Aerocapture, Entry, and Co-Delivery in Uncertain Planetary Atmospheres

by

S. W. Albert

B.S., Purdue University, 2018

M.S., University of Colorado Boulder, 2020

A thesis submitted to the
Faculty of the Graduate School of the
University of Colorado in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
Ann and H.J. Smead Department of Aerospace Engineering Sciences
2023

Committee Members:
Prof. Hanspeter Schaub
Dr. Robert D. Braun
Prof. Alireza Doostan
Prof. Jay McMahon
Dr. Soumyo Dutta

Albert, S. W. (Ph.D., Aerospace Engineering Sciences)

Aerocapture, Entry, and Co-Delivery in Uncertain Planetary Atmospheres

Thesis directed by Prof. Hanspeter Schaub

Aerocapture, the method of entering orbit via a single pass through the atmosphere of a planet, is an enhancing or enabling technology for a range of interplanetary missions. Compared to propulsive maneuvers, aerocapture can reduce cruise duration while decreasing the total mass expended for orbit insertion, thus leaving more time and mass for the primary science mission. Two mission classes in particular both benefit from aerocapture and are of high priority for the next decade of planetary science: exploration of the ice giants, Uranus and Neptune, and low-cost small satellite platforms. However, despite its potential benefits, aerocapture has never been implemented in flight. This is primarily because of the large uncertainties involved, which must be modeled and adequately mitigated by closed-loop autonomous guidance onboard the spacecraft.

Aerocapture guidance has been well-studied for vehicles that control their atmospheric flight by changing the orientation of a lift vector, but is not as well developed for a class of flight vehicles that achieve control by releasing a drag device mid-flight. Known as drag modulation, this control mechanism is significantly less complex in terms of hardware and avionics than lift modulation, and is thus appealing for small satellite missions. However, the state of the art guidance solutions have a computational demand that is both high and difficult to bound. This dissertation contributes a novel guidance algorithm for drag-modulated aerocapture that achieves equivalent performance to the state of the art, but with reduced computational demand.

One of the most pernicious sources of uncertainty that aerocapture guidance must mitigate is atmospheric density, which varies over space and time. While scientific and engineering atmosphere models are available and well-characterized for on-the-ground studies, models that retain this fidelity while being significantly more compact and analytically tractable are desirable for onboard use. This dissertation develops reduced-dimensionality models of uncertain atmosphere for use onboard a spacecraft, derives a method for updating the model based on noisy measurements, and demonstrates the ability to accurately predict future state uncertainty resulting from these environmental dispersions without requiring the use of random sampling. These contributions have a range of potential applications, including incorporation into future stochastic guidance algorithms.

Many of the mission concepts most relevant to aerocapture, such as the Uranus Orbiter and Probe, involve more than one flight vehicle. These missions benefit from the ability to deliver both spacecraft to their destination with minimal disruption to the overall concept of operations. While a number of missions have successfully executed multi-vehicle architectures in the past, this "co-delivery" method has not received dedicated systematic attention. This dissertation addresses the concept as a topic in its own right, investigating the ability to co-deliver an orbiter and probe from a single approach trajectory without the need for a divert maneuver. Co-delivery of an entire network of probes from a single, non-maneuvering mothership is also investigated. Finally, expressions for relative motion in the velocity frame are derived in order to provide a mathematical model that is more intuitive than the typical rotating orbit frame for highly-elliptical orbits, as are common for aerobraking, entry, and aerocapture.

To illustrate the unifying motivation for this work, the contributions of this dissertation are applied to an example problem: the concept of reducing atmospheric uncertainty for aerocapture by including a fly-ahead probe that enters the atmosphere some time before the orbiter. While this idea has been proposed several times, the benefit conferred to the orbiter by the probe has not been quantified. The contributions of this dissertation naturally lend themselves to addressing this problem, as well as other entry, aerocapture, and co-delivery scenarios for future interplanetary missions.

Dedication

Do I dare disturb the universe?

– T. S. Eliot

To my parents and my grandparents, who taught me always to dare.

Acknowledgements

First, I am grateful to Hanspeter Schaub for his wisdom and expertise, and for exemplifying who I aspire to be as a research professional and leader. I am indebted to every committee member: to Bobby Braun, who inspired and guided my initial passion for EDL; to Alireza Doostan and Jay McMahon, both of whom went above and beyond as teachers and collaborators as their courses informed integral parts of this dissertation; and to Som Dutta, who has welcomed me into the EDL research community and shown me around the place. Angela Bowes, my NASA mentor, deserves special mention for her technical and personal advice throughout this PhD. Alex Austin, Dan Burkhart, Erik Bailey, and Breanna Johnson have also played invaluable roles. I am also grateful to Dan Dumbacher and Mike Grant for their formative mentorship during my time at Purdue.

This dissertation would not have happened without the support of the NASA Space Technology Research Fellowship, and I also owe a great deal of thanks to the Matthew Isakowitz Fellowship Program and the Stamps Family Charitable Foundation for supporting my education. Further, my participation in the Keck Institute for Space Studies workshop inspired and informed significant portions of this work. One of the great joys of this PhD has been collaboration with other students, from whom I have learned so much. Jack Ridderhof and Ethan Burnett merit a special acknowledgement; I also extend my gratitude to Melis Grace, John Martin, Kevin Bonnet, Boris Benedikter, and George Rapakoulias. To the entire AVS lab and EsDL: I cannot be sure if I would have made it through this degree without you, but I am certain it would have been less enjoyable.

Finally, my deepest gratitude goes to my family for their constant love and support. To the many people who are not listed here but should be: first round's on me.

Contents

Chapter

Tables

Table

Figures

Figure

Chapter 1

Introduction

1.1 Motivation

Accelerating a spacecraft from rest atop a launch pad to the high speeds required for an interplanetary trajectory requires a massive expenditure of energy. After launch and cruise, the vehicle enters the local vicinity of its planetary destination, speeding up as it falls into the gravity well. If no action is taken (and if the trajectory avoids impacting the planet or atmosphere) then the spacecraft will, like a rollercoaster in the absence of friction, leave the gravity well and depart with the same planet-relative energy as it arrived, thus executing a flyby. In cases where the objective is not to perform a flyby but to enter orbit, the spacecraft must reduce its energy enough to be captured by the gravity well and achieve orbit insertion. To date, initial orbit insertion has always been accomplished via a propulsive maneuver.

The Tsiolkovsky rocket equation dictates an exponential relationship between required initial total mass including propellant (i.e. wet mass) m_0 and desired change in velocity ΔV [1],

$$\Delta V = v_e \ln \frac{m_0}{m_f}, \tag{1.1}$$

where v_e is effective exhaust velocity and m_f is final total mass without propellant, i.e. dry mass. The result is that the fraction of total wet mass that must be reserved for propellant (calculated as $1 - m_f/m_0$) is very large; around 90% for launch vehicles and typically in the range of 50% for planetary spacecraft using chemical propulsion [1, 2]. This so-called tyranny of the rocket equation[1] is a fundamental reason for the high cost-per-kilogram of launching into space[2] . As a consequence, orbit insertion is a major driving factor in the design of any spacecraft destined to reach orbit around a planet or moon.

Electric propulsion technologies such as Hall effect thrusters offer a way to loosen the grip of the rocket equation, and have been successfully demonstrated on interplanetary missions including NASA's Deep Space 1 and ESA's SMART-1 [3, 4]. They accomplish this by operating at much higher efficiency, increasing the value of v_e by as much as tenfold compared to chemical propulsion [5]. However, the maximum thrust of electric thrusters is proportional to available electrical power and they often must operate continuously for long periods of time [6]; this is especially limiting for missions to the outer planets, where incident solar radiation is greatly reduced. Moreover, low-thrust transfers designed for electric propulsion tend to require longer times of flight [7].

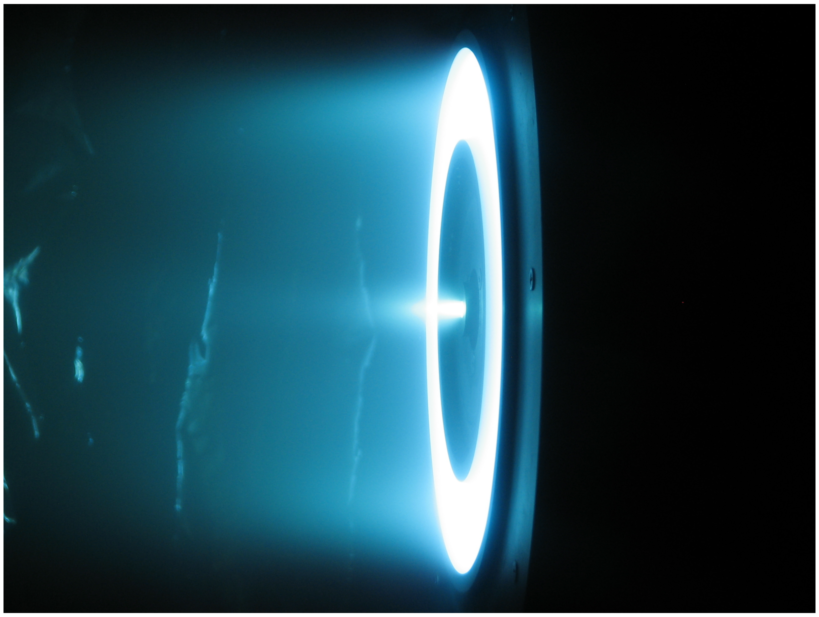


Figure 1.1: JPL's 6 kW Hall thruster[3]

Aerobraking provides another approach to reducing the total mass required for orbit insertion, and has been successfully employed at both Venus and Mars [8, 9, 10, 11]. This involves

---

[1] The Tyranny of the Rocket Equation by Don Pettit

[2] Interactive launch vehicle cost comparison from CSIS Aerospace

[3] Image: NASA/JPL-Caltech

gradually lowering the apoapsis of a high-energy initial orbit by repeatedly flying through the upper planetary atmosphere, intentionally incurring a small reduction in energy due to atmosphere drag each time, as shown in Fig. 1.2. However, insertion into the initial orbit still must be accomplished propulsively, so significant propellant mass is still required. Moreover, aerobraking missions often require hundreds of passes through the atmosphere, which takes months and requires demanding around-the-clock operations in addition to increasing mission risk [12].

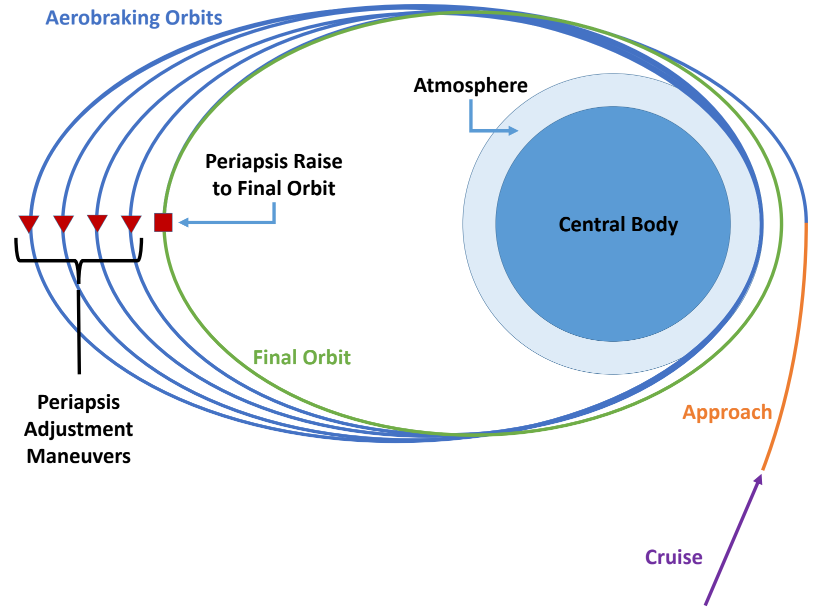


Figure 1.2: Aerobraking

Aerocapture is a promising alternative method of orbit insertion that addresses several of the shortcomings of both low-thrust propulsion and aerobraking, and will be the focus of much of the rest of this dissertation. Like aerobraking, aerocapture makes use of atmospheric drag to reduce the energy of the spacecraft to achieve the desired orbit. However, aerocapture trajectories rely on a single, lower-altitude pass through the atmosphere to capture into orbit [13, 14, 15], as shown in Fig. 1.3. As a consequence of the high heating environment encountered during hypersonic flight through mid-altitudes, aerocapture requires a protective aeroshell, much like those used for planetary entry missions. The vehicle also requires some method of flight control during atmospheric flight. This is achieved by judiciously adjusting the aerodynamic forces acting on the vehicle, and control approaches thus generally fall into two categories: lift modulation and drag modulation [16]. After exiting the atmosphere, the spacecraft executes a propulsive maneuver at the subsequent pass through apoapsis to raise periapsis out of the atmosphere, and performs other correction maneuvers as necessary. By essentially relying on a naturally-available resource (the atmosphere) rather than propellant, aerocapture could enable shorter transit times and lower total expended mass for orbit insertion for a variety of interplanetary mission concepts [17, 18, 19], including crewed mission to Mars [20, 21]. Although it has been proposed for a number of missions [22, 23, 24], aerocapture has never been flown [19].

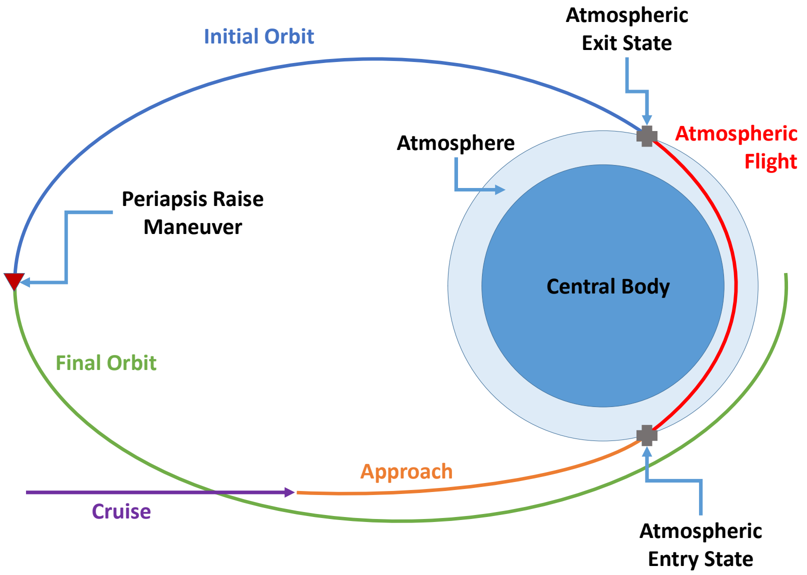


Figure 1.3: Aerocapture

Aerocapture is particularly relevant to two classes of missions of high interest for the next decade of planetary exploration: a flagship mission to the ice giants and low-cost small satellite interplanetary missions [25, 26]. To date, the ice giants, Uranus and Neptune, have each only been visited once, by Voyager 2 during flybys in the 1980s [27, 28]. The 2023-2032 Planetary Science

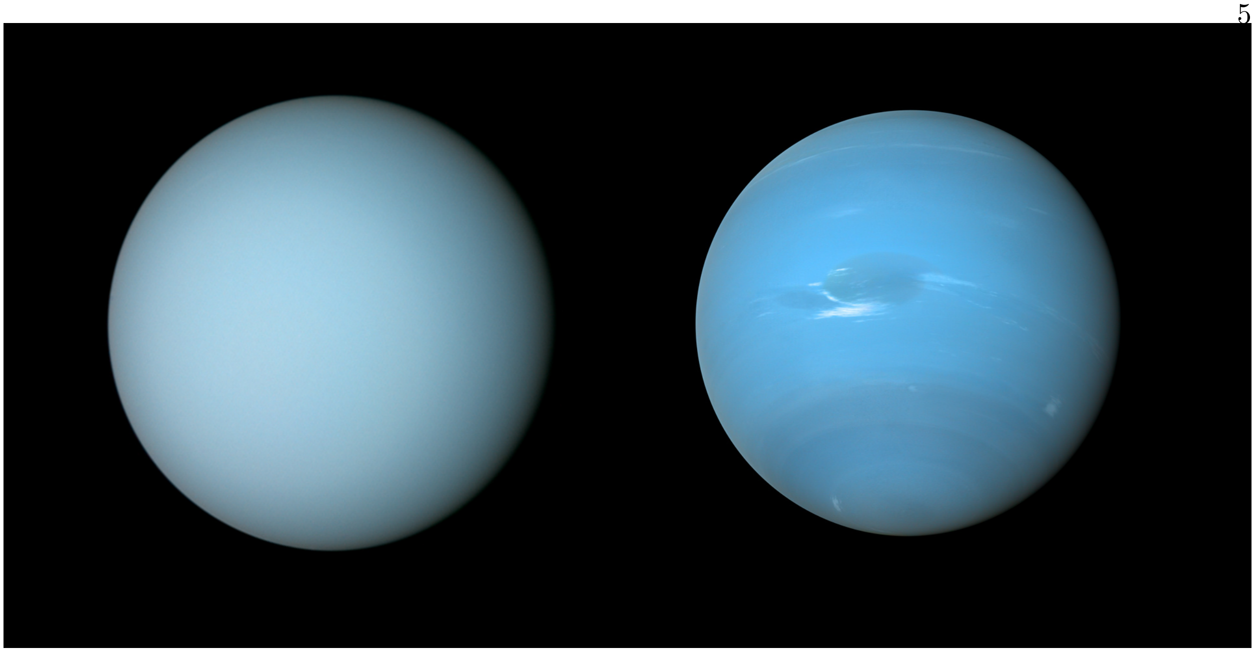


Figure 1.4: Voyager 2 portraits of Uranus (left) in 1986 and Neptune (right) in 1989[4]

Decadal Survey, *Origins, Worlds, and Life*, establishes a Uranus orbiter and probe mission as the highest priority new Flagship mission [29], delivering an in situ atmospheric probe as well as conducting a multi-year orbital tour. Trajectories to this outer planet are characterized by long trip times from Earth and large orbit insertion ΔVs [30, 31]. For traditional propulsive orbit insertion, this results in greatly reduced time and mass for the science mission. Aerocapture has been shown to reduce the total mass required for orbit insertion by some 40% for Uranus missions [32], while also reducing transit time by 2–5 years (15-30%) [25].

