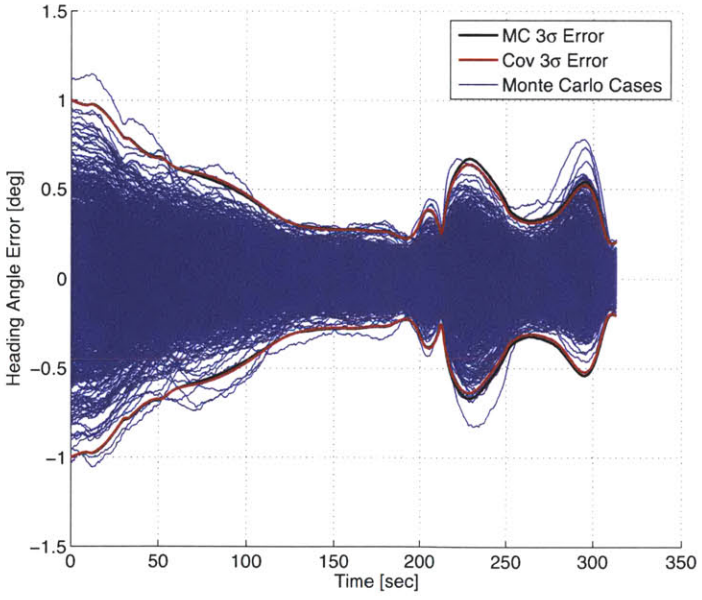

Figure 5-37: Combined Uncertainties MC Simulation - Heading Angle Dispersions

Table 5.3: MC and Covariance 3σ Error - Speed (ft/sec)

Minimum Effort			Open-Loop Shaped			Closed-Loop Shaped		
Cov	MC	% Diff	Cov	MC	% Diff	Cov	MC	% Diff
128.44	129.50	0.82	151.19	150.35	0.55	187.03	189.31	1.22
% Improvement from minEff			-17.71	-16.10		-45.61	-46.19	

Table 5.4: MC and Covariance 3σ Error - Flight Path Angle (deg)

Minimum Effort			Open-Loop Shaped			Closed-Loop Shaped		
Cov	MC	% Diff	Cov	MC	% Diff	Cov	MC	% Diff
0.72	0.81	11.82	0.54	0.52	3.69	0.64	0.84	26.60
% Improvement from minEff			25.01	35.80		10.48	-3.93	

the terminal velocity state errors from the covariance analysis and the Monte Carlo simulations as well as comparisons between the open-loop and closed-loop shaped trajectories and the minimum effort trajectory.

For the speed and heading angle errors, the linear covariance analysis and the Monte Carlo simulation results match with small percent differences, as did the position error results. However, the flight path angle has somewhat larger discrepancies in 3σ errors between the two simulation methods. There is an 11.83% difference for the minimum effort trajectory as well as a 26.60% difference for the closed-loop covari-

Table 5.5: MC and Covariance 3σ Error - Heading Angle (deg)

Minimum Effort			Open-Loop Shaped			Closed-Loop Shaped		
Cov	MC	% Diff	Cov	MC	% Diff	Cov	MC	% Diff
0.28	0.26	6.72	0.28	0.26	6.01	0.21	0.21	0.62
% Improvement from minEff			-1.23	-1.94		25.97	20.33	

ance shaped trajectory. The percent difference for the open-loop covariance shaped trajectory is still small at 3.69%. However, it was expected that the result from the covariance analysis, which use the linearized dynamics would not produce exactly the same results as the Monte Carlo simulation. In the velocity state rate equations, there are more nonlinearities introduced than in the position rate equations due to the inclusion of the aerodynamic forces. These forces change nonlinearly with the angle-of-attack, which itself can change rapidly. These nonlinearities then have the effect of producing behaviors that are not captured in the linear covariance analysis. Also, these discrepancies do not mean that the linear covariance of the velocity states cannot be used as part of a shaping cost function. The speed and heading angle errors match well with Monte Carlo results and even in cases where differences between covariance analysis and Monte Carlo simulation exist, the reducing the covariance of a state could still improve performance of the full nonlinear system.

Chapter 6

Conclusions and Future Work

In this thesis, covariance shaping was implemented as a tool in reference trajectory design for atmospheric reentry vehicles to reduce the sensitivity of trajectories to uncertainties. Both open-loop covariance and closed-loop covariance shaped trajectories were generated, along with a minimum effort trajectory created with nominal techniques to serve as a comparison baseline. These three trajectories all met the defined mission requirements but are shaped to minimize their respective performance metrics.

