

UC Irvine

UC Irvine Electronic Theses and Dissertations

Title

Robust Optimal Entry Guidance for Future Mars Landers

Permalink

<https://escholarship.org/uc/item/7bv3q5ws>

Author

Noyes, Connor David

Publication Date

2021

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA,
IRVINE

Robust Optimal Entry Guidance for Future Mars Landers

DISSERTATION

submitted in partial satisfaction of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

in Mechanical and Aerospace Engineering

by

Connor David Noyes

Dissertation Committee:
Professor Emeritus Kenneth D. Mease, Chair
Distinguished Professor Tryphon Georgiou
Professor Athanasios Sideris

DEDICATION

To my father Michael and my mother Diane.

TABLE OF CONTENTS

	Page
LIST OF FIGURES	v
LIST OF TABLES	viii
ACKNOWLEDGMENTS	ix
VITA	x
ABSTRACT OF THE DISSERTATION	xii
1 Introduction	1
1.1 Motivation	1
1.2 Dissertation Organization	6
1.3 Entry Terminal Point Controller	7
2 Mars Entry Phase Guidance Requirements	10
2.1 Entry Phase Termination	11
3 Entry Vehicle and Environment Models	14
3.1 Three Degree-of-Freedom Dynamics Model	14
3.2 System Models	17
3.2.1 Nominal Models	17
3.2.2 Uncertainty Models	17
4 Guidance Strategy	23
4.1 Independent Variable	23
4.2 Feedback Control Form	25
4.3 Defining the Objective Function	27
4.4 The Robust Optimal Guidance Problem	29
5 Solving the Robust Optimal Guidance Problem	31
5.1 Uncertainty Quantification	31
5.1.1 Linear Covariance	32
5.1.2 Unscented Transform	33
5.1.3 Linear Covariance vs Unscented Transform Example	35
5.2 Differential Dynamic Programming	38

5.2.1	Optimal Design Parameters	42
5.2.2	Second Order Dynamics Derivatives	44
5.2.3	Objective Functional Reformulation	45
5.2.4	Saturation	47
6	Guidance Assessment Conditions	49
6.1	Entry Conditions and Targets	49
6.2	Robust Optimal Control Solution Parameters	50
6.2.1	Selecting the UT Parameter	52
6.3	Simulation	52
7	Guidance Assessment	54
7.1	Reference Trajectory Optimization	54
7.1.1	Open-Loop Optimization	55
7.1.2	Closed-Loop Optimization	56
7.2	Joint Gain Optimization	57
7.3	Detailed Solution	61
7.3.1	Sensitivity Analysis	62
7.4	Convergence Properties	65
7.5	Joint Optimization of Velocity-Varying Gains	69
7.6	Comparison to MSL and Mars 2020	72
8	Sample Retrieval Lander Class Assessment	73
8.1	Open-Loop Simulation	74
8.2	Reference Trajectory Comparison	75
8.3	Standard Deviation Weighting Design Trade	79
8.4	Comparison of UT and Monte Carlo Statistics	80
9	Entry Guidance for SRP-Based EDL	85
9.1	Entry Guidance Problem for SRP-Based EDL	89
9.1.1	Entry Guidance Target Set	90
9.2	Proposed Entry Guidance	94
9.2.1	Bank Angle Parametrization	95
9.2.2	Powered Descent Propellant Mapping	99
9.2.3	Trajectory Prediction and Powered Descent Ignition	100
10	Assessment of Entry Guidance for SRP-Based EDL	102
10.0.1	Sensitivity Analysis	103
10.0.2	Monte Carlo Simulation	107
11	Conclusions	113
11.1	Future Work	114
	Bibliography	117

LIST OF FIGURES

	Page
1.1 The concept of robust optimal control.	4
2.1 Entry Descent and Landing timeline for Mars 2020, taken from [1].	11
2.2 Parachute deployment conditions on Mach number and dynamic pressure in altitude-velocity space.	12
3.1 The nominal atmospheric density.	18
3.2 The nominal aerodynamic coefficients and L/D profile.	19
3.3 Density variations from 1000 Mars Climate Database samples	20
5.1 Sensitivity of statistics to the unscented transform scaling parameter	37
5.2 The saturation function and smooth approximation for several values of the tuning parameter.	48
6.1 The reference control used to initialize DDP.	51
7.1 Monte Carlo statistics for the open-loop trajectory optimization.	55
7.2 Monte Carlo statistics for the closed-loop trajectory optimization with static gains.	56
7.3 Unscented Transform estimation errors relative to Monte Carlo statistics for trajectory optimization with static feedback gains.	57
7.4 Monte Carlo statistics for jointly optimized static feedback gains.	58
7.5 Unscented Transform estimation errors relative to Monte Carlo statistics for jointly optimized static feedback gains.	58
7.6 The optimal target range as a function of the performance weights.	59
7.7 The optimal target range as a function of the performance weights for an unguided vehicle.	60
7.8 The optimal feedback gains as functions of the performance weights.	61
7.9 Sample trajectories and 3σ bounds estimated by MC and UT.	63
7.10 Terminal altitude and range performance.	64
7.11 Total cost per iteration	66
7.12 Total cost per iteration	67
7.13 Maximum control gradient norm over all timesteps.	67
7.14 Backtracking linesearch stepsize per iteration.	68
7.15 Time in seconds spent computing derivatives of the cost and dynamics at each iteration.	68

7.16	Sample trajectories and 3σ bounds estimated by MC and UT for a solution with jointly optimized, velocity-varying gains.	70
7.17	Terminal altitude and range performance for a solution with jointly optimized, velocity-varying gains.	71
8.1	The sigma point trajectories for an unguided entry trajectory.	74
8.2	The profiles from deterministic optimal control (left) exhibit the well known bang-bang structure of altitude optimal trajectories. In comparison, the robust control histories (right) exhibit significantly different behavior depending on the weights applied.	76
8.3	Contours of the Monte Carlo 3σ -low altitude (left) and 3σ downrange distance error (right) as a function of the weights w_h and w_s	79
8.4	Altitude versus velocity for 500 sample trajectories with Monte Carlo- and UT-estimated 3σ bounds. The 3σ -low altitude is 2.5 km.	81
8.5	Range to go versus velocity for 500 sample trajectories with Monte Carlo- and UT-estimated 3σ bounds. The terminal 3σ range error is 1.8 km.	82
8.6	Acceleration load vs velocity for 500 sample trajectories with Monte Carlo- and UT-estimated 3σ bounds.	82
8.7	The reference control and 500 sample trajectories.	83
9.1	Schematic of the EDL phases.	89
9.2	The guidance algorithm seeks both the propellant-optimal ignition point (indicated by a star on the plot) and the trajectory whose optimal ignition point is best (indicated by the solid black line). The red, dotted trajectory is valid entry trajectory within the reachable set, but is infeasible because it does not intersect the powered descent feasible subset $\mathcal{F}_M$	95
9.3	Consider two different parametrizations of the control, P_A and P_B . While P_B allows the vehicle to reach more of the reachable set $\mathcal{R}$ than P_A , much of R_{P_B} does not intersect $\mathcal{F}_M$. Additionally, the propellant optimal point in R_{P_A} is better than the optimal point in R_{P_B} , indicated by the smaller distance between the optima, which are depicted with stars. Note that these sets, depicted notionally as convex 2-D shapes, are actually 6-D and no claim is made about their convexity.	97
9.4	Example contours of the propellant cost of pinpoint landing using two different parametrizations: a constant bank angle with one reversal, and a profile with two reversal velocities. Not only does the choice affect the propellant contours, but the reachable set is quite different as well. The downrange targeted in the constant profile is 700 km and is not reachable by the second parametrization, which instead targets 625 km.	98
9.5	Only the interval of the predicted entry trajectory satisfying constraints on velocity, altitude, and distance to the target are searched for the propellant-optimal ignition state.	101

10.1	The predicted PMF and associated ignition state may change each time the guidance algorithm is called. The solutions are less volatile when the discrepancy between the model predictions and the simulated environment is small. Larger variations in the solution due to variations in h_s are expected since they result in unknown future density variations.	104
10.2	The bank angle profiles for a series of 1-D parametric sensitivities. Despite their similar effect on drag, the controller deals with $\pm C_D$ quite differently than $\pm \rho$	105
10.3	Propellant mass fraction for the Monte Carlo samples.	107
10.4	Bank angle profiles resulting from calls to the guidance algorithm at $V = [5490, 4500, 3000, 2000, 1000]$ m/s.	107
10.5	All of the trajectories feature lofting as a means to further decelerate prior to ignition.	108
10.6	Downrange and crossrange flown from the entry state. Some trajectories fly significant crossranges due to the single planned bank reversal. Note that there are initial downrange errors in addition to than the initial crossrange errors, but due to the difference in scale they are not easily visible.	108
10.7	Downrange and crossrange to the target position. The crossrange is generally small, with 88% under 1 km, indicating the vehicle heading is well-aligned at ignition.	109
10.8	The minimum altitude constraint, indicated in red, causes low energy scenarios to trigger at longer downrange distances and higher velocities, resulting in increased PMF.	109
10.9	Ignition velocity magnitude (left) and components (right). Lower ignition velocities tend to occur at steeper flight path angles for the chosen parametrization. The correlation between velocity and PMF is evident, but the slowest ignition point (bottom right of the right plot) is nowhere near the lowest PMF value, demonstrating the importance of other state variables in determining the PMF required.	110
10.10	The optimal placement of the target from the entry interface varies considerably with minimum entry altitude constraint has a strong impact on the geometry of the propellant optimal powered descent trajectories.	112
10.11	The minimum entry altitude constraint has a strong impact on the geometry of the propellant-optimal powered descent trajectories.	112

LIST OF TABLES

	Page
5.1 Comparison of UQ methods in an open-loop scenario	36
5.2 Comparison of UQ methods in an closed-loop scenario	36
6.1 The mean state and standard deviations at the entry interface.	50
6.2 The mean state and standard deviations at the start of closed-loop guidance.	50
6.3 Algorithm parameters used in all numerical results.	51
7.1 KS statistics estimating the sensitivity of terminal altitude and range to the uncertain input variables.	65
7.2 KS statistics estimating the sensitivity of terminal altitude and range to the uncertain input variables.	69
8.1 Monte Carlo statistics for cases 1-3, with reference trajectories designed using optimal control and fixed bank margins.	78
8.2 The Monte Carlo statistics for cases 4-6, with reference trajectories designed using the proposed method.	78
8.3 Summary of the statistics from the Unscented Transform and Monte Carlo	81
10.1 Model parameters used in all simulation results presented.	102
10.2 A summary of PMFs for a series of $\pm 10\%$ uncertainties compared to a nominal scenario with no uncertainty. Scenarios with significant increases in PMF relative to the nominal scenario have been highlighted.	104
10.3 A summary of the propellant-optimal ignition states for a series of $\pm 10\%$ uncertainties. Optimal trajectories generally feature low crossrange, and alti- tudes close to the minimum altitude constraint.	105
10.4 The mean and 3σ values of the inputs used for the Monte Carlo simulation.	107

ACKNOWLEDGMENTS

I would first like to thank my advisor Kenneth D. Mease, whose expertise shaped not only the course of my research, but also my entire career.

I would like to thank my fellow UCI colleagues, especially Drs. Guangfei Duan, Eric Trumbauer, and Alessandro Bombelli, for many fruitful discussions during our time together.

I am also indebted to my committee members, Professor Sideris and Professor Georgiou, for their review and suggestions.

I would also like to thank my colleagues at the Jet Propulsion Laboratory for their advice, and for their patience. My experience at JPL profoundly shaped the direction of my research culminating in this dissertation. Special thanks to Joel Benito, whose guidance and mentorship taught me so much in so little time, and whose friendship I am continually grateful for. I am a better engineer for having known you.

I gratefully acknowledge the Holmes Endowed Fellowship for its financial support.

Finally, I am immensely grateful to my mother for all her love and support.

VITA

Connor David Noyes

EDUCATION

Doctor of Philosophy in Aerospace Engineering University of California	2021 <i>Irvine, CA</i>
Master of Science in Aerospace Engineering California Polytechnic State University	2013 <i>San Luis Obispo, CA</i>
Bachelor of Science in Aerospace Engineering California Polytechnic State University	2014 <i>San Luis Obispo, CA</i>

ENGINEERING EXPERIENCE

Graduate Intern Jet Propulsion Laboratory	2015 <i>Pasadena, California</i>
Guidance and Control Engineer Jet Propulsion Laboratory	2015-present <i>Pasadena, California</i>

RESEARCH EXPERIENCE

Graduate Research Assistant University of California, Irvine	2013–2021 <i>Irvine, California</i>
--	---

TEACHING EXPERIENCE

Teaching Assistant California Polytechnic State University	2013 <i>San Luis Obispo, CA</i>
Teaching Assistant University of California	2015 <i>Irvine, CA</i>

PUBLICATIONS

Mars Entry Guidance for High Elevation via Robust Optimal Control Journal of Spacecraft and Rockets (Submitted for Review)	2021
Entry Guidance for Propellant Optimal Powered Descent on Mars AAS/AIAA Astrodynamics Specialist Conference	Aug 2020
A Convex Optimization Approach to Mars Entry Trajectory Updating AAS/AIAA Astrodynamics Specialist Conference	Aug 2018
High Ballistic Coefficient Mars EDL With Supersonic Retropropulsion AAS Guidance and Control Conference	Feb 2017
Sensitivity Analysis and Uncertainty Quantification of a Mars Ascent Vehicle Concept ASME Verification and Validation Symposium	Aug 2017

ABSTRACT OF THE DISSERTATION

Robust Optimal Entry Guidance for Future Mars Landers

By

Connor David Noyes

Doctor of Philosophy in Mechanical and Aerospace Engineering

University of California, Irvine, 2021

Professor Emeritus Kenneth D. Mease, Chair

Mars landings at higher elevations than achieved to date are desired for scientific pursuits. The phases of atmospheric flight are entry, descent, and landing. The research reported here concerns the guidance for the entry phase. We distinguish between the current generation of Mars landers, defined by its continued reliance on Viking-era space technology, and future missions, such as those carrying humans, characterized by significantly higher entry masses and landing ellipse requirements on the order of 100 m, roughly two orders of magnitude smaller than the current generation.

To support higher elevation landing, entry guidance must deliver the entry vehicle to the required altitude with the required horizontal accuracy at the end of the entry phase. The state-of-the-practice entry guidance cannot both raise the final altitude and achieve the required horizontal accuracy at the end of the entry phase. By formalizing the entry guidance objectives as a robust optimal control problem, we seek both to increase the final altitude and to improve the horizontal accuracy. In this approach, we consider only the longitudinal motion and investigate the feasibility of determining a reference trajectory that, in closed-loop reference-trajectory-based guidance, will yield the robust performance required for higher elevation landing. To address robustness, the state variables and uncertain parameters in the entry dynamics are treated as random variables using the unscented transformation to

approximate their means and variances and state the performance index in terms of these statistics. Differential dynamic programming is used to solve the robust optimal control problem. Case studies of two different classes of entry vehicle in the current generation demonstrate both the robust performance of the longitudinal entry guidance and the computational feasibility of the design method.

One technology hypothesized to be enabling for landing high mass (and therefore high ballistic coefficient) future missions is supersonic retropropulsion. In the latter part of this dissertation we consider the role of entry guidance in missions where powered descent follows the entry phase directly, without an intermediate parachute deceleration prior to ignition of the retropropulsion. In parachute-based architectures, Mars entry guidance algorithms are judged on their ability to control range errors while reaching the safe parachute deployment set. In contrast, in chuteless missions where pinpoint landing is achieved via powered descent, we posit that performance will instead be based on the required propellant to land the vehicle. A predictor-corrector algorithm, designed to minimize the predicted propellant, is presented. Feasible solutions to the powered descent problem are used to define the entry guidance target set. By computing and storing a mapping from ignition states in the target set to propellant required, the entry guidance algorithm maintains a predicted ignition state that varies over time as various perturbations alter the reachable set of the vehicle. The guidance algorithm updates the bank profile in order to track the propellant optimal reachable state. Simulation results are presented for a hypothetical vehicle with a ballistic coefficient approximately double that of Mars 2020.

Chapter 1

Introduction

1.1 Motivation

NASA has successfully landed five rovers on Mars to date, including the Mars 2020 rover Perseverance. Each of these missions chose low elevation targets between -1 and -4 km relative to the Mars areoid surface, commonly referred to as MOLA, after the instrument that was used to map Mars' topography. The thin Martian atmosphere makes aerodynamic deceleration ineffective except at lower altitudes where the atmosphere is denser. Coupled with the increasing entry mass required to deliver larger, more capable rovers to the surface, this makes high elevation landing on Mars a challenging problem. Nevertheless, targets above 0 km MOLA, like Terra Sirenum in the Southern Highlands, are motivated by scientific interest [2]. In order for descent and landing operations to have sufficient timeline margin [3, 4] when targeting such elevations, the entry phase must be terminated at a sufficiently high altitude.

In addition to altitude, guidance designers must consider other objectives, such as controlling the size of the landing footprint. Both MSL and Mars 2020 had landing ellipse

requirements of 25 km x 20 km [4, 1]. Often these multiple objectives are competing, as for example was the case on the Mars Science Laboratory (MSL) mission, where the entry guidance designers noted that targeting landing site elevations above -1 km MOLA would incur a larger landing ellipse [4]. Future missions, including sample return [5] and manned, will place even greater emphasis on robust performance than the current generation, such as pinpoint landing requirements [6, 7], defined as sub-100 m accuracy. As a component of a possible sample return campaign, Mars 2020 will collect samples for potential return to Earth by a future mission with the campaign, so pinpoint landing is not some far off dream but rather an imminent need. Such a drastic reduction of the footprint will require advances in entry, descent, and landing (EDL) guidance, navigation, and control (GNC). For pinpoint landing, powered descent guidance will be required to “fly out” dispersions arising during earlier phases, including entry and time spent on the parachute. Powered descent guidance for pinpoint landing has been studied extensively [8, 9, 10, 11, 12].

EDL technology for all previous US Mars landed missions, including MSL and Mars 2020, can trace its heritage to Viking-era space qualification technology from the 1970s [3]. The 70° sphere cone aeroshell used on these missions provides only low lifting capability in the entry phase, during which ballast mass is used to offset the center of mass of the spacecraft to create an angle of attack, thereby generating lift. The bank angle, which orients the lift vector around the velocity vector, is the control variable by which the vehicle trajectory can be adjusted during the entry phase. Thus, the role of entry guidance is to generate bank angle commands that deliver the vehicle to suitable conditions for descent and landing operations. The limited lift available means the control authority is also limited, and as a result, saturation in feedback control is a key issue in Mars entry guidance [13].

The limited vehicle capability and stringent mission requirements have generated interest in extremizing various performance measures of interest. This is the purview of optimal control theory. Due to the interest in high elevation landings, guidance algorithms de-

signed to maximize altitude at parachute deploy have been studied using optimal control [14, 15, 16, 17]. For hypothetical future missions in which powered descent follows the entry phase directly, i.e., without a parachute phase, propellant-optimal entry guidance has been investigated [18, 19, 20]

In addition to the nominal challenges of Mars EDL, there exists significant uncertainty in the vehicle aerodynamics, initial entry state, and the Martian atmosphere. Thus, in addition to achieving high performance in nominal circumstances, entry guidance must be robust to uncertainty. One systematic way to balance nominal performance and robustness is to formulate an optimal control that weights a nominal (or mean) optimization objective with minimization of a function of the state covariance matrix. Throughout this dissertation, we refer to such optimal control formulations as robust optimal control problems (ROCP) [21]. There appears to be no standard nomenclature in the literature, where such problems are also sometimes referred to as uncertain optimal control [22], optimal trajectory design under uncertainty [23, 24], Riemann-Stieltjes optimal control [25], and more. Due to the close relationship between the sensitivity matrix (also called the state transition matrix) and the state covariance matrix, we also include desensitized optimal control [26, 27] under the umbrella of robust optimal control. Figure 1.1 depicts the concept of Mars entry as an ROCP.

Approaches to entry guidance based on robust optimal control have been studied in [28, 29, 23, 30, 31]. The literature on ROCPs can be divided into open- and closed-loop control strategies. Open-loop solutions to ROCPs have been proposed when feedback is unavailable [25], or for use as a reference trajectory in closed-loop trajectory tracking [29]. Open-loop covariance shaping relies on nonlinearities of the system models; for linear systems, the state covariance does not depend on the mean state, and thus in the absence of feedback, no covariance shaping can be performed. References [28, 29] explored open-loop robust optimal control for entry guidance, where only the path through the atmosphere is

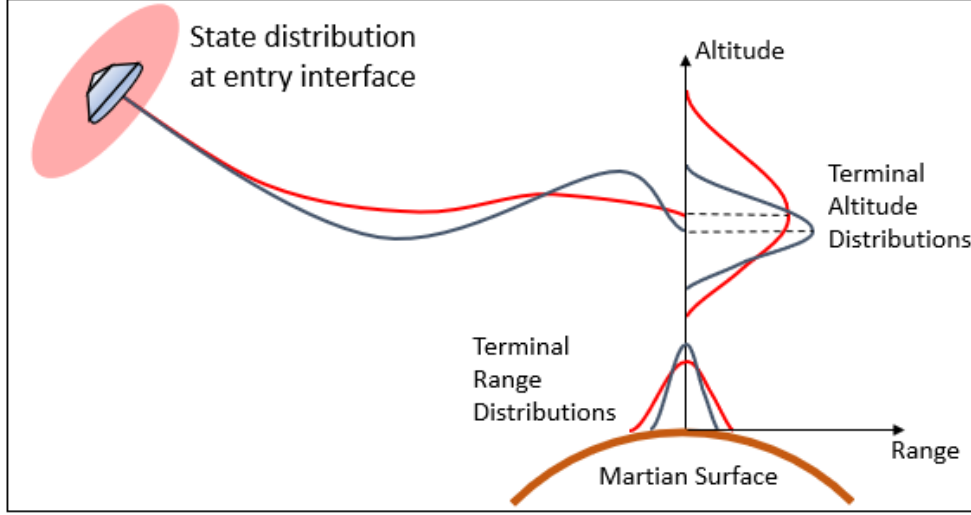


Figure 1.1: The concept of robust optimal control.

used to effect change on the terminal state covariance. In closed-loop ROCPs, a feedback control law $\mathbf{u}(t, \mathbf{x})$ is modeled. Closed-loop solutions to ROCPs have been discussed in the context of entry in [23] and [30].

A key reason to differentiate between open- and closed-loop is due to constraints on the control variables. In open-loop ROCP, path constraints on the controls are treated in the deterministic fashion. That is, because the control is purely a function of the independent variable, enforcing a general nonlinear constraint for the nominal trajectory ensures it is satisfied for any trajectory. In closed-loop formulations, one way to proceed is to treat the constraint as a probabilistic chance constraint, which ensures the constraint is satisfied with at least some chosen probability. However, control constraints are often related to hard physical limits, such as the maximum thrust available to a spacecraft, and probabilistic satisfaction may not be appropriate. The most common examples are norm-bounded controls, and individual control limits. To strictly enforce such constraints, a new piecewise continuous control can be defined by treating the constraint as part of the feedback control law such that the control constraint is guaranteed to be satisfied. As a result of the hard enforcement in the ROCP, the optimal controls will balance the distance of the open-loop controls from

the constraint in order to balance the mean performance and robustness as measured by some function of the state variance.

All of the existing applications of robust optimal control to entry guidance have utilized linear methods of determining the state covariance or sensitivity matrices, and little discussion of the effects of control saturation are found among the references. Control saturation is a nonlinear, non-differentiable phenomenon. A natural question is whether linear methods are sufficiently accurate in the presence of control saturation. Due to the limited control authority of Mars entry vehicles, we incorporate hard control saturation in our problem model, and as a consequence also consider whether or not linear methods are appropriate for our goals.

For current generation missions using parachute-based EDL, we restrict our attention to the same class of guidance law as Mars 2020, affine state-feedback. The specific objectives of this dissertation are 1) for given models of the vehicle and planet (including parametric uncertainty), determine a reference control, reference trajectory, and feedback gains that, when used for closed-loop reference tracking guidance, will optimize a performance index chosen based on mission requirements, and 2) to enable fast, efficient guidance design that is free of computationally expensive Monte Carlo simulations in the loop and hand tuning of key guidance parameters. This is motivated by the approach used for MSL, in which design maps are generated via iteration over guidance parameters with Monte Carlo simulation performed for each combination of parameters. A more detailed review of the MSL design process is deferred to Section 1.3.

To meet our first objective, this dissertation builds on the robust optimal control approach to entry guidance. Our formulation differs from the existing formulations in the form of our performance index, the use of velocity as the independent variable, and, in light of the previous discussion, careful consideration of the limited control authority and the effects of control saturation on the state distribution.

The solution of nonlinear optimal control problems is, in general, not a simple task. Add to that the complexity of quantifying uncertainty at each iteration of the solution process, and it is easy to see that solving ROCPs is a challenging problem. To meet our second objective, we propose and demonstrate an efficient solution method for the entry guidance ROCP based on Differential Dynamic Programming (DDP) and the Unscented Transform (UT). The UT is chosen because it is better suited to propagating the mean and state covariance through the nonlinear dynamics than linear methods. A numerical example is given to illustrate this point. DDP is chosen for its suitability to large-scale optimal control problems, and its robust convergence properties. Starting from the control-limited DDP algorithm presented in [32], we examine an iterated Linear-Quadratic-like simplification [33] that is enabling in solving the ROCP. Next, we extend the algorithm to jointly optimize the static feedback gains concurrently with the reference control for improved performance. Using the MSL and Mars 2020 missions as our benchmark, the performance of the guidance solutions is assessed numerically.

1.2 Dissertation Organization

This remainder of this dissertation is organized as follows: The remainder of this chapter further reviews the state-of-the-practice guidance flown on MSL and Mars 2020.

Chapter 2 begins with an overview of the EDL timeline. A high level discussion of the entry guidance problem, especially target conditions, is presented.

Chapter 3 discusses the models of the vehicle and environment used to derive the guidance algorithm.

Chapter 4 outlines the guidance strategy, including formally posing the problem as a ROCP.

Chapter 5 discusses uncertainty quantification and presents a numerical example demonstrating the benefit of UT depending on whether the control strategy is open-loop or closed-loop. An algorithm based on constrained DDP is modified to solve the ROCP. Two modifications to the ROCP are made in order for the DDP algorithm to be applicable.

Chapter 6 presents the assessment conditions, including algorithm parameters, details of the Monte Carlo simulations.

Chapter 7 presents an assessment of the guidance algorithm for MSL-class vehicle. A negative result associated with joint optimization of velocity-varying feedback control gains is presented. Chapter 8 presents a similar assessment for a heavier vehicle in the class of the Sample Retrieval Lander.

In Chapter 9 we direct our attention to chuteless, SRP-based EDL envisioned for future missions. A numerical predictor-corrector entry guidance algorithm is developed to minimize the predicted propellant during the powered descent phase. Chapter 10 presents a numerical assessment of the guidance approach.

In Chapter 11 the conclusions are stated, and several avenues for future work are discussed.

1.3 Entry Terminal Point Controller

The state-of-the-practice for Mars entry guidance, used on MSL and Mars 2020, is a modified version of the Apollo command module entry guidance [4] called the entry terminal point controller (ETPC). The three phases of the ETPC are prebank, range control, and heading alignment. In the prebank phase, there is no guidance, but the vehicle is rotated to the expected initial bank angle for the start of the range control phase. The range control

guidance begins when the sensed drag deceleration exceeds a threshold value, and consists of longitudinal guidance and lateral guidance. Longitudinal guidance modulates the bank angle magnitude to control the range flown, and reduce range dispersions at parachute deployment. Lateral guidance operates concurrently with longitudinal guidance during range control to command bank reversals when a measure of the crossrange to the target exceeds the deadband threshold, which is a quadratic function of velocity. At a specified velocity, the guidance switches from range control to heading alignment, during which the bank angle is commanded to reduce the crossrange error. During heading alignment, the bank angle magnitude is restricted to 30° . This ensures that most of the lift vector is opposite gravity, which results in increased parachute deploy altitude. For MSL and Mars 2020, the vehicle was required to land in an ellipse of $25 \text{ km} \times 20 \text{ km}$ [4, 1]. Due to the expansion of the ellipse during time spent on the parachute, the entry guidance requirement was to deploy the parachute with 10 km of the planned parachute deploy target [4]. The ETPC delivered MSL to an elevation of -4.4 km with a target miss distance of 2.3 km and a 99% confidence landing ellipse of $19.1 \text{ km} \times 6.9 \text{ km}$ [4].

The entry guidance for these latest Mars entry vehicles [4, 1] is designed by multiple iterations of two steps. The first step is to design a reference bank profile and compute the corresponding reference trajectory via open-loop simulation. The reference bank angle profile is specified by a small number of parameters, a modification from the original Apollo algorithm, which assumed a constant bank angle reference profile. This change allows for more flexibility in trajectory design, including enabling higher parachute deploy altitudes [4]. Because the ETPC range control feedback gains depend on the reference trajectory [34], the flexibility in the design of the reference trajectory is the primary means of affecting the overall performance of range control. The parameters specifying the reference control must be chosen to provide control margin for both longitudinal and lateral guidance, while also achieving high altitude. In the second step, some of the more severe off-nominal entries are simulated in closed-loop using the reference trajectory, ETPC gains, and heading alignment

guidance to estimate off-nominal performance prior to Monte Carlo simulation [4]. The results are judged based on the vehicle state at parachute deployment. The free parameters of the guidance algorithm, including the overcontrol gain and heading alignment gain, are then tuned as necessary to improve the performance. When the required performance cannot be achieved through tuning, the reference trajectory is changed and the process repeats. The iterative two-step process ends up requiring consideration of 100s or 1000s of reference trajectories [4]. Eliminating this iterative process is a key objective of this dissertation.

In Chapter 4, we propose an alternative to the longitudinal guidance operating during the range control phase. However, because our alternative guidance does not focus exclusively on range, we prefer to refer to it as second phase guidance. Similarly, the third phase guidance for MSL and Mars 2020 was heading alignment guidance, and alternative third phase guidance approaches has been proposed [35].

Chapter 2

Mars Entry Phase Guidance Requirements

Figure 2.1 depicts the EDL timeline for the Mars 2020 mission. The entry phase begins when the vehicle enters the atmosphere at 125 km altitude, and ends when the sequence leading to parachute deployment is commanded. As the discussion of MSL in the introduction highlighted, there are several requirements that entry guidance must meet. The vehicle must be delivered to safe parachute deployment conditions, generally given as acceptable ranges of dynamic pressure and Mach number, and the deployment must occur high enough to provide timeline margin for terminal descent. The parachute deploy range dispersion limit is flowed back from the landing footprint requirement. MSL determined that the entry phase had to be terminated within 10 km of the parachute deploy target to achieve the desired touchdown ellipse requirement of 25 x 20 km [4]. Mars 2020 had the same touchdown requirement [1]. Thus, broadly speaking, the goals are to achieve high terminal altitude and accurate terminal range.

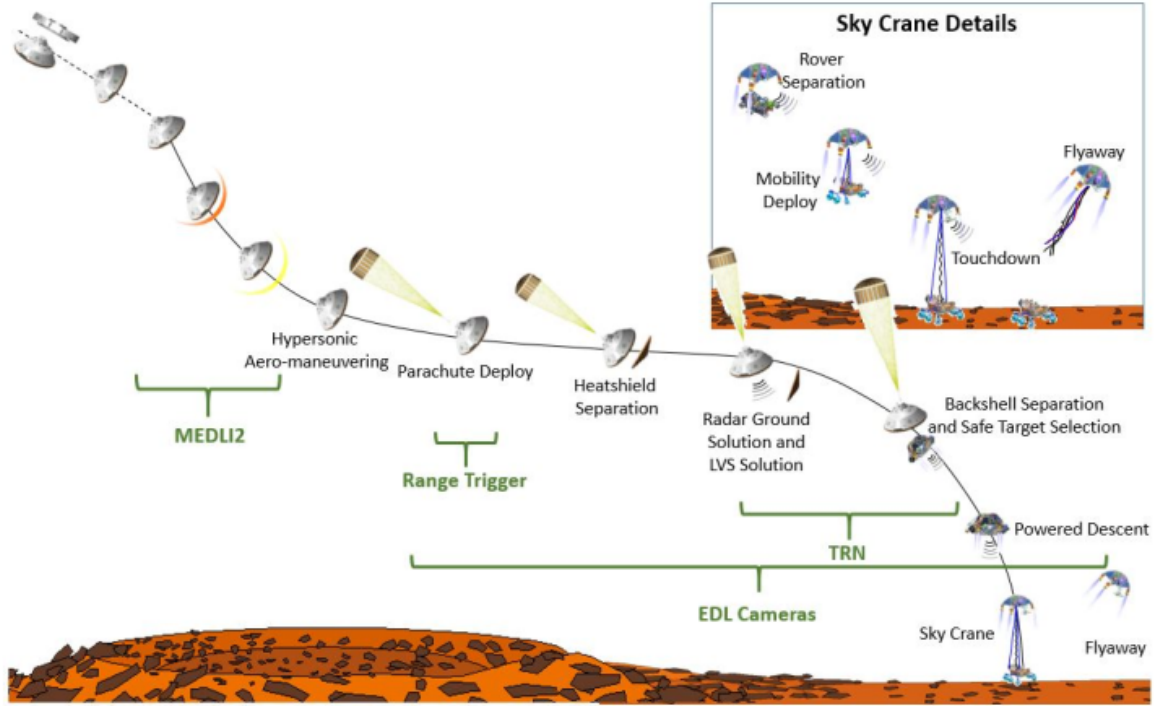


Figure 2.1: Entry Descent and Landing timeline for Mars 2020, taken from [1].

2.1 Entry Phase Termination

Generally speaking, the entry phase of Mars EDL terminates when the sequence leading to parachute deployment begins. For MSL, this sequence was triggered when the onboard estimated vehicle velocity reached a specified value [4]. Mars 2020 instead triggered at a fixed downrange distance [36]. Future missions, especially those intending to land without the use of a parachute, may consider a different function of the vehicle state [18, 20]. The choice of trigger has a strong impact on the terminal distribution. Naturally, triggering based on the onboard estimated velocity reduces the velocity spread down to approximately the spread in velocity knowledge error. Similarly, a range trigger will naturally minimize the range variations at the trigger point, but will result in larger velocity variations (though the increased to the spread in Mach number is not as large due to wind correlations [36]). We remark that designing entry guidance to reduce range errors is not a moot point in

the presence of a range trigger. Large errors in range will turn into large variations in the velocity at which the target range is reached, resulting in Mach variations that may exceed parachute qualification limits. Regardless of the trigger, the vehicle must meet parachute deployment constraints. Given models of the atmospheric density and local speed of sound on Mars, bounds on the acceptable dynamic pressure and Mach number can be converted into a feasible set in altitude-velocity space. See Fig 2.2 for an example using the Mars 2020 values [1], including an additional constraint that the deploy altitude is above 6 km.