Small satellites (smallsats), especially CubeSats, have accounted for an increasingly large share of satellites launched each year since around 2012 [33]. Technological innovations, including the miniaturization of electronics and availability of commercial-off-the-shelf hardware, have led to a steady increase in the capabilities possible in these small form-factors, and CubeSat missions have now moved beyond serving a primarily educational role to making numerous notable scientific contributions [34]. A 2014 study sponsored by the Keck Institute for Space Studies (KISS) presented space science mission concepts "uniquely enabled by the small satellite platform," and recommended

---

[4] Image: NASA/JPL-Caltech/Björn Jónsson

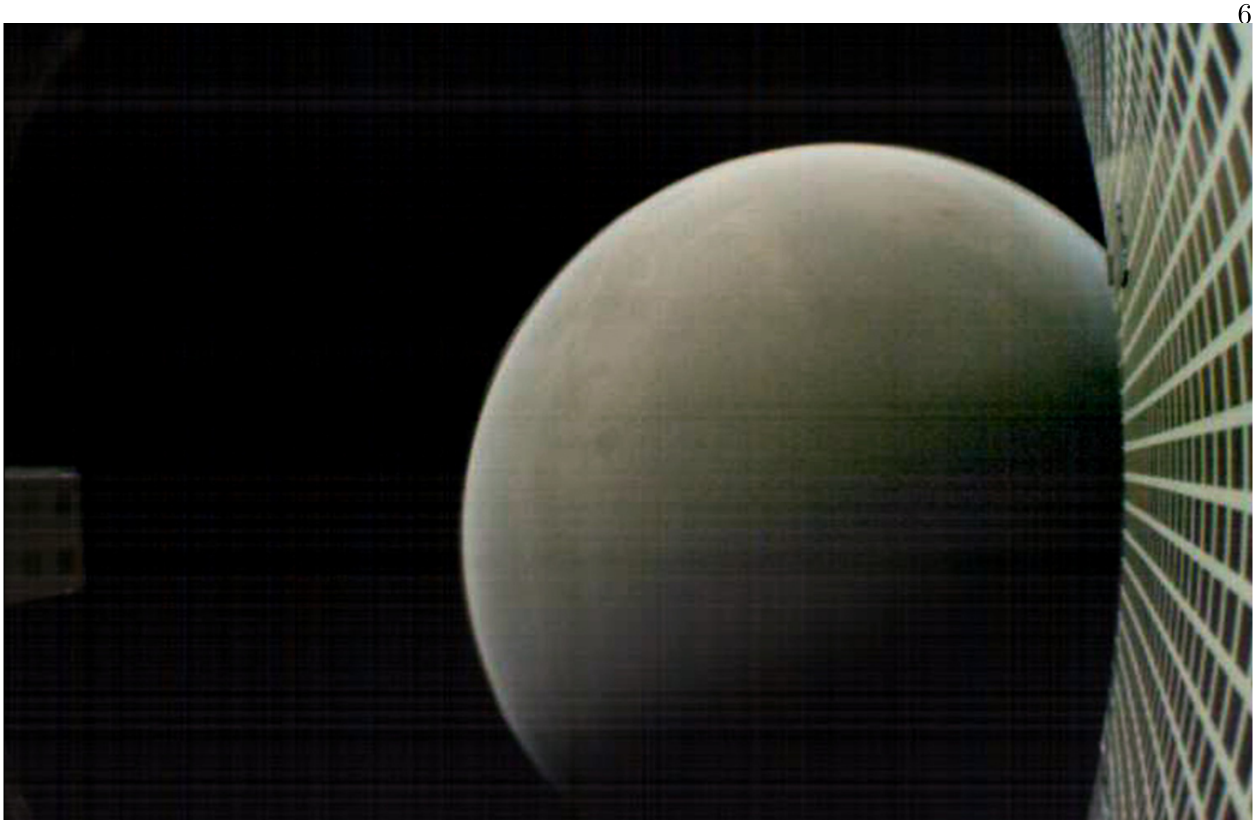


Figure 1.5: Image of Mars taken by MarCO-B after a successful InSight landing[5]

including small spacecraft as secondaries on all missions beyond low Earth orbit [35]. NASA has also studied a variety of mission concepts through its Planetary Science Deep Space SmallSat Studies program [36]. In November 2018 MarCO-A and MarCO-B, the twin CubeSat communications relays accompanying the InSight Mars lander, successfully demonstrated the merit of smallsats in deep space applications [37]. Aerocapture offers significant benefit to smallsats launched via rideshare with a primary mission, enabling orbit insertion despite the lack of high-ΔV systems at smallsat scale and reducing sensitivity to primary mission trajectory design [26, 38]. The NASA Science Mission Directorate (SMD) has re-established the Small Innovative Missions for Planetary Exploration (SIMPLEx) class of competed missions, which solicits smallsat missions for rideshare opportunities with primary SMD missions[6] . The 2022 Strategic Framework[7] released by the NASA

---

[5] Image: NASA/JPL-Caltech

[6] https://soma.larc.nasa.gov/simplex/

[7] https://techport.nasa.gov/framework

Space Technology Mission Directorate states that "an Earth-based aerocapture demonstration will reduce perceived risk and mature guidance and control methods" for aerocapture at other planetary destinations. Drag-modulated aerocapture is an especially good fit for smallsat missions because it could be significantly easier to integrate on a small spacecraft than other methods of atmospheric flight control [38, 39].

Of course, aerocapture does not deliver the benefits described above for free. The required aeroshell imposes significant packaging constraints on the spacecraft, and the hardware required to survive hypersonic atmospheric flight adds complexity, but perhaps the largest complication is the inherent risk in the maneuver. By definition, aerocapture trajectories tread a careful line between flying too low (possibly disintegrating in the atmosphere or impacting the surface) and too high (possibly failing to capture into orbit altogether). Hypersonic flight mechanics are nonlinear and highly sensitive to perturbations in the atmosphere, error in the vehicle state at entry, uncertainty in the modeled vehicle aerodynamics, and more. While the guided hypersonic entries of Apollo [40], Orion [41], Mars Science Laboratory (MSL)[42], and Mars 2020 [43] provide some degree of flight-heritage for aerocapture, it is often still perceived as high risk [44]. **Thus, advancing the state-of-the-art for aerocapture-related technologies could reduce the associated risk (both real and perceived) and improve the effective technology readiness level [19]; this is the underlying motivation for this thesis.** The following paragraphs introduce the specific areas in which this dissertation makes novel contributions; namely,

- efficient onboard aerocapture guidance,
- onboard modeling of uncertain atmospheres, and
- co-delivery concepts.

Closed-loop autonomous guidance is one of the critical challenges for aerocapture. Unlike planetary entry, passive trajectories or open-loop control are generally not feasible options for aerocapture because of the narrow window of success, on top of the significant sources of uncertainty and highly-sensitive nonlinear dynamics that are also characteristic of entry missions. Thus, the central objective for aerocapture guidance is to adapt to current conditions in order to guide the vehicle to the desired final orbit with minimal error. For example, onboard accelerometer measurements may result in a current state estimate that is higher-velocity than planned, possibly indicating that the encountered density and/or vehicle drag coefficient is lower than anticipated; in this scenario, the guidance needs to adjust the upcoming control profile to fly lower in the atmosphere than originally planned. The limited capacity of flight-heritage radiation-hardened onboard computers, combined with the requirement for the guidance algorithm to rapidly respond to new information, means that computational efficiency is a major performance metric (and potential limiting factor) for aerocapture guidance.

Another of the key technical challenges for aerocapture is that hypersonic flight trajectories are highly sensitive to variation in atmospheric density. Planetary atmospheres are characterized by high epistemic and aleatory uncertainty; that is, our ability to accurately predict local density is limited by insufficient data as well as by the inherent random variability in the system. Thus, the selection of an atmosphere model is an important consideration for uncertainty quantification (UQ) analysis of aerocapture trajectories, both for modeling and simulation during mission design and for rapid onboard predictions of uncertainty. The two most common choices are an exponential model or a semi-empirical atmosphere model such as a Global Reference Atmosphere Model (GRAM) from NASA or the Mass Spectrometer and Incoherent Scatter radar (MSIS) model from the US Naval Research Laboratory. The exponential model is useful because it provides a reasonably accurate approximation of how density varies with altitude while reducing the density profile to a function of two scalar parameters, surface-level density and atmospheric scale height [45]. Uncertainty can then be modeled by dispersing these two parameters, typically as Gaussian random variables. This approach is primarily useful as a simplified approximation enabling analytical approaches; the actual density profiles of planetary atmospheres have significant disagreement with an exponential curve, and dispersing only scale height and surface density will always retain the same exponential shape of the density profile and simply scale the result in either direction. The other common approach is using a model like GRAM that generates characteristic density profiles based on a

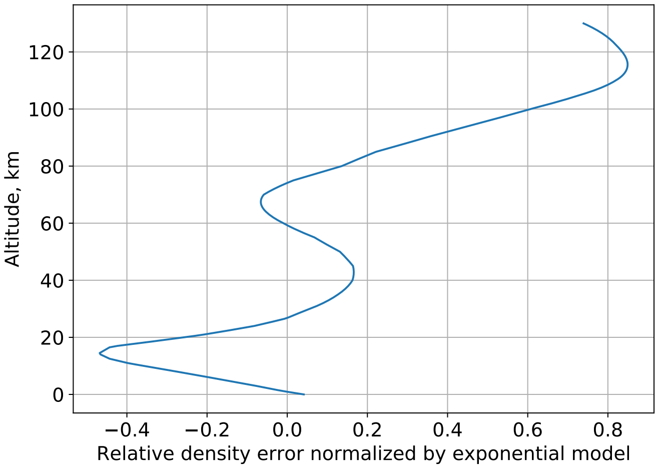


Figure 1.6: Relative error between nominal density profiles from the exponential model and Earth-GRAM2016

combination of data and modeling. These models are generally more realistic and more flexible than a simple exponential model, and have a built-in capability to randomly generate profiles with physically realistic perturbations. In the context of uncertainty quantification, the major limitation of GRAM and models like it is that they are often treated as a black box, such that a new density profile is generated for each trial in a Monte Carlo analysis.

The issue is that many UQ techniques, such as stochastic collocation [46, 47] and polynomial chaos expansion (PCE) [48, 49], require low stochastic dimensionality (i.e., a relatively small number of dispersed input parameters). By relying on random sampling techniques like Monte Carlo and simply selecting full pre-generated profiles, the analyst has implicitly forgone the implementation of other UQ techniques which, in some cases, may have outperformed Monte Carlo. This motivates the development of a parametric model of an uncertain atmosphere that is higher-fidelity than a simple exponential model, while reducing dimensionality compared to a discretized semi-empirical model like GRAM (for which the dimensionality is equal to the number of altitude points). Furthermore, onboard density models can benefit greatly from the ability to update density predictions based on in-flight measurements, so the developed model should accommodate such a method.

Returning now to the discussion of ice giants exploration and low-cost planetary science missions, observe that these two mission categories have another commonality: they either inherently require, or would significantly benefit from, more than one flight vehicle. In the case of the Uranus orbiter and probe, this statement is self-evident. In the latter category, networks of multiple small, fixed landers have been identified as a potential pathway to Mars surface exploration at reduced cost; this was a conclusion of a recent KISS workshop titled "Revolutionizing Access to the Martian Surface" as well as a follow-on "Low-Cost Science Mission Concepts for Mars Exploration" workshop [50, 51]. Other concepts are based on the idea of pathfinder probes that return data about a planetary atmosphere to primary mission, such as a probe released prior to crewed Mars landing from orbit or a small probe sent into the atmosphere of Uranus in advance of an aerocapture mission [19]. Finally, the "carry your own relay" architecture pioneered by the addition of the MarCO CubeSats to the InSight mission could be significantly enhanced if a combination of co-delivery and aerocapture enabled the relay spacecraft to enter orbit rather than continue on a flyby trajectory [37, 52]. While some co-delivery mission concepts have been either studied or successfully executed, the multi-vehicle architecture results in a number of unique mission design considerations [53, 52] that, currently, lack dedicated study. This motivates systematic study of co-delivery concepts, as well as a quantification of the extent to which a pathfinder probe would reduce risk to the primary mission.

1.2 State-of-the-Art Overview

1.2.1 Aerocapture

Aerocapture has been studied for decades [13, 14, 15, 17] and planned until various stages of development for missions including the Aeroassist Flight Experiment [22], Mars Surveyor 2001 orbiter [23], and Mars Sample Return [24]; however, it has never been implemented in flight. The Mars Polar Lander and Mars Climate Orbiter failures in 1999 led to the removal of aerocapture from the mission design for the 2001 Mars orbiter [54], and had the lasting impact of a relatively conservative approach by NASA to Mars missions and entry, descent, and landing technology [20].

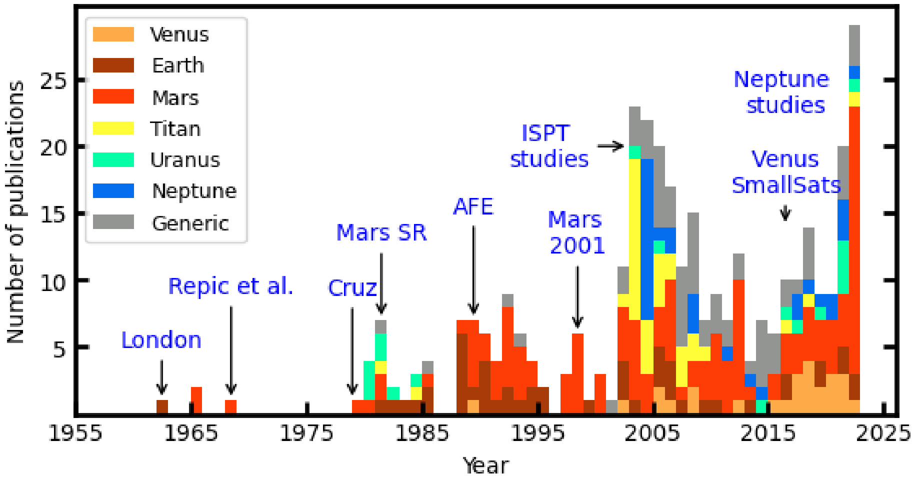


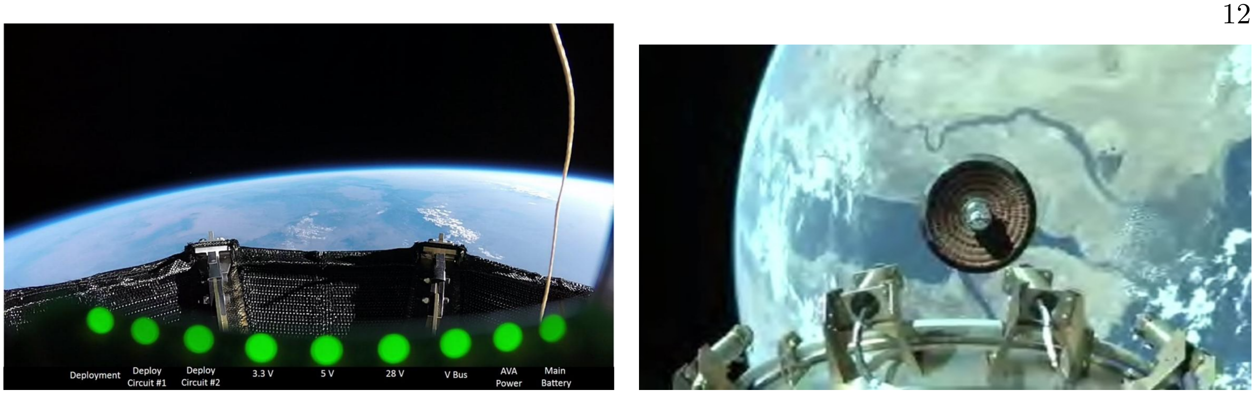
Figure 1.7: Number of aerocapture-related publications per year colored by target planet (ISPT refers to the multi-center NASA aerocapture system analysis studies)[8]

Between 2002 and 2006, a multi-center NASA team completed systems analyses for aerocapture missions to Mars [55], Venus [56], Titan [57], and Neptune [58]. These studies advanced the state-of-the-art of aerocapture at a systems engineering level, built around a set of design reference missions and quantitatively-informed assumptions. While the study team concluded that heritage blunt-body aeroshells would be sufficient for aerocapture at Mars, Venus, and Titan, they argued that a novel, higher lift-to-drag ratio (L/D) aeroshell would be necessary at Neptune. This conclusion – that aerocapture at the ice giants would require design and qualification of an entirely new class of entry vehicle as opposed to the incremental improvements to Viking-era technology that has characterized the NASA Mars program since the 1990s [20] – has posed a major programmatic barrier to the implementation of aerocapture.

Aerocapture has received considerable study in the nearly two decades since the multi-center NASA studies, particularly in the last five years, as summarized in Fig. 1.7. Significant hardware development has advanced deployable entry vehicle technology, enabling a much larger drag area than could otherwise fit in a launch vehicle fairing and reducing aeroheating compared to smaller rigid aeroshells with similar mass [59]. Suborbital flight tests have been successfully conducted

---

[8] From Wikimedia via A. Girija



(a) ADEPT sounding rocket test 1, 2018[10]

(b) HIAD suborbital test (LOFTID), 2022[11]

Figure 1.8: Successful flight tests of deployable entry vehicle technologies

for the Adaptive Deployable Entry and Placement Technology (ADEPT) and Hypersonic Inflatable Aerodynamic Decelerator (HIAD) platforms [60, 61][9] , bringing these technologies closer to readiness for mission infusion. Improved thermal protection systems, including the Heatshield for Extreme Entry Environments Technology (HEEET), have been developed and matured to a technology readiness level of 6 [62]. Advances have been made for guidance, navigation, and control for aerocapture, including: improved atmospheric estimation methods [63], more capable aerodynamic control mechanisms [64], development and systematic analysis of high-performance deterministic guidance algorithms [65, 54], and investigation of stochastic trajectory optimization methods applied to aerocapture [66, 67]. Tools for rapid conceptual design of aerocapture missions have become available [68]. Optical navigation techniques and improved estimation methods have also improved the accuracy possible in deep-space navigation [69]. As a result of all of this, aerocapture is becoming an increasingly feasible proposition [26, 25]. A 2016 study at the NASA Jet Propulsion Laboratory concluded that, while aerocapture technology readiness is destination-dependent, no prior flight demonstration would be needed to implement aerocapture at Titan, Mars, and possibly Venus [19]. Recent studies have argued that aerocapture at Neptune (a more stressing case than Uranus) is feasible with heritage blunt-body aeroshells [70, 71], avoiding the need to develop novel aeroshells as was recommended by the 2006 NASA study [58].

---

[9] LOFTID Flight Test Successful

1.2.2 Guidance

Significant investment has been dedicated to developing aerocapture guidance schemes, especially centered around the aforementioned missions that planned to use aerocapture but were eventually either cancelled or redesigned around propulsive orbit insertion. One well-studied algorithm, known as terminal point control (TPC), is derived from Apollo entry guidance and relies on linear feedback with gains derived using calculus of variations and evaluated about a pre-defined reference trajectory [72]. TPC benefits from very little onboard computational demand, and was selected from among other algorithms for the Mars Surveyor Program 2001 Orbiter as well as the French-contributed orbiter, Mars Premier, for a previous incarnation of Mars Sample Return [73, 74]. Versions of this algorithm were also used for entry guidance on the MSL and Mars 2020 missions [75, 43]. While the algorithm is robust and lightweight, some drawbacks of TPC – typical of linear control laws of this kind – are the need to manually tune feedback gains and the requirement to select a nominal profile before flight [54]. Another algorithm that has received extensive testing is the Hybrid Predictor-Corrector Aerocapture Scheme (HYPAS) [76], originally known as Analytic Drag Aerocapture Guidance [77]. HYPAS combines reference-based tracking guidance similar to TPC with an analytical predictor-corrector approach based on solutions to the planar equations of motion under equilibrium glide assumptions [54, 78, 77]. After a comparison campaign based on six degree-of-freedom simulations, HYPAS was selected for the Aeroassist Flight Experiment and was under development until cancellation of that mission [74, 22], and has been subsequently used for numerous aerocapture studies including the multi-center NASA studies mentioned earlier [54]. Notably, HYPAS does not require any pre-defined reference trajectories, leading to efficient code [74], but it may be less robust to uncertainties and relies on at least some portions of the flight profile being well-approximated by equilibrium glide.

More recently, much of the literature on aerocapture guidance has focused on numerical predictor-corrector (NPC) algorithms, which make predictions by numerically propagating the

---

[11] ADEPT image

[11] LOFTID image

nonlinear equations of motion rather than evaluating an analytical approximation. This means that NPC algorithms should be able to make the most accurate predictions of the future state, but this comes at the expense of increased computational demand as well as a lack of convergence guarantees. **These drawbacks of NPC guidance algorithms are a major motivation for the work detailed in Chapter 2.** NPC algorithms were studied for both the Aeroassist Flight Experiment and Mars Surveyor orbiter [79, 80, 81, 82], but while they were successfully demonstrated they were, at the time, not considered competitive with the TPC and HYPAS algorithms [73]. However, a combination of algorithmic improvements and a steady increase in availability of onboard computational power has led to a shift in the EDL literature to largely focus on NPCs [54]. Perhaps most notably, the Fully-Numerical Predictor-Corrector Aerocapture Guidance (FNPAG) presented in [65] and applied to mid-L/D Mars aerocapture in [83] improved over previous aerocapture NPC guidance schemes by incorporating the fact that, for deterministic in-plane motion, the optimal flight control law for aerocapture[12] is bang-bang.