Linear covariance analysis was conducted on each of the trajectories to see the expected response of each to uncertainties. This showed that open-loop covariance shaping should result in smaller open-loop covariance when only nominal controls are used. When flown using a guidance algorithm, however, the open-loop covariance shaped trajectory resulted in larger terminal position errors than the minimum effort trajectory. The closed-loop covariance shaped trajectory was expected to have the smallest closed-loop error of the three trajectories.

To validate these results, Monte Carlo simulations were used. First, the contributions of insertion errors, process noise, density uncertainty and aerodynamic coefficient uncertainty on trajectory errors were examined individually to ensure that the linear covariance results matched the Monte Carlo simulation results for each uncertainty type. Finally, Monte Carlo simulations were run using all uncertainty types for each trajectory to determine if the performance predicted by covariance analysis

was matched when simulated using the nonlinear dynamics. These tests show that linear covariance analysis provides an accurate representation of expected errors and can be used as a proxy to reduce the errors seen in the Monte Carlo simulations.

In examining all three reference trajectories developed, the results from the linear covariance analysis and Monte Carlo simulations agree that closed-loop covariance shaping is an effective method for reducing the sensitivity of a trajectory to uncertainty. This study demonstrates that expected terminal position errors could be reduced but the methodology used can be applied to minimize errors in any state variable at any point along the trajectory. Additionally, this testing revealed that using the open-loop covariance to shape a trajectory can result in increased sensitivity to uncertainty and larger expected errors when the trajectory is flown with a feedback guidance law. Taking the performance of the guidance law into account, the closed-loop covariance shaping method is shown to be a promising way to improve trajectory performance during the trajectory design process.

6.1 Future Work

6.1.1 Navigation System

When considering uncertainty during trajectory design, state uncertainties are not the only factors that should be considered. The navigation system on a reentry vehicle can also be a source of noise that can greatly contribute to errors. The benefit of using linear covariance analysis as a method for shaping trajectories is that it can be easily modified to model a navigation system, where noise and errors can affect navigation sensors. The covariance dynamics can be easily extended to include sensor dynamics and noise so that the covariance matrix includes effects from these sources. The covariance shaping techniques then function in the same way as the current formulation, adjusting the trajectory to minimize expected errors, so that the trajectory is now shaped in a way to minimize the effects of uncertainty on sensors in the navigation system as well as state uncertainties.

6.1.2 Guidance Refinement

The guidance algorithm used in this study was an LQR scheme that was chosen for its simplicity of implementation while providing fairly good tracking capabilities. The performance of the guidance algorithm was not a major point of concern because the hypothetical nature of the mission studied did not necessitate strict performance requirements. However, to further improve performance, there are several modifications that could be made to the guidance design. First, as noted in [2] and [19], the use of time as the independent variable for LQR guidance does not generally provide the best results. Instead, the system dynamics could be rewritten in terms of another variable, such as energy, which is monotonically decreasing. Additional robustness could also be achieved through the introduction of integral states [4]. The accumulated error in vehicle states can then be penalized, providing better performance.

Beyond simply modifying the current LQR guidance algorithm, there are a variety of other types of guidance algorithms and analysis techniques available to create a more effective guidance law. While LQR techniques can generally provide system stability, other performance characteristics such as response rate and accuracy can only be designed indirectly through the state and control weighting matrices. With pole placement techniques, along with Bode analysis, the performance capabilities of a guidance law can be more exactly designed so that mission specifications are met. Regardless of the guidance law chosen, however, it will have to be tailored depending on the vehicle and mission requirements.

6.1.3 6 DOF Modeling

Finally, in both the optimization modeling and the Monte Carlo simulations, 3 DOF dynamics were used to describe the motion of the reentry vehicle, while the vehicle attitude was used to define control inputs. In reality, the attitude of the vehicle is also defined by its own set of dynamics and it controlled by vehicle control surfaces, such as flaps. While 3 DOF dynamics are simpler to use and computationally faster for simulation, full 6 DOF dynamics that include both translational and rotational

motion provide a much more realistic model of vehicle motion. To ensure acceptable real-world performance, 6 DOF simulation should at least be used for Monte Carlo simulation. Reference trajectory design and linear covariance analysis could initially be conducted using the original 3 DOF models, but then 6 DOF Monte Carlo simulations could be used to validate that the 3 DOF representation provides an accurate enough model. If it does not, then the reference trajectory design process and linear covariance analysis might have to be altered to include 6 DOF dynamics as well. This would add a significant computational burden to the design process and could drastically lengthen design times.