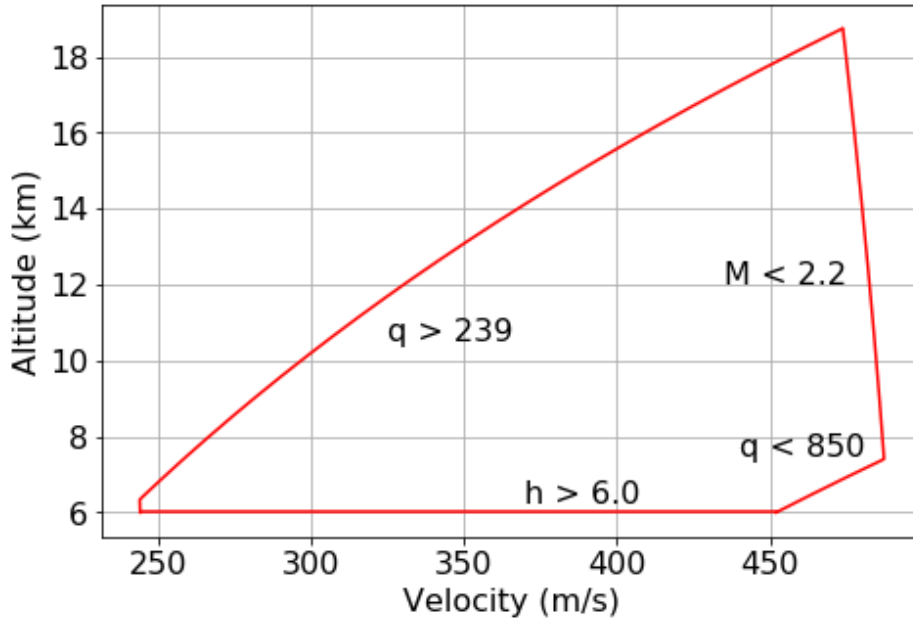


Figure 2.2: Parachute deployment conditions on Mach number and dynamic pressure in altitude-velocity space.

One key point is that, despite using different triggers to terminate the entry phase, MSL and Mars 2020 used the same design process involving parametrization of the reference control profile that terminates at a fixed velocity. Together, the control profile and the fixed final velocity define the downrange distance that will be used to trigger the termination of the entry phase in a range trigger design. Thus, the reference design is independent of the trigger used to terminate the entry phase. Throughout this dissertation it is assumed the entry phase is terminated based on velocity, using the terminal velocity $v_f = 460$ associated

with a parachute-based landing, corresponding approximately (because Mach also depends on altitude) to deployment at Mach 2. Like MSL and Mars 2020, the role of the longitudinal entry guidance is to command the vertical lift to steer the vehicle to the target location at the terminal velocity and with sufficient altitude for subsequent landing operations. We will formalize these objectives, and present the remainder of the problem formulation, in Chapter 4.

Chapter 3

Entry Vehicle and Environment Models

In this chapter we present the models of the entry vehicle and Mars environment used in the development and assessment of the entry guidance algorithm. Vectors are denoted by boldface and are assumed to be columns. $(\cdot)^T$ denotes the transpose of a vector. The trace of a square matrix is the sum of its diagonal entries and is denoted $\text{tr}(\cdot)$. The mean, or expected value, of a random variable, $\mathbf{y}$, is denoted by $\mathbb{E}[\mathbf{y}] = \bar{\mathbf{y}}$, and its covariance matrix by $\mathbb{V}[\mathbf{y}] = C_{\mathbf{y}}$. The standard deviation of a scalar random variable y is $\sigma_y \triangleq \sqrt{\mathbb{V}[y]}$. A normal distribution with mean $\bar{\mathbf{y}}$ and covariance matrix $C_{\mathbf{y}}$ is $\mathcal{N}(\bar{\mathbf{y}}, C_{\mathbf{y}})$.

3.1 Three Degree-of-Freedom Dynamics Model

The three degree-of-freedom translational dynamics for a Mars entry vehicle are defined relative to a Mars-fixed coordinate frame. In this three degree-of-freedom modeling, we have assumed that the velocity frame and the body frame are aligned, and do not consider the

vehicle attitude. The dynamics of an entry vehicle in an atmosphere are modeled by Vinh's equations [37]:

$$\dot{r} = v \sin \gamma \quad (3.1)$$

$$\dot{\theta} = \frac{v \cos \gamma \cos \psi}{r \cos \phi} \quad (3.2)$$

$$\dot{\phi} = \frac{v}{r} \cos \gamma \sin \psi \quad (3.3)$$

$$\dot{v} = -D - g \sin \gamma \quad (3.4)$$

$$\dot{\gamma} = \frac{L}{v} \cos \sigma + \left(\frac{v}{r} - \frac{g}{v} \right) \cos \gamma + C_\gamma \quad (3.5)$$

$$\dot{\psi} = -\frac{L}{v \cos \gamma} \sin \sigma - \frac{v}{r} \cos \gamma \cos \psi \tan \phi + C_\psi \quad (3.6)$$

where r is the distance from the center of Mars to the center of mass of the entry vehicle, and θ and ϕ are latitude and longitude, respectively; the velocity vector is defined by the magnitude of the planet-relative velocity, v , the flight path angle γ , and the heading angle ψ , defined counterclockwise from East. The bank angle, σ , is the clockwise rotation angle between the lift vector and the vertical plane containing the velocity vector; thus $\sigma = 0$ corresponds to the full lift-up orientation. The magnitude of the gravitational acceleration is $g = \mu/r^2$, and $\mu = 42,828 \text{ km}^3/\text{s}^2$ is the gravitational constant of Mars. Mars is assumed to be a sphere with planet radius $r_p = 3396.2 \text{ km}$, and the altitude above the surface is $h = r - r_p$. The aerodynamic accelerations are

$$L = \frac{1}{2} \rho v^2 \frac{S}{m} C_D \quad (3.7)$$

$$D = \frac{1}{2} \rho v^2 \frac{S}{m} C_L \quad (3.8)$$

where ρ is the atmospheric density of Mars, S is the vehicle surface area, and m is the vehicle mass. The Coriolis terms, C_γ and C_ψ , are

$$\begin{aligned} C_\gamma &= 2\omega_p \cos \psi \cos \phi \\ C_\psi &= 2\omega_p (\tan \gamma \sin \psi \cos \phi - \sin \phi) \end{aligned} \tag{3.9}$$

with planet rotation rate $\omega_p = 7.08 \times 10^{-5}$ rad/s.

Separation of the longitudinal and lateral guidance has proven effective in the state-of-the-practice approach, and we follow this when developing the robust optimal entry guidance algorithm. In place of the coordinates (θ, ϕ) , the distance along the great circle arc defined by the vehicle heading is substituted. Ignoring the offset between the great circle azimuth and the instantaneous vehicle heading, this distance is computed by integrating

$$\dot{s} = v \cos \gamma \tag{3.10}$$

and the longitudinal state vector is $\mathbf{x}_{\text{lon}} = [r, s, v, \gamma]^T$. An important quantity related to s is range-to-go. For a terminal target range, s_{target} , the range-to-go is $R = s_{\text{target}} - s$. During flight, the target distance is fixed, and thus $\dot{R} = -\dot{s}$. The longitudinal dynamics are

$$\dot{h} = v \sin \gamma \tag{3.11}$$

$$\dot{s} = v \cos \gamma \tag{3.12}$$

$$\dot{v} = -D - g \sin \gamma \tag{3.13}$$

$$\dot{\gamma} = \frac{L}{v} \cos \sigma + \left(\frac{v}{h + r_p} - \frac{g}{v} \right) \cos \gamma \tag{3.14}$$

3.2 System Models

3.2.1 Nominal Models

The density of the Martian atmosphere is modeled by an exponential function

$$\rho(h) = \rho_0 \exp\left(-\frac{h}{h_s}\right) \quad (3.15)$$

where $\rho_0 = 0.0158 \text{ g/m}^3$ is the reference density and $h_s = 9.354 \text{ km}$ is the scale height. This profile is shown in Fig. 3.1. The speed of sound is also modeled as a function of altitude.

The vehicle is assumed to be a low lift entry capsule. In the numerical assessment presented in Chapter 6, the vehicle is in the same class as MSL, with a lift-to-drag ratio $L/D \in [0.24, 0.31]$. A heavier vehicle with a nominal $L/D = 0.28$ is considered in Chapter 8. The aerodynamic coefficients are, in general, functions of Mach number and angle of attack. As is common in entry guidance developments, the angle of attack profile is assumed to be a fixed function of Mach number, and thus the coefficients depend solely on Mach number. The nominal aerodynamic coefficient profiles are curve-fitted from 500 MSL profiles [38]. The nominal coefficients, and their ratio $L/D = C_L/C_D$, are shown for the MSL-class vehicle in Fig. 3.2.

3.2.2 Uncertainty Models

Typically, the uncertain initial state and model parameters are discussed in the context of assessing the guidance algorithm. However, in this dissertation, a key component of the guidance approach is incorporating uncertainty during the reference trajectory design, and thus they are given here. All uncertain variables are assumed to be normally distributed, but we note that the guidance approach developed in this dissertation applies to any distributed

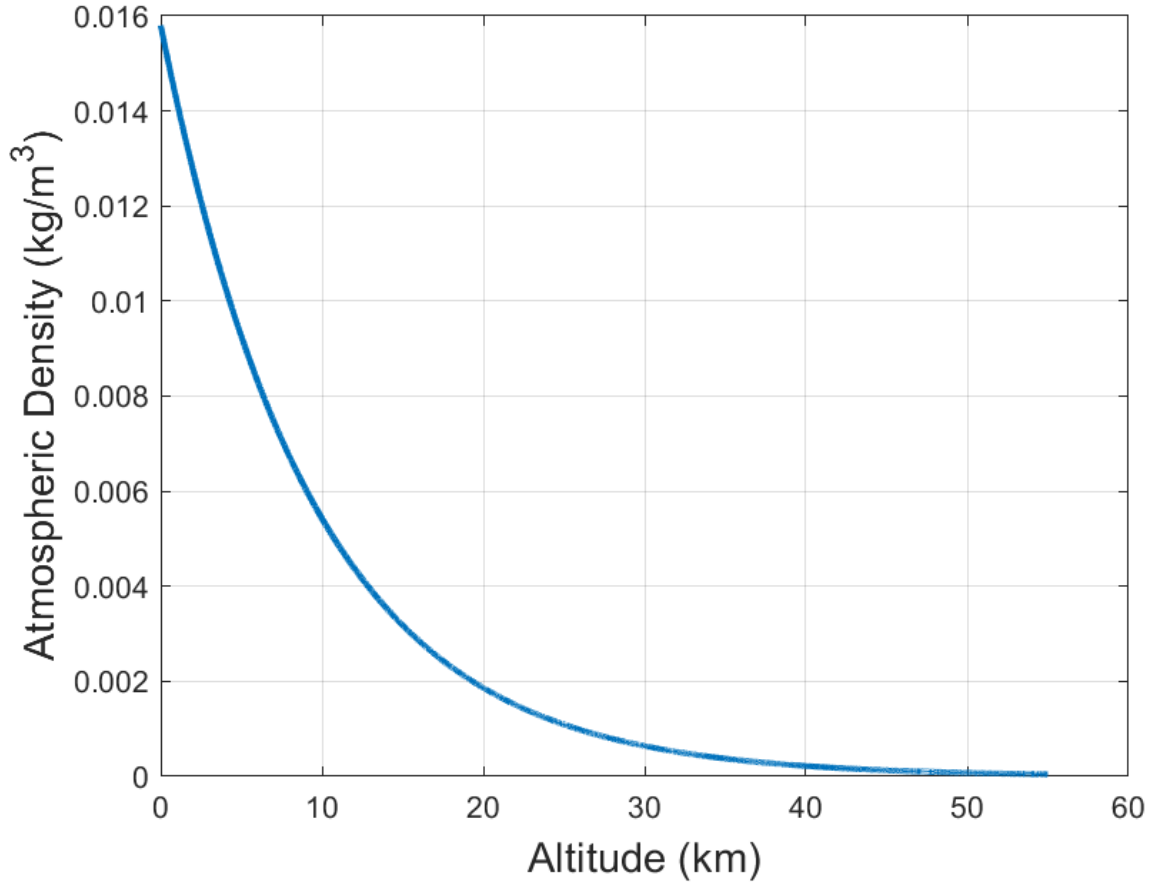


Figure 3.1: The nominal atmospheric density.

parameters with known mean and covariance.

The approach phase, prior to the entry phase, targets a nominal entry state, but errors in the insertion maneuver and navigated state result in an uncertain entry state

$$\mathbf{x}(t_0, \delta \mathbf{x}_0) = \bar{\mathbf{x}}_0 + \delta \mathbf{x}_0 \quad (3.16)$$

$$\delta \mathbf{x}_0 \sim \mathcal{N}(\mathbf{0}, C_{\mathbf{x}_0})$$

Two key quantities in the study of entry vehicles are the lift-to-drag ratio, L/D , and ballistic coefficient, $\beta = \frac{m}{SC_D}$. Often, uncertainty is modeled in these values rather than in the aerodynamic coefficients, C_L and C_D . For this reason, it is useful to rewrite Eqs. (3.7)-(3.8)

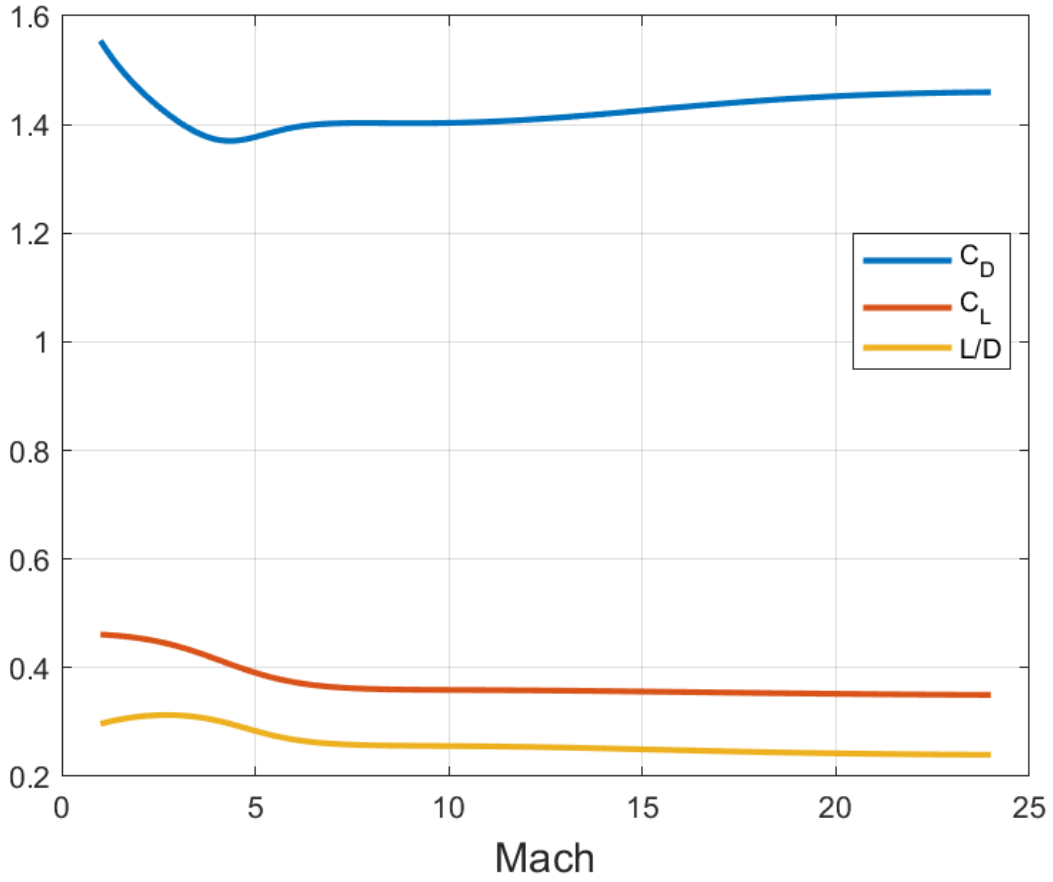


Figure 3.2: The nominal aerodynamic coefficients and L/D profile.

in terms of these quantities and their uncertainties:

$$\begin{aligned} D &= \frac{1}{2} \frac{\rho}{\beta(1 + f_\beta)} v^2 \\ L &= D \left(\frac{L}{D} \right) (1 + f_{\frac{L}{D}}) \end{aligned} \tag{3.17}$$

where $f_i \sim \mathcal{N}(0, \sigma_i^2)$ are constant fractional offsets from the mean profile.

Another key source of uncertainty in the entry dynamics is the atmospheric density model. The deviations are modeled as

$$\rho(h) = \bar{\rho}(h) + f_\rho \delta \rho(h) \tag{3.18}$$

where $\bar{\rho}(h)$ is given by Eq. (3.15). In Eq. (3.18), the deviation $\delta\rho(h)$ is free to vary independently from the mean density. Note that this assumes that a constant fraction $f_\rho \sim \mathcal{N}(0, \sigma_\rho^2)$ of the standard deviation is applied throughout the trajectory. This can be used to bound the altitude-varying, possibly asymmetric density disturbances of models like MarsGRAM [39] or the Mars Climate Database (MCD) [40]. See Fig. 3.3 for an example where the boundary is determined by 1000 samples from the MCD [35] and a symmetric model is created to bound the MCD model. Symmetry is required due to the Gaussian uncertainty assumption.

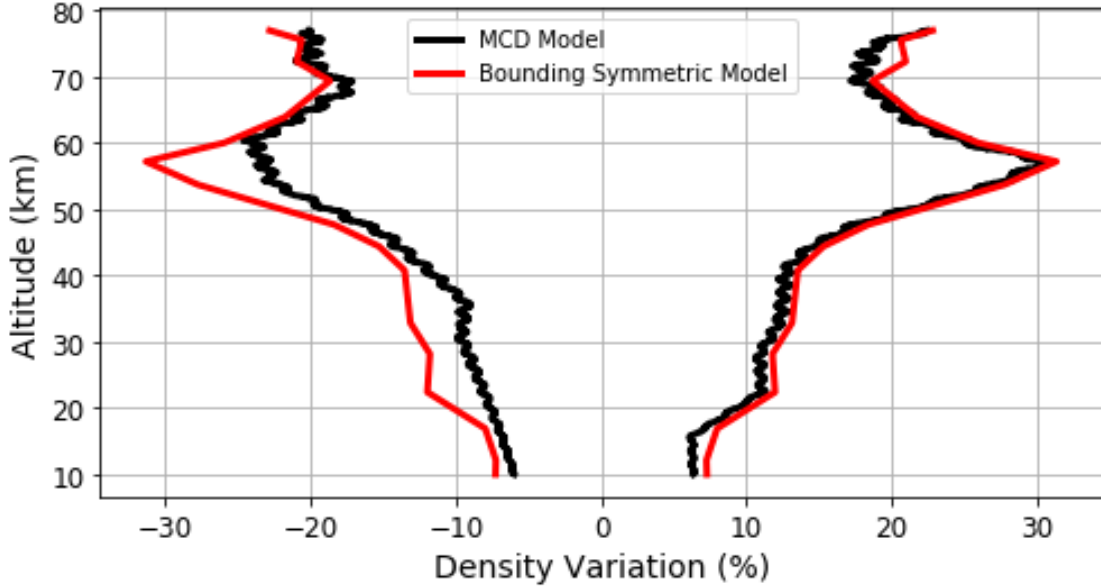


Figure 3.3: Density variations from 1000 Mars Climate Database samples

Collecting these uncertainties in the vector-valued random variable $\mathbf{p} = [\delta\mathbf{x}_0, f_{\frac{L}{D}}, f_\beta, f_\rho]^T$, the uncertain vehicle dynamics are written compactly as

$$\dot{\mathbf{x}}(t, \mathbf{p}) = \mathbf{f}(\mathbf{x}(t, \mathbf{p}), u(t, \mathbf{x}), \mathbf{p}), \quad \mathbf{x}(t_0, \mathbf{p}) = \mathbf{x}_0(\mathbf{p}) \quad (3.19)$$

where $\mathbf{p} \sim \mu = \mathcal{N}(\mathbf{0}, C_{\mathbf{p}})$ and

$$C_{\mathbf{p}} = \begin{bmatrix} C_{\mathbf{x}_0} & & & \mathbf{0} \\ & \sigma_{\frac{L}{D}}^2 & & \\ & & \sigma_{\beta}^2 & \\ \mathbf{0} & & & \sigma_{\rho}^2 \end{bmatrix} \quad (3.20)$$

Note that, for a given realization of $\mathbf{p}$, the dynamics are a deterministic function of the state and control. A nominal trajectory is defined as one in which $\mathbf{p} = \mathbf{0}$. For nonlinear systems like the entry dynamics, the mean trajectory is not generally equal to the nominal trajectory, except at the initial time. The Taylor series of a generic scalar variable, $x(t, \mathbf{p})$, expanded around $\mathbf{p} = \mathbf{0}$, is

$$x(t, \mathbf{p}) = x(t, \mathbf{0}) + \Phi_1 \mathbf{p} + \frac{1}{2} \mathbf{p}^T \Phi_2 \mathbf{p} + \dots \quad (3.21)$$

where Φ_i , $i = 1, 2, 3, \dots$ are the i^{th} partial derivatives of x . Truncating the series at second order and taking the expectation yields

$$\mathbb{E}[x(t, \mathbf{p})] = \mathbb{E}\left[x(t, \mathbf{0}) + \Phi_1 \mathbf{p} + \frac{1}{2} \mathbf{p}^T \Phi_2 \mathbf{p}\right] \quad (3.22)$$

$$= x(t, \mathbf{0}) + \mathbb{E}\left[\frac{1}{2} \mathbf{p}^T \Phi_2 \mathbf{p}\right] \quad (3.23)$$

where the second line follows from the linearity of the expectation operator and $\mathbb{E}[\mathbf{p}] = \mathbf{0}$. This latter fact also means $C_{\mathbf{p}} = V[\mathbf{p}] = \mathbb{E}[\mathbf{p}\mathbf{p}^T]$. The trace is also a linear function, so the expectation and trace can be swapped, i.e., $\mathbb{E}[\text{tr}(A)] = \text{tr}(\mathbb{E}[A])$. Together with the identity $\mathbf{p}^T A \mathbf{p} = \text{tr}(A \mathbf{p}\mathbf{p}^T)$, we arrive at

$$\mathbb{E}[x(t, \mathbf{p})] = x(t, \mathbf{0}) + \frac{1}{2} \text{tr}(\Phi_2 C_{\mathbf{p}}) \quad (3.24)$$

From Eq.(3.24) it can be seen that the nominal trajectory provides a first-order estimate of

the mean.

Chapter 4

Guidance Strategy

In this chapter we present the guidance strategy, which is to pose the guidance problem as a robust optimal control problem. The guidance law for longitudinal range control has an affine feedback structure, consisting of a reference vertical L/D profile and a feedback control law, subject to physical- and safety-related constraints. Thus, to design the guidance means to design the reference vertical L/D and corresponding longitudinal reference trajectory, and the feedback gains.

4.1 Independent Variable

Following from the assumption that the entry phase is terminated at a fixed velocity, it is essential to use velocity as the independent variable. If time were used, at best we could constrain the mean velocity at the final time to be equal to the terminal velocity, but there would nevertheless be a distribution of final velocities. Using velocity as the independent variable naturally guarantees that the entire distribution of trajectories will terminate at the same velocity. Redefining $\mathbf{x}_{\text{lon}} = [h, s, \gamma]^T$ to remove the velocity state variable, the

dynamics with respect to velocity, with $(\cdot)'$ denoting the derivative with respect to velocity, are

$$\mathbf{x}'_{\text{lon}}(v, \mathbf{p}) = \mathbf{f}_v(v, \mathbf{x}_{\text{lon}}(v, \mathbf{p}), u(v), \mathbf{p}) \quad (4.1)$$

$$= \begin{bmatrix} \frac{\dot{h}}{\dot{v}} \\ \frac{\dot{s}}{\dot{v}} \\ \frac{\dot{\gamma}}{\dot{v}} \end{bmatrix} \quad (4.2)$$

One issue with this choice, however, is that velocity may not be monotonically decreasing from the entry interface because, at high altitudes, $D \approx 0$ and so $\dot{v} = -D - g \sin \gamma$ is dominated by the gravity term until the vehicle descends into denser atmosphere. Because the gains used by MSL and M2020 during range control were interpolated as functions of the velocity magnitude, they had to contend with a similar issue.

Recall that MSL and Mars 2020 had a pre-bank phase prior to range control; this phase ended when the magnitude of vehicle sensed drag acceleration exceeded 1.96 m/s^2 [4]. With Martian gravity approximately 3.71 m/s^2 , this is sufficient drag to ensure monotonicity for γ as steep as -30° , noting that both MSL and Mars 2020 had entry flight path angles around -15.5° .

Because drag is a function of both altitude and velocity, the point at which D reaches its threshold value is not at a single velocity for all realizations of $\mathbf{p}$. Thus, we can consider a very similar solution except that pre-bank ends at a fixed velocity rather than a fixed drag magnitude. Denoting this velocity by v_0 , this value is essentially the highest velocity for which $D(v_0, \mathbf{x}, \mathbf{p}) > D_{\text{threshold}}$ and is driven by the lowest drag scenario (high altitude, low density, high ballistic coefficient). Since the entry uncertainties are generally specified at the entry interface and not at the start of range control, an ensemble of points is propagated from the entry interface to v_0 , and the mean and covariance matrix at this point become the initial state and covariance used in the robust guidance problem.

4.2 Feedback Control Form

As a result of considering uncertainty, the state variables are random variables and the goal is to optimize a distribution of trajectories rather than a single trajectory. In designing the reference trajectory, the effects of feedback control on the distribution are accounted for. Thus the closed-loop control u consists of both a reference control u_{ref} , and a feedback control δu , in contrast to open-loop design methods where $u = u_{\text{ref}}$. Because the control variable is the cosine of an angle, its magnitude must be bounded by one. Due to the low lift capability of current generation Mars entry capsules, it may be prudent to further restrict the vehicle from bank angles that orient the lift vector downward, so in our application the lower bound is taken to be zero. The limits apply to the reference control, so the path constraint is

$$0 \leq u_{\text{ref}}(v) \leq 1 \quad (4.3)$$

which disallows reference bank angle magnitudes greater than 90° . The control law is assumed to be to a saturated linear state feedback

$$u(v, \mathbf{x}_{\text{lon}}) = \text{sat}_{[0,1]} \left(\frac{\left(\frac{L}{D}\right)_{\text{ref}} u_{\text{ref}}(v)}{\frac{L}{D}} + \delta u(v, \mathbf{x}_{\text{lon}}) \right) \quad (4.4)$$

$$\delta u = k_D \delta D + k_\gamma \delta \gamma + k_s \delta s \quad (4.5)$$

where we note that, consistent with state-of-the-practice EDL operations on Mars, drag acceleration has been used as a feedback term in place of altitude. The saturation function is defined

$$\text{sat}_{[a,b]}(x) = \begin{cases} a, & x < a \\ x, & a \leq x \leq b \\ b, & b < x \end{cases}$$

The saturation function is required to ensure that, regardless of the value of the reference control u_{ref} , the feedback control $u(v, \mathbf{x})$ always satisfies the control limits. Since vertical L/D is equal to $\frac{L}{D}u$, the saturation serves the same purpose as the L/D limiter used on MSL and Mars 2020 [4]. Equation (4.3) limits the reference control, while the saturation function in Eq. (4.4) limits the closed-loop control. These bounds need not be equal; ad hoc control margin could be had by imposing tighter bounds in Eq. (4.3) than in the saturation function. However, as pointed out in Ref. [4], some saturation is not necessarily undesirable, so instead we choose the reference limits and closed-loop limits to be equal and allow the optimization process, presented in the following chapter, to determine the robust optimal margin along the trajectory.

In the numerical results presented later, the gains $[k_D, k_\gamma, k_s]$ in Eq. (4.5) are chosen to be constant values, but they may, in general, be functions of velocity. The MSL and Mars 2020 velocity-dependent gains could be used; they are designed, based on linearized dynamics, to fly a trajectory neighboring the reference which ends with the desired range. Our constant gain feedback control is always driving the perturbed trajectory back to the reference. The state deviations in Eq. (4.5) are computed with respect to the reference state at the current velocity, e.g., $\delta D(v) = D(v) - D_{\text{ref}}(v)$. The form of Eq. (4.4) may be unfamiliar but notice that when the control is not saturated, Eq. (4.4) may be rearranged as

$$\frac{L}{D}u(v, \mathbf{x}_{\text{lon}}) = \frac{L}{D}\bigg|_{\text{ref}} u_{\text{ref}}(v) + \frac{L}{D}\delta u(v, \mathbf{x}_{\text{lon}}) \quad (4.6)$$

Just like the ETPC, the control in Eq. (4.4) commands the reference vertical L/D , rather than a reference fraction of the available L/D , which turns out to be a more robust choice in the presence of aerodynamic uncertainty. Compared with the ETPC (see Eq. (2) in Ref. [4]), the difference lies only in the feedback control δu . Flight path angle is used as a feedback term in place of altitude rate. The rationale is that, when using velocity to interpolate the reference trajectory, only the difference in flight path angle is important. To see this, note

$\delta\dot{r}(v) = v \sin \gamma - v \sin \gamma_{\text{ref}} = v(\delta \sin \gamma)$. If the onboard implementation requires that altitude rate be used, we can determine a velocity-varying gain $k_{\dot{r}}$ such that $k_{\gamma}\delta\gamma \approx k_{\dot{r}}(v)\delta\dot{r}$ as follows

$$k_{\dot{r}}(v)\delta\dot{r}(v) = k_{\gamma}\delta\gamma(v) \quad (4.7)$$

$$k_{\dot{r}}(v) = k_{\gamma} \frac{\delta\gamma(v)}{\delta\dot{r}(v)} \quad (4.8)$$

$$\dot{r} = v \sin \gamma \quad (4.9)$$

$$\delta\dot{r} = v \delta \sin \gamma \quad (4.10)$$

$$k_{\dot{r}}(v) = \frac{k_{\gamma}}{v} \frac{\delta\gamma}{\delta \sin \gamma} \quad (4.11)$$

$$\delta \sin \gamma = 2 \cos \frac{\gamma + \gamma_{\text{ref}}}{2} \sin \frac{\gamma - \gamma_{\text{ref}}}{2} \quad (4.12)$$

$$\delta \sin \gamma \approx \cos \gamma_{\text{ref}}(\gamma - \gamma_{\text{ref}}) = \cos \gamma_{\text{ref}} \delta\gamma \quad (4.13)$$

$$k_{\dot{r}}(v) \approx \frac{k_{\gamma}}{v \cos \gamma_{\text{ref}}(v)} \quad (4.14)$$

The error in the approximation in Eq. 4.13 is small when γ is close to γ_{ref} . For a large flight path angle deviation of 15° , the error for any $\gamma_{\text{ref}} \in [-20^\circ, 5^\circ]$ is less than 3%.

4.3 Defining the Objective Function

The first objective is to maximize the mean altitude at the final velocity while minimizing the altitude standard deviation

$$\max J_h = \bar{h}(v_f) - w_h \sigma_h(v_f) \quad (4.15)$$

where $w_h \geq 0$ is a penalty on the standard deviation. Maximizing Eq. (4.15) for $w_h = 0$ results in an optimal mean altitude, while $w_h > 0$ maximizes a measure of the low end of the altitude distribution. For example, $w_h = 3$ maximizes the 3σ -low altitude.

The second objective, consistent with the range control objective, is to minimize the standard deviation of range

$$\min J_s = w_s \sigma_s(v_f) \quad (4.16)$$

In contrast to Eq.(4.15), the mean range is not included in the objective. This is because for altitude, achieving a very tight altitude distribution is not sufficient; the mean altitude must also be high for the low end of the distribution to have sufficient timeline margin. Regardless of the length of the trajectory flown, the goal is to minimize the standard deviation to achieve a tight distribution. While for a given mission the target location on the ground may be fixed, during mission planning the entry point can adjusted to accommodate the optimal trajectory length.

The overall performance objective is simply the sum of J_h and J_s , posed as a minimization problem

$$\min J = -\bar{h}(v_f) + w_h \sigma_h(v_f) + w_s \sigma_s(v_f) \quad (4.17)$$

Equation (4.17) may also be interpreted as the sum of a performance objective (mean altitude) and a robustness objective (the standard deviations). Regardless of the interpretation, the weights offer a simple way to adjust closed-loop performance. Due to the nonlinear dynamics, the state distribution will not remain normally distributed; nevertheless we assume that standard deviations remain an appropriate measure of the spread of the terminal distribution. That is, that a reduction in standard deviation will lead to a tighter distribution regardless of the shape of the distribution. Either variances or standard deviations may be used in Eq. (4.17). In practice, appropriate choices of the weights can produce equivalent results, but standard deviations are preferred because they are in the same units as the state variables, and allow for more natural interpretations of the weights. An additional consideration is that the square root is not differentiable at zero, but in practice, the terminal standard deviations are never exactly zero. Additionally, a small term $\epsilon \ll 1$ can be added

to safeguard against this issue in the event the terminal covariance matrix is only positive semidefinite.

4.4 The Robust Optimal Guidance Problem

In summary, the robust optimal guidance problem (ROGP) is to determine $u[v_0, v_f]$, parametrized by $u_{\text{ref}}(v)$ and $K(v)$, that minimizes

$$\begin{aligned}
\min J &= -\bar{h}(v_f) + w_h \sigma_h(v_f) + w_s \sigma_s(v_f) \\
&\text{subject to} \\
\mathbf{x}'_{\text{lon}}(v, \mathbf{p}) &= \mathbf{f}_v(v, \mathbf{x}_{\text{lon}}(v, \mathbf{p}), u(v), \mathbf{p}), \quad \mathbf{x}_{\text{lon}}(v_0, \mathbf{p}) = \mathbf{x}_0(\mathbf{p}) \\
\mathbf{p} &\sim \mathcal{N}(\mathbf{0}, C_{\mathbf{p}}) \\
0 &\leq u_{\text{ref}}(v) \leq 1 \quad \forall v \in [v_0, v_f]
\end{aligned} \tag{4.18}$$

The robust optimal guidance solution consists of the reference control and the reference trajectory, and the feedback gains. The reference trajectory is defined to be the mean trajectory, $\bar{\mathbf{x}}_{\text{lon}}(v)$, rather than the nominal trajectory, $\mathbf{x}_{\text{lon}}(v, \mathbf{0})$.