1.2.3 Uncertainty Quantification

Every guidance algorithm discussed above, as well as all flight-heritage algorithms from guided entry missions, could be described as deterministic in nature. That is, they implicitly control uncertainty by updating commands based on new estimated of the current state, and the effectiveness of this approach is estimated in uncertainty quantification studies of the closed-loop dynamics, namely via Monte Carlo analyses. An alternative approach that has recently gained attention in the literature is to *explicitly* control uncertainty by considering the effect of present and future control decisions on the trajectory uncertainty evolution, referred to as stochastic guidance or stochastic optimal control. Stochastic aerocapture guidance could directly incorporate an updated model of density variability, and can be tuned to quantitatively balance nominal performance and performance at the margin (e.g. 3σ values of a targeting parameter, where σ is standard deviation). This motivates development of a stochastic guidance architecture for aerocapture that is efficient enough

---

[12] Specifically, for bank angle modulated aerocapture

to feasibly run onboard an flight vehicle. Specifically, because the defining source of uncertainty for aerocapture is spatially-dependent variability in atmospheric density, stochastic aerocapture guidance should include a method of accurately modeling an uncertain atmosphere and updating this model based on noisy measurements. **The development of high-fidelity onboard models of uncertain atmosphere presented in Ch. 3 is directly motivated by the potential for incorporation of this model into an autonomous stochastic guidance framework for aerocapture.**

Onboard guidance often does include some function for updating the predicted density profile in-flight based on deceleration data. Typical approaches estimate either a physical parameter such as atmospheric scale height or a density scale factor that is multiplied with the a priori nominal profile [84, 85], possibly including a fading-memory filter applied to the estimated parameter. More recent approaches include machine learning [86, 87] and ensemble filtering [88]; see Ch. 6 of Ref. [63] for further discussion. What all of these approaches have in common is that they update the expected value of density, without explicitly modeling the random variability present in the system. Recent works have applied more advanced uncertainty propagation methods to aerocapture, including analytical polynomial chaos expansion and other spectral methods [89, 90], the Perron-Frobenius operator [91], and the stochastic Liouville equation [92]. A stochastic terminal point control problem is solved in [93] for an uncertain atmosphere and entry state. The aforementioned works are all limited, however, by only considering an exponential model of atmospheric density. Several approaches to stochastic numerical predictor-corrector guidance are proposed in [94] using the unscented transform for uncertainty propagation, and a similar stochastic retargeting method is applied to aerocapture in [95] using a Gaussian mixture model for uncertainty propagation. Other approaches focus on linearization and convexification techniques to obtain stochastic optimal solutions in a local neighborhood [96, 97, 98]. **See Sec. 7.3.1 for an extended discussion of potential future work leveraging the recent developments in the literature and the contributions of this dissertation to develop stochastic guidance for aerocapture.**

1.2.4 Co-Delivery

A number of interplanetary missions featuring a co-delivery architecture have been flown or proposed. The Galileo and Cassini-Huygens missions each successfully delivered an atmospheric probe along with a larger orbiter [99, 100]. The Deep Space 2 Mars Microprobes were delivered to entry by the Mars Polar Lander [101], and the Russian Mars 96 mission included two small landers plus two small penetrators all delivered by a larger orbiter [102]; unfortunately, these missions all ended in failure, and the NASA failures scrapped plans for later Mars Surveyor landers and accompanying Mars Micromissions [101]. The sample return missions of Genesis [103], Stardust [104], Hayabusa [105], Hayabusa-2 [106], and OSIRIS-REx[13] all successfully delivered sample return capsules to Earth entry from a hyperbolic carrier [107], as will the Earth Entry System component of the Mars Sample Return campaign [108]. Finally, the Pioneer Venus mission delivered four probes (one large and three identical small probes) to entry at Venus from a single spacecraft bus [109], flinging the small probes out toward their respective entry points by spinning up the main bus [110].

Despite these mission precedents, it is difficult to provide a "state-of-the-art overview" for interplanetary co-delivery concepts for the very reason that they have received little dedicated systematic study before this dissertation, but rather a series of independent mission design studies as needed. One exception is a 2013 study that demonstrates a unique method of co-delivery wherein two Phoenix-class landers enter the atmosphere together and then separate, one lander with a drag skirt and the other without [111]; however, this method requires separation between two flight vehicles during hypersonic flight, a high-risk event. Recent work for the Aeolus mission concept presents a design that co-delivers 20 probes to a global network on Mars from a single hyperbolic carrier, but assumes that the carrier maneuvers after each probe deployment [112]; this assumption is typical to previous studies of Mars network missions. One might expect the field of spacecraft formation flying to provide a set of useful tools for co-delivery trajectory design up until atmospheric

---

[13] NASA press release regarding successful return of OSIRIS-REx sample capsule on 9/24/23
[14] Pioneer Venus image: NASA/Paul Hudson

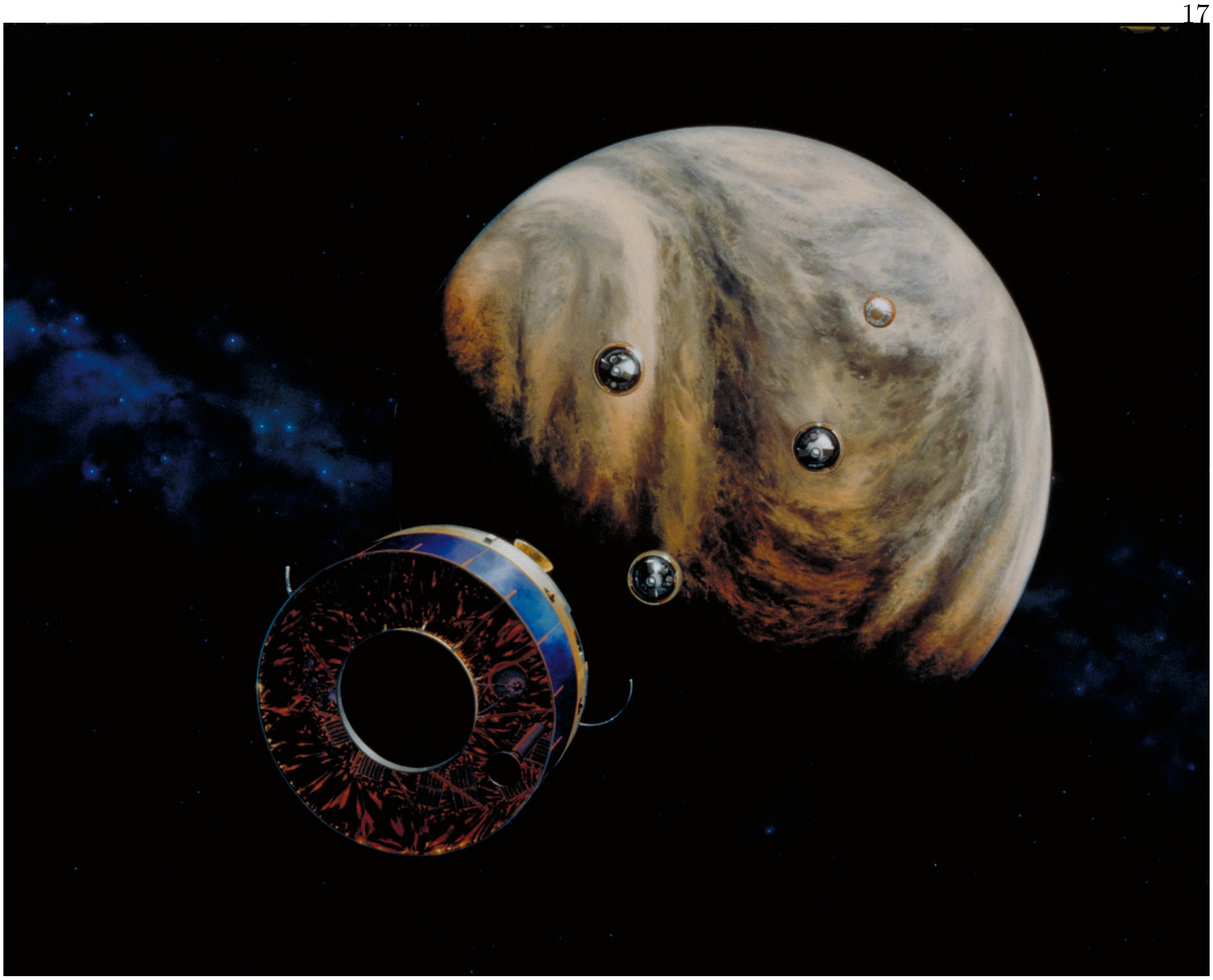


Figure 1.9: Artist's illustration of Pioneer Venus probes just after deployment from Multiprobe[14]

flight, but the existing literature is primarily concerned with motion about circular or near-circular elliptical orbits [113]. While some relative motion descriptions do accurately model motion about a highly-eccentric chief, they do not necessarily provide an intuitive representation in the way that the evolution of the relative position vector in the rotating orbit frame does for a circular chief [114]. **Motivated by the lack of dedicated study of co-delivery as its own concept, this dissertation provides a systematic investigation of co-delivery for missions that combine a probe and orbiter via aerocapture (Ch. 4) or co-deliver multiple probes (Ch. 5) and, furthermore, derives relative motion models that are more intuitive for motion about a highly-eccentric orbit or entry trajectory (Ch. 6).**

1.3 Summary of Contributions

The driving motivation for this thesis is to advance the state-of-the-art for aerocapture and co-delivery and to provide the first dedicated systematic study of co-delivery as a mission architecture; chapters 2 – 6 each describe one contribution toward this overarching aim. In Ch. 2, an efficient guidance algorithm is developed for a proposed smallsat aerocapture technology demonstration at Earth, and through high-fidelity simulation is shown to equal the performance of the state-of-the-art NPC solution while reducing computational demand. Because a defining characteristic of aerocapture is the requirement to mitigate spatially-dependent uncertainty in atmospheric density, in Ch. 3 multiple models are investigated and quantitatively compared for their ability to recreate high-fidelity atmosphere models while remaining compact enough for onboard use. A methodology is introduced for updating one of these onboard models according to incoming noisy estimates of density. A method is then demonstrated for incorporating this model of environmental uncertainty into approximate analytical predictions of state uncertainty. This model of uncertainty, combined with a predictive model for a probabilistic future state, are motivated in part by the potential for constructing a stochastic guidance scheme for aerocapture using these component parts. This concept is beyond the scope of this dissertation but is addressed in an extended future work section.

Having described contributions related to single-vehicle aerocapture, the dissertation transitions to studies of co-delivery. In Ch. 4, a novel concept is defined for co-delivering a probe and orbiter by using aerocapture for orbit insertion and designing the two vehicles to require the same entry conditions. The feasibility of this concept is quantitatively assessed across a large trade space for multiple planetary destinations, and a representative scenario is simulated in more detail including closed-loop guidance implementation. In Ch. 5, the more common idea of co-delivering a network of probes to the Martian surface also receives systematic treatment, and a linearized targeting method is developed for maneuver design for regional networks. Motivated by these co-delivery concepts, expressions for relative motion in the velocity frame are derived in Ch. 6 for both exoatmospheric and hypersonic flight. This is shown to be a more intuitive way of understanding motion about a highly-eccentric chief than the traditional orbit frame solutions. The dissertation concludes by outlining a case study evaluating the utility of a fly-ahead probe for aerocapture at Uranus. While not a contribution of its own, this example serves to illustrates how the tools developed by this thesis can be brought together to address a relevant problem.

1.4 List of Related Peer-Reviewed Publications

- *(submitted Jul. 2023)* **Albert, S.W.**, Doostan, A., and Schaub, H., "Dimensionality Reduction for Onboard Modeling of Spatially-Varying Atmospheres," AIAA Journal of Spacecraft and Rockets

- **Albert, S.W.,** Burnett, E., Schaub, H., Burkhart, P.D., and Austin, A., "Energy Reference Guidance for Drag-Modulated Aerocapture," Advances in Space Research. https://doi.org/10.1016/j.asr.2023.09.034 *(in press)*

- **Albert, S.W.** and Schaub, H., "Co-Delivery of a Martian Probe Network," AIAA Journal of Spacecraft and Rockets. https://doi.org/10.2514/1.A35560 (*in press*)

- **Albert, S.W.** and Schaub, H., "Relative Motion in the Velocity Frame for Atmospheric Entry Trajectories," AIAA Journal of Spacecraft and Rockets, Vol. 60, No. 5 (2023), pp. 1614-1624. https://doi.org/10.2514/1.A35753

- **Albert, S.W.**, Schaub, H., and Braun, R.D., "Flight Mechanics Feasibility Assessment for Co-Delivery of Direct-Entry Probe and Aerocapture Orbiter," AIAA Journal of Spacecraft and Rockets, Vol. 59, No. 1 (2022), pp. 19-32. https://doi.org/10.2514/1.A34953

Chapter 2

Efficient Onboard Guidance for Drag-Modulated Aerocapture

2.1 Introduction

Aerocapture is a technology that could enable shorter transit times and lower total expended mass for orbit insertion for a variety of interplanetary missions [17, 18, 19]. To perform aerocapture, the spacecraft executes a single pass through the atmosphere of a planet or moon, dissipating enough energy to reach the desired target orbit upon exit from the atmosphere. During the subsequent pass through apoapsis the spacecraft performs a propulsive maneuver to raise periapsis out of the atmosphere, and performs other correction maneuvers as necessary. For missions to the ice giants Uranus and Neptune, aerocapture can potentially reduce cruise duration by 2–5 years while reducing mass for orbit insertion by some 40% [25, 32]. Aerocapture also offers significant benefit to small satellites (SmallSats) launched via rideshare with a primary mission, enabling orbit insertion despite the lack of high-ΔV systems at SmallSat scale and reducing sensitivity to primary mission trajectory design [26, 38]. Although it has been proposed for a number of missions [22, 23, 24], aerocapture has never been flown [19].

Variability in the spacecraft state at atmospheric entry, atmospheric density, aerodynamic properties of the vehicle, and other day-of-flight dispersions require a spacecraft performing aerocapture to autonomously control its trajectory through the atmosphere. During this hypersonic flight phase, control is achieved by judiciously adjusting the aerodynamic forces acting on the vehicle, and control approaches thus fall into two main categories: lift modulation and drag modulation. Lift modulation involves changing the attitude of the vehicle to reorient the lift vector, typically either by banking about a fixed trim angle (bank angle modulation) or by independently modulating angle of attack and side-slip angle (direct force control) [115, 116]. Note that direct force control also involves changes in the drag and side force components, but the primary control authority is obtained by modulating lift, and thus this technique is categorized with lift modulation for the purposes of this discussion. Lift modulation, particularly bank angle modulation, is well-studied in the literature [73, 65, 116, 54], and has relevant flight heritage from guided hypersonic entry of blunt-body aeroshells including the Apollo [40], Orion [41], Mars Science Laboratory [42], and Mars 2020 [43] missions, all of which relied on some form of closed-loop lift modulation.

Recent work has studied drag modulation as a potentially simpler method of achieving control for aerocapture [63, 85]. Typically, a drag-modulated vehicle is assumed to be axisymmetric and to fly at zero angle of attack, thus generating no lift. The trajectory is influenced by adjusting the ratio of mass to effective drag area, termed ballistic coefficient; when this ratio is low, the vehicle rapidly dissipates energy through drag, and vice-versa. This can take a variety of forms, including devices that achieve continuously-variable drag [117], release of a trailing inflatable drag device [118], and jettison of one or more rigid drag skirts [119]. Single-event jettison, defined here as a single discrete change in ballistic coefficient caused by the jettison of a rigid drag skirt, is the control architecture that will be the focus of this work. This represents a limiting case, because after jettison the vehicle flies passively for the remainder of atmospheric flight and the vehicle lacks any out-of-plane control. However, for a sufficiently large change in ballistic coefficient, single-event jettison can achieve a total control authority comparable to lift modulation with heritage blunt-body aeroshells [16]. Compared to lift modulation, single-event jettison drag modulation may be less complex because the vehicle can be passively spin-stabilized, rather than requiring a high-rate reaction control system that must operate during atmospheric flight [120]. Moreover, ballast masses are not required to create an offset center of gravity, as is typically the case for lift-modulated axisymmetric vehicles [42].

A limited number of guidance algorithms for single-event jettison drag-modulated aerocapture exist in the literature. The simplest solution in terms of computational expense is to trigger jettison when the instantaneous value of a measured state exceeds some threshold, such as a velocity trigger [121]. To reduce error caused by noisy measurements, the observed state can be filtered and jettison can be triggered based on some polynomial function of the state. For example, the algorithm implemented in [122] triggers jettison when the total integrated ΔV exceeds a polynomial function of the filtered instantaneous or maximum sensed acceleration. The deceleration curve fit algorithm used for Mars Pathfinder parachute deploy [123] and applied to drag-modulated aerocapture in [124] also triggers based on deceleration measurements. In this case, two measurements are taken a set time apart, and a pre-computed curve fit between the second deceleration measurement and time until jettison is consulted to set a jettison timer. All of these approaches require only minimal onboard computation and memory, but each is also shown to have poor performance when relevant uncertainties are applied. The predictive trigger approach applied in [125] is more computationally-intensive; in this case, the energy of the spacecraft at atmospheric exit is predicted by numerically propagating the equations of motion, and jettison is commanded when the predicted final energy is less than or equal to the desired final energy. Machine learning-based guidance schemes have been successfully developed for entry and aerobraking problems [126, 127, 128, 129], but have yet to be applied to single-event jettison drag-modulated aerocapture other than for the purpose of atmospheric estimation [86, 87].

While the algorithms summarized above share the benefit of relatively low onboard computational burden, the current state of the art guidance for drag-modulated aerocapture is the numerical predictor-corrector (NPC) approach [85]. This algorithm also predicts the final state by numerically propagating the equations of motion, then takes the additional step of making a correction to the jettison time. This two-step procedure is applied iteratively, such that the algorithm should converge to an optimal jettison time each guidance call. NPC has two key differences with the predictive trigger:

(1) The NPC solves for jettison time rather than directly commanding jettison, so the release timing can operate at significantly higher resolution; this is under the assumption that a

simple controller releases the drag skirt when the jettison time is reached, operating at a higher rate than the guidance algorithm itself.

(2) The NPC is significantly more computationally expensive than the predictive trigger because multiple numerical propagations may be required in each step.

In summary, NPC guidance is significantly more accurate in the presence of uncertainties than the other algorithms discussed here [85, 121, 124], but is also much more computationally demanding. A more detailed description of the NPC algorithm is given in Sec. 2.3.

This work investigates a guidance algorithm for single-event jettison drag-modulated aerocapture, with the goal of achieving the same level of accuracy as the NPC but with significantly less computational demand. The reference mission for this study is an Earth flight test of aerocapture with a SmallSat using a rigid deployable drag skirt; that is, the drag skirt is stowed during launch and deployed during cruise, but does not change its shape during atmospheric flight. Assumptions regarding modeling of dynamics and uncertainties are discussed, and key physical parameters defined. The baseline NPC algorithm is described in detail, including an approach to the correction step that improves computational efficiency, and targeting results under relevant uncertainties are estimated. The proposed algorithm is also described, and compared directly with NPC. A parameter study is presented that gives insight into optimal tuning and tradeoffs between memory and performance for the proposed algorithm. Finally, results are discussed along with a number of avenues for potential future work.

2.2 Methodology

2.2.1 Reference Mission

Researchers from the NASA Jet Propulsion Laboratory (JPL), NASA Ames, and the University of Colorado Boulder have been studying drag-modulated aerocapture for small satellites [38, 130], including concepts for an Earth flight test of the technology [124]. This idea is supported by the 2022 Strategic Framework[1] released by the NASA Space Technology Mission Directorate, which states that "an Earth-based aerocapture demonstration will reduce perceived risk and mature guidance and control methods" for aerocapture at other planetary destinations. Motivated by these developments, single-event jettison drag-modulated aerocapture at Earth by a SmallSat is the reference mission considered in this work. As summarized in Fig. 2.1, the spacecraft is launched into a geosynchronous transfer orbit, then performs a maneuver to lower periapsis into the atmosphere, achieving the desired entry state. Based on the JPL reference mission, the spacecraft targets an apoapsis of 5000 km and performs a maneuver at the next pass through apoapsis to raise periapsis to 200 km. Autonomously raising periapsis during the first pass through apoapsis in order to achieve a near-term stable orbit is a significant component of successful aerocapture; however, specific consideration of the on-orbit maneuver guidance and control is beyond the scope of this study.