Bibliography

[1] K. P. Bollino, High-Fidelity Real-Time Trajectory Optimization For Reusable Launch Vehicles, Ph.D. Thesis, Naval Postgraduate School, 2006.

[2] G. A. Dukeman, "Profile-Following Entry Guidance Using Linear Quadratic Regulator Theory," AIAA Paper 2002-4457, 2002.

[3] P. Lu, "Regulation About Time-Varying Trajectories: Precision Entry Guidance Illustrated," *Journal of Guidance, Control, and Dynamics*, 22(6):784-790, 1999.

[4] C. C. Tracy, "Integrated Entry Guidance and Control for Autonomous Reusable Launch Vehicles," Master's Thesis, MIT Department of Aeronautics and Astronautics, 1999.

[5] S. Bairstow, "Reentry Guidance with Extended Range Capability for Low L/D Spacecraft," Master's Thesis, MIT Department of Aeronautics and Astronautics, 2006.

[6] J. C. Harpold and D. E. Gavert, "Space Shuttle Entry Guidance Performance Results," *Journal of Guidance, Control, and Dynamics*, 6(6):442-447, 1993.

[7] K. Clarke, "Performance Optimization Study of a Common Aero Vehicle Using a Legendre Pseudospectral Method," Master's Thesis, MIT Department of Aeronautics and Astronautics, 2003.

[8] M. J. Abrahamson, "Boost Through Reentry Trajectory Planning for Maneuvering Reentry Vehicles," Master's Thesis, MIT Department of Aeronautics and Astronautics, 2008.

[9] T. V. Small, "Optimal Trajectory-Shaping with Sensitivity and Covariance Techniques," Master's Thesis, MIT Department of Aeronautics and Astronautics, 2010.

[10] A. Undurti, "Optimal Trajectories for Maneuvering Reentry Vehicles," Master's Thesis, MIT Department of Aeronautics and Astronautics, 2007.

[11] H. Seywald and R. Kumar, "Desensitized Optimal Trajectories," *AIAA/AAS Spaceflight Mechanics Meeting*, AAS Paper 96-107:103-115, 1996.

[12] H. Seywald, "Desensitized Optimal Trajectories with Control Constraints," *AIAA/AAS Spaceflight Mechanics Meeting*, AAS Paper 03-147:737-743, 2003.

[13] D. K. Geller, "Linear Covariance Techniques for Orbital Rendezvous Analysis and Autonomous Onboard Mission Planning," *Journal of Guidance, Control, and Dynamics*, 29(6):1404-1414, 2006.

[14] R. Zanetti, D. C. Woffinden, and A. Sievers, "Multiple Event Triggers in Linear Covariance Analysis for Spacecraft Rendezvous," *Journal of Guidance, Control, and Dynamics*, 35(2):353-366, 2012.

[15] D. R. Vander Stoep, "Trajectory Shaping for the Minimization of State-Variable Estimation Errors," *IEEE Transactions on Automatic Control*, AC-13:284-286, 1968.

[16] S. Zimmer, C. Ocampo, and R. Bishop, "Reducing Orbit Covariance for Continuous Thrust Spacecraft Transfers," *IEEE Transactions on Aerospace and Electronic Systems*, 46:771-791, 2010.

[17] J. T. Betts, "Survey of Numerical Methods for Trajectory Optimization," *Journal of Guidance, Control, and Dynamics*, 21(2):193-207, 1998.

[18] I. M. Ross, "A Beginner's Guide to DIDO," 1998.

[19] A. De Virgilio, G. R. Wells, and E. E. Schiring, "Optimal Guidance for Aerodynamically Controlled Re-Entry Vehicles," *AIAA Journal*, 12(10):1331-1337, 1974.

[20] A. E. Bryson and Y. C. Ho, *Applied Optimal Control: Optimization, Estimation and Control*, Taylor and Francis, 1975.

[21] *1976 U.S. Standard Atmosphere*, U.S. Government Printing Office, Washington D.C., 1976.

[22] C. G. Justus and F. W. Leslie, "NASA/TM2011216467 - The NASA Marshall Space Flight Center Earth Global Reference Atmospheric Model 2010 Version", NASA, 2011.

[23] M. W. Reichelt and L. F. Shampine, "ode45.m," Version 5.74.4.10, MathWorks, Inc., 2009.

[24] N. J. Kasdin, "Runge-Kutta Algorithm for the Numerical Integration of Stochastic Differential Equations," *Journal of Guidance, Control, and Dynamics*, 18(1):114-120, 1995.