The succinct form of Eq. (4.17) perhaps belies the challenging, highly nonlinear nature of the objective functional, which features multiple integrals over the uncertainty space

$$\begin{aligned}
\min J &= - \int_{\mathbb{R}^{n_p}} h(v_f, \mathbf{p}) \mu(\mathbf{p}) d\mathbf{p} \\
&\quad + w_h \left[\int_{\mathbb{R}^{n_p}} (h(v_f, \mathbf{p}) - \bar{h}(v_f))^2 \mu(\mathbf{p}) d\mathbf{p} \right]^{\frac{1}{2}} \\
&\quad + w_s \left[\int_{\mathbb{R}^{n_p}} (s(v_f, \mathbf{p}) - \bar{s}(v_f))^2 \mu(\mathbf{p}) d\mathbf{p} \right]^{\frac{1}{2}}
\end{aligned} \tag{4.19}$$

where we recall that $\mu = \mathcal{N}(\mathbf{0}, C_{\mathbf{p}})$ is the probability density function of $\mathbf{p}$. The following

chapter will discuss the use of the unscented transform to form a tractable, finite-dimensional approximation to the objective functional. Part of the motivation behind using the mean state to define the reference trajectory is that, if the unscented transform estimates the mean range accurately, then by using the mean range as the target range, the mean range error should be very nearly zero in the Monte Carlo assessment results.

Chapter 5

Solving the Robust Optimal Guidance Problem

Solving the robust optimal guidance problem presents a number of challenges. In this chapter we first discuss our choice of uncertainty quantification method used to compute the mean and standard deviation of key variables. Next we present the differential dynamic programming algorithm used to numerically solve the guidance problem. Finally, we detail several modifications to the problem to make it amenable to solution via DDP.

5.1 Uncertainty Quantification

A key issue in solving the robust optimal guidance problem is computing the expected values and standard deviations in the objective functional and feedback terms. Broadly speaking, uncertainty quantification (UQ) methods trade between accuracy and the amount of computation required. For example, linear covariance propagation is one of the most efficient UQ methods for computing the first two probability moments, but its accuracy

depends on the nonlinearity of the system dynamics. At the other extreme, Monte Carlo simulation can estimate higher order moments to arbitrary accuracy, but may require a huge number of samples in order to do so. Since UQ will be performed at each optimal control solver iteration, the method chosen must strike a careful balance these two aspects. For very fast but inaccurate methods, the solution may not perform as expected in a higher fidelity UQ, such as Monte Carlo simulation, and the benefit of the approach may be diminished or lost entirely. On the other hand, accurate methods will result in very long solution times. In this work, the unscented transform (UT) [41] is chosen to compute the required statistics. The following subsections review linear covariance and unscented transform methods. Then, via numerical example, we demonstrate that the saturation nonlinearity, introduced by the feedback control formulation Eq. (4.4), is more accurately accounted for using the unscented transform.

5.1.1 Linear Covariance

In the linear covariance methodology, the entry dynamics are linearized around a nominal state-control trajectory, and the mean trajectory is assumed to be equal to the nominal trajectory, i.e., $\mathbb{E}[\mathbf{x}(v, \mathbf{p})] \approx \mathbf{x}(v, \mathbf{0})$. Compared with Eq.(3.24), we see this neglects a term related to the product of the covariance matrix and the trajectory curvature. Based on our definition of $\mathbf{p}$, let

$$\mathbf{y} = \begin{bmatrix} \bar{\mathbf{x}} \\ \mathbf{0} \end{bmatrix} + \mathbf{p} \quad (5.1)$$

$$\mathbf{y}' = \begin{bmatrix} \mathbf{f}_v(v, \mathbf{x}, u, \mathbf{p}) \\ \mathbf{0} \end{bmatrix} = \mathbf{f}_y(v, \mathbf{y}, u) \quad (5.2)$$

The covariance matrix is propagated by integrating the Lyapunov differential equation

$$C'_{\mathbf{y}}(v) = A(v)C_{\mathbf{y}}(v) + C_{\mathbf{y}}(v)A^T(v) \quad (5.3)$$

$$A(v) = \frac{\partial \mathbf{f}_{\mathbf{y}}}{\partial \mathbf{y}} + \frac{\partial \mathbf{f}_{\mathbf{y}}}{\partial u} \frac{\partial u}{\partial \mathbf{y}} \quad (5.4)$$

subject to the initial condition $C_{\mathbf{y}}(v_0) = C_{\mathbf{p}}$. The partial derivatives are evaluated along the nominal trajectory. Note that the linearization around the nominal trajectory is performed using the dynamics written with respect to velocity given in Eq. (4.1).

5.1.2 Unscented Transform

The unscented transform is a method to approximate the first two moments of a nonlinear transformation of a probability distribution. Consider a scalar quantity $q \in \mathbb{R}$ resulting from a nonlinear transformation $q = F(\mathbf{y})$ of a random vector $\mathbf{y} \in \mathbb{R}^n$ with known mean $\bar{\mathbf{y}}$ and covariance $C_{\mathbf{y}}$. A set of $2n + 1$ sigma points and associated weights are computed

$$\begin{aligned} \mathbf{y}_0 &= \bar{\mathbf{y}} \\ \mathbf{y}_i &= \bar{\mathbf{y}} + \left(\sqrt{(\alpha + n)C_{\mathbf{y}}} \right)_i, \quad i = 1, \dots, n \\ \mathbf{y}_{i+n} &= \bar{\mathbf{y}} - \left(\sqrt{(\alpha + n)C_{\mathbf{y}}} \right)_i, \quad i = 1, \dots, n \\ w_0 &= \frac{\alpha}{\alpha + n} \\ w_i &= w_{i+n} = \frac{1}{2(\alpha + n)}, \quad i = 1, \dots, n \end{aligned}$$

where α is a scaling parameter, and $\left(\sqrt{(\alpha + n)C_{\mathbf{y}}} \right)_i$ is the i^{th} column of the matrix square root of $(\alpha + n)C_{\mathbf{y}}$. The sigma points are then mapped through the transformation

$$q_i = F(\mathbf{y}_i), \quad i = 0, \dots, 2n \quad (5.5)$$

and finally, the mean and variance are estimated using the weights and transformed sigma points

$$\bar{q} \approx \sum_{i=0}^{2n} w_i q_i$$

$$\sigma_q \approx \left(\sum_{i=0}^{2n} w_i (q_i - \bar{q})^2 \right)^{\frac{1}{2}}$$

In our application, $\mathbf{y} = \mathbf{p}$, the nonlinear transformation is the integration of the longitudinal entry dynamics to the terminal velocity, and the quantities of interest are downrange distance and altitude. For a given covariance matrix and a chosen value of α , the ensemble of sigma points may be collected into a single extended vector,

$$\mathbf{x}_{\text{UT}} = \begin{pmatrix} \mathbf{x}^{(0)} \\ \vdots \\ \mathbf{x}^{(2n)} \end{pmatrix} \quad (5.6)$$

and the ROGP is now a deterministic problem in the extended state space. When applying the unscented transform, the scaling parameter α is important in precisely estimating the statistics. For any value of α , the sigma point distribution has the same mean and covariance as the initial distribution. Increasing the value of α places the sigma points $\mathbf{y}_i$ further from the nominal sigma point $\mathbf{y}_0$ and reduces their weight. Using a small value of α results in sigma points with only small deviations from the nominal, and for the entry guidance problem in this paper, the effects of the controller and saturation nonlinearity will not be accurately quantified. In our numerical studies, it appears that no single value of α minimizes estimation errors for all control profiles and all quantities of interest. However, it is important to recall that estimating statistics is not the purpose of the proposed trajectory optimization. So long as the reference trajectories designed using the UT confer benefits in Monte Carlo simulation, the UT statistics are sufficiently accurate, as will be demonstrated by the results

of the numerical assessment.

5.1.3 Linear Covariance vs Unscented Transform Example

In this subsection we present a numerical example demonstrating the shortcoming of linear covariance propagation for our application to closed-loop entry guidance. We simulate the performance of an MSL class entry vehicle using the feedback control law defined by Eq. (4.4). The example considers a single, non-optimized reference control and trajectory, and two sets of static feedback gains, $K_1 = [0, 0, 0]^T$ and $K_2 = [0.0725, -0.025, -0.004]^T$. K_1 corresponds to the open-loop case. The ballistic coefficient at the initial velocity is $\beta = 120 \text{ kg/m}^3$ and $L/D = 0.24$. The reference control is a linear-in-velocity ramp from 0.35 to 1. Additional details such as mean initial state and state covariance are given in Chapter 6.

The terminal statistics are computed with three different UQ methods: linear covariance propagation (LC), unscented transform (UT), and Monte Carlo (MC). While the MC statistics are themselves estimates of the true mean and variance, accurate estimates can be by using a sufficiently large sample size. Thus they are the values against which statistics computed by LC and UT are compared. The UT scaling parameter $\alpha = 15$ is used; sensitivity to this parameter is discussed below. Table 5.1 presents the terminal statistics for the results using K_1 ; in this open-loop problem both LC and UT perform well. LC underestimates mean altitude by 1.2 km and altitude standard deviation by 200 m, underestimates mean range by 1 km, and overestimates range standard deviation by 1.1 km. In contrast, the UT overestimates the mean altitude by 160 m, the altitude standard deviation by 65 m, underestimates mean range by 800 m, and overestimates range standard deviation by 180 m. While the UT estimates are better in each variable, the LC estimates could be considered sufficient for the purposes of optimization.

Table 5.2 presents the terminal statistics for the results using K_2 . Linear propagation

	Mean Altitude (km)	Std Altitude (km)	Mean Range (km)	Std Range (km)
Linear Propagation	10.566	0.739	307.367	25.747
Unscented Transform	11.917	0.985	307.707	24.780
Monte Carlo	11.754	0.920	308.485	24.596

Table 5.1: Comparison of UQ methods in an open-loop scenario

	Mean Altitude (km)	Std Altitude (km)	Mean Range (km)	Std Range (km)
Linear Propagation	10.742	0.189	307.367	3.270
Linear Prop, No Sat	10.948	0.293	307.367	2.437
Unscented Transform	11.938	1.064	302.667	3.511
Monte Carlo	11.820	0.941	301.790	3.163

Table 5.2: Comparison of UQ methods in an closed-loop scenario

is listed twice; the first case linearizes the dynamics including the saturated control law defined in Eq. (4.4), while the second linearizes the control law without the saturation nonlinearity. In essence, the latter choice assumes the feedback control is not subject to the control limits, and as a result overpredicts the efficacy of the controller, as seen by the large underpredictions of the standard deviation of both altitude and range. Like the open-loop results, both linearizations once again underpredict the mean altitude substantially. The UT overpredicts the mean range by 900 m and otherwise is once again accurate to within a few hundred meters or less for the remaining statistics. Note that this control profile has significant margin throughout the trajectory except close to the final velocity; in this case control saturation is likely not as significant as for reference control profiles that spend more time at or near the control limits. The UT is expected to produce greater benefits over LC in such scenarios.

Figure 5.1 shows how the UT-estimated means and standard deviations vary with dif-

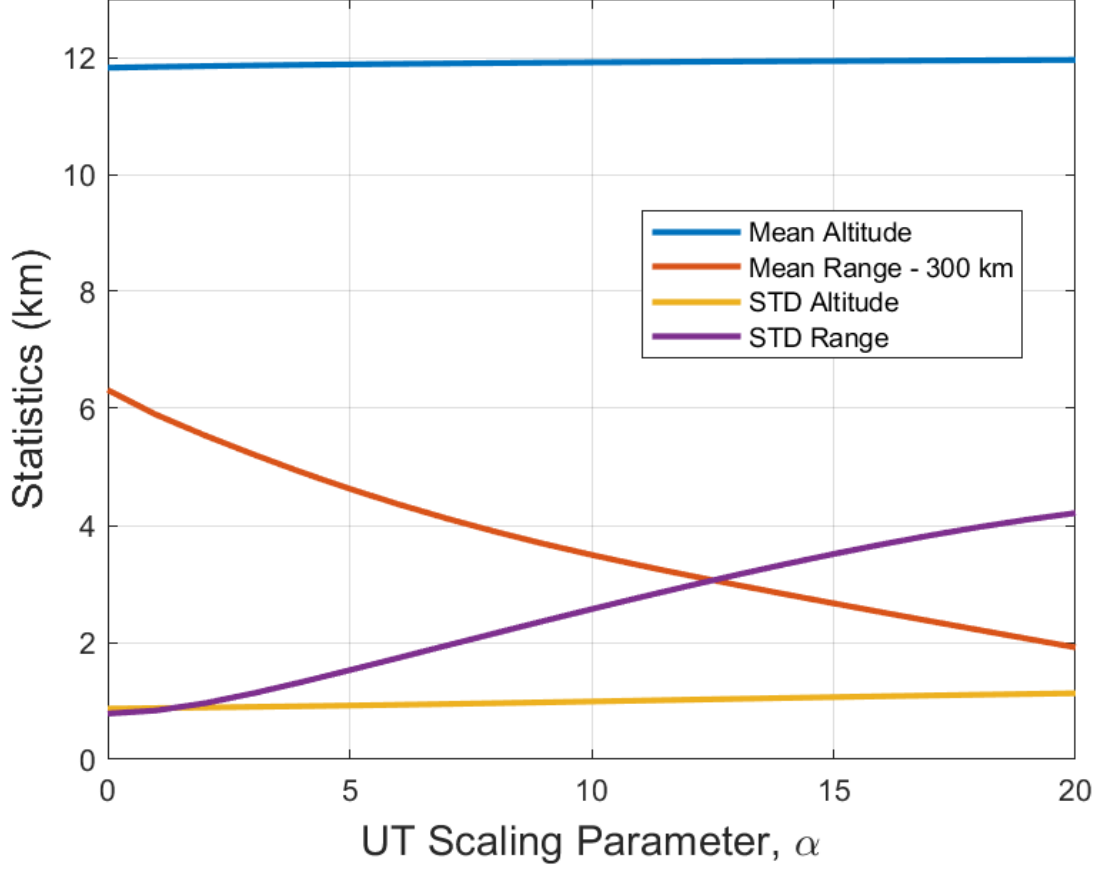


Figure 5.1: Sensitivity of statistics to the unscented transform scaling parameter

ferent choices of the scaling parameter. Note that the mean range has been shifted down 300 km to fit it in the same figure. The altitude statistics are quite stable and vary only slightly with α , while the effect is much stronger for range. This is a general phenomenon we have observed across numerous control profiles; as a result we recommend selection of α based on range considerations. Both the mean range and its standard deviation vary monotonically with α . In this example, the estimation error in $\bar{s}$ is minimized for $\alpha = 20$, while σ_s is approximately optimized for $\alpha = 13$. Our choice $\alpha = 15$ strikes a balance between these two. While the tuning parameter α is important, we note that in this example the UT estimates of $\bar{h}$, σ_h , and $\bar{s}$ are more accurate than either LC method for all of α values considered, and σ_s is better for several values. This example is meant to motivate our choice of the UT over LC; more detailed examples utilizing optimal solutions are presented in Chapter 7.

5.2 Differential Dynamic Programming

Differential Dynamic Programming (DDP) is a shooting method originally devised to solve unconstrained nonlinear optimal control problems that has since seen numerous extensions, including to constrained problems [32, 42, 43, 44, 45], and stochastic problems [33, 46, 47, 48]. DDP is a second-order algorithm that iteratively solves for a locally optimal trajectory and control starting from a nominal trajectory and control. Like Newton’s method, DDP can achieve quadratic convergence rates [49].

In this section by reviewing the control-limited DDP method, proposed in Ref. [32], that will be used as the basis for solving the robust optimal guidance problem. The following subsections discuss modifications to the algorithm and the problem formulation that will finally enable the solution of the problem. In the first subsection we present an extension of the algorithm that allows for the joint optimization of static design parameters. The following subsection proposes a simplification that vastly reduces computational time and memory requirements at the expense of lower convergence rate. This simplification is instrumental in solving the large-scale problem. Next, we discuss a reformulation of the performance objective into a Lagrange (or running) cost instead of a Mayer (or terminal) cost. Finally, we present a differentiable approximation to the saturation nonlinearity in the controller, Eq. (4.4). Since DDP requires derivatives of the dynamic equations with respect to the reference control, state variables, and potentially the feedback gains, replacing the non-differentiable saturation function is essential to the numerical solution.

This particular DDP algorithm is formulated in discrete time, so the continuous time dynamics must be discretized. In our numerical examples, Euler integration is used:

$$\mathbf{x}_{i+1} = \mathbf{f}(v, \mathbf{x}_i, \mathbf{u}_i) = \mathbf{x}_i + \mathbf{x}'_i \Delta v \quad (5.7)$$

A trajectory $\{\mathbf{X}, \mathbf{U}\}$ is a sequence of controls $\mathbf{U} = \{\mathbf{u}_0, \mathbf{u}_1, \dots, \mathbf{u}_{N-1}\}$ and corresponding states $\mathbf{X} = \{\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_N\}$ determined by integrating (5.7) from $\mathbf{x}_0$. Although the ROBG as posed in Eq. (4.17) considers only a terminal cost, in this section we consider a generic cost function J consisting of a sum of running costs l and a terminal cost l_N :

$$J(\mathbf{x}_0, \mathbf{U}) = \sum_{i=0}^{N-1} l(\mathbf{x}_i, \mathbf{u}_i) + l_N(\mathbf{x}_N) \quad (5.8)$$

Let $\mathbf{U}_i$ be the tail of the control sequence, $\{\mathbf{u}_i, \mathbf{u}_{i+1}, \dots, \mathbf{u}_{N-1}\}$, and the cost-to-go J_i , defined as the partial sum of costs from i to N is

$$J_i(\mathbf{x}, \mathbf{U}_i) = \sum_{j=i}^{N-1} l(\mathbf{x}_j, \mathbf{u}_j) + l_N(\mathbf{x}_N) \quad (5.9)$$

The value function at timestep i is the optimal cost-to-go at $\mathbf{x}$

$$V_i(\mathbf{x}) = \min_{\mathbf{U}_i} J(\mathbf{x}, \mathbf{U}_i) \quad (5.10)$$

and at the final timestep the value function is equal to the terminal objective. The dynamic programming principle reduces the problem of minimization over $\mathbf{U}_i$ to a sequence of minimization problems over u at each timestep

$$V(\mathbf{x}) = \min_{\mathbf{u}} [l(\mathbf{x}, \mathbf{u}) + V^+(\mathbf{f}(\mathbf{x}, \mathbf{u}))] \quad (5.11)$$

where V^+ is the value at the next time step. Let the pseudo-Hamiltonian $Q(\delta\mathbf{x}, \delta\mathbf{u})$ be the change in value function as a function of perturbations to the pair $(\mathbf{x}, \mathbf{u})$:

$$Q(\delta\mathbf{x}, \delta\mathbf{u}) = l(\mathbf{x} + \delta\mathbf{x}, \mathbf{u} + \delta\mathbf{u}) + V^+(\mathbf{f}(\mathbf{x} + \delta\mathbf{x}, \mathbf{u} + \delta\mathbf{u})) \quad (5.12)$$

The second-order expansion of Q is given by

$$Q_{\mathbf{x}} = l_{\mathbf{x}} + \mathbf{f}_{\mathbf{x}}^T V_{\mathbf{x}}^+ \quad (5.13)$$

$$Q_{\mathbf{u}} = l_{\mathbf{u}} + \mathbf{f}_{\mathbf{u}}^T V_{\mathbf{x}}^+ \quad (5.14)$$

$$Q_{\mathbf{xx}} = l_{\mathbf{xx}} + \mathbf{f}_{\mathbf{x}}^T V_{\mathbf{xx}}^+ \mathbf{f}_{\mathbf{x}} + V_{\mathbf{x}}^+ \mathbf{f}_{\mathbf{xx}} \quad (5.15)$$

$$Q_{\mathbf{ux}} = l_{\mathbf{ux}} + \mathbf{f}_{\mathbf{u}}^T V_{\mathbf{xx}}^+ \mathbf{f}_{\mathbf{x}} + V_{\mathbf{x}}^+ \mathbf{f}_{\mathbf{ux}} \quad (5.16)$$

$$Q_{\mathbf{uu}} = l_{\mathbf{uu}} + \mathbf{f}_{\mathbf{u}}^T V_{\mathbf{xx}}^+ \mathbf{f}_{\mathbf{u}} + V_{\mathbf{x}}^+ \mathbf{f}_{\mathbf{uu}} + \lambda I \quad (5.17)$$

where the subscripts denote partial derivatives with respect to that quantity, and the final term in each of the Hessian equations (5.15)-(5.17) are tensor-vector contractions, and λ is a regularization parameter.

The optimal control modification $\delta \mathbf{u}^*$ for some perturbation $\delta \mathbf{x}$ is a locally-linear feedback control $\delta \mathbf{u}^* = \mathbf{k} + K \delta \mathbf{x}$ obtained by minimizing the quadratic model subject to linear bounds on the controls

$$\mathbf{k} = \arg \min_{\delta \mathbf{u}} Q(\delta \mathbf{x}, \delta \mathbf{u}) \quad (5.18)$$

subject to

$$\mathbf{u}_{\min} \leq \mathbf{u} + \delta \mathbf{u} \leq \mathbf{u}_{\max} \quad (5.19)$$

In general, the Projected-Newton method proposed in [32] is used to solve this problem and any controls at one of its limits have their associated directions in the feedback gain $K = -Q_{\mathbf{uu}}^{-1} Q_{\mathbf{ux}}$ set to zero. When the control is a scalar (as in the problem this algorithm will be used to solve in this dissertation), the solution is given in closed form by

$$k^* = \text{sat}_{[u_{\min}-u, u_{\max}-u]} \left(-\frac{Q_u}{Q_{uu}} \right) \quad (5.20)$$

and $K = \mathbf{0}$ whenever u is saturated. Substituting the optimal control into the expansion of

Q , a quadratic model of V is obtained with derivatives

$$\begin{aligned} V_{\mathbf{x}} &= Q_{\mathbf{x}} + K^T Q_{\mathbf{uu}} \mathbf{k} + Q_{\mathbf{ux}}^T \mathbf{k} + K^T Q_{\mathbf{u}} \\ V_{\mathbf{xx}} &= Q_{\mathbf{xx}} + K^T Q_{\mathbf{uu}} K + Q_{\mathbf{ux}}^T K + K^T Q_{\mathbf{ux}}. \end{aligned} \quad (5.21)$$

Note these derivatives are different from those given in Ref. [32], but agree with other DDP developments [50, 44]. The backward pass is performed by initializing V and its derivatives with the value and derivatives of the terminal objective, then recursively computing the optimal control policy and Eq. (5.21).

The forward pass uses the newly computed control policy to integrate the new trajectory, subject to a backtracking line search

$$\hat{\mathbf{x}}_0 = \mathbf{x}_0 \quad (5.22)$$

$$\hat{\mathbf{u}}_i = \mathbf{u}_i + \epsilon \mathbf{k}_i + K_i(\hat{\mathbf{x}}_i - \mathbf{x}_i) \quad (5.23)$$

$$\hat{\mathbf{x}}_{i+1} = \mathbf{f}(v, \hat{\mathbf{x}}_i, \hat{\mathbf{u}}_i) \quad (5.24)$$

where ϵ is a search parameter initialized to 1 and reduced until the Armijo rule [51] is satisfied, i.e., the value function shows sufficient improvement, as measured by the ratio of actual improvement to expected improvement

$$\frac{V(\mathbf{U} + \epsilon \delta \mathbf{U}) - V(\mathbf{U})}{\Delta V} \leq \epsilon_{improve} \quad (5.25)$$

$$\Delta V = \sum_{i=0}^N (-\epsilon \mathbf{k}_i^T Q_{\mathbf{u}} - \alpha^2 \mathbf{k}_i^T Q_{\mathbf{uu}} \mathbf{k}_i) \quad (5.26)$$

where $\epsilon_{improve}$ is the minimum improvement ratio, and is generally taken to be quite small

[52]. When the step is accepted, the regularization parameter λ is decreased

$$\kappa = \min\left(\frac{\kappa}{\eta}, \frac{1}{\eta}\right) \quad (5.27)$$

$$\lambda = \kappa\lambda \quad (5.28)$$

where κ is an additional parameter used to schedule λ , initialized to 1 in the first iteration, and $\eta > 1$ is a constant value for scaling κ . In the event the smallest acceptable stepsize $\epsilon = \epsilon_{\min}$ fails to produce sufficient improvement, λ is increased according to

$$\kappa = \max(\kappa\eta, \eta) \quad (5.29)$$

$$\lambda = \max(\kappa\lambda, \lambda_{\min}) \quad (5.30)$$

and the backward pass is repeated to find a new control update. If Eq. (5.25) cannot be satisfied and the update Eq. (5.30) results in $\lambda > \lambda_{\max}$, the algorithm terminates early. The backward-forward iterations are repeated until convergence to a locally optimal trajectory. The algorithm terminates when either the cost improvement magnitude is small, or the largest norm of the gradient $Q_{\mathbf{u}}$ is small, i.e.,

$$|\Delta J| < \epsilon_J \quad (5.31)$$

$$\max_i \|Q_{\mathbf{u}}^{(i)}\| < \epsilon_{tol} \quad i = 1, 2, \dots, N \quad (5.32)$$

where ϵ_J and ϵ_{tol} are user-defined threshold values.

5.2.1 Optimal Design Parameters

Previously, we considered the state vector to be a function of the independent variable and the uncertainty vector $\mathbf{p}$. However, we may wish to treat the state as a function of some ad-

ditional constant design parameters, $\mathbf{d}$, $\mathbf{x}(v, \mathbf{p}, \mathbf{d})$. The design parameters are different from the uncertain parameters because they do not increase the number of sigma points required. Additionally, in our formulation, the design parameters are appended only to the extended vector of all sigma points. Thus, given uncertainty dimension n_p , and design parameter dimension n_d , the total number of states is $n_p(2n_p + 1) + n_d$. We define an augmented state vector $\mathbf{x}_a = [\mathbf{x}_{\text{UT}}, \mathbf{d}]^T$ and augment the entry dynamics with trivial dynamics

$$\mathbf{x}'_a = \begin{bmatrix} \mathbf{x}'_{\text{UT}} \\ \mathbf{0} \end{bmatrix} \quad (5.33)$$

DDP is now applied to the augmented state vector with an additional modification that we will explain. In [53], Lagrange multipliers are used to enforce terminal constraints in a continuous-time DDP formulation. At the end of each backward pass, the Lagrange multipliers are updated to reduce the cost function. Although the setting was quite different, we notice that the form of the update to the Lagrange multipliers actually applies to any constant parameter and is the inspiration for our proposed modification to the discrete-time algorithm.

During the backward pass, the first and second partial derivatives of the value function are computed recursively from the final velocity to the initial velocity. The gradient and hessian of the value function can be written in block notation as

$$V_{\mathbf{x}_a} = \begin{bmatrix} V_{\mathbf{x}} & V_{\mathbf{d}} \end{bmatrix} \quad (5.34)$$

$$V_{\mathbf{x}_a \mathbf{x}_a} = \begin{bmatrix} V_{\mathbf{xx}} & V_{\mathbf{xd}} \\ V_{\mathbf{dx}} & V_{\mathbf{dd}} \end{bmatrix} \quad (5.35)$$

The quadratic expansion of the value function at the initial velocity is

$$V(v_0, \mathbf{x}_a + \delta \mathbf{x}_a) = V(v_0, \mathbf{x}_a) + V_{\mathbf{x}_a}^T \delta \mathbf{x}_a + \frac{1}{2} \delta \mathbf{x}_a^T V_{\mathbf{x}_a \mathbf{x}_a} \delta \mathbf{x}_a \quad (5.36)$$

At the end of the backward pass, we set $\delta \mathbf{x}_a = \mathbf{0}$ because we are optimizing a trajectory from a fixed initial state. However, when including design parameters that can be altered, we can derive an update $\delta \mathbf{x}_a = [\mathbf{0}, \delta \mathbf{d}]^T$ to reduce the value function based on the expansion (5.36), which simplifies to

$$V(v_0, \mathbf{d} + \delta \mathbf{d}) = V(v_0, \mathbf{d}) + V_{\mathbf{d}}^T \delta \mathbf{d} + \frac{1}{2} \delta \mathbf{d}^T V_{\mathbf{d} \mathbf{d}} \delta \mathbf{d} \quad (5.37)$$

The $\delta \mathbf{d}$ that minimizes Eq. (5.37) is determined by the solution to $V_{\mathbf{d}} + V_{\mathbf{d} \mathbf{d}} \delta \mathbf{d} = 0$, or

$$\delta \mathbf{d} = -V_{\mathbf{d} \mathbf{d}}^{-1} V_{\mathbf{d}} \quad (5.38)$$

In practice, the update must be treated like the control updates by using a line search $\delta \mathbf{d} = -\epsilon_{\mathbf{d}} V_{\mathbf{d} \mathbf{d}}^{-1} V_{\mathbf{d}}$. While a different line search parameter could be utilized, in our implementation we update the control and design parameters with the same stepsize, i.e., $\epsilon_{\mathbf{d}} = \epsilon$. There is no guarantee that $V_{\mathbf{d} \mathbf{d}}$ is positive definite, and thus regularization of the matrix may be required, i.e., $\delta \mathbf{d} = -\epsilon_{\mathbf{d}} \tilde{V}_{\mathbf{d} \mathbf{d}}^{-1} V_{\mathbf{d}}$ where $\tilde{V}_{\mathbf{d} \mathbf{d}} = V_{\mathbf{d} \mathbf{d}} + \lambda I$.

5.2.2 Second Order Dynamics Derivatives

The second derivatives $\mathbf{f}_{\mathbf{xx}}$, $\mathbf{f}_{\mathbf{ux}}$ in Eqs. (5.15)) and (5.16) are $n \times n \times n$ and $m \times n \times n$ tensors, respectively, that must either be stored at N timesteps, or else recomputed in the event the backward pass must be repeated with increased regularization. For large n , both computing and storing these quantities is problematic. Eliminating these terms entirely, as is done in the iterative Linear Quadratic Gaussian method [33], leads to poor performance for the problem

under consideration for some values of (w_h, w_s) and some guesses at the optimal control. In particular, the first-order algorithm has a slower convergence rate and a smaller radius of convergence, and different guesses for the initial control sequence can lead to different local minima. In contrast, the full second-order DDP algorithm consistently converges to a single minimum from most initial guesses. We propose a compromise in which only the $n \times m \times m$ tensor $\mathbf{f}_{\mathbf{uu}}$ is retained in the above equations. Numerical experiments, an example of which is presented in Section 7.4, suggest the convergence rate with this modification approaches that of the full algorithm with all second-order terms included at a huge reduction in computation time. For the situation at hand, in which the number of controls m is fixed and small ($m = 1$), and the augmented state dimension, which depends on the number of uncertainties under consideration, is large ($n = 78$ for $\mathbf{p} \in \mathbb{R}^6$), this modification proves to be enabling for $N \geq 100$. A limited-memory version of the Quasi-Newton approximations to these terms proposed in [54] may also be a good choice to obtain superlinear convergence properties of the approximated second-order algorithm while reducing the memory required.

5.2.3 Objective Functional Reformulation

In order for DDP to be applied to our robust optimal control problem, it is essential to reformulate some or all of the objective terms as running costs rather than terminal costs. This is because the terminal formulation results in singularity of the Hessian of the pseudo-Hamiltonian with respect to the control, and this quantity must be invertible for the algorithm to proceed. The reformulation is based on the relationship

$$J(\mathbf{x}(v_f)) = J(\mathbf{x}(v_0)) + \int_{v_0}^{v_f} \left(\frac{\partial J}{\partial v} + J' \right) dv \quad (5.39)$$

For a fixed initial state $\mathbf{x}(v_0) = \mathbf{x}_0$, the term $J(\mathbf{x}_0)$ is constant and cannot be affected by the control profile. For our particular J in Eq. (4.17), $\frac{\partial J}{\partial v} = 0$. As such,

$$u^* = \arg \min J(\mathbf{x}(v_f)) \quad (5.40)$$

$$= \arg \min \int_{v_0}^{v_f} J' dv \quad (5.41)$$

Thus, we need an expression for J' for the guidance objective defined in Eq. (4.17). Due to linearity of the expectation operator, we have the following relationships for the rate of change of the first two moments of a scalar random variable

$$\frac{d}{dv} \mathbb{E}[x] = \mathbb{E}[x'] \quad (5.42)$$

$$\frac{d}{dv} \mathbb{V}[x] = \mathbb{E}[2xx'] - 2\mathbb{E}[x] \mathbb{E}[x'] \quad (5.43)$$

$$\sigma_x = \mathbb{V}[x]^{\frac{1}{2}} \quad (5.44)$$

$$\frac{d}{dv} \sigma_x = \frac{1}{\sigma_x} (\mathbb{E}[xx'] - \mathbb{E}[x] \mathbb{E}[x']) \quad (5.45)$$

Applying these to altitude and downrange distance, and discretizing the resulting integral, we arrive at the reformulated discrete-time running cost

$$l(\mathbf{x}, \mathbf{u}) = \left(-\mathbb{E}[h'] + \frac{w_h}{\sigma_h} (\mathbb{E}[hh'] - \mathbb{E}[h] \mathbb{E}[h']) + \frac{w_s}{\sigma_s} (\mathbb{E}[ss'] - \mathbb{E}[s] \mathbb{E}[s']) \right) \Delta v \quad (5.46)$$

While reformulating the mean altitude objective is essential to solving the problem, especially when $w_s = w_h = 0$, reformulating the standard deviation terms is not. However, in numerical experiments we found that while both formulations result in the same solution, using running costs for the standard deviations allows DDP to converge in noticeably fewer iteration. Running costs are used in the results in this dissertation.

5.2.4 Saturation

The saturation function in Eq. (4.4) is not differentiable at the saturation limits, which produces numerical difficulties for the DDP algorithm. To circumvent this problem, we substitute the following smooth approximation when solving the robust optimal guidance problem

$$\text{sat}_{[0,1]}(x) \approx \frac{1}{2} + \frac{1}{4M} \log \left(\frac{\cosh(2Mx)}{\cosh(2M(x-1))} \right)$$

where $M > 0$ is a tuning parameter. Larger values of M reduce approximation error but increase the derivative magnitude at the saturation limits. $M = 20$ reduces the maximum error to less than 1% and is used in our implementation. Figure 5.2 shows how the approximation varies with M ; at $M = 20$, the approximation is visually indistinguishable from the true function.