The drag skirt in this study is modeled as the Adaptable Deployable Entry and Placement Technology (ADEPT), an umbrella-like deployable structure for entry probes currently under development at NASA Ames [131]. During launch, ADEPT is in the retracted configuration, significantly reducing fairing volume required for the spacecraft and enabling stowage in the standard ESPA envelope [131, 132]. The drag skirt is fully deployed between separation from the launch vehicle and atmospheric entry, and remains rigidly deployed until it is jettisoned by the guidance algorithm.

The initial epoch for simulation of this mission is defined as 10 minutes before nominal atmospheric entry, which is the time of the final orbit determination (OD) update to the spacecraft from ground control. From this point onward, the navigated states are based on propagation with only IMU data. The nominal entry state, defined at the atmospheric interface altitude of 125 km, has a planet-relative velocity u of 9.9 km/s and flight-path angle γ of $-4.6\,^\circ$, where flight-path angle is the angle between the planet-relative velocity vector $\boldsymbol{u}$ and the local horizontal plane. The nominal entry point is at a geocentric latitude ϕ of -7.4° and longitude θ of 14.8° with a heading of 118.9°, where heading angle ψ is defined as the angle between the horizontal projection of the

---

[1] https://techport.nasa.gov/framework

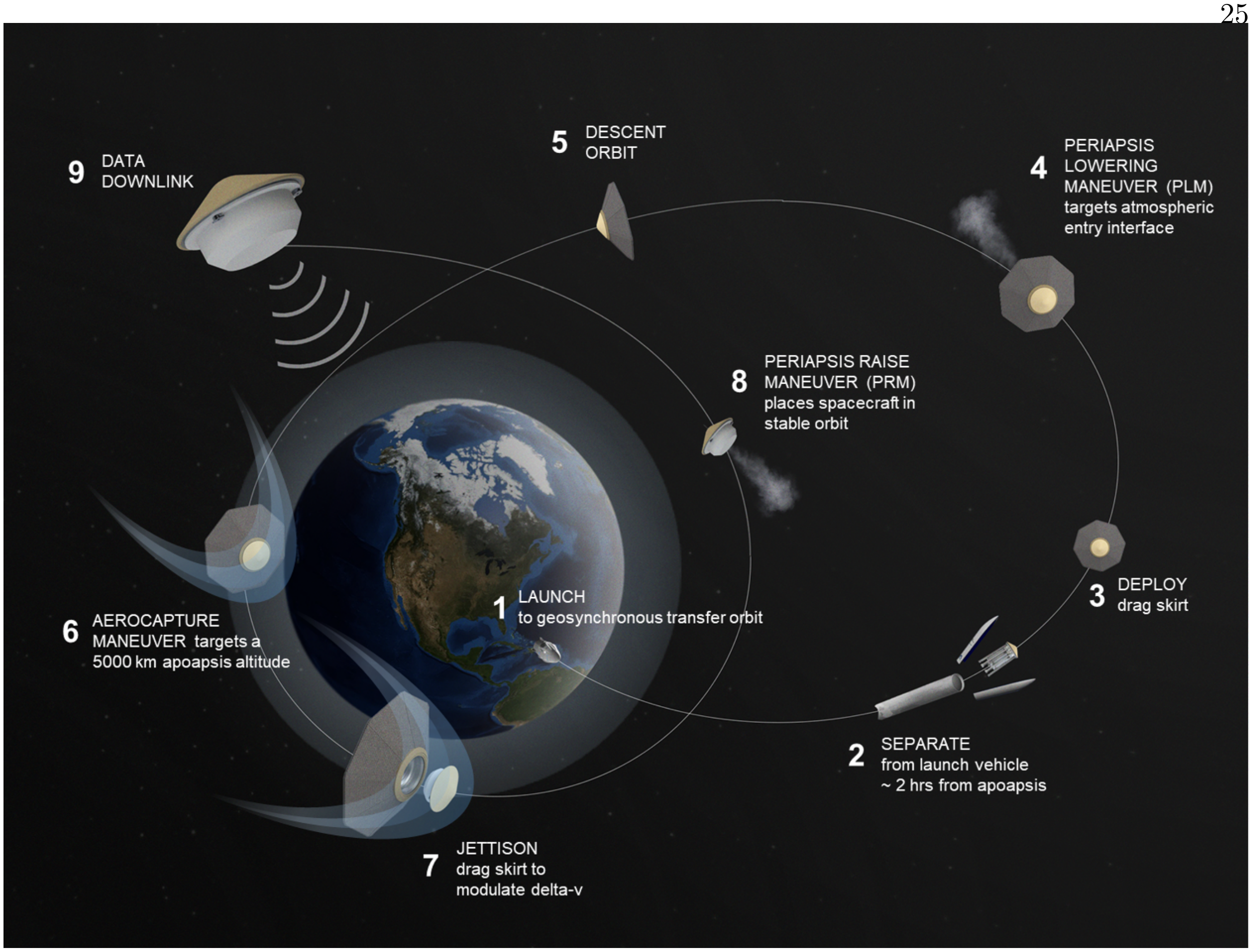


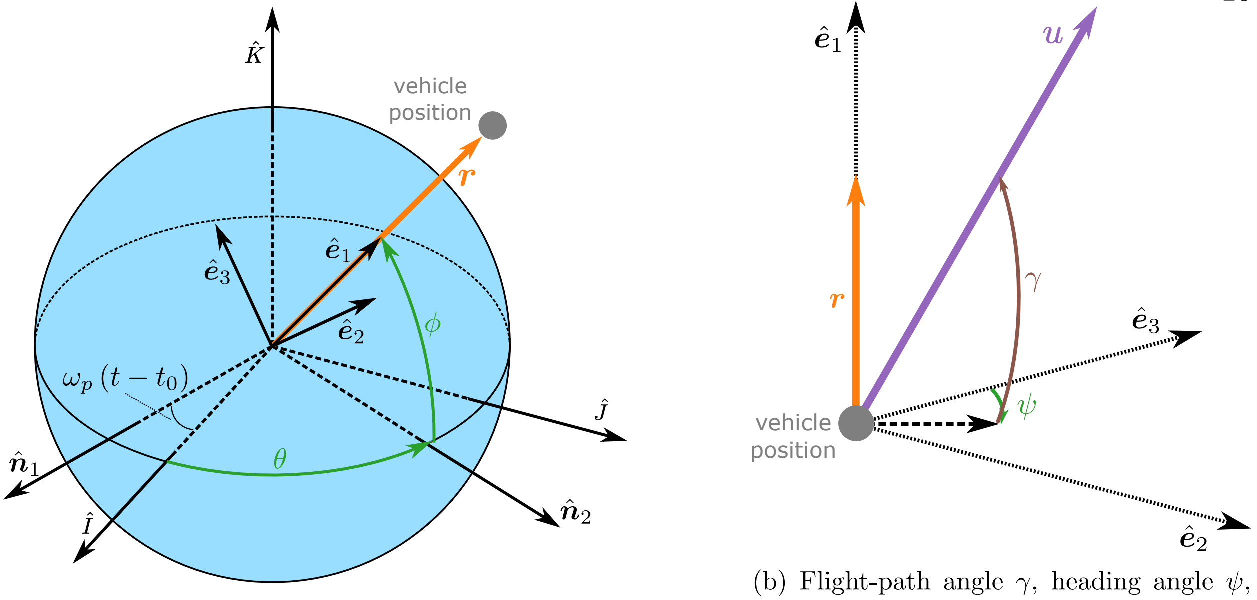
Figure 2.1: Aerocapture Earth Flight Test

velocity vector and a due-North vector in that same plane. These definitions are illustrated in Fig. 2.8, where the unit vector bases $\{\hat{n}_1, \hat{n}_2, \hat{K}\}$, $\{\hat{I}, \hat{J}, \hat{K}\}$, and $\{\hat{e}_1, \hat{e}_2, \hat{e}_3\}$ define inertial, planet-fixed, and position frames, respectively. The vector from the central body to the vehicle is denoted $\boldsymbol{r}$, and $\hat{\boldsymbol{r}} = \boldsymbol{r}/r$ is the associated unit vector.

2.2.2 Problem Dynamics

2.2.2.1 Simulation

In this work, the performance of each guidance algorithm is quantified through testing in a high-fidelity simulation environment implemented in the Dynamics Simulator for Entry, Descent, and Surface Landing (DSENDS) software developed by the DARTS lab at NASA JPL [133]. The



(a) Latitude ϕ, longitude θ, and position vector $\boldsymbol{r}$

(b) Flight-path angle γ, heading angle ψ, and flow velocity vector $\boldsymbol{u}$

Figure 2.2: Frame definitions

gravity model includes point-mass and spherical harmonics of degree and order 8 for the Earth, as well as point-mass gravity from the Moon and the Sun. Atmospheric density is modeled using the Earth Global Reference Atmospheric Model (EarthGRAM) 2010 [134], such that the value of density depends on 3D position and time.

The vehicle shape is a 60-degree sphere-cone both with and without the drag skirt, such that the drag skirt extends the conical section at the same angle. The aerodynamics model used in simulation includes drag and aerodynamic moments. No lift is modeled; the vehicle is axisymmetric and passively-stabilized, such that the axis of symmetry remains approximately aligned with the freestream velocity vector. Thus, while the simulation is 6 degree-of-freedom, oscillations in vehicle attitude are small and have only a minor effect on the vehicle trajectory.

2.2.2.2 Predictor Model

Both guidance algorithms presented in this work rely on numerical propagation of the relevant equations of motion to predict trajectories onboard. These equations constitute a simplified version of the dynamics modeled in the full "truth" simulation. Specifically, the modeled forces include point-mass gravity, J_2 oblateness, and drag, resulting in the following equation for inertial acceleration [135]:

$$\ddot{\boldsymbol{r}} = -\frac{\mu}{r^2}\hat{\boldsymbol{r}} - \frac{3\mu J_2 R^2}{2r^4}\left(\left(1 - 5\left(\hat{\boldsymbol{r}} \cdot \hat{\boldsymbol{K}}\right)^2\right)\hat{\boldsymbol{r}} + 2\left(\hat{\boldsymbol{r}} \cdot \hat{\boldsymbol{K}}\right)\hat{\boldsymbol{K}}\right) - \frac{\rho u^2}{2\beta}\hat{\boldsymbol{u}} \tag{2.1}$$

where $\boldsymbol{r}$ is the vector from the central body to the vehicle, μ is the gravitational parameter, J_2 is the oblateness coefficient, ρ is atmospheric density, R is the planetary equatorial radius, $\hat{\boldsymbol{K}}$ is the polar axis unit vector, and $\beta = m/(C_D A)$ is the ballistic coefficient of the vehicle. The quantities m, C_D, and A are the mass, drag coefficient, and reference area of the vehicle, respectively. The quantity $\boldsymbol{u}$ is the flow velocity, or the velocity of the spacecraft with respect to the planetary atmosphere, which is assumed to be rotating with the planet with angular velocity $\boldsymbol{\omega}_p$ between initial time t_0 and current time t,

$$\boldsymbol{u} = \dot{\boldsymbol{r}} - \boldsymbol{\omega}_p \times \boldsymbol{r}, \tag{2.2}$$

where $\dot{\boldsymbol{r}}$ is the inertial velocity vector. The predictor models density by linearly interpolating from a table of density vs. altitude output by EarthGRAM that represents a nominal atmosphere profile. Note that the predictor thus assumes the same density is experienced in the descending and ascending portions of the aerocapture trajectory, other than as modified by the atmospheric scale factor as discussed later, whereas the DSENDS simulation incorporates dependence of density on latitude and longitude. A table of C_D vs. dynamic pressure is similarly used by the predictor to compute β. However, note that this latter step is likely higher-fidelity than necessary because C_D changes little in the relevant flight regime for this scenario; constant C_D would be a reasonable approximation. The values of μ, J_2, and R used in both the predictor and simulation are provided in Table 2.1. The average ballistic coefficient for each phase is also listed, where β_1 and β_2 are the values pre- and post-jettison, respectively. The predictor uses fourth-order Runge-Kutta integration to numerically propagate the equations of motion, with a fixed time step of 0.125 s.

Table 2.1: Nominal Simulation Parameters

Parameter	Value
μ	$3.9860 \times 10^5\,\mathrm{km^3/s^2}$
ω_p	$7.2921 \times 10^{-5}\,\mathrm{rad/s}$
J_2	0.0010826
R	6378.1 km
β_1	$32\,\mathrm{kg/m^2}$
β_2	$137\,\mathrm{kg/m^2}$

2.2.3 Models of Uncertainty

The variability of atmospheric density is modeled by EarthGRAM, which has a built-in Monte Carlo framework for generating realistic dispersions [134]. The vehicle aerodynamics are dispersed based on experience with blunt-body aeroshells [136], resulting in a standard deviation of about $\sigma = 0.015$ for C_D near peak dynamic pressure, where σ is standard deviation and the nominal value is 1.38. The entry state is dispersed according to a navigation assessment performed at JPL that was then scaled to match the project requirement of entry flight-path angle delivery error with a standard deviation value of $3\sigma = 0.2^\circ$ at the atmospheric interface altitude of 125 km. The time required for the drag skirt to fully separate from the capsule is assumed to be uniformly dispersed along a range from 0.05 s to 0.14 s. The vehicle mass and area are not dispersed, nor are gravitational parameters.

Importantly, the predictor does not operate on the true state of the spacecraft. Noisy measurements from an inertial measurement unit (IMU) are modeled and fed into a navigation filter, and the predictor operates on these filtered state estimates. The navigation filter uses the same dynamics model as the predictor, Eq. (2.1).

2.3 Numerical Predictor-Corrector Guidance

NPC guidance is treated as the baseline solution in this work due to both its state-of-the-art targeting performance and its previous application as part of the JPL SmallSat aerocapture project [85, 38, 137]. The implementation discussed here is similar to that presented in [85], but with a more computationally-efficient correction method. The algorithm is summarized by Fig. 2.3 and outlined in detail in this section; performance results are given in Sec. 2.5.

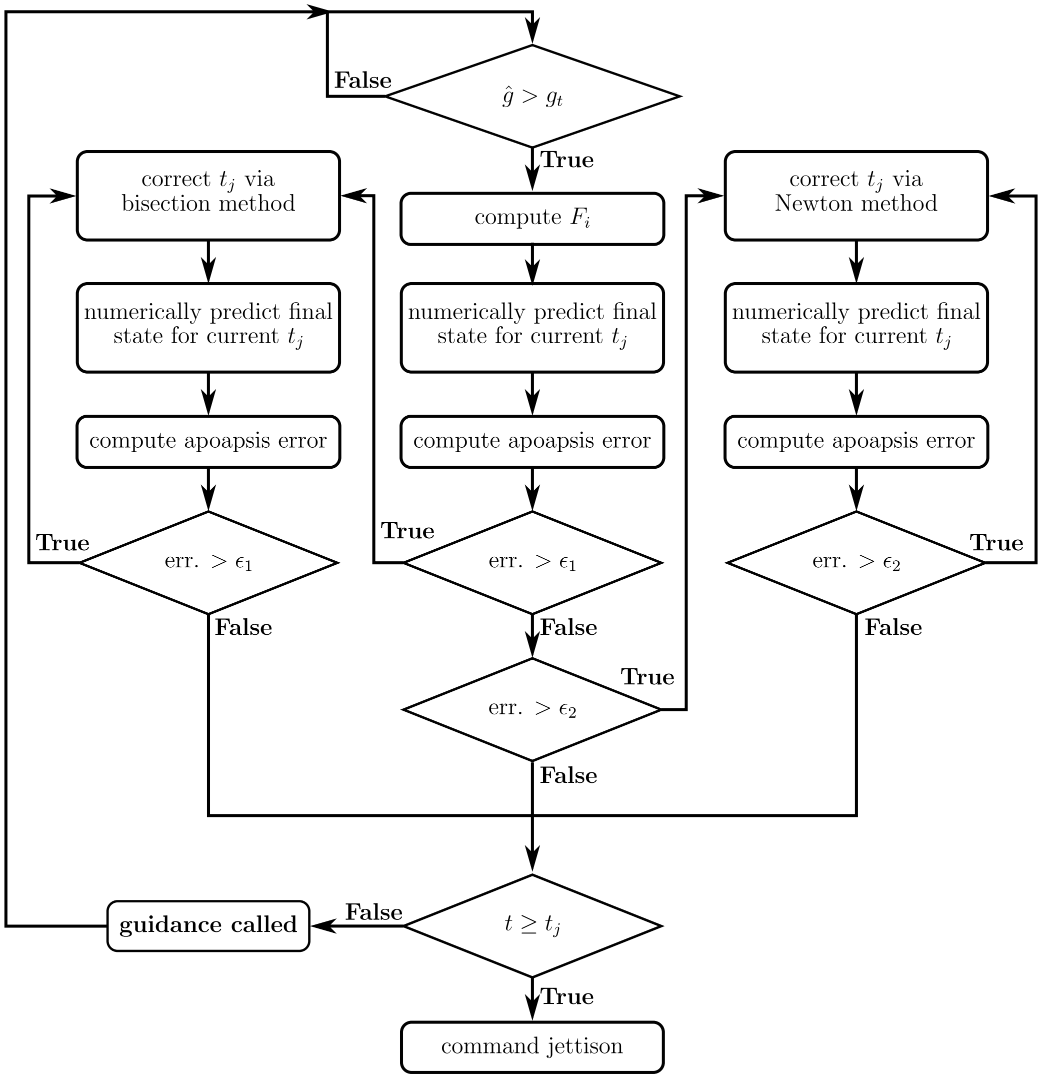


Figure 2.3: NPC Guidance Diagram

IMU measurements are used to generate an estimate of sensed acceleration (or g-load), $\hat{g}$, and when this exceeds some threshold value g_t the guidance routine is initiated. In the subsequent step, nominal density at the navigation-estimated altitude is used with the navigation-estimated state to compute an estimate of the dynamic pressure:

$$q_{\text{est}} = \frac{1}{2}\rho_{\text{nom}}(r)u^2 \tag{2.3}$$

where the estimated dynamic pressure is used to obtain an estimate of the drag coefficient via interpolation of stored data of C_D vs. q. Next, the density is estimated from a re-arranged expression for acceleration due to aerodynamic drag (which equals $\hat{g}$ since the vehicle is assumed to have no lift):

$$\rho_{\text{est}} = 2\frac{m_1 \hat{g}}{A_1 C_{D,1} u^2} = 2\frac{\beta_1 \hat{g}}{u^2} \tag{2.4}$$

where m_1, A_1, $C_{D,1}$, and β_1 are the pre-jettison values of those variables. The density estimate is used to compute the i^{th} density scale factor F_i:

$$F_i = \rho_{\text{est}}(t_i)/\rho_{\text{nom}}(r(t_i)) \tag{2.5}$$

This value is then filtered via a low-pass filter:

$$\overline{F}_i = (1-k)\overline{F}_{i-1} + kF_i \tag{2.6}$$

As the gain k is decreased, this filter will increasingly reject small disturbances. Sensible values of k depend on the frequency of density scale factor measurement updates. Alternatively, the density scale factor could also be filtered with a moving average filter, detailed below:

$$\overline{F}_n = \frac{1}{n}\sum_{i=1}^{n} F_i \tag{2.7}$$

where n is a memory parameter, and again the chosen value of n should be tuned based on the density scale factor update frequency. In this work, the low-pass filter is implemented with $k = 0.05$ for a guidance update rate of 8 Hz. The nominal density profile is then re-scaled by F_i for all subsequent numerical propagations within that guidance call, as follows:

$$\rho_{\text{pred}}(r) = \overline{F}_i \rho_{\text{nom}}(r) \tag{2.8}$$

This form of density re-scaling significantly improves targeting results compared to ignoring atmospheric estimation altogether [84], but is limited to linearly shifting the entire profile and thus fails to capture the more complex atmospheric perturbations that occur in reality. Other methods, such as exponentially correlating the scale factor, ensemble correlation filtering [88], machine learning [86, 87], or modeling density as a Gaussian random field [138] may improve the atmospheric estimation component of NPC guidance.

Once the density scale factor is computed, the navigation-estimated state is numerically propagated until the altitude of the spacecraft either exceeds the atmospheric interface altitude or decreases below some minimum. This prediction uses the jettison time computed by the previous guidance call or, in the case of the first guidance call, a pre-defined initial guess, set to 700 s in this case. The radius of apoapsis is then computed from the final state using Keplerian relations, and error is computed as the difference between the predicted and desired apoapsis radii. In the case of an escape trajectory, apoapsis radius is poorly-defined and the error is set equal to positive infinity. In the case of an impact trajectory, in which the spacecraft reaches the surface instead of exiting the atmosphere, the Keplerian apoapsis is computed from the final state as normal; the value will badly undershoot the target and thus the guidance algorithm behaves as expected. As an aside, note that for certain, more extreme mission scenarios an edge case is possible in which the vehicle reaches the minimum altitude bound while still hyperbolic in terms of orbital energy, and care should be taken to correctly categorize these cases as undershoots, despite their hyperbolic Keplerian state.

The error magnitude is then compared against two tolerance values, ϵ_1 and ϵ_2, where $\epsilon_1 > \epsilon_2$. The purpose of the dual tolerances is to direct the algorithm to an appropriate root-finding subroutine for the correction step. If the error exceeds both tolerances, bisection method is selected; if the error is between the two tolerance values, Newton's method is selected; finally, if the error is below both tolerances, no updates to jettison time are required and the algorithm skips the correction step entirely. In this work, tolerances were defined as $\epsilon_1 = 500\,\text{km}$ and $\epsilon_2 = 25\,\text{km}$, selected based on a trial-and-error process in order to achieve a good balance between accuracy and speed. These tolerances would need adjustment for a significantly different apoapsis target or central body.

The bisection method subroutine begins with lower and upper bounds on the optimal jettison time, selected *a priori* without any dependence on the solution from the previous guidance call. These values should span the duration of the longest atmosphere pass that is expected based on dispersions and are strongly scenario-dependent. For this work, bounds of 600 and 900 s are selected, noting that $t = 0$ is defined as 10 minutes prior to atmospheric entry. The jettison time is then set equal to the midpoint of these bounds, and the predictor numerically propagates to the final state and computes an apoapsis error. If the magnitude of this error is below the tolerance ϵ_1, the algorithm exits the bisection subroutine with a converged solution. Otherwise, the bounds on jettison time are updated based on the sign of the error. In an overshoot case with positive error, the upper bound is set equal to the current value of the jettison time; in the undershoot case, the lower bound is similarly updated. The subroutine then repeats, using the updated midpoint as the new jettison time, and continues until either the error magnitude is below the tolerance ϵ_1 or a maximum number of iterations is reached. The subroutine also includes logic to recognize cases in which the jettison time converges against the original upper or lower bound. This can occur in cases where, due to dispersions, the target state is unreachable and the best-case scenario is to jettison as early or as late as possible.

Newton's method begins by perturbing an initial guess for the jettison time by some predetermined amount; in this work, a perturbation of $\delta t_j = 0.5\,\mathrm{s}$ is used and the initial guess is set to $700\,\mathrm{s}$. For numerical consistency, the perturbation should be a multiple of the time step used by the predictor for fixed-time step integration. The apoapsis radius corresponding to this perturbed jettison time is then numerically predicted; note that this propagation is not explicitly represented in Fig. 2.3. The derivative of the objective function, in this case the slope of apoapsis radius as a function of jettison time $r_a'(t_j)$, is then approximated via first-order finite differencing as shown in Eq. (2.9). The updated jettison time is then computed via Eq. (2.10), which finds the x-intercept of the tangent line. The apoapsis radius resulting from the updated jettison time $t_{j,i+1}$ is numerically predicted, and the error is computed and checked against the tolerance ϵ_2. For a sufficiently accurate linearization and a nonzero slope of $r_a(t_j)$, the error should decrease each step. The subroutine repeats until either converging within the tolerance ϵ_2 or reaching a maximum number of iterations.

$$r'_a(t_j) \approx \frac{r_a(t_j + \delta t_j) - r_a(t_j)}{\delta t_j} \tag{2.9}$$

$$t_{j,i+1} = t_{j,i} - \frac{r_a(t_j)}{r'_a(t_j)} \tag{2.10}$$

The advantage of combining these two root-finding methods in a single guidance algorithm is that bisection method is robust but relatively slow, whereas Newton's method tends to converge more efficiently but requires a sufficiently-accurate initial guess. In particular, for a more typical aerocapture scenario in which the initial orbit is hyperbolic, escape cases that are still hyperbolic after exiting the atmosphere can be frequently encountered and may exist near the optimal solution for a high-energy target orbit. In these cases apoapsis radius is poorly-defined and the elliptical Keplerian equations would yield a negative value. Because the error no longer varies smoothly, the gradient is poorly-behaved and Newton's method fails to accurately converge to the solution. Bisection method, on the other hand, can handle errors of $\pm\infty$ and thus behaves as desired when escape cases are simply assigned an error of ∞. Once converged to a solution, however, the optimal jettison time (as predicted based on the navigation-estimated states) tends to require only small corrections in subsequent guidance calls. Because the initial guess is good, Newton's method can more efficiently compute these minor adjustments as long as the perturbation step is tuned appropriately. Note that a possible alternative implementation of the NPC would, during a single guidance update, call the Newton's method subroutine after the bisection method reduces the error to be between the two tolerance values. However, bisection method is typically only used in either edge cases where the solution is unreachable or at times far from the optimal jettison time, and therefore this modification would add computational expense with a negligible impact on performance.

The output of this prediction-correction loop is a jettison time t_j. In Fig. 2.3 the logic to command jettison once this time is reached or exceeded is portrayed as part of the guidance algorithm. However, note that this command is not necessarily limited to the update frequency of the guidance algorithm. Instead, t_j can be output by the guidance and a separate jettison controller can check the current time and command drag skirt jettison when t_j is reached. This controller is simple and can run at a higher rate than the guidance algorithm, enabling higher-resolution commanding of jettison and a corresponding improvement in targeting accuracy. Finally, the "guidance called" delay block in Fig. 2.3 reflects the fact that this process is called at a fixed rate rather than constantly iterating.

A significant drawback of the NPC guidance algorithm is that the number of iterations required to converge is indeterminate. That is, while an upper limit on the number of iterations can be enforced, there is no guarantee on the resulting error magnitude once this limit is reached. Each guidance call requires a minimum of one numerical propagation, used to determine whether or not the current t_j results in apoapsis error within the tolerances. The bisection subroutine requires one additional propagation per iteration, and Newton's method, while more efficient, requires two propagations per iteration (one perturbed, one corrected). The end result is that the NPC is not only computationally expensive due to the requirement of onboard propagation, but the number of operations required for convergence is in general **unknown**. In practice the number of propagations required for convergence can be approximately bounded through analysis with expected dispersions, as shown in Sec. 2.5, but the lack of a theoretical guarantee can make validation of the NPC approach difficult.

2.4 Energy Reference Guidance

The energy reference guidance (ERG) algorithm proposed in this work[2] seeks to achieve comparable performance to the NPC while reducing computational requirements. The algorithm is summarized in Fig. 2.4 and outlined in detail in this section, with results provided in Sec. 2.5. ERG is divided into two phases: a pre-compute phase that is executed after the final OD update to the spacecraft is received, and an algorithm that is executed each time guidance is called during the atmospheric flight phase.

During the pre-compute phase, a smoothly-varying family of reference trajectories is gener-

---

[2] ERG is equivalent to the simplified form of the QIC algorithm proposed in [139].

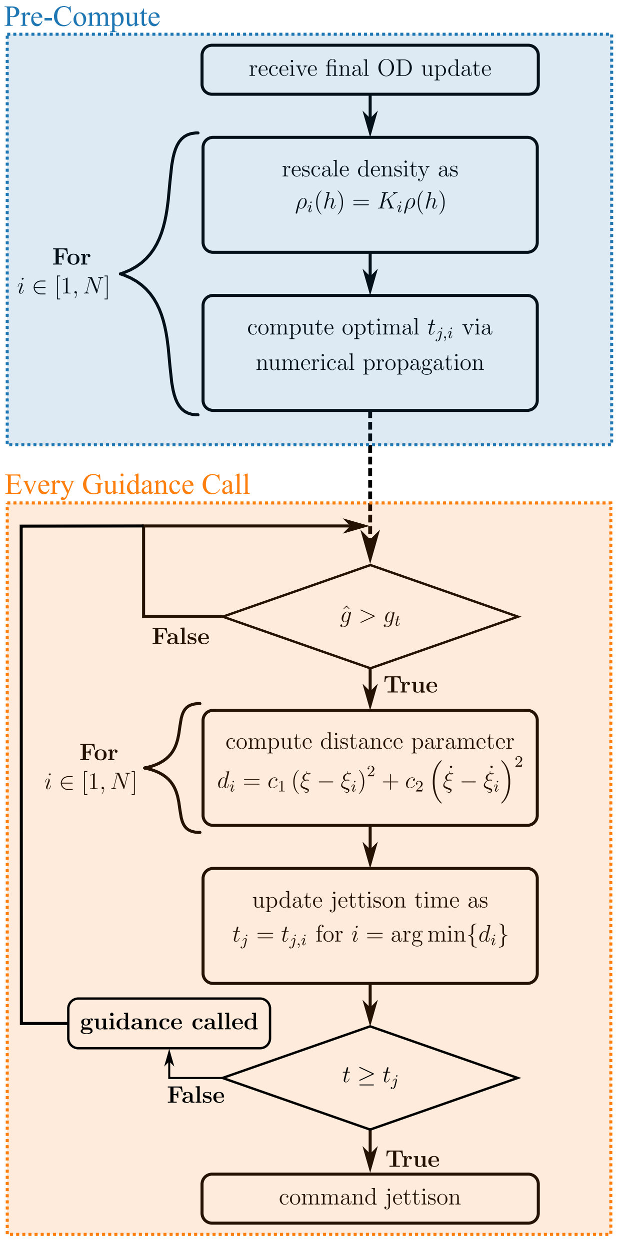


Figure 2.4: Energy Reference Guidance Diagram

ated and stored for later use. The ith reference trajectory is computed by linearly re-scaling the nominal density profile by some factor K_i, then solving for the optimal jettison time $t_{j,i}$ through an iterative prediction-correction procedure. This jettison time optimization is equivalent to the Newton's method subroutine from the NPC algorithm, and similarly relies on numerical propagation from the navigation-estimated states. Figure 2.5 shows an example set of reference trajectories,

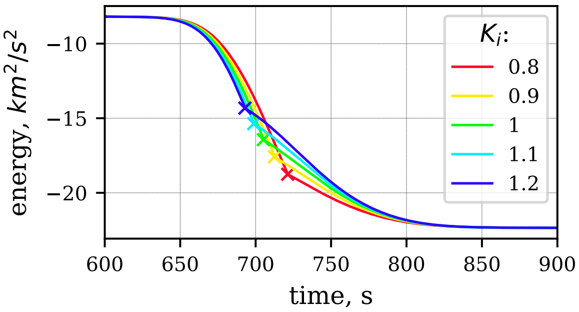


Figure 2.5: Orbital energy vs. time for family of reference trajectories, where X marks optimal jettison time

where the trajectories with earlier optimal jettison times correspond to denser atmospheres (larger K_i values).

In this work a range of $K_i \in [0.8, 1.2]$ is used based on trial and error; this range depends on the expected dispersions, atmospheric and otherwise, and is pre-defined on the ground. The smallest and largest K_i values correspond to the worst-expected overshoot and undershoot cases, respectively, based on both expected dispersions (aleatory uncertainty) and a potential lack of data at other planetary destinations (epistimic uncertainty), and can be conservative. The tradeoff for conservatism in these values is an incremental increase in memory and CPU requirements, but this has marginal effect on the CPU demand during the atmospheric flight phase. The number of K_i values, N, and the resolution at which reference trajectory data are saved are treated as tuning parameters and discussed in Sec. 2.5. It is important to note that this method of linearly re-scaling density is not meant to be a good model of how density dispersions behave in real atmospheres, in which dispersions vary with position and time. Additionally, note that it would be possible to implement ERG with other methods of modifying density to generate a family of reference trajectories, such as varying atmospheric scale height of an exponential model. In Subsection 2.5.2 ERG is tested against the higher-fidelity density dispersions provided by GRAM as described in Subsection 2.2.3.

During the atmospheric flight phase, guidance is called periodically and is active while sensed deceleration is above a threshold value, just like in the NPC guidance. Once a guidance call is initiated, the algorithm determines the reference trajectory that most closely matches the vehicle trajectory at the current time. This is accomplished via a heuristic distance parameter d:

$$d_i = c_1 \left(\xi - \xi_i\right)^2 + c_2 \left(\dot{\xi} - \dot{\xi}_i\right)^2 \tag{2.11}$$

where ξ and $\dot{\xi}$ are the energy and energy rate computed from the current navigation-estimated state, respectively, ξ_i and $\dot{\xi}_i$ are the energy and energy rate along the ith reference trajectory at the current time, and c_1 and c_2 are tuning parameters. Energy is specific orbital energy,

$$\xi = \frac{|\dot{\boldsymbol{r}}|^2}{2} - \frac{\mu}{|\boldsymbol{r}|}, \tag{2.12}$$

and energy rate is computed by differencing the current energy with the energy computed from a prior state estimate. The values along the reference trajectory are approximated for the current time by using the values at the time step immediately prior to the current time. See Sec. 2.5 for a discussion of why this method is chosen as opposed to interpolation, and for a discussion of the values of c_1 and c_2. The motivation for this choice of distance parameter is that the target orbit is associated with a particular energy value and, since the vehicle lacks any out-of-plane control authority, the guidance objective can be posed as an energy-targeting problem without loss of generality. The current energy of a trajectory gives information about the remaining energy that must be dissipated, and the current energy rate of that trajectory gives information about whether the vehicle is on track to reach the desired energy upon atmospheric exit as compared to pre-optimized reference trajectories.

Once d_i is computed for each reference trajectory, the reference with the smallest distance parameter is selected as the nearest match. Then, the algorithm simply updates the jettison time t_j to equal the jettison time that was computed for that nearest reference trajectory, $t_{j,i}$. Like NPC guidance, the algorithm outputs a jettison time that is monitored by a jettison controller that is potentially running at a higher rate.

It should be noted that a family of guidance algorithms for aerocapture and entry relying on profiles of energy along a reference trajectory already exist in the literature. Energy controller guidance was considered for the aeroassist flight experiment in the 1980's and later studied for the Mars sample return orbiter [79, 140]. In this algorithm, targeted final energy is used to analytically compute a reference energy profile, assuming an exponential atmosphere and constant aerodynamic coefficients. This reference energy is analytically converted into a reference bank angle profile which is then tracked using a second-order controller. ERG shares with this algorithm the basic concept of defining energy and energy rate along a reference trajectory optimized to reach the target final apoapsis. A key difference, however, is that ERG generates a dispersed family of these trajectories, chooses from among them based on current energy and energy rate, then flies that reference control in open loop, as opposed to analytically computing and tracking a single reference.

To summarize, in ERG a family of optimized reference trajectories is generated during a pre-compute step. Then, during atmospheric flight updates, the nearest reference is selected based on a heuristic distance parameter and the commanded jettison time is updated to equal the jettison time associated with that reference. ERG has a number of things in common with the NPC guidance algorithm. Namely, both algorithms rely on onboard numerical propagation from a navigation-estimated state, and both solve for optimal jettison time in a root-finding procedure that requires an indeterminate number of iterations to converge. The key difference, however, is that in ERG the numerical propagations occur in a pre-compute phase that occurs before atmospheric entry, and is thus significantly less time-constrained. That is, whereas the NPC requires the prediction-correction procedure to converge during a single guidance call (0.125 s in this case), ERG only requires that the procedure converge for each reference in the time between OD cutoff and atmospheric entry. In fact, if the link budget and timing of the mission design allow, the pre-compute step could be performed on the ground and the relevant data could be uplinked along with the final OD update. Moreover, OD cutoff could be shifted earlier if necessary to allow a longer time for the pre-compute phase. An earlier OD cutoff does result in higher navigation error at entry, though, so this creates a tradeoff between accuracy and onboard computation requirements.

Quantitatively comparing the computational demand of these two algorithms would require hardware-in-the-loop simulation of a flight software-like implementation of each algorithm, which is beyond the scope of this study. While logged CPU time on a research computer is sometimes used as a basis of comparison in the literature, this approach can result in misleading data. The implementations of these two algorithms are developed as proofs-of-concept, not designed to emulate a flight software implementation and optimized for efficiency; additionally, other processes can draw from the same computing resource and affect the CPU time required. Nevertheless, the ERG algorithm has two clear advantages over the NPC in terms of CPU demand. During the atmospheric flight phase of ERG, no numerical propagation or iterative root-finding is required; the algorithm simply evaluates a mathematical expression for the distance parameter associated with each reference trajectory, then selects the minimum from among these values. It is clear that, when the algorithms are tuned for comparable performance, ERG requires significantly fewer computer operations per guidance call than the NPC and is less demanding of CPU capacity as a result. A second important feature of the ERG algorithm is that it is computationally well-posed, in that the number of individual operations required per guidance call can be predicted exactly. In contrast, the NPC requires an indeterminate number of numerical propagations to reach a given convergence tolerance as part of its root-finding procedure during each guidance call. Reduced algorithmic complexity and an ability to closely theoretically constrain CPU demand are significant advantages of the ERG over the NPC when it comes to verification and validation of flight software, especially in the case of radiation-hardened avionics with limited capacity.

Although ERG is less demanding of CPU capability, this is traded-off by a higher memory requirement compared to NPC. The time, energy, and energy rate at each point along each reference trajectory must be stored in memory and remain accessible to the guidance algorithm. Thus, the total memory required is a product of the number of reference trajectories, the number of datapoints per trajectory, and the memory required per value (e.g. 64 bits for double-precision numbers). In the following section, the relationship between targeting performance and required memory is quantitatively explored.

2.5 Results

2.5.1 NPC Performance

Figure 2.6 shows the histogram of apoapsis altitudes achieved using the NPC guidance in a 5001-trial Monte Carlo analysis, modeling the scenario and uncertainties as described in Sec. 2.2. The mean and standard deviation apoapsis altitude achieved by NPC are 5057 km and 357 km, respectively; recall that the target is 5000 km. The data are approximately Gaussian, with the exception of a small right skew due to a small number of high-apoapsis outliers. Note that one cause of these outliers is that the dispersions assumed in this work sometimes exceed the total control authority of the vehicle. For example, in cases where the atmospheric density is below nominal and simultaneously navigation errors result in delivery with a shallower entry flight-path angle than desired, the vehicle may overshoot the target orbit even if the drag skirt is never jettisoned.

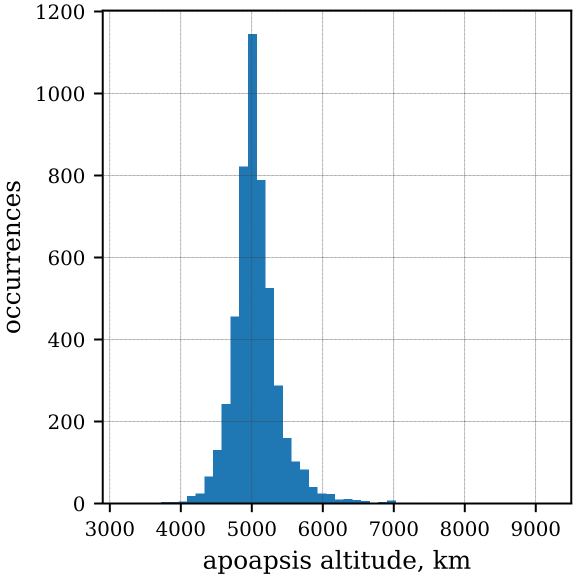


Figure 2.6: Targeting results for NPC, 5001-trial Monte Carlo analysis

In order to roughly assess the computational demand of the NPC algorithm, the number of propagations required per guidance call is counted and the maximum of this value is recorded for each trial; denote this maximum $\mathrm{p}_{\mathrm{max}}$ for convenience. In 88% of cases $\mathrm{p}_{\mathrm{max}} = 7$, and in all but 2 of the 5001 trials $\mathrm{p}_{\max} \leq 7$; the maximum observed value was 11. Because the NPC lacks guarantees on the number of iterations required for convergence, this type of numerical analysis would be required to bound the required computational capacity. The statistics of $\mathrm{p}_{\max}$ are affected by the incoming trajectory, target orbit, assumed dispersions, tuning of the guidance algorithm, and a number of other implementation details.