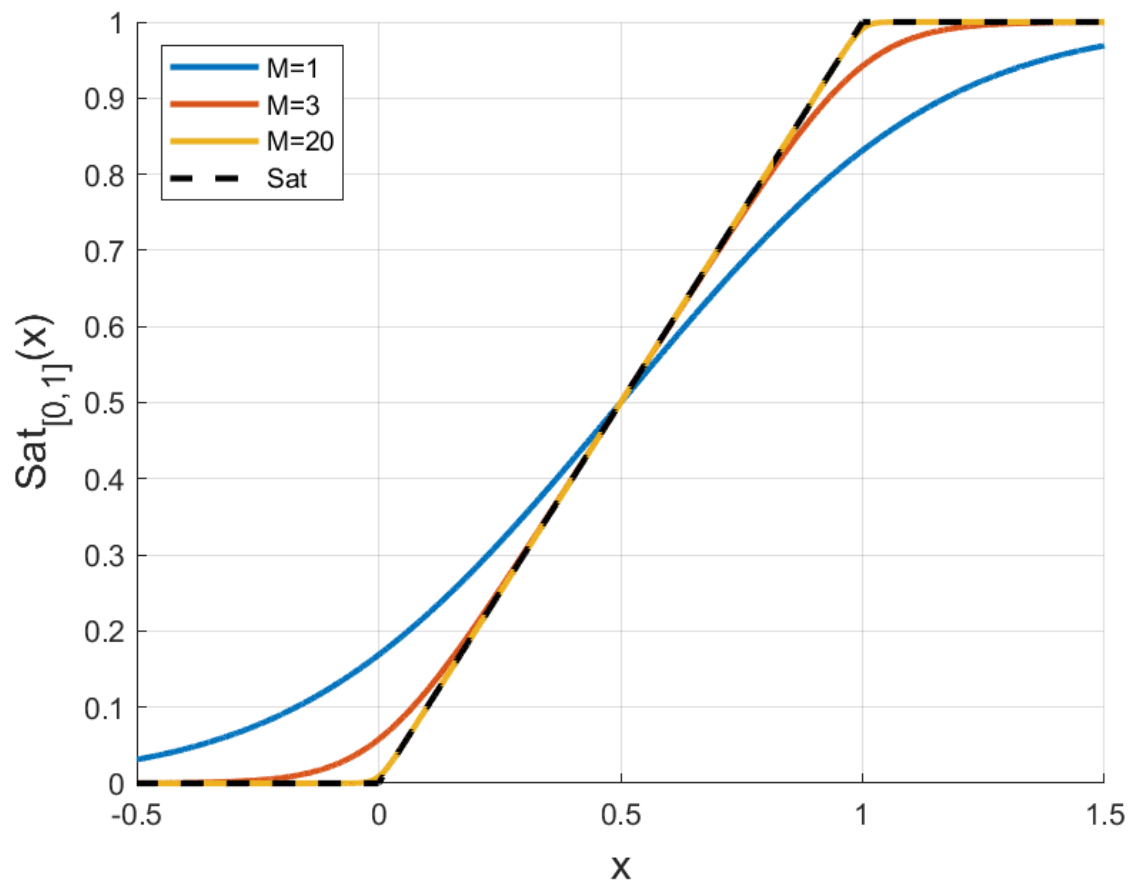


Figure 5.2: The saturation function and smooth approximation for several values of the tuning parameter.

Chapter 6

Guidance Assessment Conditions

6.1 Entry Conditions and Targets

The entry guidance strategy is assessed for an MSL-class vehicle with mass $m = 2804$ kg, and area $S = 15.8 \text{ m}^2$. The entry state and associated standard deviations at the entry interface are given in Table 6.1. By assumption, altitude determines the entry interface point, and thus $\sigma_h = 0$. Given the mean and standard deviation of the entry interface state, a specified prebank angle, and a chosen v_0 at which phase 2 guidance begins, the mean and covariance of the state at v_0 are determined by integrating Eq. (3.11)-(3.14) until $v < v_0$. We assume that the distribution remains approximately normal. In practice this appears to be true; the limited aerodynamic nonlinearities at high altitudes do not cause significant changes to the distribution. Note that the parametric uncertainties are also included in determining the mean and covariance at v_0 .

The resulting mean state and deviations at the start of closed-loop guidance are given in Table 6.2. These are the values used in the solution of the optimal control process. Note that we set $s(v_0) = 0$ for simplicity of discussion, but the total range from the entry

interface to the target is 223 km + the optimal range of the solution to the ROGP. Similar to the altitude at the entry interface, after propagation to a fixed velocity, $\sigma_v = 0$. The 3σ uncertainty in the ballistic coefficient and lift-to-drag ratio is 15%. Atmospheric density uncertainty is modeled with the symmetrized Mars Climate Database profile in Fig. 3.3, with the boundary interpreted as the 3σ value.

State Variable	Mean Value	Standard Deviation	Units
Altitude	120	-	km
Velocity	5505	1	m/s
Range	0	5	km
FPA	-15.5	0.2	deg

Table 6.1: The mean state and standard deviations at the entry interface.

State Variable	Mean Value	Standard Deviation	Units
Altitude	39.5	2.5	km
Velocity	5460	-	m/s
Range	0	10	km
FPA	-10.6	0.25	deg

Table 6.2: The mean state and standard deviations at the start of closed-loop guidance.

6.2 Robust Optimal Control Solution Parameters

Table 6.3 summarizes the algorithm parameters governing the linesearch, regularization, and stopping conditions. The smooth saturation approximation uses $M = 20$. The trajectory is discretized into $N = 250$ velocity steps, with four Euler integration steps per velocity step. The value of N and the number of integration steps were chosen by validating the terminal altitude and downrange distance against adaptive stepsize integration using MATLAB's *ode45* function. The acceptable error tolerance across all sigma points was 100 m.

Parameter	Value
Initial Regularization Value, λ	1
Initial Regularization Multiplier, κ	1
Scale Factor, η	1.8
Minimum Regularization, $\lambda_{\min}$	10^{-6}
Maximum Regularization, $\lambda_{\max}$	10^8
Cost Tolerance, ϵ_j	10^{-7}
Gradient Tolerance, ϵ_{tol}	10^{-4}
Minimum Stepsize, $\epsilon_{\min}$	10^{-3}
Minimum Improvement Ratio, $\epsilon_{\text{improve}}$	10^{-4}

Table 6.3: Algorithm parameters used in all numerical results.

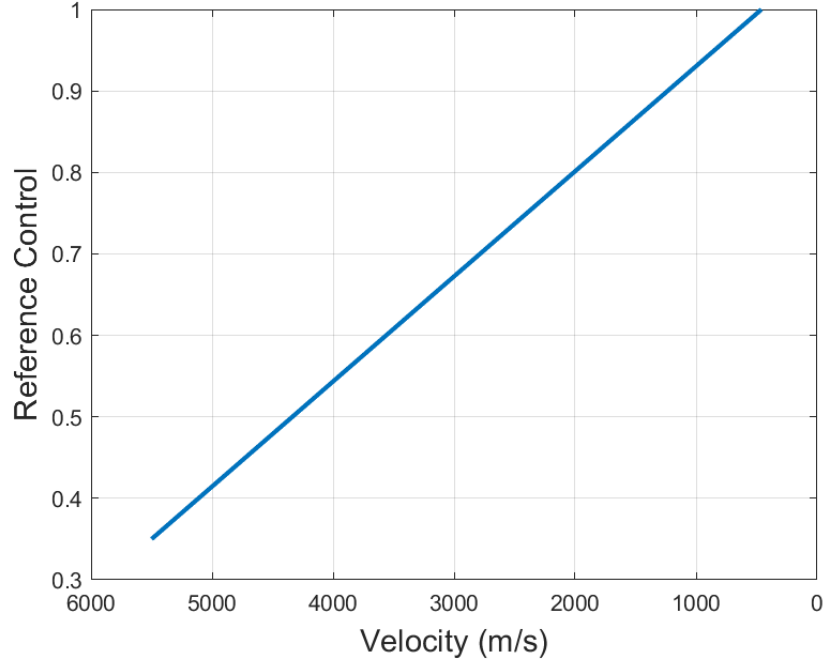


Figure 6.1: The reference control used to initialize DDP.

A trajectory-control pair is required to initialize the DDP algorithm. In all numerical examples, the initial guess at the optimal reference control is the linear ramp depicted in Fig 6.1. This guess is motivated by experience with altitude optimal trajectories, which feature a minimum lift arc followed by a full lift up arc. Such a control has zero margin (in

one direction) at every velocity. Our choice is motivated by a similar progression from lift down to lift up, but more gradual and with significant margin throughout. We remark that the solutions are not sensitive to the initial guess. A numerical demonstration is given in Section 7.4.

All solutions consider static gains. In the first section we consider two sets of gains, $K_1 = [0, 0, 0]$ and $K_2 = [0.13, -0.04, -2.5]$, to highlight the impact of the reference trajectory. K_2 was chosen to provide to minimize the UT-estimated value of σ_s for the initial reference control. Later results use jointly optimized gains. For these cases, the initial guess at the optimal gains is K_2 . In all examples, unless otherwise noted, the DDP algorithm is applied with the simplification proposed in Subsection 5.2.2.

6.2.1 Selecting the UT Parameter

Our earlier example in Section 5.1.3 showed that proper selection of α is important, especially for the range variable. In the numerical assessments presented in the Chapter 7, we consider weight combinations on the grid $w_h \times w_s = [0, 3] \times [0, 3]$. As such, we selected α to minimize the UT-estimation error in σ_s for the optimal solution with $w_h = w_s = 1.5$, i.e., at the center point of the grid. We considered integer values of α between 6 and 20, and solved the ROGP for each value. A Monte Carlo simulation was performed for each α and the optimal choice was $\alpha = 15$.

6.3 Simulation

Unless otherwise noted, each Monte Carlo simulation is conducted with 3000 sample trajectories. The samples are drawn using Latin Hypercube sampling. The equations of motion are simulated in the time domain, and the bank angle command is updated at a rate of 1

Hz. The bank angle magnitude is subject to the same limits applied during optimization, $0^\circ \leq |\sigma| \leq 90^\circ$, and the magnitude of the bank angle rate is limited to $20^\circ/s$. No bank angle acceleration limit is applied. Range error is the distance between the terminal downrange distance defined by the reference trajectory and the downrange distance flown. The primary metrics we will examine are the 3σ -low altitude $= \bar{h} - 3\sigma_h$ and 3σ range error $= 3\sigma_s$ at the final velocity, $v_f = 460\text{m/s}$. The terminal state distributions are generally non-Gaussian, and in practice percentiles are often used to specify mission requirements. However, metrics based on the mean and standard deviation are presented, because these are the values estimated by the unscented transform, which allows for a direct comparison with the Monte Carlo results.

Chapter 7

Guidance Assessment

A series of numerical experiments is conducted to assess the robust optimal guidance. In the first section, the feedback gains are held constant while sweeping over the weights (w_h, w_s) to determine the influence of the reference trajectory alone, in both open- and closed-loop formulations. In the section second we utilize the design parameter optimization outlined in Section 5.2.1 to jointly optimize the feedback gains. The third section presents a detailed look at a single robust optimal solution. Section 4 empirically examines the convergence properties of the DDP algorithm. Section 5 presents a negative result in which joint optimization of velocity-varying gains produces poor results. The final section discusses the results in the context of the MSL and Mars 2020 missions.

7.1 Reference Trajectory Optimization

In this section we solve the ROGP for a variety of objective weights (w_h, w_s) while holding the gains constant. This demonstrates the extent to which the reference trajectory can impact the solution for a fixed set of gains. In the first subsection, we consider the open-loop

($K = K_1 = \mathbf{0}$) scenario to gauge the level of dispersions that must be mitigated by the entry guidance, and the extent to which open-loop covariance shaping can impact performance. MSL entry guidance designers performed a similar exercise [4]. The second subsection repeats the sweep over the weights in a closed-loop scenario with $K = K_2$.

7.1.1 Open-Loop Optimization

Figure 7.1 presents contours of the Monte Carlo estimated 3σ -low altitude and range errors. The 3σ range errors vary from 59 to 66 km, while the 3σ -low altitude varies from 7.8 km to 10.6 km. Increasing the weight w_s reduces range errors as intended, but comes at a penalty to altitude performance. The results also highlight the large magnitude of the terminal state variations arising from the variations in the initial state and model uncertainty, and the limited extent to which the reference trajectory can alter the terminal statistics without feedback.

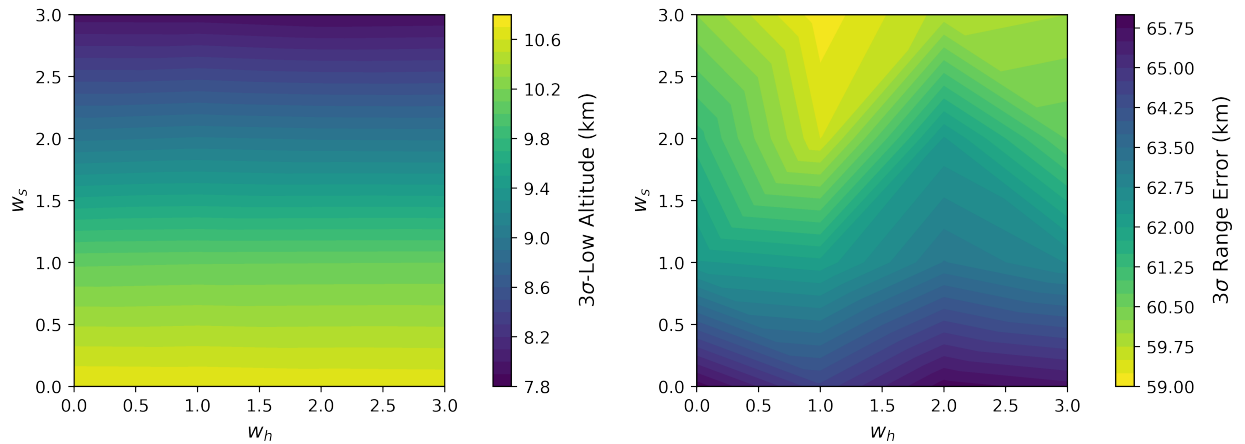


Figure 7.1: Monte Carlo statistics for the open-loop trajectory optimization.

7.1.2 Closed-Loop Optimization

Figure 7.2 presents contours of the Monte Carlo estimated 3σ -low altitude and range errors for the closed-loop guidance with $K = K_2$. Introducing feedback allows the guidance to achieve 3σ range errors as low as 5.6 km while also keeping the 3σ -low altitude above 9 km. This low altitude can be raised to 9.7 km at the expense of growing the 3σ range errors to 14.6 km. The largest range errors occur when optimizing purely for mean altitude, $w_h = w_s = 0$. Figure 7.3 shows the contours of the errors in the UT statistics compared to the Monte

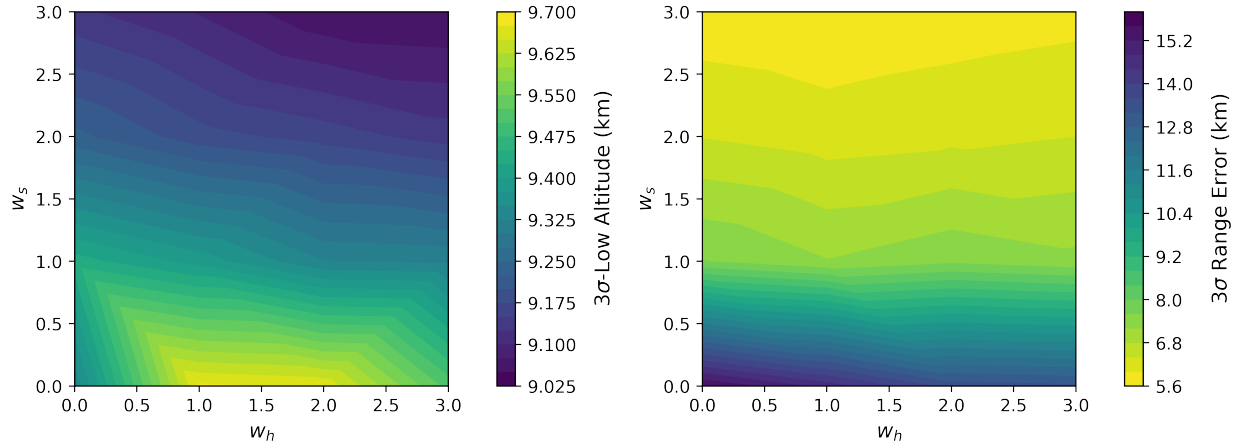


Figure 7.2: Monte Carlo statistics for the closed-loop trajectory optimization with static gains.

Carlo results. The error in 3σ -low altitude varies between a 120 m overestimate and a 360 underestimate. For a majority of (w_h, w_s) pairs, the low altitude is underestimated. Although the estimation errors are generally small, they are large enough to affect the optimality of the solutions. The 3σ -low altitude should be maximized when $w_h = 3$ and $w_s = 0$, but instead the highest 3σ -low altitude is achieved for $w_h = 1.5$. Notice that the suboptimality is also small, the solution for $w_h = 3$ is only 150 m lower than for $w_h = 1.5$. In contrast, the smallest range errors do occur for the $w_h = 0, w_s = 3$.

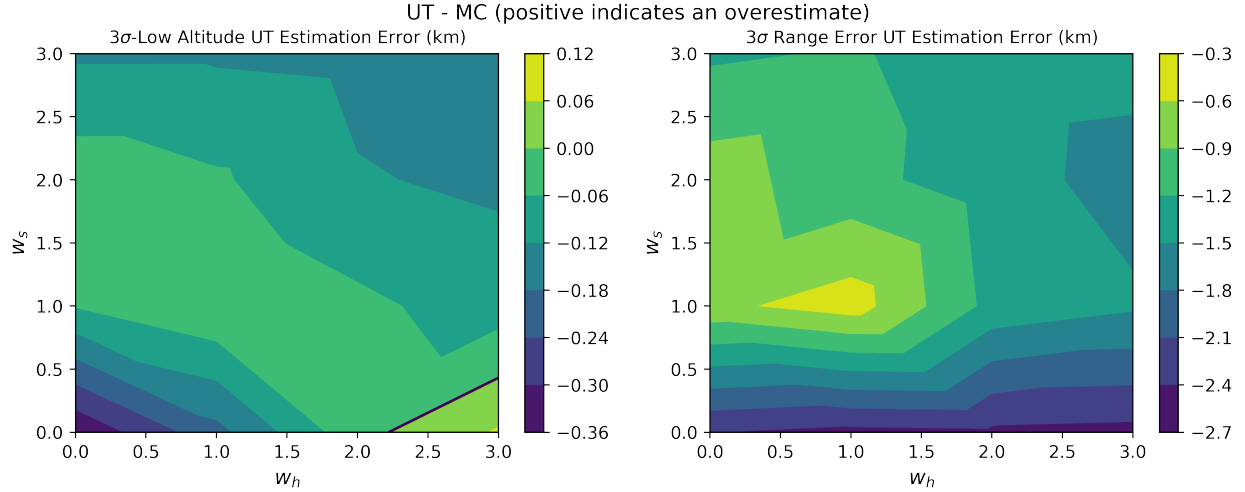


Figure 7.3: Unscented Transform estimation errors relative to Monte Carlo statistics for trajectory optimization with static feedback gains.

7.2 Joint Gain Optimization

In this section, the ROGP is again solved for a variety of weights but now the gains are treated as design parameters, and the DDP modification presented in Subsection 5.2.1 is used to jointly optimize them alongside the reference trajectory and control. As one might intuitively expect, the effect on the range of solutions is much more extreme with joint optimization than without.

Figure 7.4 shows the contours. An additional 800 meters of low altitude can be gained (relative to the previous solutions with unoptimized gains), but at the expense of large 3σ range errors (above 40 km). However, a very tight range distribution of $3\sigma_s = 3$ km can be achieved while still maintaining a 9 km 3σ -low altitude for a variety of weights. The range error stops improving at approximately $w_s = 1$, but the low altitude continues to decrease as w_s is raised. Thus, an effective compromise might be $w_h = w_s = 1$. Figure 7.5 shows the contours of the errors in the UT statistics compared to the Monte Carlo results. The dark line in each plot is the zero contour, where the UT and MC agree with approximately no error. Comparing Fig. 7.5 to the previous error contour, Fig. 7.3, we can see the effect to

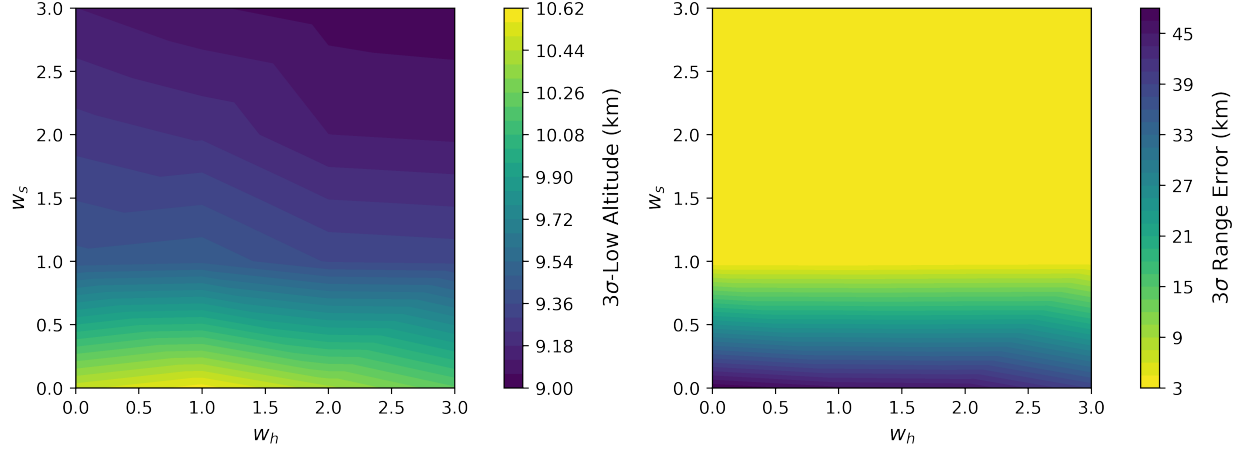


Figure 7.4: Monte Carlo statistics for jointly optimized static feedback gains.

selecting the UT scaling parameter to minimize the estimation error in range error. For the fixed gain results in Fig. 7.3, the range errors were uniformly underestimated, while for the optimized gains, the zero contour passes through $(w_h, w_s) = (1.5, 1.5)$. The error in 3σ -low altitude varies between a 120 m overestimate and a 200 underestimate. Like the fixed gain results, the low altitude is underestimated for a majority of (w_h, w_s) pairs. Estimation errors once again affect the optimality of the solutions. The highest 3σ -low altitude is achieved for $w_h = 1$. Figure 7.6 depicts how the optimal downrange distance resulting from the solution to

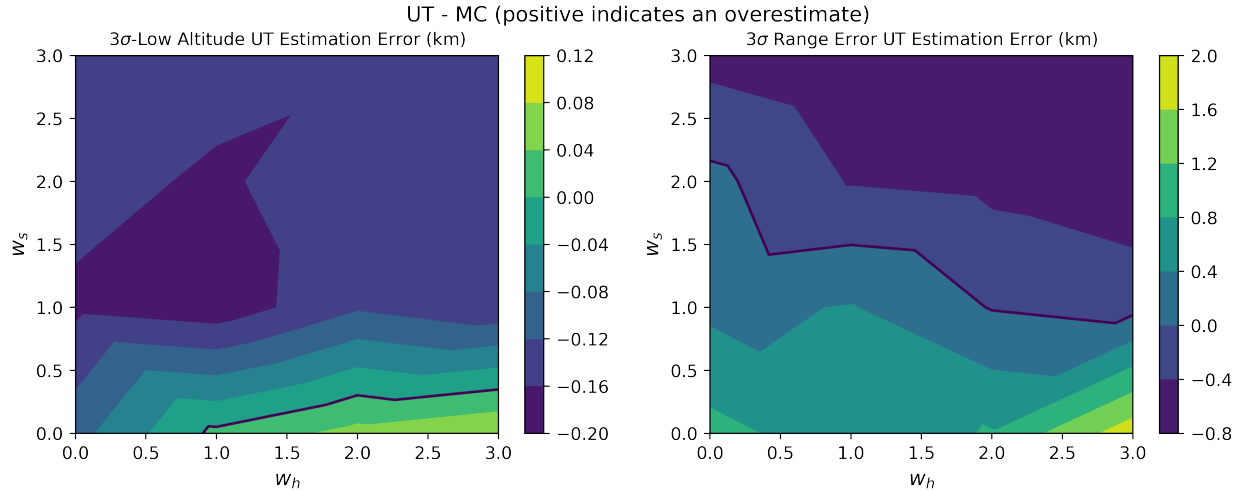


Figure 7.5: Unscented Transform estimation errors relative to Monte Carlo statistics for jointly optimized static feedback gains.

the ROGP varies with the weights. A very clear general trend is that more robust solutions, i.e., those with higher weight on either standard deviation, tend to target longer downrange distances. The total variation in target distance is about 30 km. The feedback gains have a large impact on these results. For comparison, Fig 7.7 shows the target distances arising from the unguided open-loop optimization (corresponding to the contour in Fig. 7.1). In the unguided case, there is virtually no change in target distance for different w_h , and the variation is exclusively with respect to w_s . The target distances are also noticeably shorter than in the guided case, roughly 40 km shorter on average.

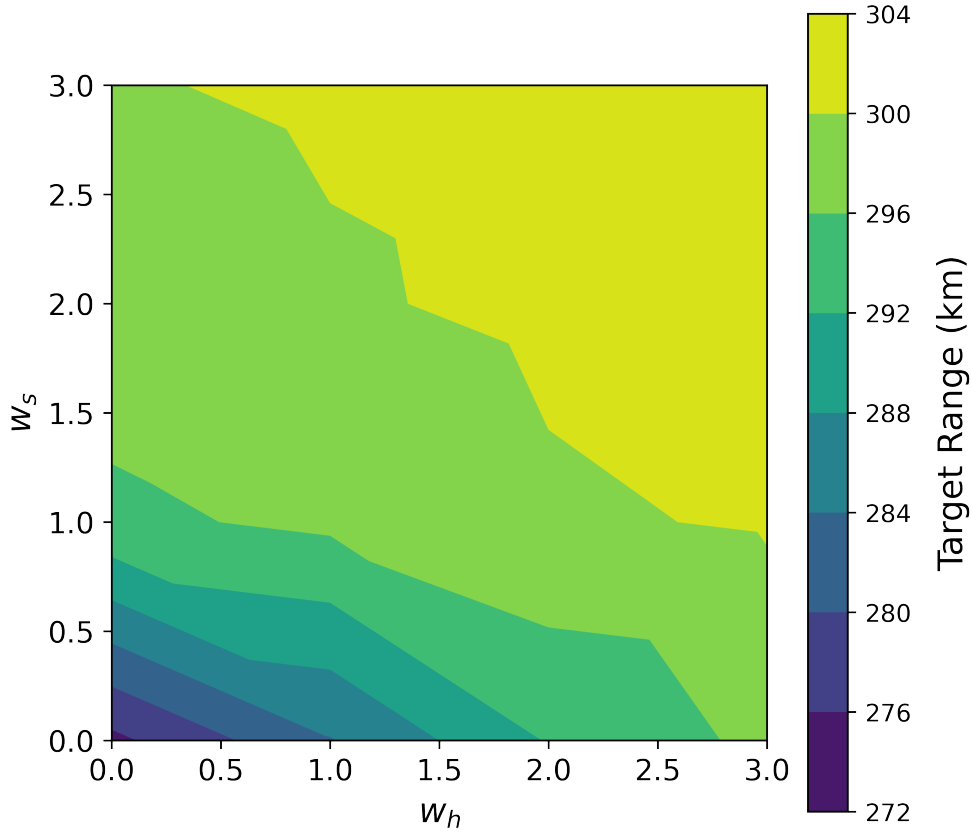


Figure 7.6: The optimal target range as a function of the performance weights.

Figure 7.8 shows the contours of the optimal value of each feedback gain for the weights. A number of trends may be readily identified. When range performance is ignored $\sigma_s = 0$, the range gain $k_s \approx 0$. For the special case of $w_h = w_s = 0$, the drag gain $k_d \approx 0$. The drag

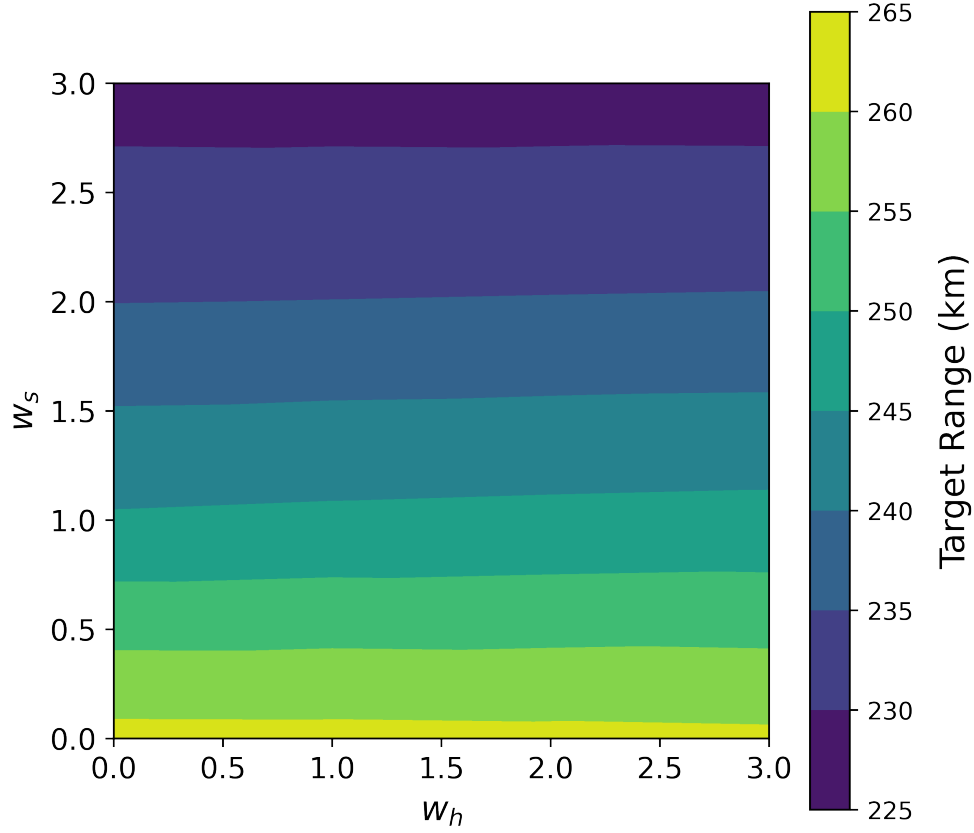


Figure 7.7: The optimal target range as a function of the performance weights for an unguided vehicle.

gain grows with increasing w_s until $w_s = 2$ before decreasing again at $w_s = 3$. A result seen in both k_D and k_s is the opposing trends as the weights are varied. Increasing w_s increases the gain magnitudes but increasing w_h lowers them, except for small values of w_s where they are essentially independent of w_h . The FPA gain k_γ is nearly independent of w_h except when $w_s \approx 0$. Interestingly, the FPA gain is approximately zero for $w_s = 1$ and increases for both large and smaller values of w_s . We remark that the gains used to initialize each optimization, K_2 , given in Chapter 6, are not enclosed by the ranges of the optimal gains except k_s . This is not surprising, as they were optimized for the initial reference control and not any of the robust optimal solutions.

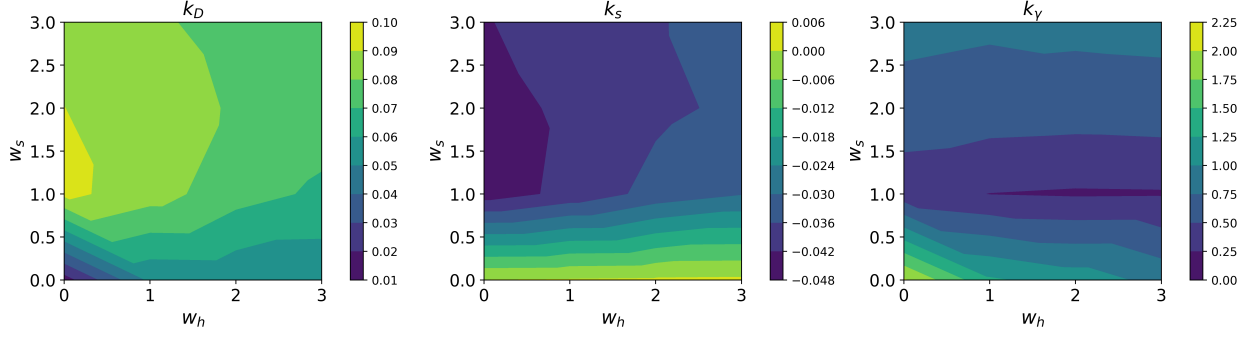


Figure 7.8: The optimal feedback gains as functions of the performance weights.

7.3 Detailed Solution

The previous sections considered the design space of the ROGP, examining the tradeoffs enabled by varying the weights in the performance index. In this section we present an in-depth look at a single solution, including visualizing the sample trajectories. We also perform further analysis to determine which uncertainties most strongly affect performance by applying the global sensitivity analysis technique known as Monte Carlo Filtering (MCF) [55]. We will examine the solution for $w_h = w_s = 1$ with joint optimization of the feedback gains.

The evolution of the marginal distribution of each state variable is shown in Fig. 7.9, including 500 of the Monte Carlo sample trajectories, and the 3σ bounds estimated via UT and MC. The UT bounds are nearly indistinguishable from the MC bounds. The 3σ -low altitude is 9.5 km and the 3σ range error is 3.2 km. The lower right plot of Fig. 7.9 shows the reference control profile and 500 closed loop control profiles from the samples. The reference control is saturated in the full lift up orientation for $v < 1000\text{m/s}$. As noted in Ref. [4], such a maneuver results in higher deploy altitudes but also deteriorates the range performance as dispersed MC samples fall short of the target. However, the probabilistic nature of the performance index accounts for the expansion of the range errors when choosing the velocity at which to begin to command full lift up. In some sense, this result agrees

with the MSL conclusion, which was that range control is less effective at velocities under 1100, and thus maintaining altitude by limiting the magnitude of the bank angle while also reducing cross range errors using heading alignment was an effective strategy to increase deploy altitudes without significantly compromising range accuracy. Here the robust optimal solution naturally leads to a similar approach by saturating the reference control, resulting in nearly all trajectories applying $u \geq 0.8$ for the remainder of entry.

Figure 7.10 shows a scatter plot of the terminal altitude and range, as well as 99% confidence ellipses estimated via UT and MC. Here the UT estimation error is more clearly visible compared to the trajectory histories, but is small. Note also that while the UT has some estimation error in the covariance elements that results in a rotation of the confidence ellipse, these elements are not used in the problem formulation. The samples that terminated short of the target (positive range error) demonstrate the earlier point about the “brittleness” of solutions that do not leave margin at low velocities. However, we remark that this is remedied by applying a larger weight on σ_s , which will naturally result in less saturation.

7.3.1 Sensitivity Analysis

The numerical results show a clear tradeoff between terminal altitude and range performance. In this section we utilize MCF to determine which uncertainties are the dominant source of low altitude or high range error. MCF is a density-based, global sensitivity analysis method compatible with MC data.