2.5.2 Baseline ERG Performance

The targeting performance for ERG under the same circumstances is shown in Fig. 2.7, where $N = 17$ reference trajectories are generated. In this case the mean and standard deviation apoapsis altitude are 5009 km and 355 km, respectively, as summarized in Table 2.2. Statistically speaking, these targeting results are approximately equivalent; the ERG algorithm achieves targeting performance almost identical to that of the baseline NPC algorithm. Though the mean apoapsis altitude of the ERG has lower error than that of the NPC, this difference is insignificant in the context of a 5000 km target apoapsis and standard deviation of more than 350 km. This is remarkably good performance considering that ERG can only choose from a set of 17 options for jettison time, whereas the NPC guidance refines jettison time to within a small tolerance.

Table 2.2: Apoapsis altitude statistics for baseline NPC and ERG

Algorithm	Mean, km	σ, km
NPC	5057	357
ERG	5009	355

Figures 2.8a and 2.8b provide a comparison that gives some insight into how ERG is able to accurately target a final orbit. In both cases a single jettison time is chosen before atmospheric entry and used in every trial. In Fig. 2.8a, t_j is optimized *a priori* based on the nominal scenario, whereas in Fig. 2.8b t_j is optimized using simulations beginning from the navigation-estimated state after OD cutoff 10 minutes prior to entry. Put differently, the former case is open-loop control and the latter case is equivalent to ERG with only a single reference trajectory.

In the open-loop case shown in Fig. 2.8a, targeting performance is very poor. A significant

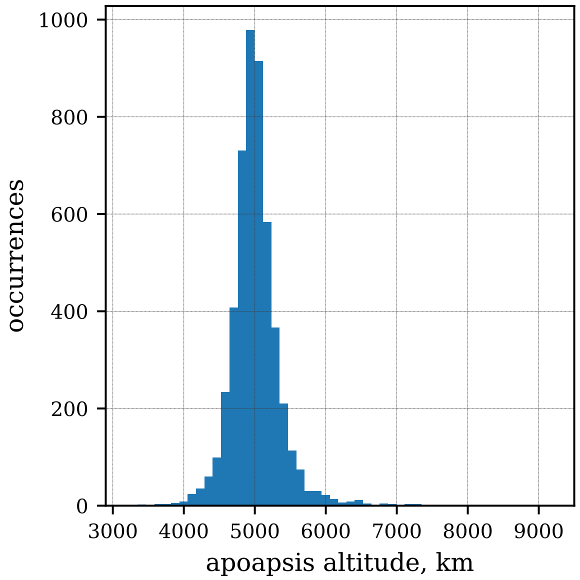
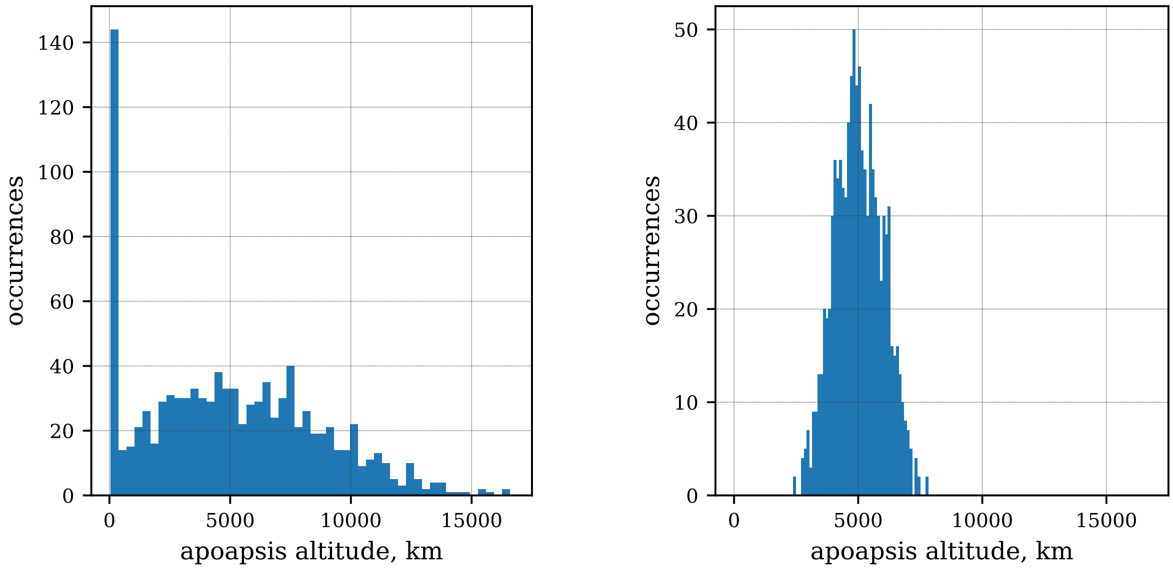


Figure 2.7: Targeting results for ERG, 5001-trial Monte Carlo analysis



(a) Jettison time optimized *a priori*, 1001-trial Monte Carlo analysis

(b) Jettison time optimized after OD cutoff, 1001-trial Monte Carlo analysis

Figure 2.8: Targeting results for fixed-time jettison

number of cases either impact the planet or have apoapsis altitudes so low that the vehicle is doomed to re-enter before having a chance to maneuver, with 13.3% of cases reaching an apoapsis below 200 km. There is also a high number of overshoot cases and a wide spread to the data. The case in Fig. 2.8b, shown with the same x-axis scaling, stands in sharp contrast. With a standard deviation of 971 km it is significantly worse than the case with 17 reference trajectories shown in Fig. 2.7, but performs far better than the case in Fig. 2.8a, avoiding any impact cases or any apoapsis altitudes above 8000 km.

This comparison serves to illustrate the following point. The state of the vehicle at atmospheric entry is subject to two distinct types of dispersions: delivery error and navigation error. The former is the difference between the pre-planned nominal entry state and true state, whereas the latter is the difference between the onboard best-estimate of the state, based on filtered navigation data, and the true state. Under the assumptions for this mission scenario, delivery error generally exceeds navigation error; that is, the spacecraft is delivered to entry with limited accuracy, but navigation filters produce a fairly accurate state estimate by the time of OD cutoff. The results in Fig. 2.8a use a jettison time based on the nominal entry state and are thus subject to both delivery and navigation errors. The Fig. 2.8b results, in contrast, use a jettison time based on the navigated state at OD cutoff, which effectively removes most of the delivery error. Therefore, it is clear that much of the benefit from the ERG algorithm is simply a result of re-computing a reference trajectory (in this case, a jettison time) onboard the spacecraft using an updated state estimate.

2.5.3 ERG Tuning

Recall that the ERG algorithm can be tuned by adjusting the values of c_1 and c_2 in the distance parameter, Eq. (2.11). A parametric study was carried out to find values of these parameters that offer reasonable performance, with results shown in Table 2.3. In order to eliminate other factors, these cases used 401 reference trajectories with 8000 datapoints per trajectory. A tuning of $c_1 = 1, c_2 = 10$ is selected based on its minimum standard deviation result, and is used for all following results as well as for the baseline case in Fig. 2.7. It is interesting to note that the minimum-variance case occurs when c_1 and c_2 are of similar magnitude, and that when either

Table 2.3: Apoapsis altitude statistics for varying distance parameter tuning

c_1	c_2	Mean, km	σ, km
1	0	6034	601
100	1	5036	389
10	1	5033	382
1	1	5027	361
1	10	5019	338
1	100	5022	340
1	1000	5053	416
0	1	5121	578

parameter is set to zero performance degrades significantly. This highlights the fact that energy and energy rate are both necessary for the best match with a reference trajectory.

2.5.4 Memory vs. Performance Trade-Offs

Although the ERG algorithm is significantly less demanding of CPU capability, it is significantly *more* demanding of memory space accessible to the guidance algorithm. It is therefore of interest to quantify trade-offs between memory and performance for the ERG algorithm. The storage required is estimated as the product $3 \times N \times M \times D$ where N is the number of reference trajectories, M is the number of datapoints per reference trajectory, and D is the required memory per datapoint, and where 3 is pre-multiplied because each reference trajectory requires storing time, energy, and energy rate at each datapoint.

In Figure 2.9, the number of reference trajectories is varied from 1 to 401 and the apoapsis altitude results are compared, with a 1001-trial Monte Carlo analysis performed in each case. $M = 8000$ datapoints are recorded for each reference trajectory. The mean and standard deviation of apoapsis altitude for these same trials are listed in Table 2.4. From these results, it is clear that increasing the number of reference trajectories above 81 makes no discernable difference in performance. From 81 to 17 there is a small increase in standard deviation, then from 17 to 9 a larger increase in variability and the first noticeable change in the histogram. For fewer than 9 reference trajectories, performance significantly degrades. Note that the mean remains centered for all cases, as overshoot and undershoot cases increase at approximately the same rate as the number of reference trajectories is decreased. Based on this analysis, a reasonable balance between memory and performance seems to be $N = 17$ reference trajectories. Note that this inflection point may change for differing mission scenarios.

A similar analysis is presented in Fig. 2.10 and Table 2.5, where in this case the number of datapoints per trajectory is varied from 8000 to 500 while holding the number of reference trajectories constant at 81. Note that the numerical propagation always occurs with a timestep of 0.125 s, meaning that for $M = 4000$ a datapoint is recorded every other step, for $M = 2000$ every 4 steps, etc., assuming propagation for 1000 s total.

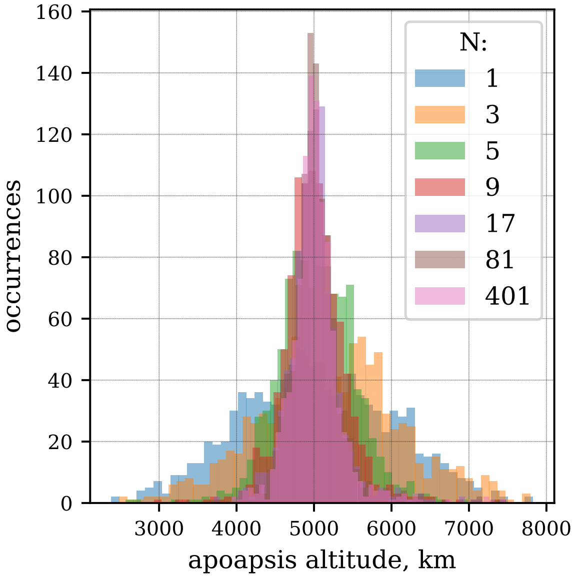


Figure 2.9: Performance comparison for varying number of reference trajectories, 1001-trial Monte Carlo analysis

Whereas the data in Fig. 2.9 remain centered while the spread increases, in this case there is a shift to the right combined with an increased spread each time that M is decreased. That is, recording fewer datapoints results in a bias toward overshoot cases as well as increasing variability. Moreover, in these results targeting performance begins to degrade immediately, without a clear inflection point.

Table 2.4: Apoapsis altitude statistics for varying number of reference trajectories

N	Mean, km	σ, km
401	5019	338
81	5020	338
17	5010	361
9	5008	415
5	5036	511
3	5210	914
1	5026	971

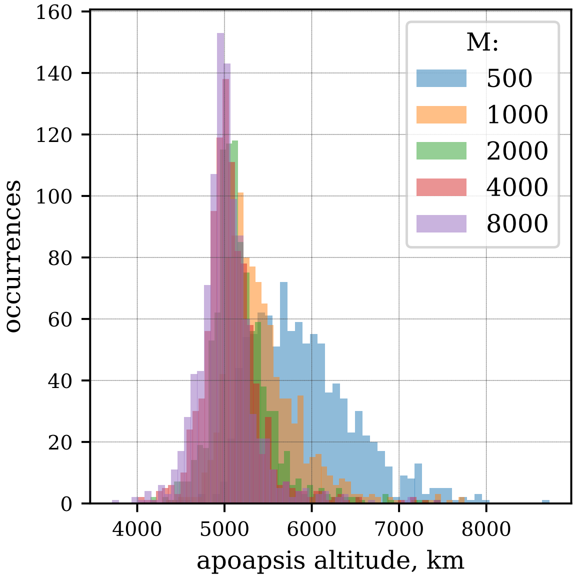


Figure 2.10: Performance comparison for varying reference trajectory resolution, 1001-trial Monte Carlo analysis

Table 2.5: Apoapsis altitude statistics for varying number of datapoints per reference trajectory

M	Mean, km	σ, km
8000	5020	338
4000	5076	345
2000	5190	365
1000	5425	433
500	5912	588

To understand these trends, recall that during the atmospheric flight phase the reference energy and energy rate values are approximated by using data from the time step immediately prior to the current time. Thus, for a resolution of 500 datapoints, the values used to compute the distance parameters are associated with a point on the trajectory up to two seconds earlier than the current time. This effectively inflates the energy of every reference point, and the result is that the matched trajectory has a higher density scale factor than it otherwise would, leading to an earlier jettison time and ultimately the skew toward overshoot cases observed in Fig. 2.10. It may seem as though using interpolation to compute energy and energy rate of the reference trajectory at the current time would address this issue. However, energy rate changes as a step function at the moment of drag skirt jettison. In short, interpolating across this discontinuity disrupts the ability of the algorithm to successfully match with the reference trajectory that would actually yield optimal performance. Therefore, in this work values from the previous time are used and the requirement for high-resolution reference trajectory data is accepted; a value of $M = 8000$ is taken to be the baseline configuration shown in Fig. 2.4. As a point of reference, if double-precision values of 64 bits each are assumed for this baseline configuration, a total of about 3.3 MB of memory would be required. In comparison, the Sphinx avionics platform, which was developed at JPL for SmallSat missions and now has flight heritage from the Lunar Flashlight spacecraft [141], includes 256 MB of synchronous dynamic RAM [142] This suggests that the 3.3 MB requirement is well within reason. Thus, while the tradeoff of reduced CPU demand for the ERG is increased memory requirement, this increase is not likely to represent a significant detriment to the overall design.

2.6 Discussion

A notable limitation of both algorithms presented in this work is that path constraints, such as peak heat flux and peak g-load, are not incorporated into the onboard logic. While other work does provide a method to account for these constraints in NPC guidance for aerocapture and entry [65, 143], there is currently no equivalent approach for ERG. The impact of this limitation strongly depends on the mission scenario of interest. For the small satellite demonstration mission studied here, the vehicle design is expected to have significant margin compared to the expected heating and g-loads, and thus it is likely unnecessary for the onboard guidance to directly incorporate the associated constraints. In more stressing cases for which the nominal scenario is near the limits of heating and g-loads, an additional outer loop could be added to the ERG algorithm to prohibit executing jettison times that are predicted to have an unacceptably high likelihood of resulting in path constraint violation.

The most likely barriers to implementation of this algorithm are the computation time required to generate the reference trajectories and the memory required to store the associated data. Therefore, it would be of interest to extend the approach presented in this work to achieve the same performance with fewer reference trajectories, or else improve performance with the same number.

A potential approach would be to interpolate between the reference trajectories in some way, such that the commanded jettison time does not necessarily equal one of the reference jettison times. Because the current vehicle state will generally not equal the state at that time along even the nearest reference trajectory, the difference between the current and reference state could inform a correction to the jettison time of that reference trajectory. One could accomplish this by computing linear sensitivities of jettison time with respect to each relevant state component, then computing the correction term as the product of this sensitivity and the state difference. The altitude, velocity magnitude, and flight-path angle could be considered a sufficient set of state components since the primary concern is planar motion. However, there are two significant issues with that approach. First, this would require computing and storing sensitivity values at each time along each reference trajectory, resulting in a major increase in CPU demand and, assuming three state sensitivities, doubling the amount of memory required. Second, even setting aside the computational challenges, the dynamics are nonlinear and the true state tends to diverge significantly from any of the reference trajectories over time, leading to inaccurate linearization.

One possible workaround is the incorporation of quasi-initial conditions. These fully represent the current state by back-propagating through a nominal model, effectively defining a nonlinear coordinate transformation. Quasi-initial conditions have been shown to be a more linear state representation than the state at a given time for aerocapture [144]. This state representation also removes the requirement of computing sensitivities at each time, since they need only be computed once in quasi-initial condition space, although a single back-propagation per guidance call is then required during atmospheric flight. Preliminary work by the authors incorporates quasi-initial conditions into an extension of the ERG algorithm presented here [139]. While early results are promising, it is difficult to guarantee reliable and accurate linearization in the presence of dispersions, whereas the simpler approach presented here performs well. Furthermore, note that while the computational burden of the quasi-initial condition approach is far less than a linearization based on the current state, it does still require numerically computing three sensitivity values for each reference trajectory, meaning that the number of numerical propagations during the pre-compute phase increases by roughly a factor of four.

Another interesting avenue for future work is some method of nonlinear corrections to the reference jettison time. This could be combined with the previous concept, such that some nonlinear interpolation surface is generated in quasi-initial condition space during the pre-compute step and then used to guide corrections during the atmospheric flight phase. This could potentially alleviate issues related to inaccurate linearization, although it would likely require a commensurate increase in computational cost.

2.7 Conclusions

It is worth returning here to the single-event jettison concept itself. This control architecture inherently sacrifices performance in pursuit of simplicity. By relying on the jettison of a single rigid drag skirt, the vehicle lacks any out-of-plane control authority, forgoes continuous control and, perhaps most importantly, is coasting without any control authority for the remainder of atmospheric flight once the drag skirt is jettisoned. A range of other approaches address one or more of these shortcomings, including continuously-variable drag modulation [117], jettison of multiple drag skirts [119], and lift modulation [115, 116]. However, each of these architectures adds complexity in terms of flight hardware and, in most cases, flight software. The motivation to use single-event jettison drag-modulation is not to achieve orbit insertion as accurately as possible; rather, the goal is to reliably reach the target orbit within some reasonable error bounds while keeping the aerocapture subsystem as simple as possible. This is appropriate either for missions that can tolerate a range of apoapsis altitudes or for cases where the spacecraft has sufficient propellant to clean up the expected targeting errors.

This broader motivation should inform the choice of guidance algorithm and the interpretation of results. In this work a guidance algorithm, ERG, is presented that achieves equivalent targeting performance to the baseline NPC. Both algorithms have a standard deviation of about 355 km and in some outlier cases reach an apoapsis several thousand kilometers higher than the target. However, the choice of an inherently limited control architecture limits the ability of any guidance algorithm to accurately target a final orbit. The fact that the two distinct algorithms achieve nearly-identical results could suggest that both are operating near the ceiling of performance for this scenario. The ERG algorithm achieves this result with significantly reduced CPU demand, albeit with an increased demand for accessible memory. The simplicity of the atmospheric flight phase of the ERG algorithm aligns well with the broader motivation to reduce complexity for this type of mission scenario.

Chapter 3

Onboard Modeling of Uncertain Atmospheres

3.1 Introduction

Hypersonic flight mechanics are characterized by nonlinear dynamics and high sensitivity to variations in atmospheric density. Furthermore, the behavior of planetary atmospheres is complex and difficult to predict. Appropriate modeling of density is thus key to the analysis of hypersonic trajectories, including in the context of onboard modeling for closed-loop guidance schemes. Autonomous guidance algorithms typically treat density as a known function of altitude, either in analytical form as an exponential function of altitude or by interpolating from a table [145]. In-flight measurements of sensed acceleration can be converted to estimates of current density (though this approach does treat aerodynamic properties as known), and these observations are then incorporated by multiplying the nominal profile by the ratio of observed density to expected density [84, 85]. Recent work contributes more sophisticated methods of incorporating in-flight observations, such as machine learning or an ensemble correlation filter [86, 87, 88]. However, these methods ultimately treat the density as known and update a nominal profile.

Recent and ongoing works propose stochastic approaches to closed-loop guidance with the aim of being robust to uncertainties without taking an overly-conservative approach [96, 94], and central to these methods is an onboard prediction of state and environmental uncertainty. Several non-Monte Carlo uncertainty quantification (UQ) techniques, including polynomial chaos expansion and linear covariance analysis [89, 146, 147], potentially enable onboard uncertainty propagation for hypersonic flight vehicles. However, these methods generally require a parametric, low-dimensional representation of uncertainty [148, 149]. Recent studies have explicitly incorporated a probabilistic atmosphere model into UQ approaches [150, 92, 93]; however, these approaches typically assume an exponential form for density and incorporate uncertainty by dispersing the atmospheric scale height and surface density, a method that always results in an exponential profile. The assumption of exponential density significantly limits the ability of the model to capture more complex perturbations due to its inability to capture short-period perturbations or other deviations of the density profile from the idealized exponential shape [138]. Semi-empirical models such as the Global Reference Atmospheric Models (GRAMs) from NASA provide much higher-fidelity representations of the atmosphere and its response to external factors, such as solar weather [151], but lack a convenient low-dimensional and parametric form. Estimating uncertainty using these models typically requires generating a large number of density profiles then computing statistics of the generated dataset, rather than estimating variability directly. Thus, GRAMs and similar models are not feasible for onboard uncertainty propagation purposes.

This motivates the development of a reduced-dimensionality model that retains the higher-fidelity properties of models like GRAM, and a method for in-flight updates to this model. Previous work treats density as a Gaussian random field with altitude the sole independent variable, and demonstrates a Karhunen-Loève expansion (KLE) for density [138]. Reference [146] shows that linear covariance analysis incorporating this model closely matches Monte Carlo results. This study expands on these results in the following ways. Practical implementation of the KLE is explored in greater detail, examining alternative methods of constructing the expansion. The KLE models are also compared against variational autoencoder (VAE) models, which use deep neural networks to achieve nonlinear dimensionality reduction as compared to the linear dimensionality reduction attained by KLE models, and which enable representing non-Gaussian random processes. A VAE is a generative model in that it learns and generates samples from the joint probability density function of the data. The efficiency of each approach in capturing density variability is compared both directly and through statistics of dispersed trajectories generated in Monte Carlo analyses using each model. The aim of this work is not to claim that either the KLE or VAE modeling approach is better for this application; rather, this study provides a proof of concept for each model type and discusses the benefits and drawbacks of each. New work outlining and demonstrating an expansion on the KLE model to treat density as a function of multiple variables (e.g. altitude, latitude, and longitude) is presented and its comparative utility is discussed. Finally, an approach to updating the KLE based on sequential noisy density measurements is presented and demonstrated, and the potential for onboard execution of this method is discussed.

3.2 Preliminaries

3.2.1 Review of Karhunen–Loève expansion

A random field is a function that maps a random outcome to a continuous function across a (possibly multi-dimensional) domain in space. Somewhat more formally: for some measurable space $(\Omega, \mathcal{F})$ of sample space Ω and σ-field $\mathcal{F}$ of subsets of Ω, a random field $\{\Phi(z) : z \in \mathcal{Z} \subseteq \mathbb{R}^d\}$ is a collection of random variables $(X_z)_{z \in \mathcal{Z}}$ with values that map $\Omega \mapsto \mathbb{R}$ [152]. A Gaussian random field (GRF) $\Psi(z)$ is a random field for which any finite linear combination of the random variables X_z results in a Gaussian random variable; that is, at any point z_i in the domain $\mathcal{Z}$ the probability density function of the value of the field $\Psi(z_i)$ is Gaussian [153]. A GRF is fully characterized by its mean function μ and covariance function Σ,

$$\mu(z) = \langle \Psi(z) \rangle, \tag{3.1}$$

$$\Sigma(z_1, z_2) = \langle (\Psi(z_1) - \langle \Psi(z_1) \rangle)(\Psi(z_2) - \langle \Psi(z_2) \rangle) \rangle, \tag{3.2}$$

where $\langle \rangle$ is the expectation operator.

A Karhunen–Loève (also known as Kosambi-Karhunen–Loève) expansion represents a random field through an infinite linear combination of orthogonal basis functions (a Fourier expansion), in such a way that, when truncated to a fixed number of terms, the choice of the basis functions minimizes the mean-square error [154, 155]. This definition is shown by Eq. (3.3) where Φ is the random field, z is the independent variable, and λ_i and $\phi_i(z)$ are the eigenvalues and eigenfunctions of the covariance function of the random field $\Sigma(z_1, z_2)$, respectively, as shown in Eq. (3.4). Finally, each Y_i is a random variable described by Eq. (3.5):

$$\Phi(z) = \langle \Phi(z) \rangle + \sum_{i=1}^{\infty} \sqrt{\lambda_i} \phi_i(z) Y_i; \tag{3.3}$$

$$\int_0^T \Sigma(z_1, z_2) \phi_i(z_2) \mathrm{d}z_2 = \lambda_i \phi_i(z_1); \tag{3.4}$$

$$Y_i = \frac{1}{\sqrt{\lambda_i}} \int_0^T \Phi(z) \phi_i(z) \mathrm{d}z. \tag{3.5}$$

In practice, the eigenvalues and eigenfunctions are sorted by descending magnitude of the eigenvalues and then the sum in Eq. (3.3) is truncated after some d_K number of sufficient terms. Determining the required d_K is problem-dependent, but in general it is chosen such that the mean-square norm of the approximation is within some relative error of the exact mean-square norm. Eq. (3.6) gives one heuristic method, where k is some sufficiently large number and α is close to 1 based on the desired level of permissible error (for a relative mean-square norm error of $(1 - \alpha) \times 100\%$).

$$d_K = \min \left\{ j : \frac{\sum_{i=1}^{j} \lambda_i}{\sum_{i=1}^{j+k} \lambda_i} \geq \alpha \right\} \tag{3.6}$$

In the case where $\Phi(z)$ is a GRF $\Psi(z)$, the Y_i's are all independent and identically distributed (i.i.d.) standard normal random variables:

$$Y_1, Y_2, ... \sim \mathcal{N}(0, 1) \text{ i.i.d.} \tag{3.7}$$

Often the probability density function of a random field is not known exactly, but some sufficiently large dataset is available. In this case the sample covariance matrix is computed,

$$\boldsymbol{C}_{ZZ} \approx \boldsymbol{Q}_{ZZ} = \frac{1}{m-1} \boldsymbol{F}\boldsymbol{F}^T, \tag{3.8}$$

where $\boldsymbol{Q}_{ZZ}$ is the unbiased estimate of the sample covariance matrix $\boldsymbol{C}_{ZZ}$, $\boldsymbol{F} \in \mathbb{R}^{n \times m}$ is a matrix such that each column is an observation vector less the sample mean, n is the number of datapoints per observation vector, and m is the number of observation vectors in the dataset. Having computed a covariance matrix, it is straightforward to find the eigenvalues and eigenvectors of that matrix and sort them according to descending order of the eigenvalues, and the results are the $\{\lambda_i\}$ and

$\{\boldsymbol{\phi}_i\}$ in Eq. (3.3), respectively, where each $\boldsymbol{\phi}_i$ is now a vector rather than a function. The discrete KLE form of a GRF $\boldsymbol{\Psi} \in \mathbb{R}^n$ is thus summarized below:

$$\boldsymbol{\Psi} \approx \langle \boldsymbol{\Psi} \rangle + \sum_{i=1}^{d_K} \sqrt{\lambda_i} \boldsymbol{\phi}_i Y_i, \tag{3.9}$$

$$Y_i, ..., Y_d \sim \mathcal{N}(0, 1) \text{ i.i.d.}$$

3.2.2 Review of Variational Autoencoder

An autoencoder is a type of latent variable model that provides a method of nonlinear dimensionality reduction, consisting of an encoder and a decoder connected sequentially. The encoder takes the input data and, through one or more neural network layers, converts the data into a lower dimensional encoding vector – i.e., set of latent variables – representing some learned features of the data. The decoder, through a symmetric set of neural network layers, then attempts to reconstruct the original input from the latent variables. By forcing the input data through a bottleneck, the autoencoder learns a latent space that can be used for compressed representation of the data. The use of deep neural networks for the encoder and decoder enables the autoencoder to take advantage of *nonlinear* relationships in the input data. In fact, it can be shown that a linear autoencoder (one which lacks nonlinear activation functions in the neural networks) will learn the same latent space as a KLE applied to discrete data, commonly known as principal component analysis [156].

While autoencoders are useful in applications such as denoising and anomaly detection, they are limited in their utility as generative models. Because the latent space constructed by an autoencoder is not necessarily smooth or continuous, interpolation or randomly sampling from the latent space with the goal of generating new synthetic data can produce unrealistic results. A variational autoencoder addresses this limitation by describing the encoder, decoder, and the latent variables in terms of probability distributions rather than individual deterministic entities [157]. More specifically, a type of distribution is assumed *a priori* and then, given an input vector that is not necessarily Gaussian, the encoder outputs encoding vectors for the parameters describing that distribution; often, a Gaussian distribution is assumed, and the encoder thus outputs the mean vector and covariance matrix. During the reconstruction process, latent variables are drawn as samples from these (potentially correlated) probability distributions before being passed through the decoder.

This probabilistic description encourages local smoothness in the latent space, but without additional constraints the distributions can become narrow and sparse, resulting in overfitting. To compensate, VAEs incorporate Kullback-Leibler (KL) divergence as a regularization term. KL divergence essentially measures the divergence between two probability distributions [158]. By penalizing divergence between the learned latent variable distributions and a target distribution (often the standard normal), the encodings are attracted toward the center of the latent space and sufficient variance is encouraged. The loss function can then be written as the weighted sum of these two terms,

$$\mathcal{L}(\boldsymbol{x}, \hat{\boldsymbol{x}}) + \beta_{\mathrm{KL}} \sum_{j} \mathrm{KL}(q_j(\boldsymbol{z}|\boldsymbol{x})||p(\boldsymbol{z})), \tag{3.10}$$

where $\boldsymbol{x}$ is the input vector, $\hat{\boldsymbol{x}}$ is the reconstructed output vector, $\boldsymbol{z}$ is the latent variable, $q_j(\boldsymbol{z}|\boldsymbol{x})$ is the learned distribution for each dimension j of the latent space and $p(\boldsymbol{z})$ is the assumed prior distribution [159]. $\mathcal{L}()$ is the likelihood function penalizing reconstruction error, typically evidence lower-bound (ELBO) [160], and KL() is the KL divergence acting as a regularizing term. Finally, β_{KL} determines the weight of the KL divergence term relative to the reconstruction loss, where the subscript is used here to distinguish from ballistic coefficient.

To briefly summarize, a VAE is a probabilistic method of nonlinear dimensionality reduction that is a popular choice for generative modeling. The derivation of a VAE can also be understood as applying Bayesian variational inference to the latent variable distributions of an autoencoder. For a more thorough mathematical treatment of VAEs, the reader is directed to Refs. [159, 160].

3.3 Methodology

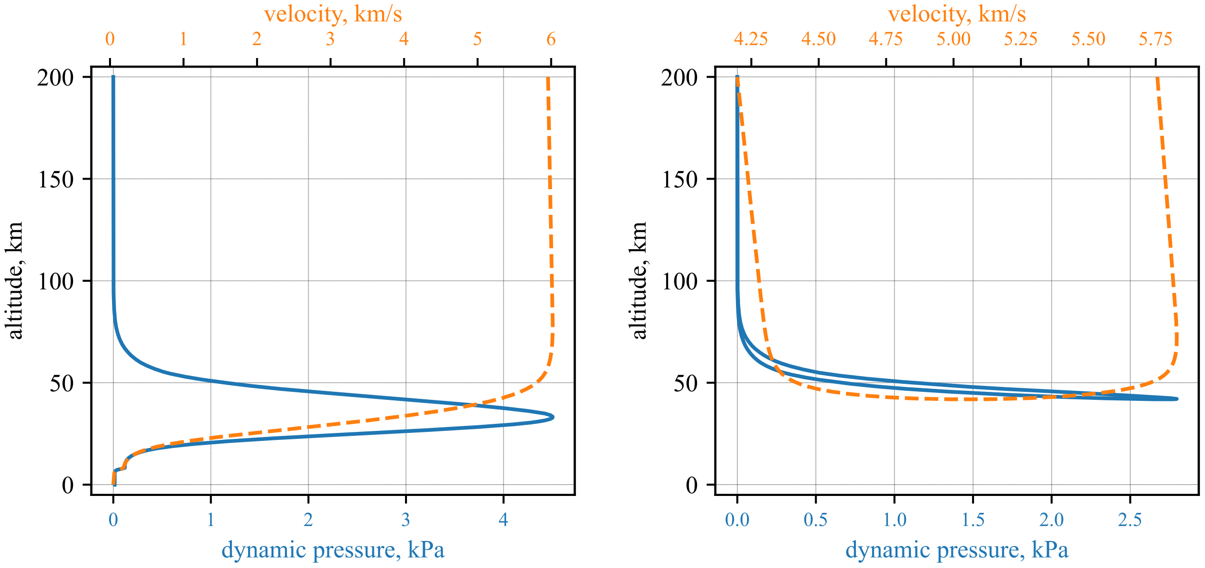
3.3.1 Simulation Description

This section briefly describes the methodology for trajectory simulation used in this study and summarizes relevant vehicle parameters. Trajectories are simulated by numerically propagating the three degree-of-freedom equations of motion for atmospheric flight about a rotating ellipsoidal planet via explicit Runge-Kutta integration of order 4(5). Density is modeled using MarsGRAM 2010 [151], interpolating from a resulting table of density vs. altitude unless stated otherwise. Mars is assumed to have gravitational parameter $\mu = 4.305 \times 10^4\ \mathrm{km}^3\,\mathrm{s}^{-2}$, equatorial radius $R = 3397.2\,\mathrm{km}$, oblateness spherical harmonic coefficient $J_2 = 0.001964$, and a planetary rotation period of $\omega_p = 1.02595675$ days [161]. Mach number is defined as the ratio of vehicle speed to the speed of sound $M = v/a$, where sound speed a for the Martian atmosphere is interpolated from a nominal tabular model [162]. Heat flux is modeled by computing convective heat flux $\dot{q}$ at the stagnation point assuming a fully catalytic surface using the Sutton-Graves expression shown in Eq. (3.11), where ρ is density and a value of the heating coefficient $k = 1.904 \times 10^{-4}\,\mathrm{kg}^{0.5}/\mathrm{m}$ is used based on nominal atmospheric composition at Mars [163]. Dynamic pressure q is defined by Eq. (3.12).

$$\dot{q} = k\sqrt{\frac{\rho}{R_n}}V^3 \tag{3.11}$$

$$q = \frac{1}{2}\rho v^2 \tag{3.12}$$

There are two types of trajectories used as representative examples in this study. The first is a steep direct entry trajectory at Mars for the Small High Impact Energy Landing Device, or SHIELD, a small, mostly-passive probe under development at NASA JPL intended for low-cost access to the Martian surface [164]. Once reaching subsonic conditions, SHIELD deploys a drag skirt, then jettisons the heatshield shortly thereafter. The drag coefficient C_D during each configuration varies with Mach number and is linearly interpolated from tabular data provided by the JPL SHIELD team. Ballistic coefficient $\beta = m/(C_D A)$ describes the ratio of inertial forces to aerodynamic forces, where m is vehicle mass and A is reference area; the ballistic coefficient for SHIELD ranges from about 20 kg m^{-2} shortly after entry to around 5 kg m^{-2} near the surface after drag skirt deployment and heatshield jettison. SHIELD has a lift-to-drag ratio of $L/D = 0$, and an assumed nose radius of $R_n = 0.85$ m. The trajectory considered in this study is defined by an entry velocity of 6 km/s and an entry flight-path angle (EFPA) of $-18\,^\circ$ at the atmospheric interface altitude of 125 km, entering due-East at $0\,^\circ$ latitude and $0\,^\circ$ longitude, where flight-path angle is defined as the angle between the air-relative velocity of the vehicle and the local horizontal. The reference SHIELD direct-entry trajectory is shown in Fig. 3.1a.



(a) SHIELD direct-entry

(b) aerocapture

Figure 3.1: Dynamic pressure and velocity magnitude vs. altitude for reference trajectories. Note identical y-axis scaling, different x-axis scaling.

The other trajectory considered here is aerocapture at Mars by a vehicle similar to the Mars Science Laboratory (MSL) aeroshell. A ballistic coefficient of $\beta = 130$ kg m^{-2} and lift-to-drag ratio of $L/D = 0.24$ are assumed [165], and the vehicle flies full-lift-up for the duration of the trajectory. The entry is again due-East at $0\,^\circ$ latitude and longitude, in this case with entry velocity of 5.8 km/s and EFPA of $-11\,^\circ$. A nose radius of $R_n = 1$ m is assumed, which conveniently normalizes the value of $\dot{q}$ for re-scaling to other vehicles. The reference aerocapture trajectory is shown in Fig.

3.1b.

3.3.2 VAE Architecture and Training

This section summarizes the architecture of the deep neural networks used to construct the VAE models in this work, and describes the approach taken to training. It is not a claim of this work that this particular architecture or training methodology is optimally suited to representing atmospheric density; rather, confronted with a large number of tunable parameters, this is an approach that was found to work well over the course of trial-and-error experimentation, and it is detailed here for reproducibility.

The encoder is built from a 6-layer deep neural network with the following numbers of nodes: 256, 256, 128, 128, 64, and 64; the decoder is also 6 layers such that the order of dimensions is reversed, going from 64 to 256. The latent space is limited to only 4 dimensions; this is the dimensionality that directly corresponds to the number of terms in the KLE models. All neural networks use the Gaussian Error Linear Unit (GELU) nonlinear activation function [166], and are implemented using the open-source tool PyTorch [167, 168].

The models are trained with batch size 1024 for 100,000 epochs, long enough that the loss curve was observed to plateau. A weighting parameter of $\beta_{\mathrm{KL}} = 0.15$ is selected, and the loss function is normalized by the batch size. The learning rate is initially set to 1×10^{-3}, and a learning rate scheduler is implemented to reduce the learning rate after a period of time once the loss is observed to plateau. Specifically, the learning rate is reduced by a factor of 0.9 if no improvements are observed after 500 consecutive epochs, with a threshold for improvement of 1×10^{-5}. Moreover, a cooldown period of 2500 epochs is required to pass before resuming normal operations after each time the learning rate is reduced[1] .

---

[1] ReduceLROnPlateau via PyTorch

3.4 Columnar Atmosphere Model Comparison

In this section, atmospheric density is approximated as a random field as a function of only altitude. In reality, atmospheres vary across 3D position and time, and are affected by external factors such as space weather. However, for applications like entry and aerocapture which traverse tens of vertical kilometers within the atmosphere, the dominant factor in density change is altitude. Thus, a columnar atmosphere model is assumed in this section, such that $\rho(h, \phi, \theta, t) \approx \rho(h)$ where ρ is density, h altitude, ϕ latitude, θ longitude, and t elapsed time. See Section 3.5 for a discussion of density variation with latitude and longitude.

While a random field is a theoretically appropriate choice for modeling density [138, 169, 170], it is an infinite-dimensional object. In contrast, the non-Monte Carlo methods for onboard uncertainty propagation discussed earlier require a parametric, finite-dimensional representation of density variability [148, 149]. Thus, some form of dimensionality reduction is required to go from either raw data or a more complex model to a parametric, low-dimensional model appropriate for onboard use. In this section, KLE and VAE approaches are both applied to construct density models, and the results are compared for accuracy in their generative modeling as well as, crucially, their accuracy in predicting quantities of interest such as peak heat flux.