Density-based sensitivity analysis considers the PDF or CDF of an output variable, in contrast to variance-based methods that consider the second moment of the output. In the context of sensitivity analysis, global means to consider input variations in the entire feasible space, in contrast to local methods, like partial derivatives, that are evaluated at a single point.

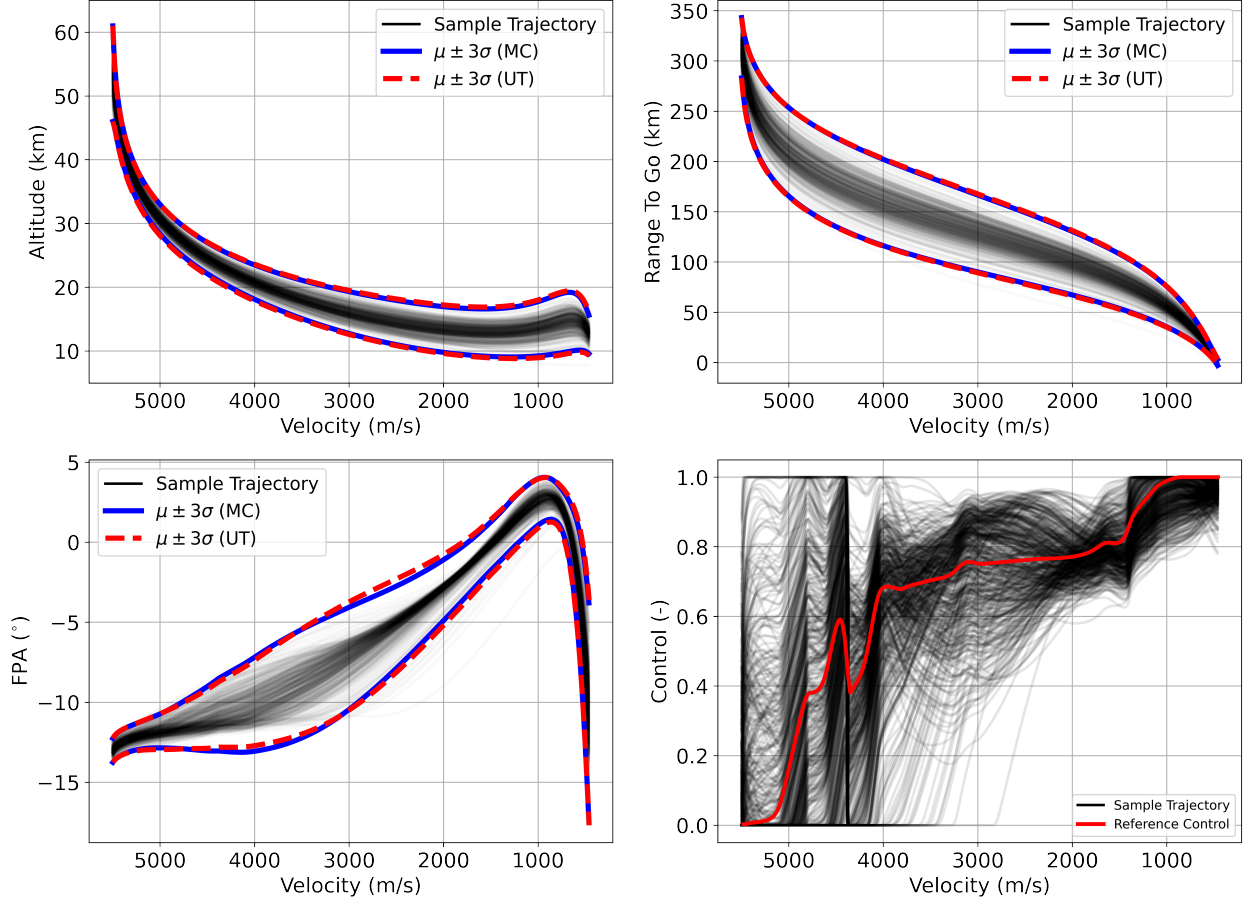


Figure 7.9: Sample trajectories and 3σ bounds estimated by MC and UT.

MCF proceeds as follows: given an output variable of interest, y , divide the MC samples into two groups based on a selected criteria. For each input variable, x_i , compute the ECDF of each group. Next, the two-sample Kolmogorov–Smirnov statistic, a measure of the distance between two ECDFs, is computed as

$$KS = \max_{x_i} |F_1(x_i) - F_2(x_i)| \quad (7.1)$$

where F_i is the ECDF of the i^{th} input. Notice the the KS test is sensitive to both the shape and location of the CDF. Finally, the inputs are ranked according to their KS statistic. In our application the outputs are terminal altitude and range. For altitude, the criteria is that the sample altitude is less than the mean altitude of the samples. For range, the criteria is

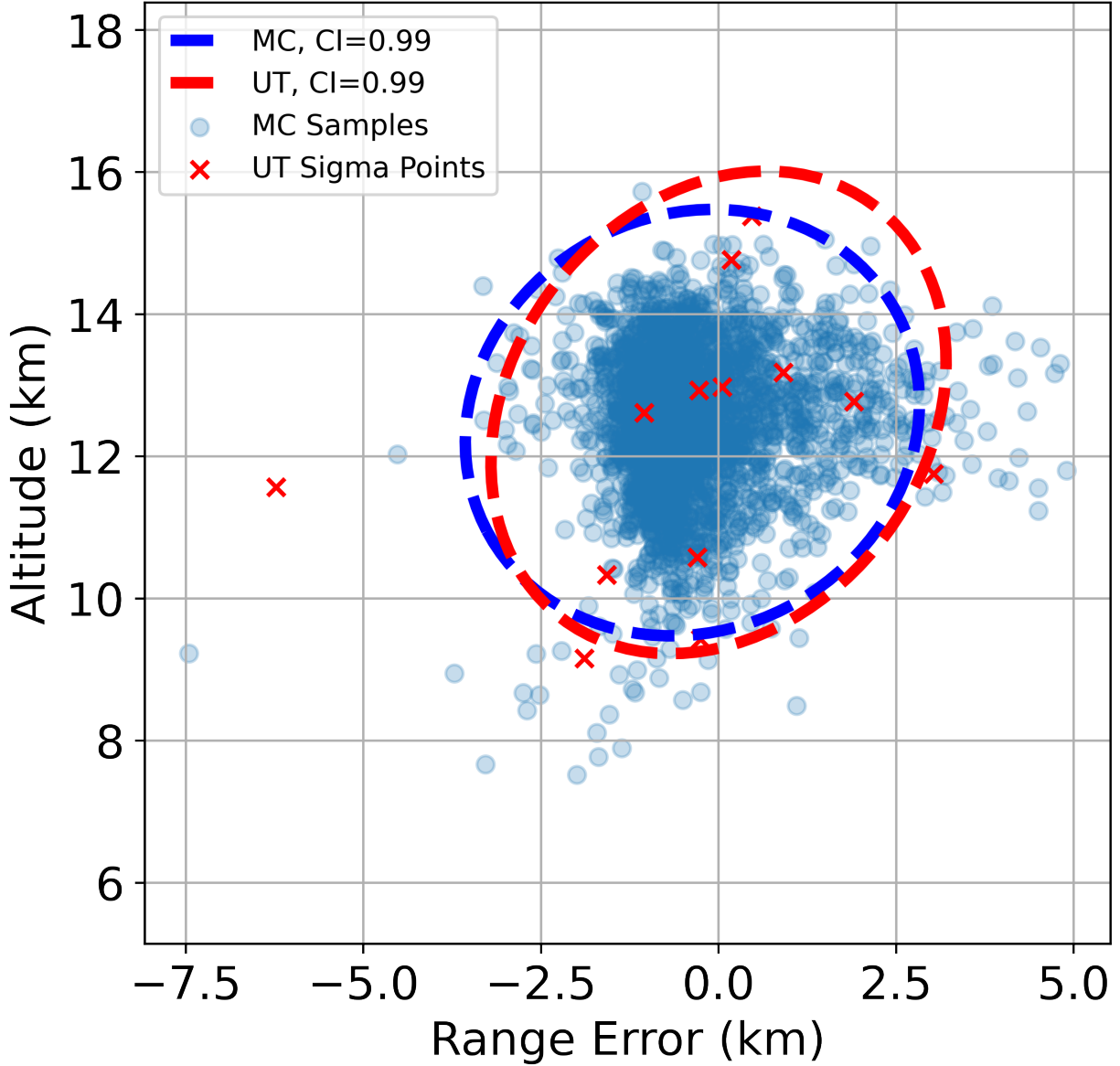


Figure 7.10: Terminal altitude and range performance.

that the absolute error is greater than the mean absolute error. Note that the two groups do not (and need not) contain the same number of samples. Performing this test for the two different outputs and criteria are summarized in Table 7.1. The dominant sources of lower than average altitude are density and ballistic coefficient, and to a lesser extent initial errors in range and altitude. MSL engineers also determined that aerodynamics and atmospheric uncertainty are the dominant factors in altitude loss [4]. The entry flight path angle has a

limited effect. Conversely, the initial state errors in altitude, and range significantly impact the guidance performance. The flight path angle and lift-to-drag ratio have a lesser effect on the range errors, and density and ballistic have little to no impact. From these results, it is clear that the uncertainties affect our quantities of interest in different ways; the only uncertainty in the top 3 of both outputs is the initial altitude. One conclusion is that in order to simultaneously achieve high altitude and low range error, entry guidance must effectively mitigate all of the different sources of uncertainty.

Low Altitude		High Range Error	
Uncertainty	KS Statistic	Uncertainty	KS Statistic
$\delta\rho$	0.52	s_0	0.41
$\delta\beta$	0.36	h_0	0.22
h_0	0.15	γ_0	0.11
s_0	0.14	$\delta(L/D)$	0.10
$\delta(L/D)$	0.11	$\delta\rho$	0.05
γ_0	0.05	$\delta\beta$	0.04

Table 7.1: KS statistics estimating the sensitivity of terminal altitude and range to the uncertain input variables.

7.4 Convergence Properties

In this section we examine the convergence of solutions using 1) the full DDP algorithm, 2) DDP with the $\mathbf{f}_{\mathbf{xx}} = \mathbf{0}$, $\mathbf{f}_{\mathbf{ux}} = \mathbf{0}$ modification proposed in Subsection 5.2.2 (labeled DDP-ControlHessian in figures), and the iterated Linear Quadratic Regulator [33] (iLQR) variation where additionally $\mathbf{f}_{\mathbf{uu}} = \mathbf{0}$. We solve the ROGP for $w_h = w_s = 1$ with each method. The primary metrics that we are interested in are the total cost, the total number of iterations, the time spent computing derivatives, and the maximum of the norm of the gradient of the

value function with respect to the controls over the entire trajectory. We will also examine the linesearch stepsizes. In this example, only the parametric uncertainties in $([L/D, \beta, \rho])$ are considered; the initial state uncertainty is set to zero. The reason is that the number of sigma points, and thus the number of states in the extended state vector, depends on the uncertainty dimension. While there is no difficulty in solving the full six dimensional problem (and indeed larger sizes) using iLQR or DDP-ControlHessian, the laptop used to solve the ROGP could not even store the full Hessian matrices for all $N = 250$ steps used in the original DDP formulation.

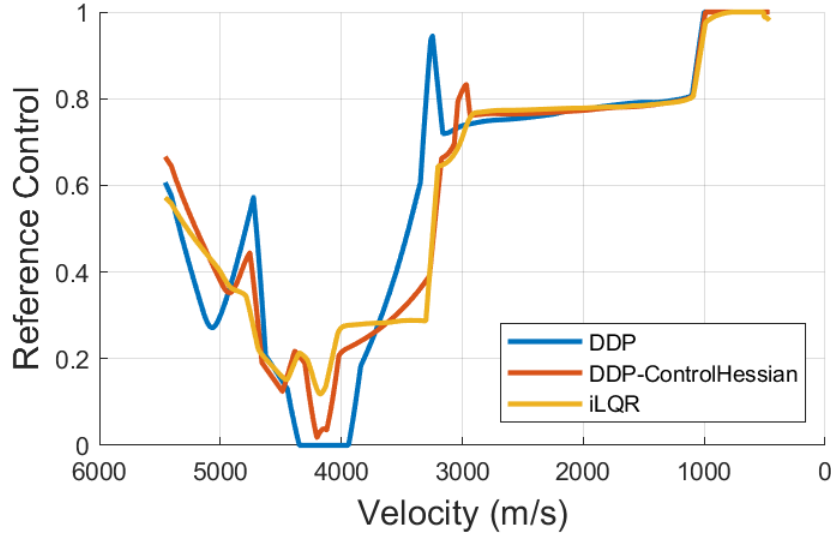


Figure 7.11: Total cost per iteration

Figure 7.11 shows the resulting reference controls. While the results are similar, the three methods do not converge to identical solutions. Figure 7.14 shows the total cost at each iteration. Full DDP achieves the lowest cost of 2.868 after 81 iterations. Control Hessian-only DDP achieves a cost of 2.870 after 46 iterations, and iLQR has the highest cost, 2.871, and still has not terminated after 250 iterations. Figure 7.13 shows the maximum gradient at each iteration. Unlike the DDP variants, the gradient never meets the exit criteria and thus is terminated based on the number of iterations. Figure 7.14 shows the stepsizes accepted in the linesearch procedure. Both DDP variants frequently accepted the full step $\epsilon = 1$, while

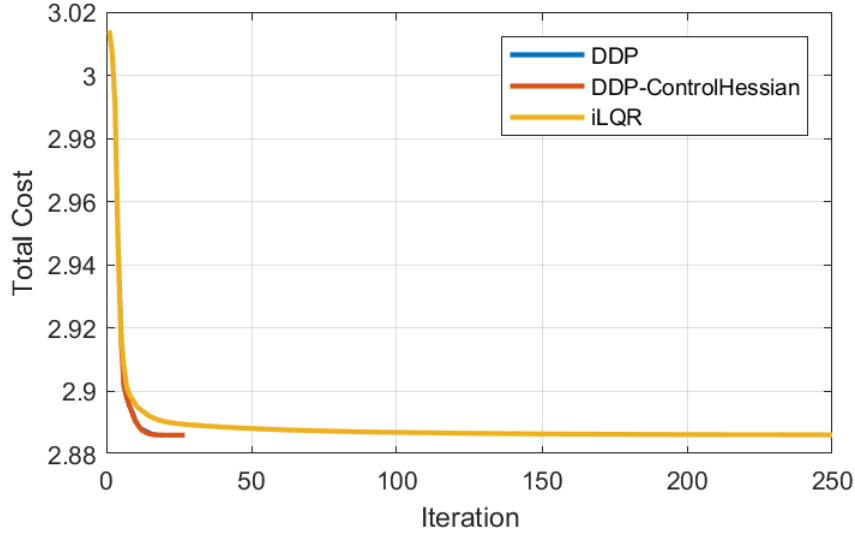


Figure 7.12: Total cost per iteration

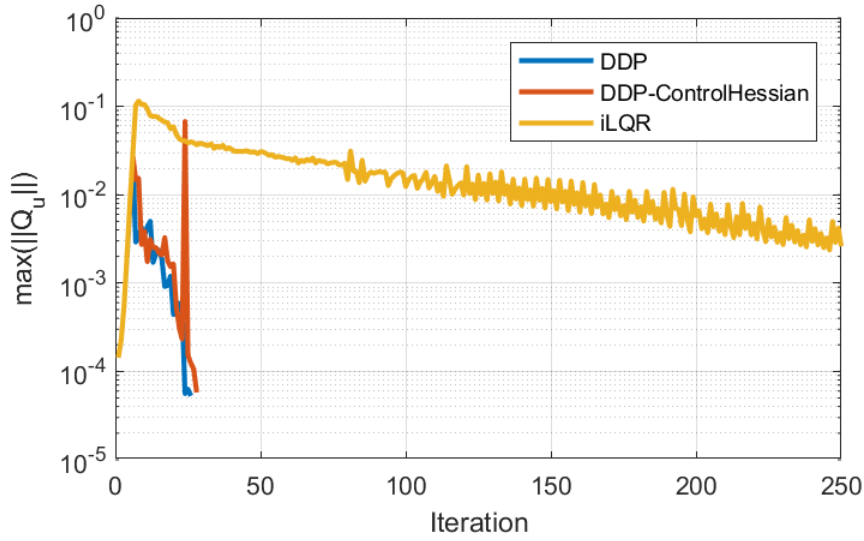


Figure 7.13: Maximum control gradient norm over all timesteps.

iLQR always took considerably smaller steps on the order of 10^{-2} . Finally, Fig. 7.15 shows the drastic, nearly 10x reduction in time spent computing the derivatives resulting from omitting Hessian terms. The average times were 47.7 s, 5.6 s, and 5.2 s, respectively. This example shows that, for this problem, retaining the second order derivatives with respect to the controls achieves nearly the same convergence rate as the full DDP algorithm while being only mildly more expensive to compute than the iLQR solution.

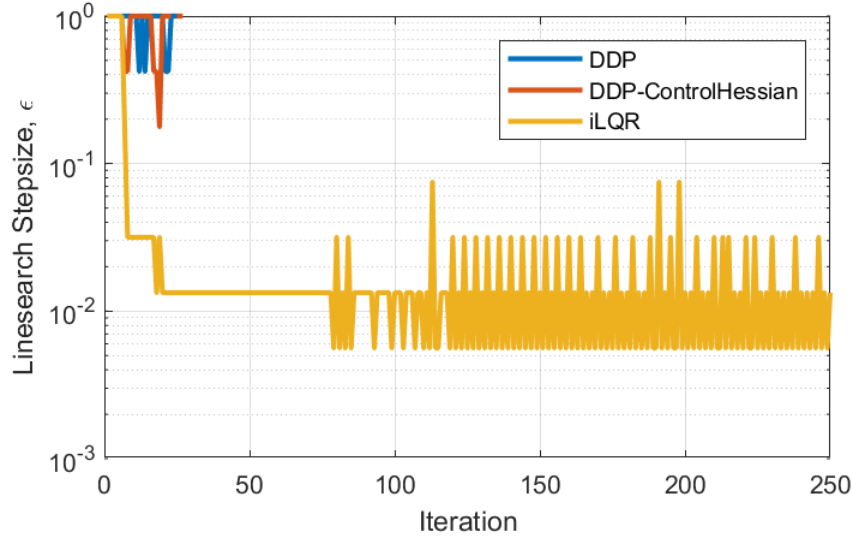


Figure 7.14: Backtracking linesearch stepsize per iteration.

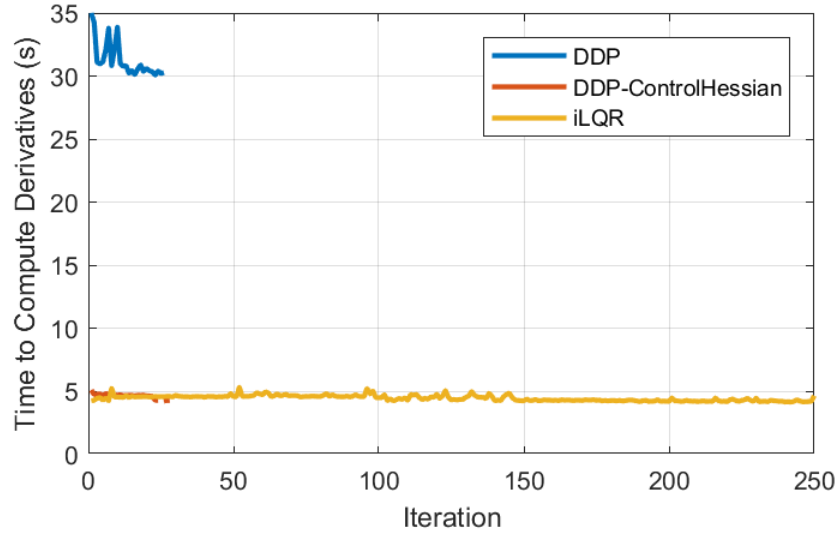


Figure 7.15: Time in seconds spent computing derivatives of the cost and dynamics at each iteration.

We also considered different initial guesses at the optimal control to determine if the solution is sensitive to the initialization. In contrast to the linear ramp up guess, we considered a linear ramp down from 1 to 0, as well as two constant guess of full lift up, $u_{\text{ref}} = 1$ and half lift up $u_{\text{ref}} = 0.5$. Only the control hessian is retained, and the final cost and total iterations are summarized in Table 7.2. All four solutions converge to nearly identical costs.

Interestingly, the ramp down profile, which is far from the converged optimal solution, finds a solution with a slightly lower cost than the other 3 solutions. The ramp up guess converged in the fewest iterations (46) while the most iterations was 76. Thus, it may be concluded that the robust optimal solution is not sensitive to the initial guess that is used.

Guess	Final Cost	Iterations
Ramp up	2.8702	46
Ramp down	2.8697	71
Full lift up	2.8702	76
Half lift up	2.8701	61

Table 7.2: KS statistics estimating the sensitivity of terminal altitude and range to the uncertain input variables.

7.5 Joint Optimization of Velocity-Varying Gains

A natural question is if DDP can be used to jointly optimize not only constant gains, but also velocity-varying gains as well. Given that constant gains are a special case of varying gains, the optimal solution with dynamic gains should be expected to perform no worse than the constant gain solution, and is expected to perform better. In this section we present an example of optimized velocity-varying gains and discuss the issues.

Figure 7.16 shows the MC results as well as the sigma points and estimated 3σ bounds. It can be seen that the solution drives the sigma points to a narrow set of terminal ranges, and as such the UT estimates a small $\sigma_s = 0.3$ km. However, the simulated performance in the Monte Carlo setting show the statistics are quite different from the UT values. Figure 7.17 shows a scatter plot of the terminal altitude and range flown as well as the 99% confidence ellipses. In this scenario, the UT is not accurately estimating the terminal statistics. This issue is reminiscent of the overfitting problem in machine learning, in which a solution that

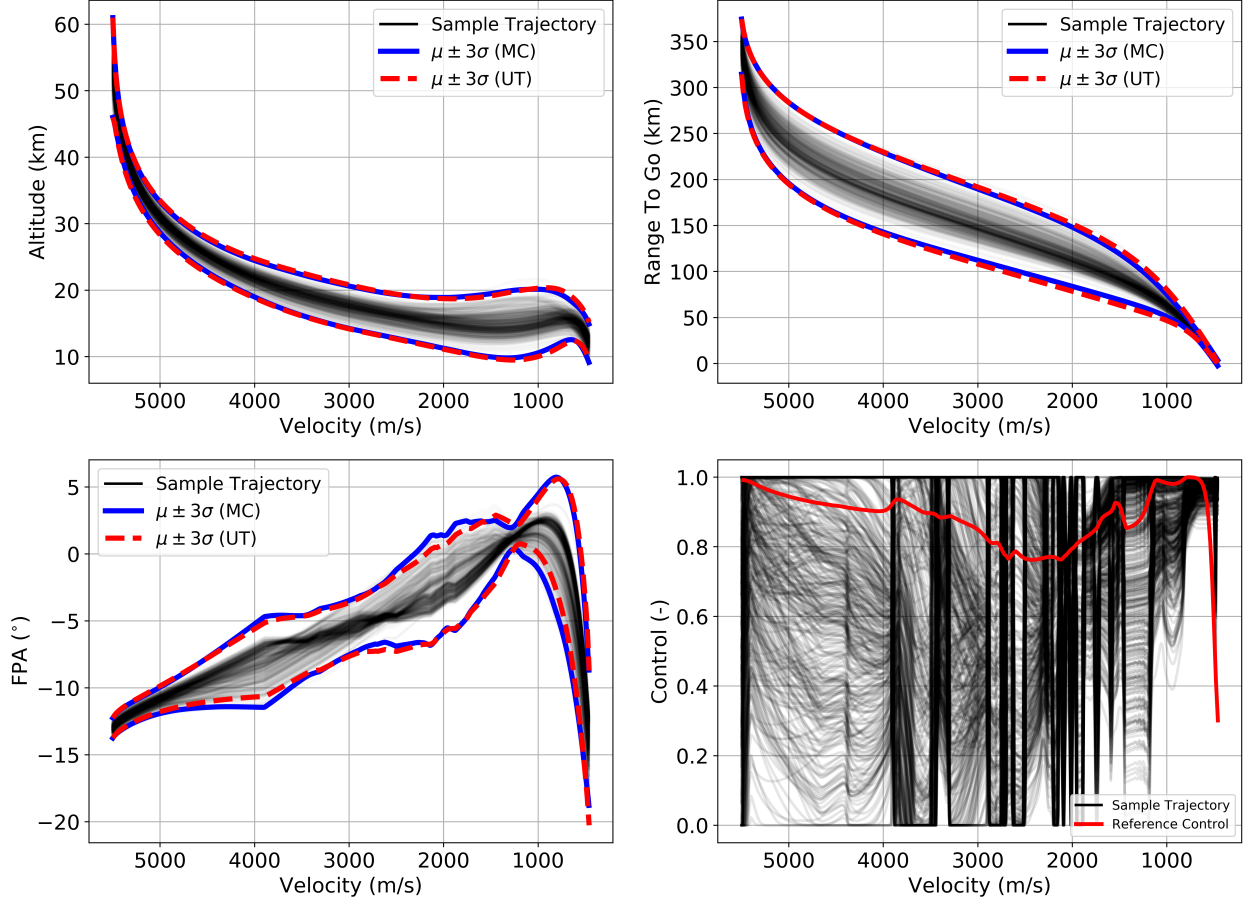


Figure 7.16: Sample trajectories and 3σ bounds estimated by MC and UT for a solution with jointly optimized, velocity-varying gains.

has a low value of the loss function over the sample points does not generalize well to samples not in the data set. The DDP solution is correctly guiding the sigma points together but fails to work well in other dispersed scenarios, such as those with a mix of the input uncertainties. It is evident the guidance does not perform as predicted and the size of the range error is not only larger than expected, but also larger than solutions with fixed gains. One apparent cause for the poor performance is the high gain nature of the solutions. Unless restricted to smaller magnitudes, the gains are, at times, quite large in magnitude, resulting in large control corrections and oscillatory commands that are not well-tracked due to the bank rate limit. Our experience with poor performance and poor agreement between the UT and MC statistics was part of the motivation to consider static feedback gains. In conclusion,

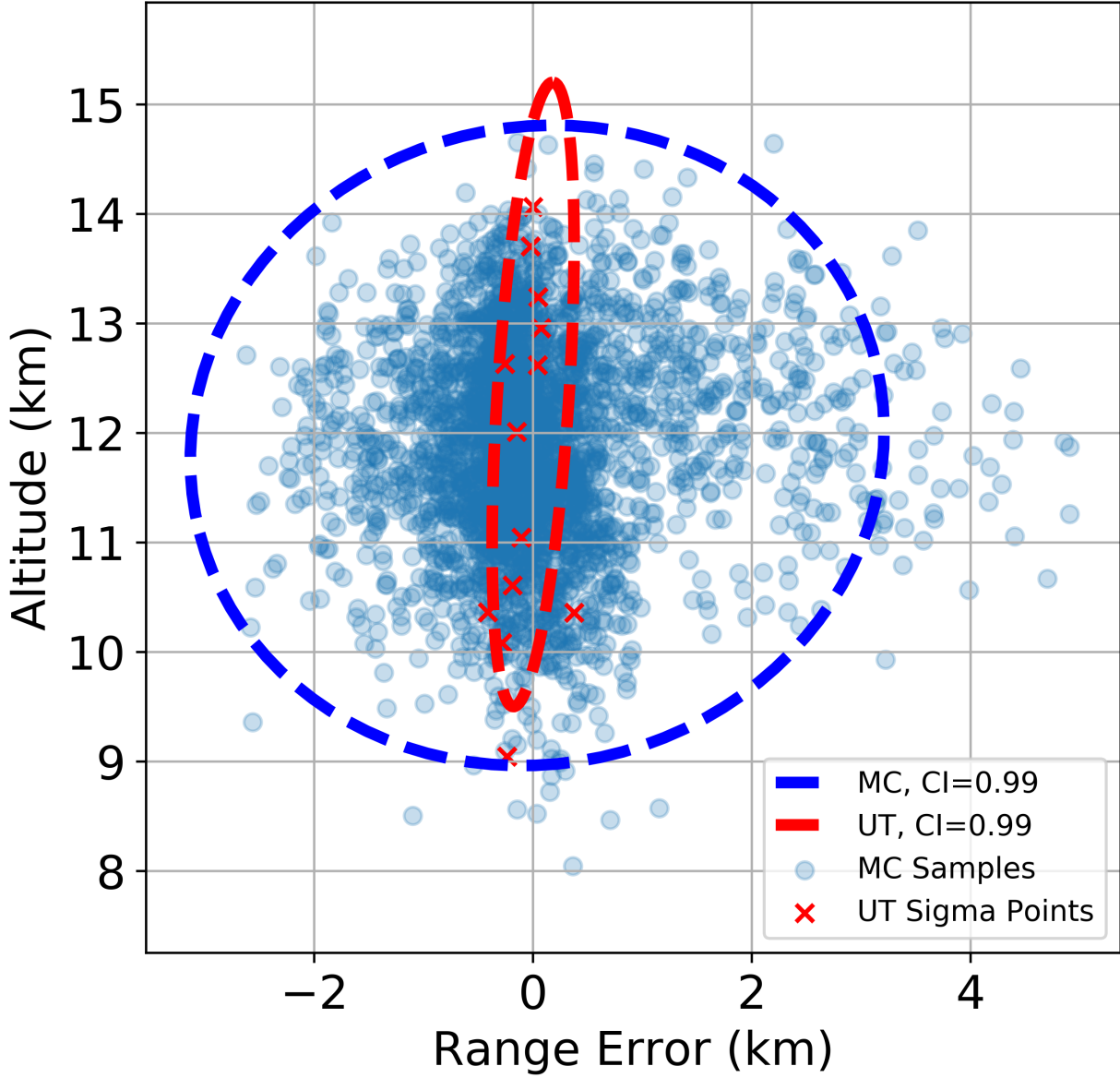


Figure 7.17: Terminal altitude and range performance for a solution with jointly optimized, velocity-varying gains.

while joint optimization of velocity-varying gains has the potential to improve performance compared to static gains, we were unable to do so, and additional work is needed to determine how to solve the issue discussed here.

7.6 Comparison to MSL and Mars 2020

The exact conditions under which MSL and Mars 2020 assess their guidance are not presented in Refs. [4, 1]; thus a precise comparison cannot be made. Nevertheless several comparisons that may be drawn. MSL designers simulated the open-loop performance and found the 3σ range error was approximately 40 km with a 3σ -low altitude of 6 km for a reference trajectory with 9 km nominal deployment altitude. Thus, the total effect of the stochastic parameters in their simulation is about 3 km 3σ altitude loss and 40 km 3σ range error. In performing the same exercise, we found that our 3σ altitude loss was 2.5-3.1 km, and 3σ range error was 59-66 km, each depending on the weights selected. With closed-loop guidance, and in the absence of onboard estimation errors, i.e., the case most similar to our assessment conditions, MSL reported a 3σ range error at chute deploy of 7 km [4]. In comparison, despite approximately 50% larger open-loop range errors and comparable altitude losses, our approach achieves 3σ range errors of 3.2 km and, like the ETPC, no additional altitude loss results from the closed-loop guidance. The 3σ -low altitude is 9.5 km, enabling targeting of site elevations above 0 km. Thus, while the results are not conclusive and additional testing is necessary, the proposed control law and design method appear promising in achieving high altitude while reducing range and altitude dispersions.

Chapter 8

Sample Retrieval Lander Class Assessment

In Ref. [56], an earlier version of the work in the dissertation, we assessed the robust optimal guidance on a vehicle in the class of the Sample Retrieval Lander (SRL) mission of the Mars Sample Return campaign[5, 57]. The ballistic coefficient of the vehicle is 200 kg/m^2 with an entry mass of 5000 kg. In order to contend with the increased ballistic coefficient, the vehicle L/D is raised to 0.28 (assumed to be constant, in contrast to the earlier results for the MSL-class vehicle). The same state and uncertainty at the entry interface presented in Chapter 6 are assumed. The resulting mean entry state is $x_0 = [54.5 \text{ km}, -11.5^\circ, 0 \text{ km}]^T$ at $v_0 = 5525 \text{ m/s}$. The same parametric uncertainty is considered except the altitude-varying Mars Climate Database model is replaced with a constant fractional offset with $3\sigma_\rho = 20\%$. In this earlier work, we had not yet developed the DDP extension to jointly optimize the static feedback gains presented in Section 5.2.1. As a result, any feedback gain optimization presented in this chapter was performed separately from the reference control using a sequential quadratic programming method to minimize the original terminal objective in Eq. (4.17), as estimated using the UT.

In the first section, the unscented transform is used to estimate the performance of an unguided entry vehicle to determine the magnitude of errors that must be mitigated by the entry guidance. In the second section, reference trajectories and controls generated by our approach are compared to those designed by nominal optimal control with fixed bank angle margins. By nominal optimal control we refer to the optimization of a single trajectory without considering uncertainty or closed-loop guidance. The third section investigates how the optimal trajectories change with the weights w_h and w_s , incorporating both initial state uncertainty and parametric uncertainty. The final subsection presents a detailed look at one solution and the corresponding Monte Carlo statistics.

8.1 Open-Loop Simulation

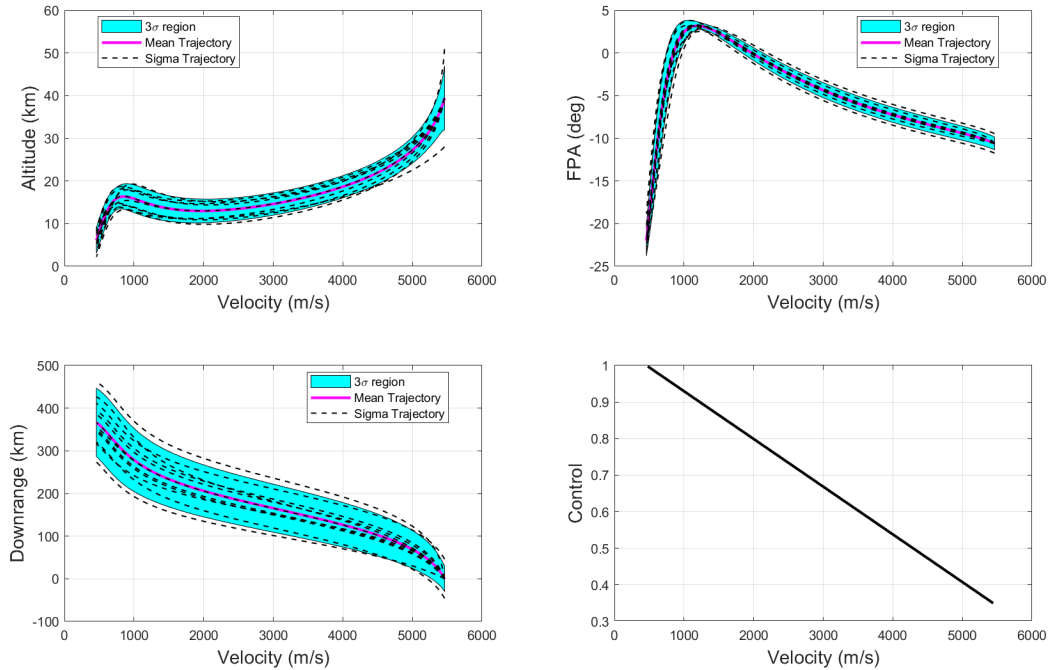


Figure 8.1: The sigma point trajectories for an unguided entry trajectory.

In order to estimate the magnitude of the range and altitude dispersions that must be

mitigated by the guidance commands, we simulate the performance of our initial trajectory with the feedback gains set to zero. Since the purpose of determining the magnitude of the terminal errors is as a point of reference, the unscented transform estimate is used rather than full Monte Carlo simulation. The altitude, range, FPA, and nominal control profiles are shown in Fig. 8.1, including the sigma point trajectories and the UT-estimated 3σ region. The control profile, shown in the bottom right of the figure, is a linear ramp from $u(v_0) = 0.35$ to $u(v_f) = 1$. The mean altitude is 6.08 km and the 3σ -low altitude is 3.09 km. At the chute deploy velocity, 460 m/s, the target range is 366.4 km and the 3σ range error is 80.3 km. Thus the robust optimal guidance has substantial potential errors to overcome.

8.2 Reference Trajectory Comparison

In this subsection we compare the performance of the robust optimal guidance using six different reference trajectory-control pairs. For all six cases, the control law in Eq. (4.4) is used with the same feedback gains and feedback control limits, Eq. (4.3). In the first three cases, a reference trajectory and control are generated by solving the optimal control problem $\min J = -h(v_f)$ subject to nominal longitudinal dynamics with different limits on the reference control to preserve margin for feedback. For the remaining three cases, the solution to the ROGP is used to generate the reference trajectory and control. Monte Carlo simulations are used to assess the guidance performance.

Case 1 considers $u_{\text{ref}} \in [0, 1] = [\cos 90^\circ, \cos 0^\circ]$, Case 2 considers $u_{\text{ref}} \in [\cos 75^\circ, \cos 15^\circ]$, and Case 3 considers $u_{\text{ref}} \in [\cos 60^\circ, \cos 30^\circ]$. Bank angle margin is important for range control, as well as for preserving control authority for lateral guidance, which is not modeled here. Cases 4-6 solve the robust optimal guidance problem with different choices of the weights to design the reference trajectories. Case 4 optimizes mean altitude ($w_h = w_s = 0$), Case 5 optimizes the 3σ -low altitude ($w_h = 3, w_s = 0$), and Case 6 optimizes a combination

of mean altitude and 3σ downrange deviation ($w_h = 0, w_s = 3$). Recall that each of the six cases may achieve a different downrange distance.

The feedback gains are constants $[k_D, k_\gamma, k_s] = [0.0725, -0.025, -0.004]$. These gains were computed by minimizing the UT-estimated downrange distance standard deviation for the initial guess at the control profile, which is a linear (in velocity) ramp from zero to one. Different initial guesses for the optimal profile would yield some variation in these gains, and superior results can be obtained by optimizing the gains for each solution, but using the same gains for all of the cases allows us to highlight the impact of the reference trajectory without confounding variables. These gains are used in the optimization process for Cases 4-6, and in the Monte Carlo simulations for all six cases.

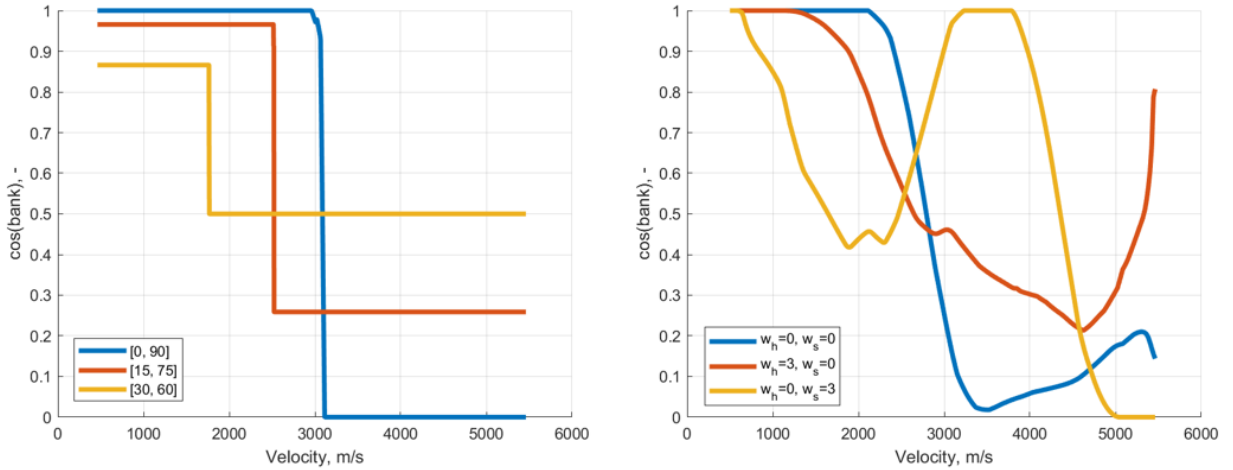


Figure 8.2: The profiles from deterministic optimal control (left) exhibit the well known bang-bang structure of altitude optimal trajectories. In comparison, the robust control histories (right) exhibit significantly different behavior depending on the weights applied.

Figure 8.2 shows the resulting control profiles for the six cases. The altitude-optimal controls resulting from nominal optimal control (Cases 1-3) exhibit the well-known bang-bang structure, while the robust optimal control profiles are quite varied depending on the weights selected. For Case 4, which optimizes mean altitude, the profile is somewhat similar to the bang-bang profiles, but retains margin at high velocities and has a slower transition to a lift up ($u = 1$) orientation. An interesting feature of the robust profiles is that they exhibit

significant margins, as solutions to the problem that considers robustness. Nevertheless, saturation still occurs at some velocities. This suggests that the robust optimal amount of margin varies along the trajectory, which implies using a fixed margin will produce inferior results, as the results presented next indicate. Although lateral guidance control authority ($L/D \sin \sigma$) is not considered in the formulation of the robust performance optimization problem, the solutions show that there is significant, though non-uniform, margin. Further consideration of compatibility with lateral guidance is left for future work.

The results are summarized in Tables 8.1-8.2. Case 1 preserves no margin at all, and as a result has the highest nominal terminal altitude of the three cases. This translates into the highest mean altitude of the three cases, despite the lack of margin. However, the lack of margin causes poor range control performance in many trajectories, leading to large mean and 3σ range errors. While the mean error can be reduced by target biasing, the 3σ range error is the largest of the 6 cases. As the amount of control margin increases in Cases 2 and 3, both the mean and 3σ range errors drop dramatically while the mean altitude also decreases. The 3σ -low altitudes display non-monotonic behavior with respect to bank margin because the mean altitude loss may or may not be offset by the reduction in altitude deviation. This is true for Case 2, which has a higher 3σ -low altitude relative to Case 1. In Case 3, the drop in mean altitude is greater than the drop in the 3σ -low altitude because the altitude deviation has decreased further.

Case 4 ($w_h = w_s = 0$) optimizes the mean altitude with knowledge of the problem uncertainty and guidance gains. The mean altitude is 50 m lower than Case 1 because the UT overestimated the mean. However, this difference is quite small, and the performance is more robust because the 3σ -low altitude is higher and the 3σ range error is smaller. Case 5 ($w_h = 3, w_s = 0$) adds additional weight to the altitude standard deviations, resulting in a 500 m increase to the 3σ -low altitude despite a 300 meter drop in mean altitude relative to Case 4. Case 5 has the highest 3σ -low altitude of the six cases, with a 200 m increase over the best

of the three profiles designed without considering uncertainty. Case 6 ($w_h = 0, w_s = 3$) places no weight on altitude standard deviation, but weights range errors heavily, and produces a 1.5 km decrease in 3σ range errors as a result. It has the lowest range error of the six cases, and compared to the large margins of Case 3, still produces a higher 3σ -low altitude.

This demonstrates the benefit of robust optimal control to designing reference trajectories. Additionally, the flexibility of the weights allows the trajectory designer to trade between range errors and altitude performance to the extent that the reference trajectory can alter closed-loop performance.

Table 8.1: Monte Carlo statistics for cases 1-3, with reference trajectories designed using optimal control and fixed bank margins.

Case	Bank Angle Limits	Mean Altitude (km)	3σ Low Altitude (km)	Mean Range Error (km)	3σ Range Error (km)
1	$[0^\circ, 90^\circ]$	6.38	2.87	-4.26	17.80
2	$[15^\circ, 75^\circ]$	6.20	3.49	-1.86	13.94
3	$[30^\circ, 60^\circ]$	4.80	2.36	0.12	3.83

Table 8.2: The Monte Carlo statistics for cases 4-6, with reference trajectories designed using the proposed method.

Case	Weights	Mean Altitude (km)	3σ Low Altitude (km)	Mean Range Error (km)	3σ Range Error (km)
4	$w_h = 0$ $w_s = 0$	6.33	3.38	-1.34	15.53
5	$w_h = 3$ $w_s = 0$	6.06	3.72	-1.01	11.85
6	$w_h = 0$ $w_s = 3$	4.84	2.58	-0.30	2.48

8.3 Standard Deviation Weighting Design Trade

From the previous subsection it is evident there exists a design space where weights on deviations produce superior results to simply optimizing the mean altitude when tail behavior of the distribution is important. This section investigates the extent to which weighting the standard deviations can alter the trajectory performance. The robust optimal control problem is solved for a variety of weights. The results are summarized in Fig. 8.3. Again the advantages of weighting the standard deviations compared with optimizing only the mean are made clear. Comparing $w_h = w_s = 0$ with, e.g., $w_h = 1$, $w_s = 1$ demonstrates that a 50% reduction in the standard deviation of downrange distance can be obtained with no loss in the low end of the terminal altitude distribution. Additionally, we can numerically determine the trade between terminal altitude and downrange robustness. The minimum downrange standard deviation achieved ($w_h = 1$, $w_s = 3$) was 0.7 km, an 85% reduction from the mean optimization with $\sigma_s(v_f) = 5.1$ km. These trajectories show clear improvements with no change to the guidance gains - simply a different reference trajectory is used to improve the results. The downrange distance shows only small improvement for weights $w_s > 1$, but the low altitude continues to drop as this weight is increased. Thus, a compromise such as $w_h = w_s = 1$ strikes a good balance between the two objectives.

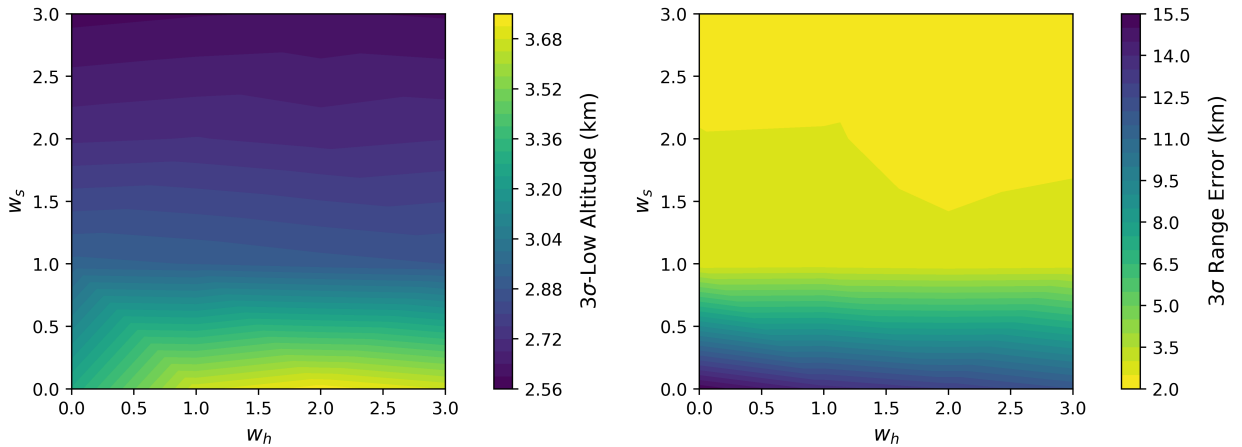


Figure 8.3: Contours of the Monte Carlo 3σ -low altitude (left) and 3σ downrange distance error (right) as a function of the weights w_h and w_s .

These results also show some inaccuracy due to using the unscented transform. The 3σ -low altitude should be maximized when the objective function is exactly this quantity, i.e., $w_h = 3$, $w_s = 0$, but due to small estimation errors this is not the case, and the highest 3σ -low altitude is achieved with $w_h = 2$, $w_s = 0$. Similarly, the 3σ range error is not minimized when $w_h = 0$, $w_s = 3$, but instead $w_h = 1$, $w_s = 3$. However, the difference in each of these cases is quite small. This issue is related to that fact that a constant UT scaling parameter $\alpha = 15$ is used in all computations. Our experience with the numerical results suggests that the optimal value of α depends on the weights selected, and larger weights should use a higher α .

These results used constant feedback gains, so it is likely that methods of generating gains that vary along the trajectory, such as the Linear-Quadratic Regulator, Apollo influence coefficients [34], or joint optimization of the feedback gains in the robust optimal control problem, could produce even greater synergistic improvements to the trajectory design. Additionally, the constant feedback gains were optimized for the initial guess trajectory, so further improvement is possible by reoptimizing the gains for the converged trajectory and control, as will be done in the following subsection.

8.4 Comparison of UT and Monte Carlo Statistics

The previous subsections looked at broader trends when using the proposed method to design reference trajectories. This subsection will focus on a single solution, and compare the UT and Monte Carlo statistics in greater detail. Additionally, the feedback gains are again constants, but for this case they are reoptimized to minimize UT-estimated downrange errors after finding the optimal reference control with the previous gains. The reoptimized gains are $[k_D, k_\gamma, k_s] = [0.058, -0.021, -0.005]$. Since each component of the objective is important and a compromise must be met, we select the weights $w_h = w_s = 1$. A Monte

Carlo simulation is conducted with 8000 samples in order generate precise statistics for comparison with the unscented transform.

Quantity	Unscented Transform	Monte Carlo
Mean Altitude (km)	4.81	4.70
3σ Low Altitude (km)	2.50	2.46
Mean Range Error (km)	-0.20	0
3σ Range Error (km)	1.09	1.79

Table 8.3: Summary of the statistics from the Unscented Transform and Monte Carlo

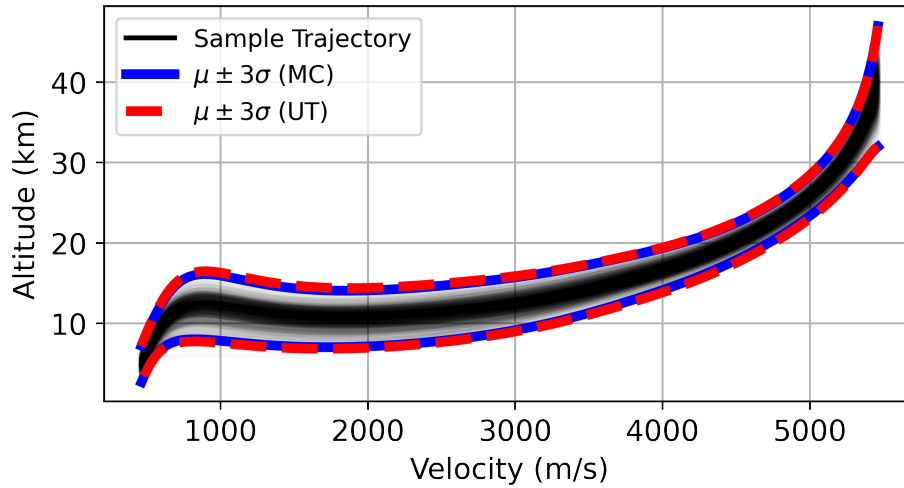


Figure 8.4: Altitude versus velocity for 500 sample trajectories with Monte Carlo- and UT-estimated 3σ bounds. The 3σ -low altitude is 2.5 km.

Table 8.3 compares statistics estimated with the unscented transform and Monte Carlo simulation. The reference trajectory and constant gains are quite effective, with the Monte Carlo 3σ range error under 2 km, despite small UT-estimation errors. This is achieved in part due to the reoptimization of the gains. This result is further strengthened by comparison with the performance of the unguided vehicle, which we recall was $3\sigma_s = 80$ km. This is very nearly a 98% reduction in 3σ range error at a cost of 630 meters in 3σ -low altitude, again compared to the unguided statistics. Relative to the Monte Carlo results, the UT overestimates the mean altitude by 100 m, the low altitude by 40 m, and underestimates

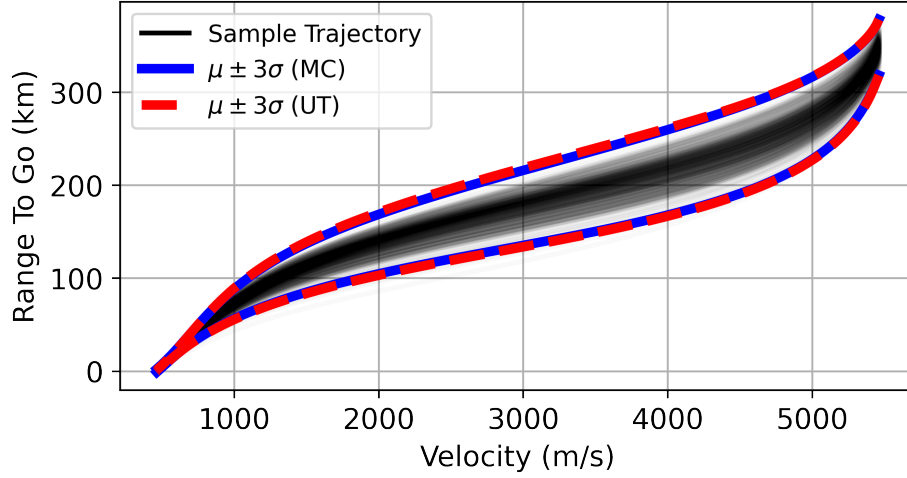


Figure 8.5: Range to go versus velocity for 500 sample trajectories with Monte Carlo- and UT-estimated 3σ range error is 1.8 km.

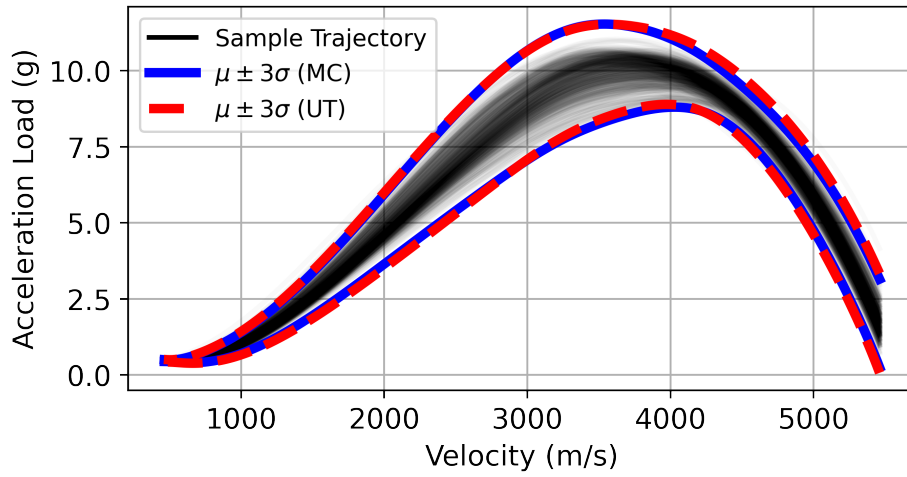


Figure 8.6: Acceleration load vs velocity for 500 sample trajectories with Monte Carlo- and UT-estimated 3σ bounds.

the 3σ range error by 700 m, or nearly 40%. Because the mean trajectory is used as the reference, the UT-estimated mean error is always zero.

Figures 8.4-8.6 show 500 sample trajectories as well as the Monte Carlo estimated 3σ bounds, and the UT-estimated 3σ bounds for comparison. Each of these figures demonstrates that the UT-estimated statistics closely approximate the Monte Carlo statistics. Like the terminal statistics in Table 8.3, the UT estimation errors do not have a consistent sign at

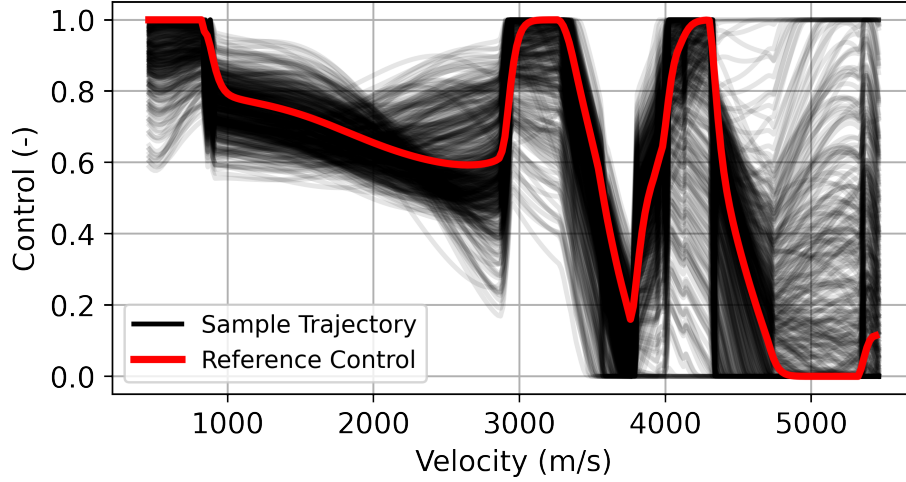


Figure 8.7: The reference control and 500 sample trajectories.

every point along a trajectory; on some intervals the UT underestimates the 3σ bounds, while on other intervals it overestimates them. More importantly, the magnitude of these errors remains small, confirming that the UT is an effective method of quantifying uncertainty in the optimization process. Figure 8.4 shows that lofting at low velocities is a key feature of the trajectories used to raise the terminal altitude. Contrary to prior studies that found range control is not effective at low velocities [4], Fig. 8.5 shows the convergence of the 3σ range errors from 25 km at $v = 1100$ m/s to less than 2 km at the $v = 460$ m/s. Figure 8.6 shows the acceleration loads which never exceed 12g. If, as with MSL, a 15g limit is imposed, then this result indicates that a steeper mean entry flight path angle can be accommodated.

Figure 8.7 shows the reference control as well as the sample controls for the same 500 trajectory subset. The reference control resembles bang-bang profiles at high velocities before transitioning gradually to a full lift up orientation. There exist intervals with significant margin, particularly between 1000 m/s and 3000 m/s, as well as intervals in which the reference control and many sample controls are saturated. This is important because a full lift up orientation leaves no control authority for lateral guidance to manage the vehicle crossrange distance to the target.

SRL will land near Mars 2020 in Jezero Crater at an altitude of -2.566 km MOLA. The range of altitudes achieved using the method demonstrates that, for the heavier vehicle under consideration, a constant $L/D = 0.28$ is not sufficient to reach the required altitude. Nevertheless, the altitude optimal solutions in the second section of this chapter indicate the maximum altitude reachable in the nominal scenario is about 6.4 km, providing a value against which the 3σ -low altitudes from the robust optimal solutions may be compared. The 3σ -low altitudes in Section 8.3 ranged from 2.6 km to 3.7 km, or altitude losses of 3.8 km and 2.7 km relative to the maximum altitude trajectories. Case 6 had a 3σ -low altitude loss of 2.25 km relative to the mean altitude with a tight 3σ range error of 2.5 km. Overall, the robust optimal solutions for this heavier class of vehicle show similar robustness to the results of the previous chapter for the MSL-class vehicle.

Chapter 9

Entry Guidance for SRP-Based EDL

In this chapter we turn our attention away from current generation EDL architectures and consider a hypothetical future scenario. This chapter and Chapter 10 are based on our work in Ref. [20]. Among the many considerable challenges involved in the Entry, Descent, and Landing (EDL) of a spacecraft on Mars, perhaps the greatest is the low density of the Martian atmosphere [3, 38]. At approximately one percent of the Earth’s atmospheric density, current generation Mars entry vehicles are incapable of achieving sufficient deceleration to reach subsonic speeds before reaching the surface, necessitating the use of EDL technologies including supersonic parachutes and retropropulsion, both of which were utilized in safely landing MSL’s Curiosity rover [58], and Mars 2020’s Perseverance [1]. As the mass (or ballistic coefficient) of these vehicles grows, the problem is exacerbated until safe deployment conditions for current supersonic parachutes are not reachable at altitudes high enough to provide sufficient timeline margin for subsequent EDL events [3]. In such missions, supersonic retropropulsion (SRP) may be relied upon to land larger, heavier vehicles relative to the current generation of landers without the use of a parachute, and as a result powered descent will play an even more significant role in the EDL process. In this chapter and the next, the emphasis is on the interplay between the entry phase and the powered phase, and

thus only these two phases are considered.

The challenges during the entry phase from a guidance perspective include entry state dispersions due to delivery errors, atmospheric winds and density uncertainty, aerodynamic uncertainty, and imperfect onboard state estimation, each of which contribute to the size of the landing footprint. MSL was the first Mars entry system to utilize hypersonic guidance to autonomously steer the spacecraft toward the target, which resulted in a landing footprint nearly an order of magnitude smaller than previous unguided ballistic entries [3]. Mars 2020 improved upon the MSL guidance design by incorporating a range trigger [36] to deliver Perseverance to a 99% touchdown ellipse of $7.1 \text{ km} \times 6.5 \text{ km}$. It is expected that a feature of SRP-based EDL is the requirement for pinpoint accuracy, defined as a sub-100 meter landing ellipse [59]. Assuming an entry guidance meets the basic requirement of delivering the vehicle to a state from which the powered guidance can achieve pinpoint landing, dispersed entry trajectories result in increased propellant requirements, rather than contribute to the landing ellipse. As a result, rather than judge an entry guidance algorithm by its landing ellipse, the propellant cost or propellant mass fraction (PMF) of the powered descent trajectory may now be considered a primary metric.

A fundamental difference of retropropulsion-based EDL from parachute-based EDL architectures, even those that utilize a vertical powered descent phase with potential for a divert maneuver beforehand, lies in the set of desirable states at the termination of the entry phase. In parachute-based EDL, the terminal entry state is constrained by safe parachute deployment conditions defined by upper and lower limits on Mach number and dynamic pressure. The constraints are often translated into constraints on altitude and velocity using a model of the Martian atmosphere. In summary, when a chute is involved, the target set for entry guidance is the parachute deployment box coupled with small downrange and crossrange errors, combined with a well-aligned heading, and no restriction on flight path angle.

Traditionally, the primary objective of bank angle modulation during entry is range control with lateral control as a secondary objective, and guidance approaches could treat the two problems as decoupled. During the mission design phase, the downrange distance is chosen to be compatible with the parachute deployment conditions and other mission parameters such as the entry flight path angle. The target set in SRP-based EDL is in some sense much larger, but with more coupling between the state variables because, for a given powered descent guidance, the final entry state determines if pinpoint landing can be achieved, and if so, the propellant mass required. All six entry state variables contribute to the propellant cost; the optimal downrange distance and altitude at which to ignite depend on both the velocity magnitude and the flight path angle, while aligning the heading eliminates crosstrack flight during powered descent.

There exist many guidance solutions to the powered descent problem, often with a focus on propellant optimality. Classes of such algorithms include G-FOLD [60, 61] and other convex optimization-based methods, polynomial-based approaches including the venerated Apollo lunar descent guidance [62], indirect optimal control methods [63], and many others.

Far less attention has been given to the role of entry guidance when followed directly by a powered descent phase, without an intermediate phase that makes use of a parachute or other decelerator. Reference [18] presents one approach to entry guidance in a chuteless EDL architecture. Reference [18] correctly identifies the potential for integrating the entry phase with the subsequent powered descent phase as an opportunity to reduce propellant consumption, and the vehicle bank angle is modulated to perform range control, utilizing a bank profile that is linear-in-energy with one free parameter. Lateral logic to determine bank reversals is applied separately after solving for the bank angle magnitude. Reference [64] investigated an alternative EDL scheme in which aerodynamic control during entry is foregone in favor of hypersonic retropropulsion, essentially eliminating the entry phase.

The proposed entry guidance for SRP-based EDL is designed to deliver the vehicle

to a target set of states from which pinpoint landing can be achieved in the subsequent powered descent phase. Of the reachable states in the target set, entry guidance steers the vehicle to the one that will require the minimum propellant mass for pinpoint landing. For the simulation testing in this paper, the target set is computed by solving optimal control problems. The possibility of computing the powered descent controllable set to be used as a target set for an entry guidance algorithm was suggested in Ref. [11], wherein a procedure based on convex optimization was designed to efficiently compute it is given.

MSL triggered its parachute deployment and descent sequence using a velocity trigger [4], while Mars 2020 will utilize a range trigger [1] to dramatically decrease the size of the landing ellipse [36]. A component of our approach is an onboard determination of when to trigger powered descent initiation. In essence, the role of such a trigger is to ensure the minimum propellant ignition point is found along any entry trajectory that has at least one feasible ignition state. The theoretical developments behind one such trigger were introduced in Ref. [63], while Ref. [18] demonstrated its benefits over triggering at a fixed state variable, such as downrange, velocity, or energy. One potential weakness of the trigger in Ref. [63] is that it assumes a propellant optimal powered descent guidance. In theory, future Mars missions, especially those transporting humans, may very well consider a guidance algorithm with a different objective, such as prioritizing safety or even passenger comfort, as the bang-bang nature of propellant optimal powered descent solutions may not be suitable, despite the importance of limiting propellant consumption.

Despite presenting an approach in which a prediction of the terminal entry state is available and being used to modulate the vehicle bank angle, in Ref. [18] only the current vehicle state is checked for triggering the powered descent initiation. Our approach to the triggering mechanism differs in that we utilize the predicted trajectory. This allows us to query the mapping from the powered descent guidance algorithm at points along the predicted trajectory to generate propellant consumption predictions. The benefit of doing so is two-fold.

Firstly, the propellant optimal ignition point along an entry trajectory can be determined for any subsequent powered descent guidance, even one that is itself not propellant optimal. Secondly, by making explicit use of these predictions in the entry guidance algorithm, the two phases become linked by more than just the onboard triggering strategy, and superior propellant performance can be achieved.

9.1 Entry Guidance Problem for SRP-Based EDL

The problem under investigation in this chapter considers a two-phase EDL sequence consisting of an entry phase and a powered descent phase. The dispersions in vehicle state arising from the approach phase are modeled as delivery errors at the beginning of the EDL sequence. In the entry phase, the vehicle is guided via bank angle modulation and the vehicle state is described in spherical coordinates, as described in Chapter 3. In the descent phase, the vehicle is guided via supersonic retropropulsion with the equations of motion given in Cartesian coordinates. The reason for the choice of two different coordinate systems is to be consistent with the literature on each respective topic.

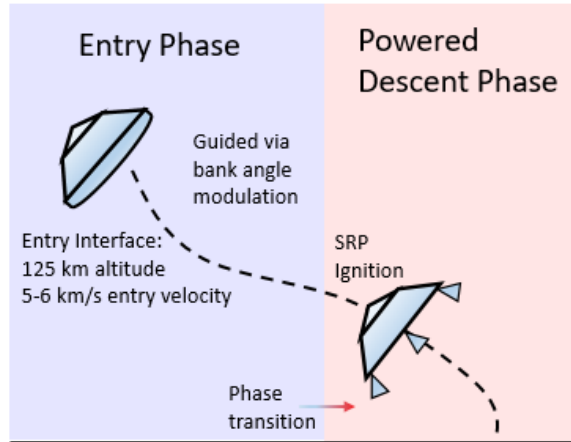


Figure 9.1: Schematic of the EDL phases.

Equations (3.1)-(3.6) will be compactly referred to as $\dot{\mathbf{x}} = f(\mathbf{x}, \sigma)$ and their solution at

time t from a state $\mathbf{x}_0$ evolving under a bank angle profile $\sigma[0, t]$ is given by the transition map $\mathbf{x}(t) = \varphi(t; \mathbf{x}_0, \sigma[0, t])$.

The entry guidance problem is to determine the entry phase duration and a bank angle profile that will deliver the vehicle from the entry interface to the propellant-optimal point in the target set, i.e., the set from which pinpoint landing can be achieved. A formal definition of the target set is given in the subsequent section.

9.1.1 Entry Guidance Target Set

The entry guidance target set is defined by the powered descent algorithm to be used during the descent phase, and the available propellant onboard the vehicle. The equations of motion for a point mass vehicle in powered flight in a surface fixed frame, neglecting aerodynamic forces and Coriolis effects, are

$$\dot{\mathbf{r}} = \mathbf{v} \tag{9.1}$$

$$\dot{\mathbf{v}} = \frac{\mathbf{T}}{m} + \mathbf{g}(\mathbf{r}) \tag{9.2}$$

$$\dot{m} = -\frac{\|\mathbf{T}\|}{I_{sp}G_0} \tag{9.3}$$

where $\mathbf{r} \in \mathbb{R}^3$ is the position vector of the vehicle, $\mathbf{v} \in \mathbb{R}^3$ is the velocity vector of the vehicle, m is its mass, $\mathbf{T} \in \mathbb{R}^3$ is the thrust vector of the vehicle, $\mathbf{g} \in \mathbb{R}^3$ is the gravitational acceleration vector, I_{sp} is the specific impulse of the rocket engine, assumed to be constant, and G_0 is the gravitational acceleration magnitude at the surface of the Earth. The full powered descent state vector in $\mathbb{R}^7$ comprises the position, velocity, and mass.

Any powered descent algorithm capable of meeting the pinpoint landing requirement may be considered. To eschew the choice of a specific algorithm in this work, the powered

descent problem is posed as the optimal control problem

$$\max m(t_f) \tag{9.4}$$

$$\mathbf{r}(t_0) = \mathbf{r}_0 \tag{9.5}$$

$$\mathbf{v}(t_0) = \mathbf{v}_0 \tag{9.6}$$

$$m(t_0) = m_0 \tag{9.7}$$

$$\mathbf{r}(t_f) = [0, 0, z_{\text{target}}] \tag{9.8}$$

$$\mathbf{v}(t_f) = [0, 0, \dot{z}_{\text{target}}] \tag{9.9}$$

$$0 < T_{\min} \leq \|\mathbf{T}(t)\| \leq T_{\max} \tag{9.10}$$

where the final time t_f is free, $T_{\min}$ and $T_{\max}$ are constant bounds on the available thrust, and z_{target} and $\dot{z}_{\text{target}}$ are the desired altitude and vertical velocity at the end of the descent phase. For a fixed initial mass, maximizing the final mass is equivalent to minimizing the propellant use. Formulated as such, the optimal control problem can be viewed as a function $\text{OCP}(\cdot)$ that maps an initial condition, i.e., ignition state, to a propellant cost. The set of initial states for which the optimal control problem has a feasible solution includes regions that are not of interest. Examples include states with initial altitudes very low to the ground, initial velocities much higher than those reachable by the entry vehicle before impacting the surface, and initial distances very far from the target, or having already overshoot the target. Denoting $\mathbf{r} = [x, y, z]^T$, the set of feasible solutions is restricted to a region of interest by defining a set of constraints

$$c_i(\mathbf{x}) \leq 0 \text{ for } i = \{1, 2, 3, 4\} \tag{9.11}$$

where

$$\mathbf{c}(\mathbf{x}) = \begin{bmatrix} x^2 + y^2 - d_{\max}^2 \\ d_{\min}^2 - x^2 - y^2 \\ z_{\text{target}} + z_{\min} - z \\ \|\mathbf{v}\| - v_{\max} \end{bmatrix} \quad (9.12)$$

and $[d_{\min}, d_{\max}]$ are the minimum and maximum distance to the target, $z_{\min}$ is the minimum altitude, and $v_{\max}$ is the upper bound on velocity. The set of all initial state vectors for which the solution to the optimal control problem is feasible and the propellant required by the solution is less than some prescribed maximum M , and satisfying these constraints is

$$\mathcal{F}_M = \left\{ \mathbf{x} \mid \max_{i=1,\dots,4} c_i(\mathbf{x}) \leq 0 \text{ and } \text{OCP}(\mathbf{x}) \leq M \right\}, \quad (9.13)$$

and is the set targeted by the entry guidance algorithm. Although the set is used an entry guidance target, it is defined in terms powered descent state variables. As the initial T/W ratio of the vehicle increases, $\mathcal{F}_M$ contracts. This is because, for a fixed I_{sp} and $T_{\min}$, increasing thrust will increase the mass flow rate and decrease the maximum time of powered flight, which naturally decreases the distance from the target the vehicle can fly. The effect is that higher T/W places more stringent requirements on the entry guidance target set unless additional propellant is added, or the value of I_{sp} or $T_{\min}$ changes.

Given an entry state and target position, the corresponding descent state is determined by assuming the current heading ψ defines the downrange direction. The downrange and

crossrange distances to the target are computed via spherical trigonometry [38]

$$d = \cos^{-1}(\sin \phi \sin \phi_{\text{target}} + \cos \phi \cos \phi_{\text{target}} \cos(\theta_{\text{target}} - \theta)) \quad (9.14)$$

$$\Psi = \pi/2 - \text{sign}(\theta_{\text{target}} - \theta) \cos^{-1} \left(\frac{\sin \phi_{\text{target}} - \sin \phi \cos d}{\cos \phi \sin d} \right) \quad (9.15)$$

$$c = \sin^{-1}(\sin d \sin(\psi - \Psi)) \quad (9.16)$$

$$DR = R_p \cos^{-1} \left(\frac{\cos d}{\cos c} \right) \quad (9.17)$$

$$CR = R_p c \quad (9.18)$$

where R_p is the planet radius. The full powered descent state vector in terms of the entry state variables is $[DR, |CR|, R - R_p, -V \cos \gamma, 0, V \sin \gamma, m_0]^T$. Because the downrange direction is defined by the current heading angle, the crossrange velocity is always zero. Under the assumed dynamics and zero crossrange velocity, the sign of the crossrange to the target does not affect the propellant cost, and its absolute value is used.

By using a downrange-crossrange coordinate system based on the current heading, an infinite number of entry states map to the same SRP state as a result of this surjective transformation. Intuitively, this reflects the rotational symmetry of the problem around the z -axis, stemming from the fact that states x and y have identical dynamics. For the powered descent problem as posed, any initial state $[x, y, z, \dot{x}, \dot{y}, \dot{z}, m]$ such that $(z, \dot{z}, m)$ are fixed

and

$$x^2 + y^2 = p^2 \tag{9.19}$$

$$\dot{x}^2 + \dot{y}^2 = v^2 \tag{9.20}$$

$$\psi = \tan^{-1}(y/x) \tag{9.21}$$

$$\dot{x} = -v \cos \psi \tag{9.22}$$

$$\dot{y} = -v \sin \psi \tag{9.23}$$

$$\tag{9.24}$$

lies along an iso-propellant contour defined by the horizontal distance and velocity magnitudes, p and v . An alternative mapping from entry states to powered descent states is to convert the entry state and target to Cartesian coordinates and subtract them. Although this is a feasible option, it is not preferable because it would result that every entry state maps to a unique SRP state. This is undesirable because it requires a larger table of solutions encompassing all the possible Cartesian states.

9.2 Proposed Entry Guidance

The objective is to deliver the vehicle to the minimum propellant ignition point in the feasible subset $\mathcal{F}_M$ that is reachable under the chosen bank angle parametrization from the current state. This is in contrast to prescribing ad hoc rules for the terminal entry set to reduce propellant consumption, such as minimizing velocity at a certain downrange distance to the target. This objective is accomplished by the guidance algorithm using a nested computational structure. At the inner level, the propellant map is used to determine the optimal ignition point along each predicted trajectory. At the outer level, the bank angle profile is optimized to find the optimal ignition state that is reachable from the current vehicle

state. To mitigate the complexity of determining the bank angle function during flight, a parametrization is chosen on the ground, and its parameters are repeatedly optimized during flight. Thus, the three key components of the approach are the powered descent propellant map, the bank angle parametrization, and the onboard determination of the vehicle state at which the entry phase ends and powered descent begins.

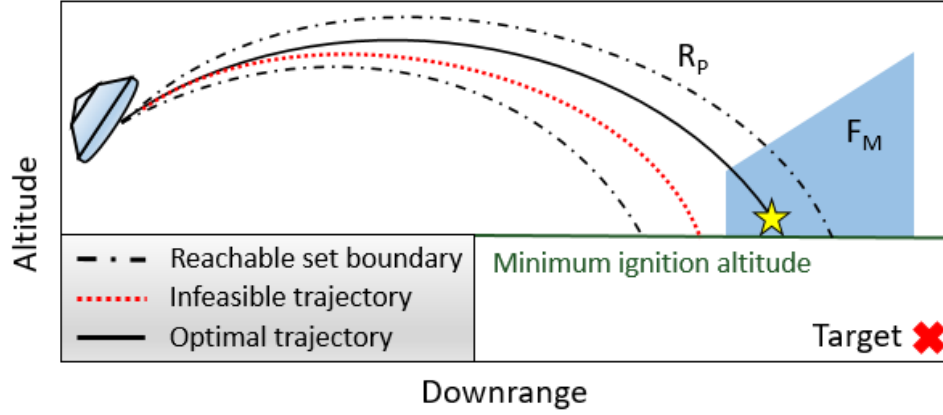


Figure 9.2: The guidance algorithm seeks both the propellant-optimal ignition point (indicated by a star on the plot) and the trajectory whose optimal ignition point is best (indicated by the solid black line). The red, dotted trajectory is valid entry trajectory within the reachable set, but is infeasible because it does not intersect the powered descent feasible subset $\mathcal{F}_M$.

9.2.1 Bank Angle Parametrization

The entry phase guidance problem is intended to be solved onboard and so motivates a low-order parametrization of the bank angle profile. Parametrizing the bank angle function converts the optimal control problem into a more tractable nonlinear optimization problem. The choice of parametrization has important implications on the entry reachable set. The entry phase reachable set from an initial state $\mathbf{x}_0$ is a tube of trajectories satisfying the equations of motion under admissible control functions

$$\mathcal{R}(\mathbf{x}_0) = \{\mathbf{x} \mid \exists t_f \text{ and } \sigma[0, t_f] \text{ with } \mathbf{x} = \varphi(t_f; \mathbf{x}_0, \sigma[0, t_f])\}$$

and the portion of the reachable set that is of interest is its intersection with the powered descent feasible subset, i.e., $\mathcal{R} \cap \mathcal{F}_M$. Conceptually, by parametrizing the bank angle function in a particular way, we restrict the possible bank angle functions that can be generated, and as result will reduce the size of the reachable set. Let R_P be the reachable set corresponding to a parametrization P . The relative size of $\mathcal{R}_P \subset \mathcal{R}$ is not the only consideration in choosing a parametrization. It is also desirable that $\mathcal{R}_P$ retains the propellant optimal point (or set) and neighboring region. As a result, one might like to compare the propellant cost associated with a parametrization with the optimal solution over all parametrizations, i.e., compute

$$\delta m = \left(\min_{x \in (\mathcal{R} \cap \mathcal{F}_M)} \text{OCP}(x) - \min_{x \in (\mathcal{R}_P \cap \mathcal{F}_M)} \text{OCP}(x) \right) \geq 0 \quad (9.25)$$

in order to assess its quality, but this may be difficult to do. Instead, the minimum propellant costs of any two parametrizations can be compared against one another directly. Let λ be a measure of the volume of a state space set. Let P_A and P_B be two different parametrizations, such that $\lambda(\mathcal{R}_{P_A}) < \lambda(\mathcal{R}_{P_B})$, and $\lambda(\mathcal{R}_{P_A} \cap \mathcal{R}) < \lambda(\mathcal{R}_{P_B} \cap \mathcal{R})$. Figure 9.3 depicts the relationship between the sets $\mathcal{F}_M$, $\mathcal{R}$, $\mathcal{R}_{P_A}$, and $\mathcal{R}_{P_B}$ as well as a possible situation to demonstrate the possibility that $\mathcal{R}_{P_A}$ is smaller than $\mathcal{R}_{P_B}$, and can reach less of R , yet has a smaller optimality gap than parametrization P_B .

The commanded bank angle profile considered here is a constant bank angle magnitude σ_c with one bank reversal at a velocity v_r and the current velocity V is used as the independent variable in place of time,

$$\sigma(V; \sigma_c, v_r) = \begin{cases} \sigma_c & V \geq v_r \\ -\sigma_c & V < v_r \end{cases} . \quad (9.26)$$

These commands are fed to a rate limiter to approximate the nature of a reaction control system tracking the commands. Because the profile includes a reversal, it offers a means to control lateral motion.

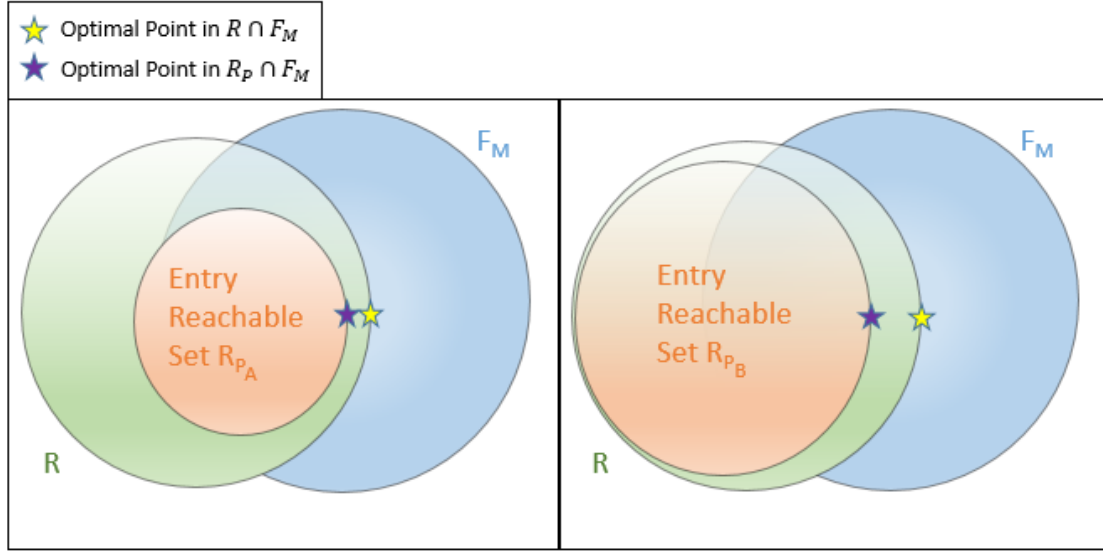


Figure 9.3: Consider two different parametrizations of the control, P_A and P_B . While P_B allows the vehicle to reach more of the reachable set $\mathcal{R}$ than P_A , much of R_{P_B} does not intersect $\mathcal{F}_M$. Additionally, the propellant optimal point in R_{P_A} is better than the optimal point in R_{P_B} , indicated by the smaller distance between the optima, which are depicted with stars. Note that these sets, depicted notionally as convex 2-D shapes, are actually 6-D and no claim is made about their convexity.

The parametrization remains the same after the reversal, but the guidance continues to update the parameter values. By setting the lower bound on v_r to be less than the ignition velocity during optimization, or greater than the current velocity, there exists the possibility to have one or no reversal planned at each update. This allows for as many reversals as there are optimization updates, but in simulations there will only be one or two reversals unless significant perturbations to the vehicle's planned heading occur after the first reversal. If an additional reversal is desired to eliminate large crossrange excursions, the parametrization is easily amended to accommodate one. A second reversal can be placed at a fixed velocity, and, exactly as the algorithm is currently applied, the first reversal timing is optimized until the reversal has been completed. Then, the second reversal is treated as the new optimization variable in addition to the bank angle magnitude. Such a “one-at-a-time” strategy was put forth in Ref. [65]. The result is, instead of one reversal with occasionally a second, all trajectories will feature two reversals with potential for a third.

For comparison, we also consider a second parametrization of the form

$$\sigma(V; v_1, v_2) = \begin{cases} 90^\circ & V \geq v_1 \\ -90^\circ & v_2 < V < v_1 \\ 0^\circ & V \leq v_2 \end{cases} \quad (9.27)$$

where the two parameters v_1, v_2 such that $v_1 > v_2$ dictate the timing of the bank reversals, and the bank angle magnitudes are fixed.

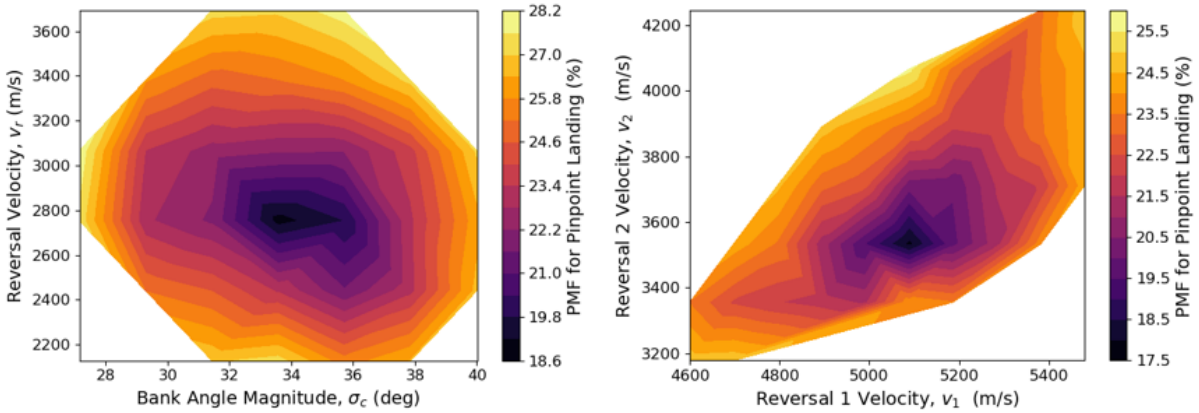


Figure 9.4: Example contours of the propellant cost of pinpoint landing using two different parametrizations: a constant bank angle with one reversal, and a profile with two reversal velocities. Not only does the choice affect the propellant contours, but the reachable set is quite different as well. The downrange targeted in the constant profile is 700 km and is not reachable by the second parametrization, which instead targets 625 km.

One key question when optimizing the parameters of the bank angle profile is whether there exist several isolated minima, or only a single global minimizer. Evaluating the objective function for a grid of the parameters defining the bank profile in each parametrization yields Figure 9.4, and indicates the objective function is likely convex, with steep gradients away from the optimum. From this, and repeated optimizations at various possible points, there is always a clear optimum. The right plot of Figure 9.4 shows that the two reversal parameters are significantly more coupled than the constant bank magnitude parametrization. This makes sense, as each reversal has a strong impact on the final heading, while in the

constant bank parametrization, the bank magnitude has a much weaker affect on the heading than the reversal velocity. Clearly the choice of parametrization affects not only the shape of these contours, but also the minimum PMF achievable, as the second parametrization has a superior optimum. Note, however, that due to the difference in reachable set between the two parametrizations, the optimal downrange distance from the entry interface is different. For simplicity, the constant parametrization is adopted for the results in this paper. A more detailed analysis of the choice of parametrization is left for future work.

The shape of these objective functions indicates that gradient-based methods should be effective in determining the optimal profile parameters, however, determining the gradient of the objective function with respect to the switching velocity is generally difficult. As a result, derivative-free methods are preferred for this application. Powell’s conjugate direction method [66] is used in all of the examples herein. It is effective because for the chosen parametrization, the contours are nearly independent, so 1-D searches over each bounded parameter can very quickly locate the optimum.

9.2.2 Powered Descent Propellant Mapping

By repeatedly solving the powered descent optimal control problem, Eq. 9.4, for different ignition states, a pointwise approximation of a subset of the pinpoint landing feasible set $\mathcal{F}_M$ is computed via sampling. The full descent state vector is 7-D, but with two degenerate dimensions: the crossrange velocity, which is always zero by definition, and the initial mass, which is assumed to be fixed. A consequence of the dimensionality of the state space is a requirement to compute a large number of solutions - 16,807 solutions are required to span each of the remaining 5 dimensions with only 7 points. From sampled solutions we construct an approximation of the propellant costs for states in $\mathcal{F}_M$. Sampling is applicable irrespective of the powered descent algorithm to be used; for the problem of constrained propellant-

optimal solutions, there exist significantly more efficient methods of computing the set $\mathcal{F}_M$ and the associated propellant cost, such as the convex optimization-based approach detailed in Ref. [11].

This mapping may be stored and interpolated in a variety of ways. Neural networks and other machine learning approaches were considered for their universal function approximation property. However, upon examination of the shape of the 5-D space together with numerical validation using additional OCP sample solutions, piecewise-linear 5-D interpolation was chosen to approximate the propellant mapping.

9.2.3 Trajectory Prediction and Powered Descent Ignition

Once the bank angle parametrization is specified, the estimated entry state is integrated numerically to generate a prediction of the remaining unpowered entry trajectory. The components of the constraint vector Eq. 9.12 used in the definition of $\mathcal{F}_M$ are then applied to the predicted trajectory to determine the interval that should be checked for the propellant-optimal transition to powered descent. These constraints greatly reduce the number of calls to the descent guidance propellant mapping required to find the optimum. As a reminder, these constraints include an upper bound on velocity corresponding to the maximum achievable deceleration with the available propellant, a minimum ignition altitude required for safe powered descent and subsequent EDL operations, and a maximum distance to the target, again related to the available propellant. Figure 9.5 depicts this process including a portion of the trajectory already flown, the entry flight prediction, constraints, pinpoint landing target, and predicted optimal ignition point.

Once the constraints have been applied, 1-D optimization is applied to the remaining trajectory interval, indicated by the dash black line, by viewing the fixed trajectory as a function of some independent variable. Our implementation uses velocity as the independent

variable over which to search, but time or distance are also possibilities. 1-D optimization is used to very quickly locate the optimum, rather than determine the solution by brute force by checking all of the remaining points. Once the solution is determined, the commanded bank angle sign and magnitude are computed as a function of the vehicle estimated velocity, and are then sent to the control system.

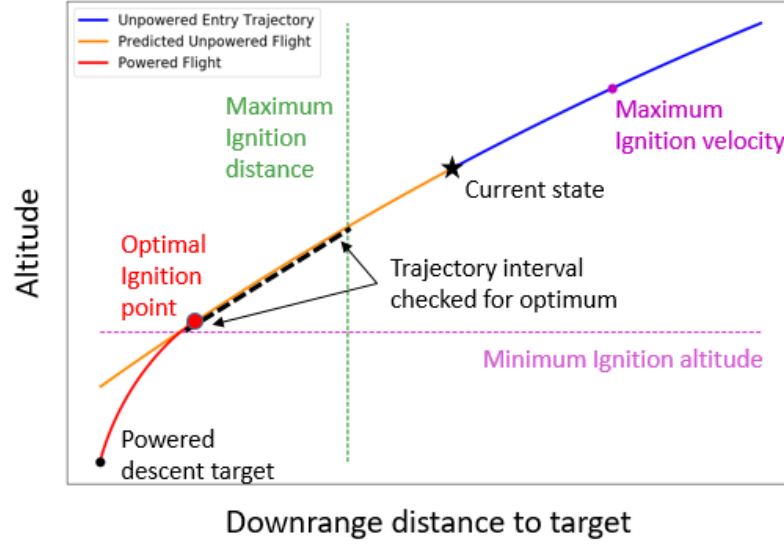


Figure 9.5: Only the interval of the predicted entry trajectory satisfying constraints on velocity, altitude, and distance to the target are searched for the propellant-optimal ignition state.

Chapter 10

Assessment of Entry Guidance for SRP-Based EDL

In order to characterize the behavior and performance of the propellant optimal entry guidance algorithm presented in Chapter 9, simulations are conducted under various circumstances. In all examples, the vehicle under consideration has a nominal $L/D = 0.24$, and a ballistic coefficient of 310 kg/m^2 , approximately double that of Mars 2020. The vehicle's propulsion model includes an initial T/W of 4 with a constant $I_{sp} = 290 \text{ s}$, and $T_{\min} = 0.5T_{\max}$. These parameters are summarized in Table 10.1. The site targeted at

Property	Value	Units
Ballistic coefficient	310	kg/m^2
Lift to drag ratio	0.24	-
Bank angle magnitude limit	90	deg
Bank angle rate limit	10	deg/s
Initial thrust to weight ratio	4	-
Specific Impulse	295	s

Table 10.1: Model parameters used in all simulation results presented.

the termination of the descent phase is 700 km downrange with $z_{\text{target}} = \dot{z}_{\text{target}} = 0$. Performance is evaluated using the propellant mass fraction (PMF), the portion of the vehicle that is propellant, as the primary metric. The bounds used in the constraints Eq. 9.12 are $z_{\text{min}} = 3$ km, $[d_{\text{min}}, d_{\text{max}}] = [0, 25]$ km, $v_{\text{max}} = 700$ m/s. The entry phase terminates at the ignition velocity last predicted by the entry guidance algorithm.

10.0.1 Sensitivity Analysis

The performance of the algorithm is first investigated in a series of one-dimensional sensitivities to parametric uncertainties, including lift and drag coefficients, and two forms of density uncertainty. For this section, the atmospheric density is modeled as

$$\rho = \rho_0 e^{-\frac{R-R_P}{h_s}} \quad (10.1)$$

where $\rho_0 = 0.0158 \text{ kg/m}^3$ is the density at the surface and $h_s = 9354.5$ m is the scale height. Uncertainty is modeled in each of these quantities. Variations in ρ_0 correspond to constant percentage shift at all altitudes, while variations in the scale height alter the density's gradient with respect to altitude, $\frac{\partial \rho}{\partial h} = -\frac{\rho}{h_s}$. The guidance algorithm is called at four fixed velocities, 5500, 4000, 2000, and 1000 m/s.

Due to its reliance on predictions, the algorithm is susceptible to model uncertainty. Reference [67] examined the performance of predictive algorithms in the presence of such uncertainties and found, like Ref. [68], that model adaptation during flight is essential to good performance in the presence of such parametric uncertainty. In the simulations presented, trajectory predictions are made aerodynamic accelerations scaled ratios of current sensed accelerations to modeled accelerations, L/L_{model} and D/D_{model} . Because the first three perturbations result in constant ratios of L/L_{model} and D/D_{model} , the predictions will match the simulated environment, and the predicted propellant consumption and ignition state

should not vary significantly with each call to the entry guidance algorithm. In contrast, the scale height uncertainty will result in aerodynamic ratios that vary with altitude, causing error in the predictions due to their inability to account for unknown future variations. Thus it is expected to see greater variation in the predicted propellant required as well as the predicted ignition state. This behavior is exhibited in Fig. 10.1.

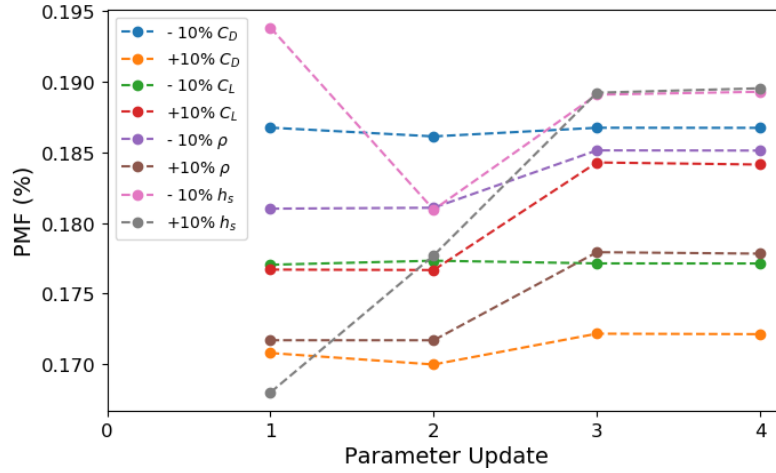


Figure 10.1: The predicted PMF and associated ignition state may change each time the guidance algorithm is called. The solutions are less volatile when the discrepancy between the model predictions and the simulated environment is small. Larger variations in the solution due to variations in h_s are expected since they result in unknown future density variations.

Scenario	PMF (%)	PMF-Nominal (%)	Ignition Velocity (m/s)
Nominal	18.07	0.00	525.7
-10% C_D	19.98	1.92	550.6
+10% C_D	17.25	-0.81	494.6
-10% C_L	18.16	0.09	516.7
+10% C_L	17.97	-0.10	526.9
-10% ρ	18.94	0.88	548.7
+10% ρ	17.74	-0.33	501.8
-10% h_s	18.92	0.86	531.6
+10% h_s	17.82	-0.24	506.1

Table 10.2: A summary of PMFs for a series of $\pm 10\%$ uncertainties compared to a nominal scenario with no uncertainty. Scenarios with significant increases in PMF relative to the nominal scenario have been highlighted.

Scenario	PMF (%)	Downrange to Target (m)	Crossrange to Target (m)	Altitude (m)	Horizontal Velocity (m/s)	Vertical Velocity (m/s)
Nominal	18.07	10390.75	473.1	3111.9	-501.3	-135.0
-10% C_D	19.98	15587.7	1125.3	3145.4	-534.2	-133.5
+10% C_D	17.25	10709.4	428.4	3363.0	-471.1	-150.5
-10% C_L	18.16	11398.7	293.0	3246.1	-496.4	-143.3
+10% C_L	17.97	12897.7	532.2	3471.2	-506.9	-143.8
-10% ρ_0	18.94	12963.6	614.5	3355.4	-532.5	-132.1
+10% ρ_0	17.74	11790.4	198.7	3502.2	-478.6	-151.0
-10% h_s	18.92	15130.6	265.2	3154.6	-513.4	-137.7
+10% h_s	17.82	10722.2	438.8	3330.6	-483.4	-149.7

Table 10.3: A summary of the propellant-optimal ignition states for a series of $\pm 10\%$ uncertainties. Optimal trajectories generally feature low crossrange, and altitudes close to the minimum altitude constraint.

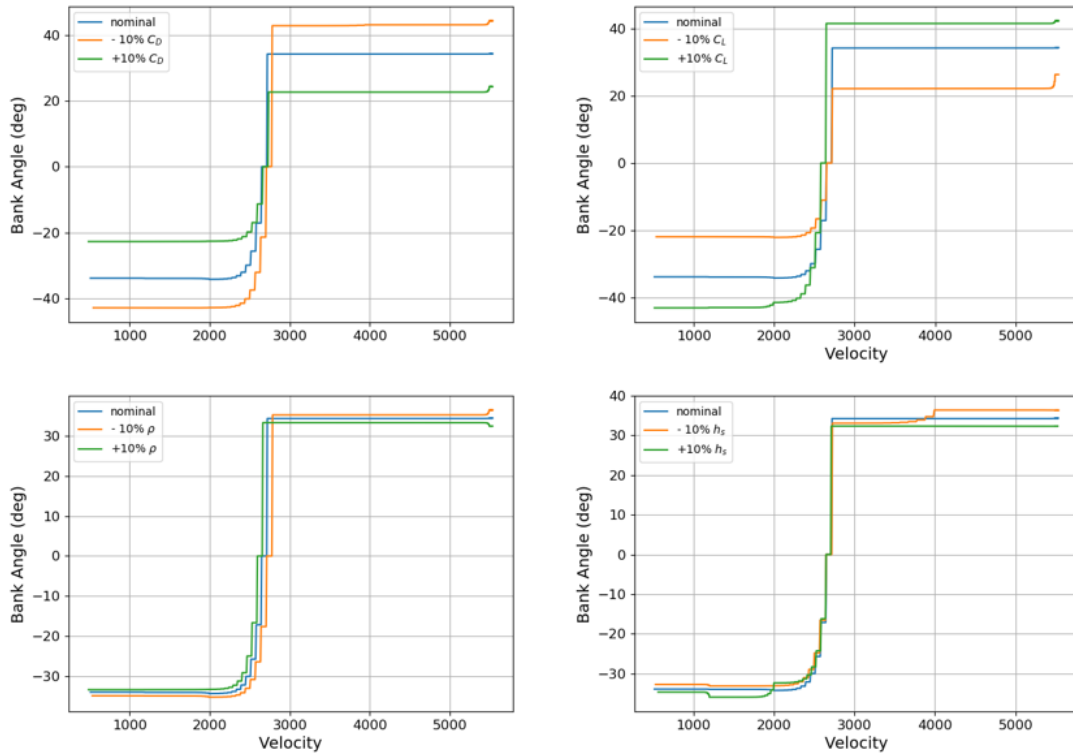


Figure 10.2: The bank angle profiles for a series of 1-D parametric sensitivities. Despite their similar effect on drag, the controller deals with $\pm C_D$ quite differently than $\pm \rho$.

Figure 10.2 shows the resulting bank angle profiles. Due to their similar effect on the vehicle L/D ratio, the perturbations pairs $(+C_D, -C_L)$ and $(-C_D, +C_L)$ result in essentially

the same update to the bank angle magnitude. The lift dispersed cases essentially fly identical altitude-velocity profiles but have slightly different reversal velocities due to the impact of lift on lateral motion.

Table 10.2 summarizes the PMF for each scenario and compares them to the nominal scenario, while Table 10.3 gives the corresponding propellant-optimal ignition states. Several trends are noticeable. The three scenarios with the highest PMF all involve lower than nominal drag acting on the vehicle, leading to higher ignition velocities and less favorable ignition states. Although the $-10\% \rho_0$ case features the same decrease in drag as $-10\% C_D$, it also preserves the L/D ratio. Unlike the drag coefficient variations, the lift coefficient variations produced very little change in PMF. There is a noticeable asymmetry present in each of the dispersions that affects drag: the minus variation, or “low drag” scenario, requires a greater increase in PMF than is saved in the positive variation. Note that a constant -10% variation in drag (from any source) over the entire trajectory is a substantial perturbation that is challenging to overcome in the already thin atmosphere. Despite this, the additional PMF required relative to the nominal scenario is modest.

All of the trajectories feature lofting near the end of the entry phase (not shown), and the terminal altitude is almost always very close to the minimum altitude constraint. Since altitude decreases monotonically with velocity after lofting, this allows the vehicle to decelerate for as long as possible, reducing the velocity that must be nulled by SRP. Scenarios in which the optimal ignition occurs higher indicate some other constraint is active, such as overshooting the optimal downrange distance at which to ignite. Although there is no separate guidance logic for heading alignment, it is clear from the small crossrange values in Table 10.3 that by targeting the optimal point in $\mathcal{F}_M$, the vehicle heading is aligned prior to ignition.

10.0.2 Monte Carlo Simulation

Quantity	Nominal	3σ
Lift coefficient	0.35	10%
Drag coefficient	1.46	10%
Entry radius	3522 km	0
Entry longitude	66.87°	0.15°
Entry latitude	21.28°	0.08°
Entry velocity	5500 m/s	3 m/s
Entry flight path angle	-15.75°	0.25°
Entry heading angle	-13.60°	0.25°

Table 10.4: The mean and 3σ values of the inputs used for the Monte Carlo simulation.

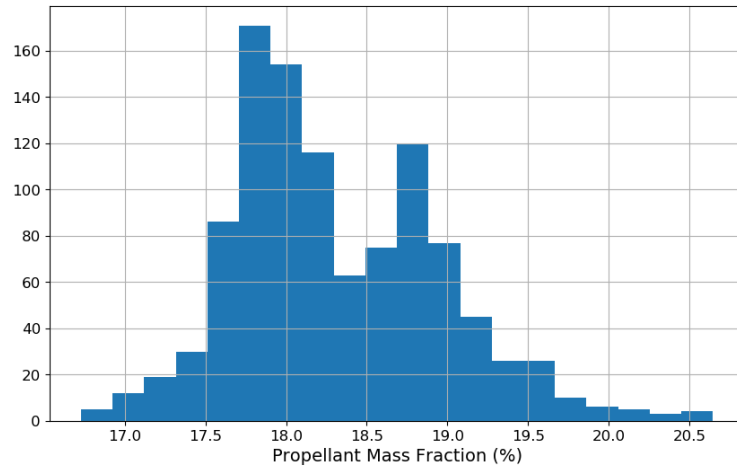


Figure 10.3: Propellant mass fraction for the Monte Carlo samples.

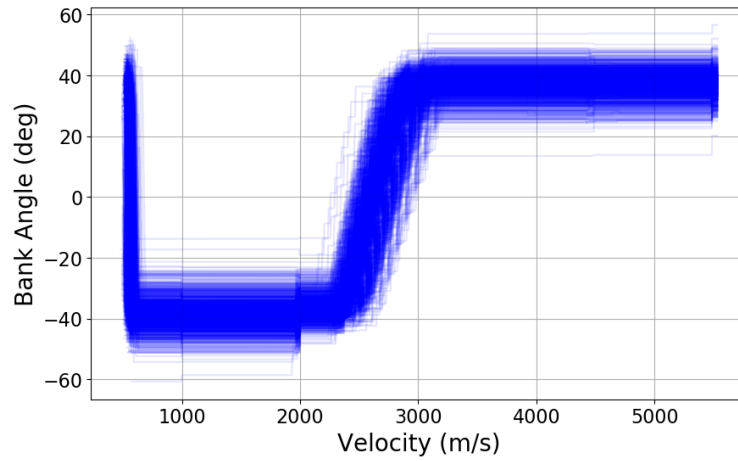


Figure 10.4: Bank angle profiles resulting from calls to the guidance algorithm at $V = [5490, 4500, 3000, 2000, 1000]$ m/s.

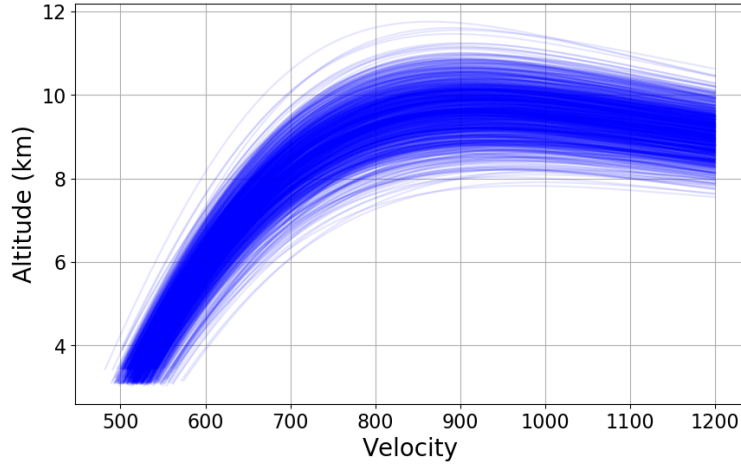


Figure 10.5: All of the trajectories feature lofting as a means to further decelerate prior to ignition.

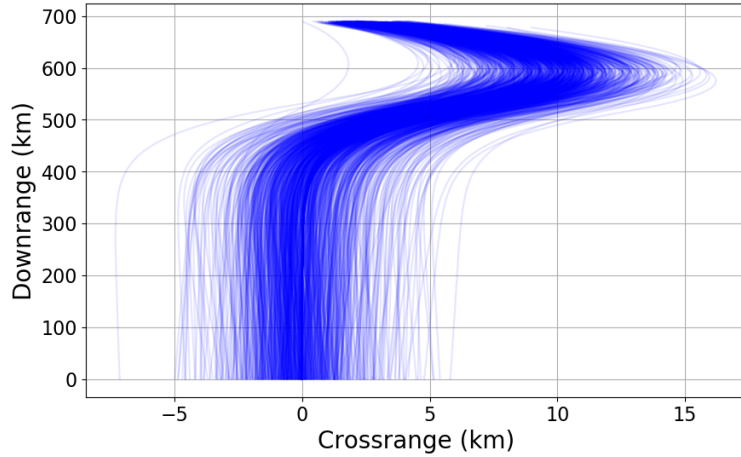


Figure 10.6: Downrange and crossrange flown from the entry state. Some trajectories fly significant crossranges due to the single planned bank reversal. Note that there are initial downrange errors in addition to than the initial crossrange errors, but due to the difference in scale they are not easily visible.

A Monte Carlo of 1000 samples is conducted with entry state delivery errors and aerodynamic dispersions, listed in Table 10.4, and a higher fidelity atmosphere model. The atmospheric density is modeled using MarsGRAM [69], rather than the exponential model employed specifically for the sensitivity study.

Figure 10.3 shows the distribution of PMFs and Figs. 10.4-10.6 display the entry trajectories. The median PMF is 18.2 %, and the 99 percentile is 20.1%. From Fig. 10.5 it is

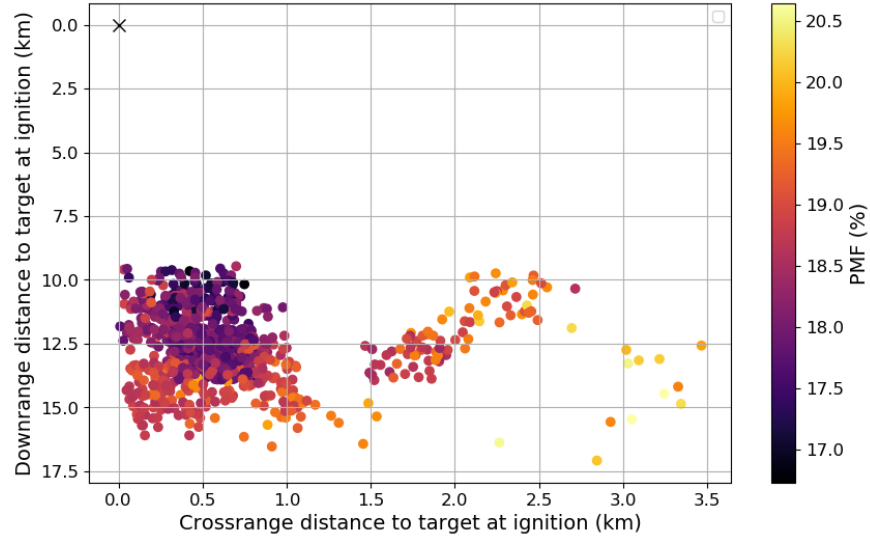


Figure 10.7: Downrange and crossrange to the target position. The crossrange is generally small, with 88% under 1 km, indicating the vehicle heading is well-aligned at ignition.

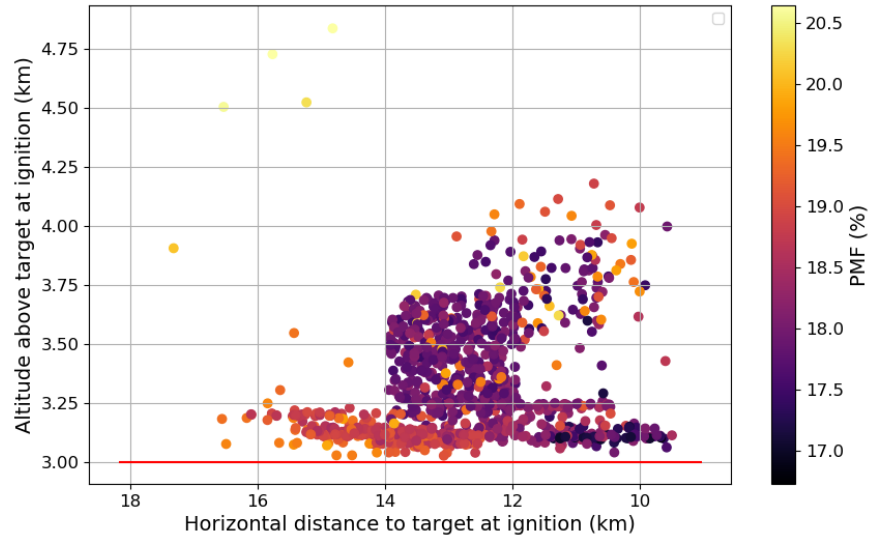


Figure 10.8: The minimum altitude constraint, indicated in red, causes low energy scenarios to trigger at longer downrange distances and higher velocities, resulting in increased PMF.

evident that all of the samples feature at least a mild lofting near the end of the entry phase. Figure 10.4 depicts the bank angle profiles. The chosen parametrization of the bank profile, coupled with the nature of the disturbances modeled, means that unless significant perturbation of the vehicle heading occurs after the first reversal, there will be only one reversal in total. In instances where the prediction was inaccurate and the reversal was poorly timed

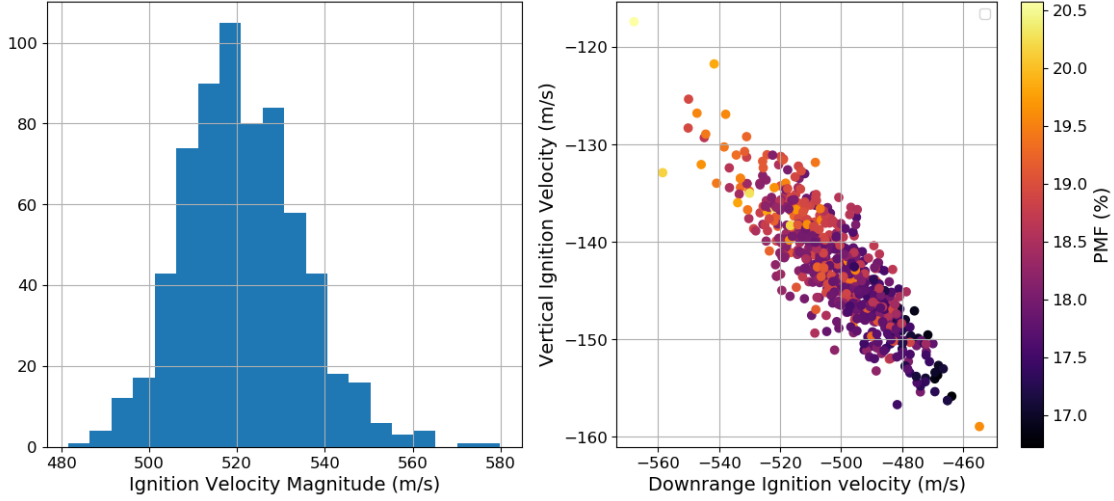


Figure 10.9: Ignition velocity magnitude (left) and components (right). Lower ignition velocities tend to occur at steeper flight path angles for the chosen parametrization. The correlation between velocity and PMF is evident, but the slowest ignition point (bottom right of the right plot) is nowhere near the lowest PMF value, demonstrating the importance of other state variables in determining the PMF required.

as a result, a late second reversal is used to correct the vehicle heading.

Figures 10.7-10.9 plot the ignition states at the ignition velocity determined by the last call to the entry guidance algorithm. For the vehicle under consideration, the optimal downrange to the target is generally between 10-15 km. Nearly all samples have a crossrange distance of less than 1 km at ignition, and the maximum crossrange samples trigger with around 3.5 km crossrange to the target. As seen in Fig. 10.9, ignition velocities occur over a range of nearly 100 m/s. PMF and velocity at ignition are naturally correlated, but the lowest velocity ignition (the point in the bottom right of the right plot) is near the upper end of the PMF range, which shows that other state variables may yet have a strong impact on the PMF required to land.

Figure 10.8 shows that altitude at ignition is consistently low, within 1-2 km of the minimum altitude constraint. This is not surprising, as after the vehicle has passed the point of lofting, the lowest velocity along an entry trajectory will occur at the minimum altitude. The consistently low ignition altitudes suggest a guidance strategy that triggers ignition at a

fixed altitude may perform nearly as well as searching for the optimal ignition state. Doing so would eliminate the need to find the optimum ignition point along a trajectory, removing the nested computation structure, and leaving only the optimization over the parameters of the bank angle profile. An additional benefit of such a strategy is a reduction of the propellant mapping interpolation to four dimensions, requiring far fewer solutions to represent the mapping with the same accuracy. Additionally, some of the suboptimality associated with an altitude trigger would be mitigated by the vehicle flying slightly differently knowing that the ignition altitude is different.

Reference [63] also noted that propellant-optimal ignitions generally occur near the last feasible entry state, but submits that propellant consumption alone is not a sufficient criterion for powered descent ignition because it does not account for operational margins. We suggest alternatively that by defining the minimum altitude constraint with operational margin in mind, the entry guidance algorithm may utilize an altitude trigger and focus exclusively on propellant consumption without issue. Another possible generalization is to trigger on a fixed glideslope angle, essentially allowing the final altitude to be lower if the vehicle is also nearer to the target.

The minimum altitude constraint also implications during mission design. Not only does it strongly affect the geometry of the powered descent phase, but it also affects the optimal entry trajectory length and the minimum propellant required. Figure 10.10 plots these quantities versus the value of the altitude constraint, while Fig. 10.11 shows how the powered descent trajectories vary with the constraint. For the vehicle considered, and a constant bank angle magnitude during entry, the unconstrained optimal ignition altitude is about 1.5 km above the target. When the targeted downrange distance is free, the growth in PMF due to the constraint is modest, but the optimal entry downrange distance grows by nearly 40 km when imposing a 5 km constraint, and the propellant-optimal downrange distance from the target at ignition nearly doubles.

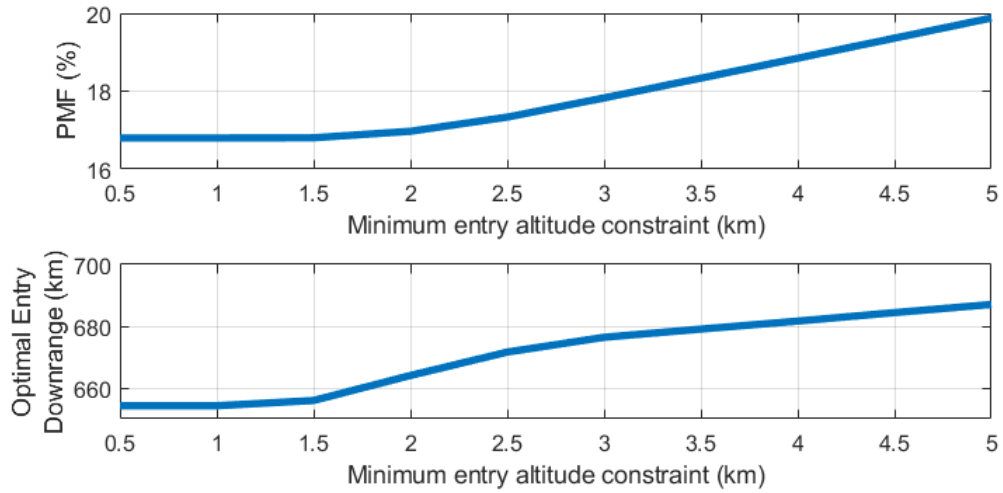


Figure 10.10: The optimal placement of the target from the entry interface varies considerably with minimum entry altitude constraint has a strong impact on the geometry of the propellant optimal powered descent trajectories.

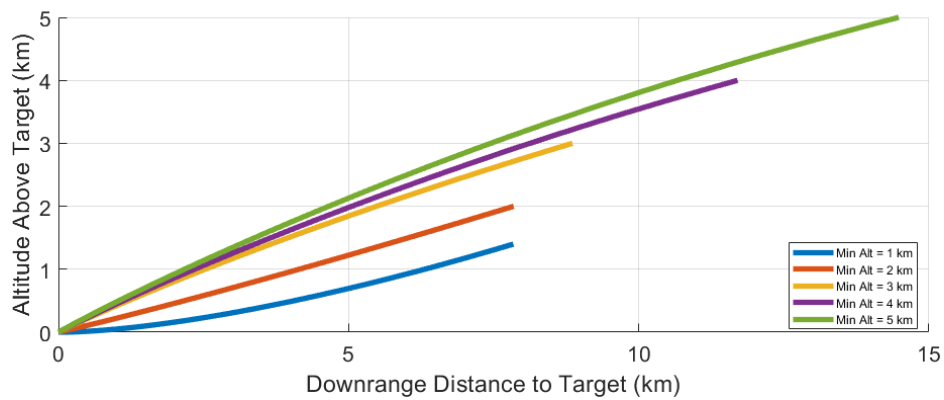


Figure 10.11: The minimum entry altitude constraint has a strong impact on the geometry of the propellant-optimal powered descent trajectories.

Chapter 11

Conclusions

In Chapters 4-8 we have examined the feasibility of designing a reference trajectory that, in closed-loop reference trajectory-based longitudinal entry guidance, yields the robust performance required for higher elevation landing than achieved to date. The entry guidance problem is posed as a robust optimal control problem with uncertain initial state and parameters for the closed-loop dynamics. The performance index is a weighted combination of maximizing mean altitude and minimizing the standard deviations of terminal altitude and downrange distance at the end of the entry phase just prior to the descent and landing phases. The weights on the standard deviations offer an intuitive way to adjust the robust performance. To significantly reduce computation relative to Monte Carlo simulation, the statistics required to evaluate the performance index are estimated with sample trajectories computed via the unscented transform. The resulting large-scale robust optimal control problem is solved by differential dynamic programming. Case studies of MSL- and SRL-class entry vehicles demonstrated both the robust performance of the longitudinal entry guidance and the computational feasibility of the design method. Post-design Monte Carlo simulations verified that the statistics estimated with the unscented transform are sufficiently accurate. An interesting feature of the generated reference bank angle is that, though not imposed a

priori, it has margins which result naturally to achieve robustness. While a full quantitative comparison with MSL and Mars 2020 is not possible due to differences in models and assumptions, the results indicate the robust optimal guidance could support higher elevation landings than achieved to date.

In Chapters 9 and 10 we examined entry guidance for a theoretical mission with a high ballistic coefficient. The primary metric in quantifying performance of the algorithm is the required propellant to achieve pinpoint landing. The target set for the entry guidance algorithm is generated by simulating the performance of the powered descent guidance algorithm. A predictor-corrector strategy is used to track the optimal ignition state. Assessment for a vehicle approximately twice the ballistic coefficient of Mars 2020 demonstrate the guidance is effective in reducing the propellant required; only modest increases in propellant are required despite the presence of large uncertainty in the atmosphere and aerodynamics.

11.1 Future Work

There are several key areas in which this research could progress. Concerning the robust optimal entry guidance for current generation missions, the first is consideration of different control laws in order to improve performance. While entry guidance using a linear control law has been flown on two Mars missions, and also produced compelling results in this dissertation, there remains the prospect of formulating the ROCP using a nonlinear control law to further improve performance. Control methods such as feedback linearization [70], nonlinear predictive control [71, 13], and sliding mode control [72, 73] have been studied in the context of entry guidance. Some of these methods are more directly applicable than others; considerations include differentiability of the control law, the number of free parameters to be chosen or, potentially, optimized, and whether the method requires integration of additional differential equations, especially if the number depends on the number of sigma

points. The first consideration is related to the solution via DDP, which requires the second derivatives to exist. Depending on the form of the controller, non-differentiable functions can be approximated by smooth functions, as was done for the saturation function in this work. The latter considerations are to retain tractability of the resulting ROCP, which can quickly grow in scale. Feedback linearization is a strong candidate because the transformation from the original state to the feedback linearized state is required to be a diffeomorphism, thereby satisfying the differentiability requirement. The method has no additional differential equations. After feedback linearization, the new system has a linear input-output structure, and the new control can be chosen as an affine state feedback, which can be readily optimized by the method presented here.

Another area to consider is further improvements and modifications to the DDP algorithm. The modified DDP algorithm has the capability to optimize the entry flight path angle in addition to the reference control and feedback gains. However, the EFPA has a first-order effect on the total acceleration and heat load. Current vehicles have total acceleration limits of 12-15g; manned missions are expected to have limits near 5g. Incorporating probabilistic state path constraints of the form $\bar{a} + w_a \sigma_a \leq a_{\max}$ would make the EPFA optimization more useful, and the weight w_a would allow for a trade between performance and constraint satisfaction.

A natural avenue for future work is the inclusion of lateral guidance during phase two and a change to third phase guidance at a fixed velocity. Including the parameters of the lateral deadband in the optimization is difficult due to the discontinuous nature of the command, which is the sign of the bank angle. The lateral guidance commands a bank reversal when the heading error (or another measure of lateral performance, such as crossrange distance)

$$|\psi(v) - \psi^*(v, \mathbf{x})| > c_0 + c_1 v + c_2 v^2 \quad (11.1)$$

where ψ^* is the heading angle that forms a great circle arc between the current position and the target position and c_i are the coefficients determining the reversal threshold. However, it may be possible to consider the full (lateral+longitudinal) state vector, and to model the lateral guidance, but not include the coefficients as part of the optimization process. The sigma point trajectories will react to the lateral guidance, commanding sign changes during the optimization process. Heading alignment guidance can then be modeled since the lateral states are included in the formulation. The heading alignment gain could be treated as an additional optimization variable. Finally, the coefficients can be optimized (or at least tuned carefully) in an outer loop.

Future avenues for the SRP-based guidance include an efficient method of computing the target set. Convex optimization-based methods to generate constrained controllable sets have been proposed in [11]. As mentioned in Chapter 9, exploring an altitude-based trigger as an alternative to the propellant optimal trigger would make the algorithm significantly less expensive computationally, but it remains to be seen if it will be as effective. Finally, the parametrization used had the advantage of simplicity, but determining an alternative low order parametrization to approximate the structure of velocity-minimizing trajectories is another area of potential improvement.

Bibliography

- [1] Adam Nelessen, Chloe Sackier, Ian Clark, Paul Brugarolas, Gregorio Villar, Allen Chen, Aaron Stehura, Richard Otero, Erisa Stilley, David Way, et al. Mars 2020 entry, descent, and landing system overview. In *2019 IEEE Aerospace Conference*, pages 1–20. IEEE, 2019.
- [2] JJ Wray, RE Milliken, Colin M Dundas, Gregg A Swayze, JC Andrews-Hanna, AM Baldridge, M Chojnacki, JL Bishop, BL Ehlmann, Scott L Murchie, et al. Columbus crater and other possible groundwater-fed paleolakes of terra sirenum, mars. *Journal of Geophysical Research: Planets*, 116(E1), 2011.
- [3] Robert D Braun and Robert M Manning. Mars exploration entry, descent and landing challenges. In *2006 IEEE Aerospace Conference*, pages 18–pp. IEEE, 2006.
- [4] Gavin F Mendeck and Lynn Craig McGrew. Entry guidance design and postflight performance for 2011 mars science laboratory mission. *Journal of Spacecraft and Rockets*, 51(4):1094–1105, 2014.
- [5] Brian K Muirhead, Austin K Nicholas, Jeffrey Umland, Orson Sutherland, and Sanjay Vijendran. Mars sample return mission concept status. *Acta Astronautica*, 2020.
- [6] Aron A Wolf, Jeff Tooley, Scott Ploen, Mark Ivanov, Behcet Acikmese, and Konstantin Gromov. Performance trades for mars pinpoint landing. In *2006 IEEE Aerospace Conference*, pages 16–pp. IEEE, 2006.
- [7] Douglas A Craig, Patrick Troutman, and Nicole Herrmann. Pioneering space through the evolvable mars campaign. In *AIAA Space 2015 Conference and Exposition*, page 4409, 2015.
- [8] Behcet Acikmese and Scott R Ploen. Convex programming approach to powered descent guidance for mars landing. *Journal of Guidance, Control, and Dynamics*, 30(5):1353–1366, 2007.
- [9] Ufuk Topcu, Jordi Casoliva, and Kenneth D Mease. Minimum-fuel powered descent for mars pinpoint landing. *Journal of Spacecraft and Rockets*, 44(2):324–331, 2007.
- [10] Behcet Acikmese, M Aung, Jordi Casoliva, Swati Mohan, Andrew Johnson, Daniel Scharf, David Masten, Joel Scotkin, Aron Wolf, and Martin W Regehr. Flight testing

- of trajectories computed by g-fold: Fuel optimal large divert guidance algorithm for planetary landing. In *AAS/AIAA spaceflight mechanics meeting*, 2013.
- [11] Utku Eren, Daniel Dueri, and Behçet Açıkmeşe. Constrained reachability and controllability sets for planetary precision landing via convex optimization. *Journal of Guidance, Control, and Dynamics*, 38(11):2067–2083, 2015.
 - [12] Zhengshi Yu, Pingyuan Cui, and John L Crassidis. Design and optimization of navigation and guidance techniques for mars pinpoint landing: Review and prospect. *Progress in Aerospace Sciences*, 94:82–94, 2017.
 - [13] Joel Benito and Kenneth Mease. Nonlinear predictive controller for drag tracking in entry guidance. In *AIAA/AAS Astrodynamics Specialist Conference and Exhibit*, page 7350, 2008.
 - [14] A Bombelli, L Soler, and K Mease. Entry trajectory planner for high elevation mars landing. In *11th International Planetary Probe Workshop*, volume 1795, 2014.
 - [15] Jiateng Long, Ai Gao, Pingyuan Cui, and Yang Liu. Mars atmospheric entry guidance for optimal terminal altitude. *Acta Astronautica*, 155:274–286, 2019.
 - [16] Yiyu Zheng, Hutao Cui, and Yuanhang Ai. Indirect trajectory optimization for mars entry with maximum terminal altitude. *Journal of Spacecraft and Rockets*, 54(5):1068–1080, 2017.
 - [17] Guangfei Duan and Kenneth D Mease. Mars entry guidance for high elevation landing with uncertainty quantification and reduction. In *22nd AIAA International Space Planes and Hypersonics Systems and Technologies Conference*, page 5315, 2018.
 - [18] Ping Lu, Ronald R Sostaric, and Gavin F Mendeck. Adaptive powered descent initiation and fuel-optimal guidance for mars applications. In *2018 AIAA Guidance, Navigation, and Control Conference*, page 0616, 2018.
 - [19] Changhuang Wan, Gangshan Jing, Ran Dai, and Jeremy R Rea. Fuel-optimal guidance for end-to-end human-mars entry, powered-descent, and landing mission. In *AIAA Scitech 2020 Forum*, page 1472, 2020.
 - [20] Connor Noyes and Ken Mease. Entry guidance for propellant optimal powered descent ignition on mars. In *AIAA/AAS Astrodynamics Specialist Conference*. AAS, 2020.
 - [21] Ryan Loxton, Kok Lay Teo, and Volker Rehbock. Robust suboptimal control of nonlinear systems. *Applied Mathematics and Computation*, 217(14):6566–6576, 2011.
 - [22] Chris Phelps. *Computational optimal control of nonlinear systems with parameter uncertainty*. PhD thesis, UC Santa Cruz, 2014.
 - [23] Benjamin Robert Saunders. *Optimal trajectory design under uncertainty*. PhD thesis, Massachusetts Institute of Technology, 2012.

- [24] John Darlington, Constantinos C Pantelides, Berç Rustem, and BA Tanyi. Decreasing the sensitivity of open-loop optimal solutions in decision making under uncertainty. *European Journal of Operational Research*, 121(2):343–362, 2000.
- [25] I Michael Ross, Ronald J Proulx, Mark Karpenko, and Qi Gong. Riemann–stieltjes optimal control problems for uncertain dynamic systems. *Journal of Guidance, Control, and Dynamics*, 38(7):1251–1263, 2015.
- [26] Hans Seywald and Renji Kumar. Desensitized optimal trajectories. *Analytical Mechanics Associates Rept*, pages 03–16, 2003.
- [27] Venkata Ramana Makkapati, Mehregan Dor, and Panagiotis Tsiotras. Trajectory desensitization in optimal control problems. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 2478–2483. IEEE, 2018.
- [28] Pingyuan Cui, Zeduan Zhao, Zhengshi Yu, and Juan Dai. Terminal altitude maximization for mars entry considering uncertainties. *Acta Astronautica*, 145:446–455, 2018.
- [29] Todd Vincent Small. *Optimal trajectory-shaping with sensitivity and covariance techniques*. PhD thesis, Massachusetts Institute of Technology, 2010.
- [30] Shuang Li and Yuming Peng. Mars entry trajectory optimization using doc and dcnlp. *Advances in Space Research*, 47(3):440–452, 2011.
- [31] Hongying Xu and Hutao Cui. Robust trajectory design scheme under uncertainties and perturbations for mars entry vehicle. In *2015 IEEE International Conference on Computational Intelligence & Communication Technology*, pages 762–766. IEEE, 2015.
- [32] Yuval Tassa, Nicolas Mansard, and Emo Todorov. Control-limited differential dynamic programming. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1168–1175. IEEE, 2014.
- [33] Emanuel Todorov and Weiwei Li. A generalized iterative lqg method for locally-optimal feedback control of constrained nonlinear stochastic systems. In *Proceedings of the 2005, American Control Conference, 2005.*, pages 300–306. IEEE, 2005.
- [34] Dallas Ives, David Geller, and Gilbert Carman. Apollo-derived mars precision lander guidance. In *23rd atmospheric flight mechanics conference*, page 4570, 1998.
- [35] Guangfei Duan. *Entry Guidance and Navigation for High Elevation Mars Landing*. PhD thesis, UC Irvine, 2019.
- [36] Soumyo Dutta and David W Way. Comparison of the effects of velocity and range triggers on trajectory dispersions for the mars 2020 mission. In *AIAA Atmospheric Flight Mechanics Conference*, page 0245, 2017.
- [37] Nguyen X Vinh, Adolf Busemann, and Robert D Culp. Hypersonic and planetary entry flight mechanics. *NASA Sti/Recon Technical Report A*, 81:16245, 1980.

- [38] Joel Benito Manrique. *Advances in spacecraft atmospheric entry guidance*. University of California, Irvine, 2010.
- [39] HL Justh, CG Justus, and HS Ramey. Mars-gram 2010: improving the precision of mars-gram. 2011.
- [40] Stephen R Lewis, Matthew Collins, Peter L Read, François Forget, Frédéric Hourdin, Richard Fournier, Christophe Hourdin, Olivier Talagrand, and Jean-Paul Huot. A climate database for mars. *Journal of Geophysical Research: Planets*, 104(E10):24177–24194, 1999.
- [41] Simon J Julier and Jeffrey K Uhlmann. New extension of the kalman filter to nonlinear systems. In *Signal processing, sensor fusion, and target recognition VI*, volume 3068, pages 182–193. International Society for Optics and Photonics, 1997.
- [42] Gregory Lantoine and Ryan P Russell. A hybrid differential dynamic programming algorithm for constrained optimal control problems. part 1: Theory. *Journal of Optimization Theory and Applications*, 154(2):382–417, 2012.
- [43] Gregory Lantoine and Ryan P Russell. A hybrid differential dynamic programming algorithm for constrained optimal control problems. part 2: Application. *Journal of Optimization Theory and Applications*, 154(2):418–442, 2012.
- [44] Zhaoming Xie, C Karen Liu, and Kris Hauser. Differential dynamic programming with nonlinear constraints. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 695–702. IEEE, 2017.
- [45] Andrei Pavlov, Iman Shames, and Chris Manzie. Interior point differential dynamic programming. *arXiv preprint arXiv:2004.12710*, 2020.
- [46] Evangelos Theodorou, Yuval Tassa, and Emo Todorov. Stochastic differential dynamic programming. In *Proceedings of the 2010 American Control Conference*, pages 1125–1132. IEEE, 2010.
- [47] Naoya Ozaki, Stefano Campagnola, Ryu Funase, and Chit Hong Yam. Stochastic differential dynamic programming with unscented transform for low-thrust trajectory design. *Journal of Guidance, Control, and Dynamics*, 41(2):377–387, 2018.
- [48] Naoya Ozaki, Stefano Campagnola, and Ryu Funase. Tube stochastic optimal control for nonlinear constrained trajectory optimization problems. *Journal of Guidance, Control, and Dynamics*, 43(4):645–655, 2020.
- [49] David H Jacobson and David Q Mayne. Differential dynamic programming. 1970.
- [50] David Q Mayne. Differential dynamic programming—a unified approach to the optimization of dynamic systems. In *Control and Dynamic Systems*, volume 10, pages 179–254. Elsevier, 1973.

- [51] Larry Armijo. Minimization of functions having lipschitz continuous first partial derivatives. *Pacific Journal of mathematics*, 16(1):1–3, 1966.
- [52] Jorge Nocedal and Stephen Wright. *Numerical optimization*. Springer Science & Business Media, 2006.
- [53] Wei Sun, Evangelos A Theodorou, and Panagiotis Tsiotras. Continuous-time differential dynamic programming with terminal constraints. In *2014 IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning (ADPRL)*, pages 1–6. IEEE, 2014.
- [54] Teresa B Culver and Christine A Shoemaker. Optimal control for groundwater remediation by differential dynamic programming with quasi-newton approximations. *Water Resources Research*, 29(4):823–831, 1993.
- [55] Andrea Saltelli. Global sensitivity analysis: an introduction. In *Proc. 4th International Conference on Sensitivity Analysis of Model Output (SAMO’04)*, pages 27–43. Citeseer, 2004.
- [56] Connor D Noyes and Kenneth D Mease. Mars entry guidance for high elevation landing via robust optimal control. *Journal of Spacecraft and Rockets*, 2021 (submitted for review).
- [57] Brian K Muirhead, Austin K Nicholasa, Chad Edwardsa, Jeffrey Umlanda, Sanjay Vijendranb, and Richard Zureka. Mars sample return campaign concept architecture.
- [58] A Miguel San Martin, Steven W Lee, and Edward C Wong. The development of the msl guidance, navigation, and control system for entry, descent, and landing. 2013.
- [59] Bradley A Steinfeldt, Michael J Grant, Daniel A Matz, Robert D Braun, and Gregg H Barton. Guidance, navigation, and control system performance trades for mars pinpoint landing. *Journal of Spacecraft and Rockets*, 47(1):188–198, 2010.
- [60] Behcet Acikmese and Scott R Ploen. Convex programming approach to powered descent guidance for mars landing. *Journal of Guidance, Control, and Dynamics*, 30(5):1353–1366, 2007.
- [61] Behcet Acikmese, M Aung, Jordi Casoliva, Swati Mohan, Andrew Johnson, Daniel Scharf, David Masten, Joel Scotkin, Aron Wolf, and Martin W Regehr. Flight testing of trajectories computed by g-fold: Fuel optimal large divert guidance algorithm for planetary landing. In *AAS/AIAA spaceflight mechanics meeting*, 2013.
- [62] Allan R Klumpp. Apollo lunar descent guidance. *Automatica*, 10(2):133–146, 1974.
- [63] Ping Lu. Propellant-optimal powered descent guidance. *Journal of Guidance, Control, and Dynamics*, 41(4):813–826, 2018.
- [64] Christopher L Marsh and Robert D Braun. Fully-propulsive mars atmospheric transit strategies for high-mass missions. *Journal of Spacecraft and Rockets*, 48(2):271–282, 2011.

- [65] Guangfei Duan, Marc Rovira Navarro, and Kenneth D Mease. Trajectory tracking and online replanning for mars entry. In *AIAA/AAS Astrodynamics Specialist Conference*, page 5444, 2016.
- [66] Michael JD Powell. An efficient method for finding the minimum of a function of several variables without calculating derivatives. *The computer journal*, 7(2):155–162, 1964.
- [67] Alexander I Kozynchenko. Analysis of predictive entry guidance for a mars lander under high model uncertainties. *Acta Astronautica*, 68(1-2):121–132, 2011.
- [68] Ping Lu. Entry guidance: a unified method. *Journal of Guidance, Control, and Dynamics*, 37(3):713–728, 2014.
- [69] HL Justh. Mars global reference atmospheric model 2010 version: Users guide. 2014.
- [70] Sanjay Bharadwaj, Anil V Rao, and Kenneth D Mease. Entry trajectory tracking law via feedback linearization. *Journal of Guidance, Control, and Dynamics*, 21(5):726–732, 1998.
- [71] Ping Lu. Constrained tracking control of nonlinear systems. *Systems & control letters*, 27(5):305–314, 1996.
- [72] Sanjay Talole, Joel Benito, and Kenneth Mease. Sliding mode observer for drag tracking in entry guidance. In *AIAA Guidance, Navigation and Control Conference and Exhibit*, page 6851, 2007.
- [73] Juan Dai and Yuanqing Xia. Mars atmospheric entry guidance for reference trajectory tracking. *Aerospace Science and Technology*, 45:335–345, 2015.