Density exhibits approximately Gaussian probability with correlation structure across a spatial domain; see Ref. [138] for detailed justification of this Gaussian assumption based on Mars-GRAM 2010 data. Thus, a KLE can be constructed under the assumption that density is a GRF, then truncated after an appropriate number of terms. The sample covariance matrix is formed from any sufficiently large dataset of density values vs. altitude; typically, it is convenient to use simulated data from a relevant model such as a GRAM. Note that, to avoid a nonzero probability of producing a negative value, the density random field should in fact be treated as a truncated Gaussian.

Figure 3.2 shows the result of constructing a KLE from a dataset of 5000 density profiles output by MarsGRAM, denoted KLE-ρ for shorthand. For the sake of later comparison, a fixed number of $d_K = 15$ terms is used for this and all subsequent KLE models in this section unless noted otherwise. The horizontal axis of this plot shows normalized density perturbation $\delta\rho$, as defined in Eq. (3.13), rather than density itself because this captures variability better even as the value of density changes by orders of magnitude across this altitude range:

$$\delta\rho = \rho/\bar{\rho} - 1. \tag{3.13}$$

The thick dashed lines show the $\pm 3\sigma$ bounds, where σ is standard deviation. In the case of MarsGRAM these bounds are computed directly from the sample profiles; for the KLE, 5000 separate realizations are generated and evaluated, then standard deviation is computed from this generated dataset. In addition, three sample profiles from each model are shown in the thin solid lines.

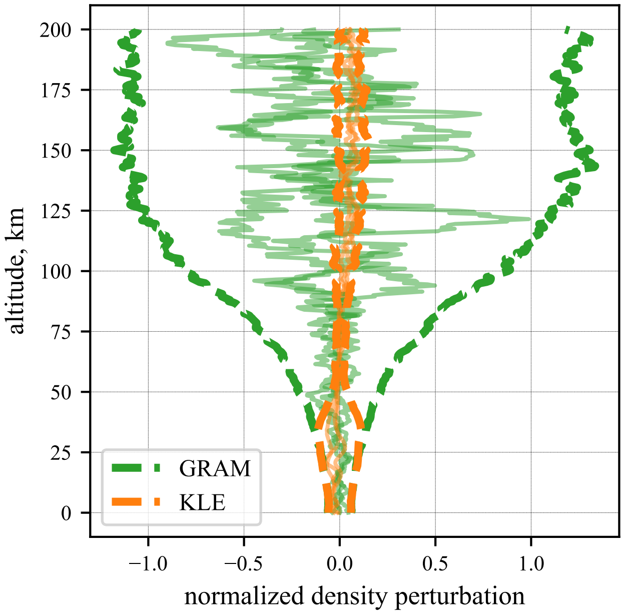


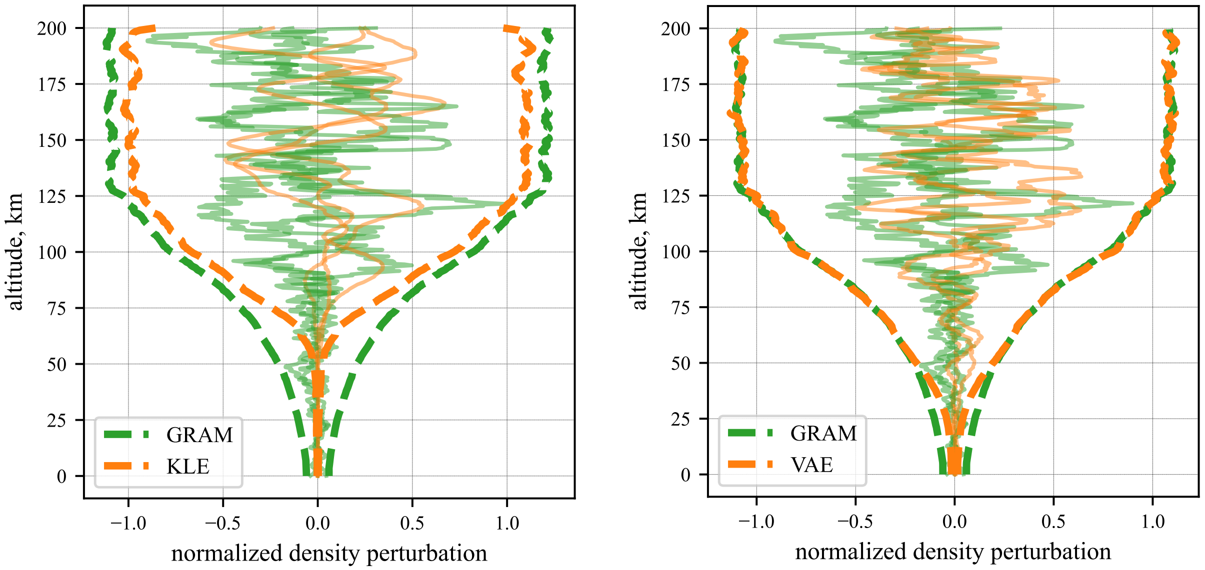
Figure 3.2: KLE vs. Mars-GRAM for KLE constructed from density values; thick dashed lines are $\pm 3\sigma$ bounds, thin solid lines are sample profiles

Notably, the KLE 3σ bounds only align with the MarsGRAM bounds up to about 35 km, badly underestimating variability at higher altitudes. This occurs because the value of density is much greater at low altitudes: for Mars, about 1×10^{-2} kg/m^3 at the surface, order of 1×10^{-5} kg/m^3

at 50 km, and order of $1 \times 10^{-10} - 1 \times 10^{-9}\,\mathrm{kg/m^3}$ at the atmospheric interface altitude of 125 km. The KLE is truncated based on eigenvalue magnitude, and the variability at low altitudes where density is high is prioritized as a result, even though as a percentage of nominal density varies more at high altitudes. For this reason, a KLE based on density values is an inefficient way to capture normalized density perturbations at high altitudes. A VAE model trained directly on density data suffers even more from essentially the same issues; because of the widely-varying magnitudes of the training data, the VAE fails to meaningfully learn density behavior at all except for at very low altitudes.

This shortcoming can be addressed by constructing the models differently. While columnar density remains the quantity of interest, the data can be pre-processed for model construction in a variety of ways, with a converse post-processing step recovering density values. For example, a model can be constructed from normalized density perturbation values in the following way. First, compute $\delta\rho$ values corresponding to each value in the dataset. In the case of a VAE, then train the model on this $\delta\rho$ data directly. In the case of a KLE, form a mean vector and covariance matrix for these $\delta\rho$ data and construct a KLE using these summary statistics. Finally, treat the outputs of this model as $\delta\rho$ values and re-arrange Eq. (3.13) to recover density values. The results of constructing KLE and VAE models in this way are shown in Fig. 3.3, denoted KLE-$\delta\rho$ and VAE-$\delta\rho$, respectively.

Figure 3.3 shows a clear improvement in terms of capturing overall density variability, and the sample profiles now look similar to the GRAM output. However, both models significantly underestimate variability below 50 km in altitude. The specific case of the KLE-$\delta\rho$ model, shown in Fig. 3.3a, does an especially poor job at capturing variability at low altitudes and also moderately underestimates variability at altitudes above 50 km. These models in some ways suffer from the opposite problem as the KLE-ρ model: because normalized density perturbations are smaller near the surface, this region is poorly captured, whereas the model performs relatively well at high altitudes. That said, the KLE/VAE-$\delta\rho$ models are more compact, meaning that for a given dimensionality they each give a better approximation of density variability with altitude than an equivalent model



(a) KLE vs. Mars-GRAM

(b) VAE vs. Mars-GRAM

Figure 3.3: Models constructed from normalized density perturbations; thick dashed lines are $\pm 3\sigma$ bounds, thin solid lines are sample profiles

trained directly on density values.

However, it is important to keep the application of interest in mind. The goal of these approximations is not to model the atmosphere as well as possible; the real goal is to provide a compact atmosphere model that results in accurate trajectory predictions when compared to trajectories predicted using MarsGRAM directly. Recall that aerodynamic force scales with dynamic pressure q. As seen in Fig. 3.1a, for a planetary entry trajectory dynamic pressure peaks at mid to low altitudes, with the particular altitude depending on the vehicle and trajectory. Above this altitude density is too low for significant dynamic pressure, and below this altitude the vehicle has slowed down to the point that dynamic pressure greatly reduces. A similar phenomenon occurs in reverse for launch vehicles. Therefore, it would be of interest for the model to prioritize density variation where it matters most for a given trajectory of interest; that is, where dynamic pressure is highest.

To that end, a scaling vector $\boldsymbol{k}_q$ is constructed based on dynamic pressure along the reference SHIELD entry trajectory, with a value corresponding to each altitude step in the discretization of the original density data. In order for the resulting training data to have consistent magnitudes, the actual dynamic pressure in Pascals is divided by 100 and the vector is further modified to have a minimum of 1,

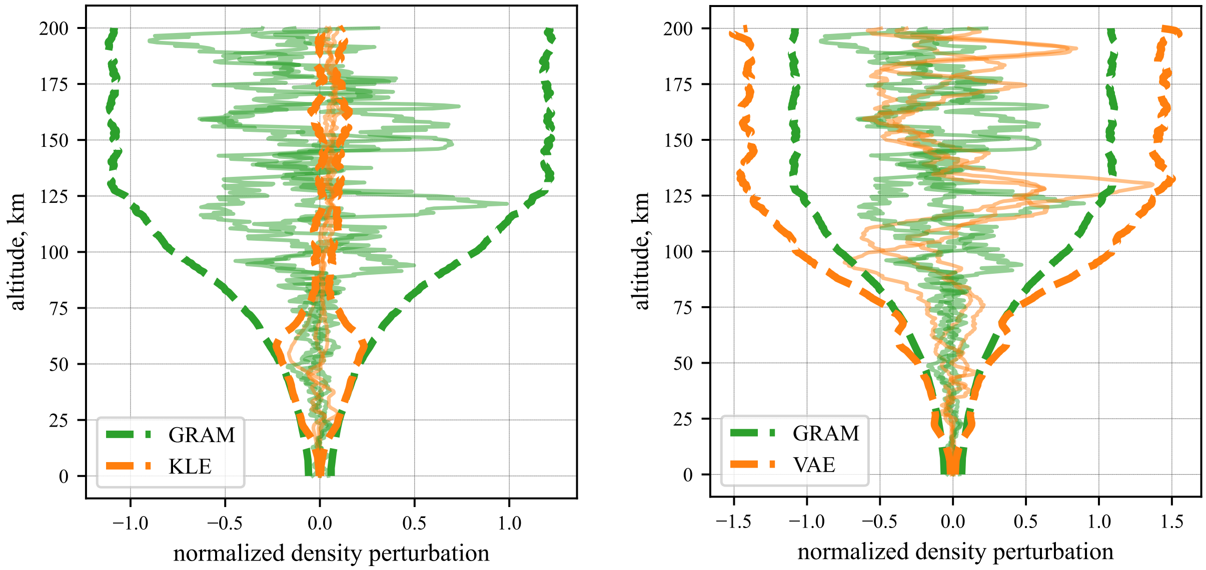
$$\boldsymbol{k}_q = \max(q/100, 1). \tag{3.14}$$

The training data are then generated by elementwise multiplying the vector of $\delta\rho$ values by the scaling vector $\boldsymbol{k}_q$, and the output of the model is then correspondingly divided by $\boldsymbol{k}_q$ before converting the normalized perturbations back to density values. In effect, this informs the reduced-dimensionality model which altitude range is most important to capture.

Figure 3.4 shows results for KLE and VAE models built from normalized density perturbations that have been scaled based on reference dynamic pressure, denoted KLE-q and VAE-q, respectively. As seen in Fig. 3.4a, the 3σ bounds computed by this KLE-q closely match GRAM from about 60 km down to about 20 km, corresponding closely to the dynamic pressure pulse shown in Fig. 3.1a. Given the fixed number of terms in the expansion, this comes at the expense of accuracy outside of that altitude range, where this expansion underestimates variability. The corresponding VAE-q model, shown in Fig. 3.4b, exhibits similar results except that, for altitudes outside of the prioritized range the model overestimates variability in some altitude regions and underestimates it in others.

In order to take a closer look at model performance at a specific altitude of interest, Fig. 3.5 shows histograms of the normalized density perturbation value predicted at 40 km altitude by the KLE/VAE-q models compared with the value given by GRAM. There are two key takeaways from this visualization. First, the KLE-q and VAE-q both do excellent jobs of recreating the empirical distribution of the training data. Second, the training data are, by inspection, well-approximated by a Gaussian distribution at this altitude. The highly-Gaussian nature of the training data explains why the KLE, which assumes an underlying GRF, does just as well as the VAE at this altitude.

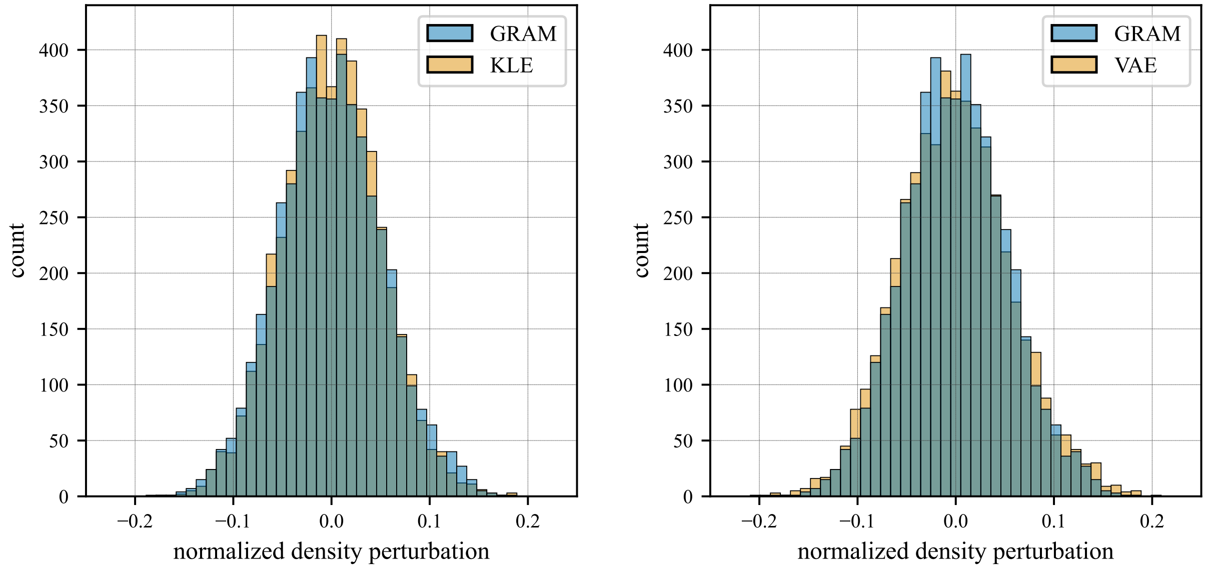
As previously mentioned, the true quality test for these density models is how well they predict dispersed trajectories compared to GRAM. To that end, a 1000-trial Monte Carlo analysis



(a) KLE vs. Mars-GRAM

(b) VAE vs. Mars-GRAM

Figure 3.4: Models constructed from normalized density perturbations scaled by SHIELD dynamic pressure profile; thick dashed lines are $\pm 3\sigma$ bounds, thin solid lines are sample profiles



(a) KLE vs. Mars-GRAM

(b) VAE vs. Mars-GRAM

Figure 3.5: Histograms of density at 40km altitude, generated by models constructed on normalized density perturbations scaled by SHIELD dynamic pressure profile

is performed for each of these models and for GRAM, where the only dispersed parameter in each analysis is density. A violin plot comparing the statistics of peak heat flux for each case is shown in Fig. 3.6. The KLE-ρ, KLE-$\delta\rho$, and VAE-$\delta\rho$ models underestimate variability to varying degrees. The KLE-q and VAE-q models have comparably good results, and both match well with the statistics predicted by GRAM directly. These results demonstrate that scaling normalized density perturbations based on reference dynamic pressure is the most compact of the modeling approaches considered here.

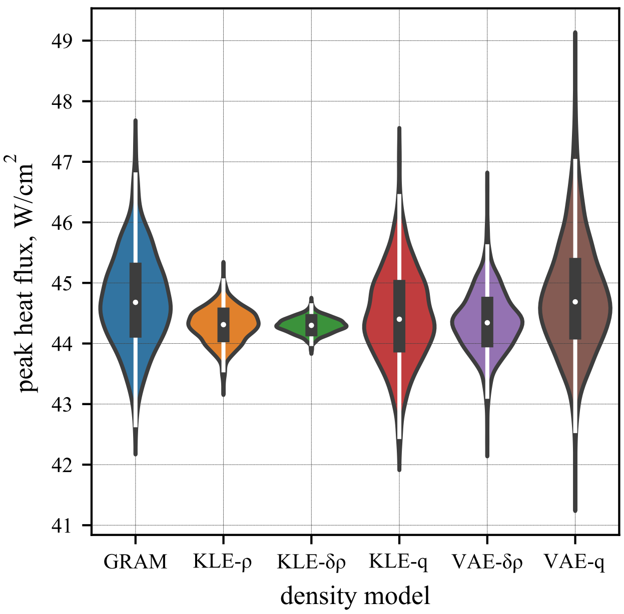
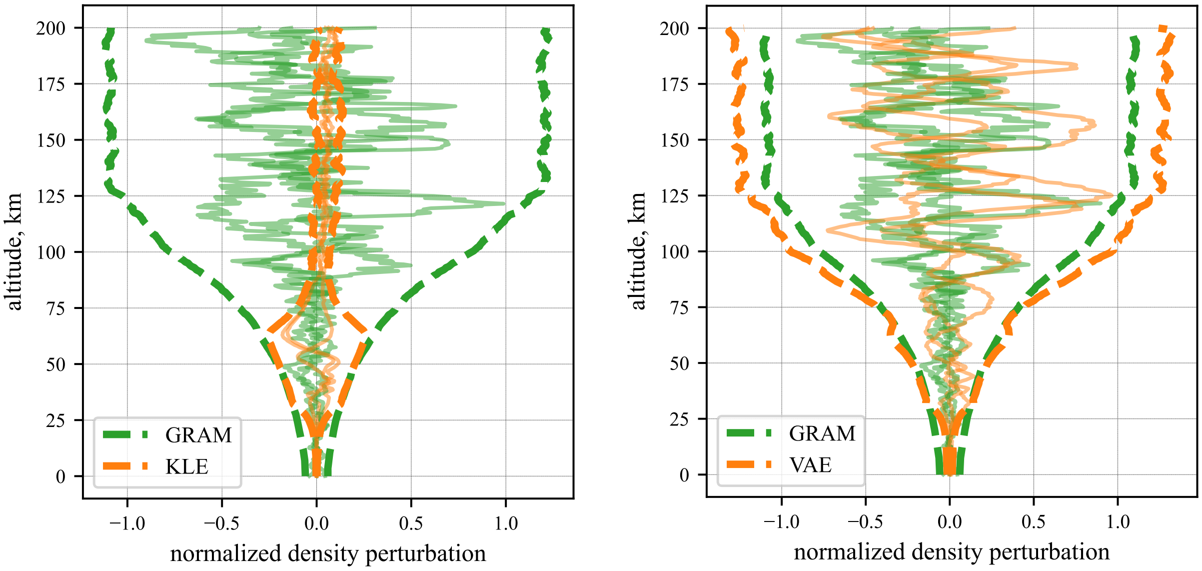


Figure 3.6: Peak heat flux statistics for SHIELD trajectories

A similar scaling approach can be applied based on the reference aerocapture trajectory. This process is slightly more involved because during aerocapture the vehicle passes through each relevant altitude twice, with differing dynamic pressures, and has a minimum altitude well above the surface, as seen in Fig. 3.1b. Recall, however, that the reference dynamic pressure is simply useful for re-scaling, and does not need to be dynamically valid. Thus, the following approach is taken in this study to form the reference dynamic pressure. Above the minimum altitude of the reference trajectory, the dynamic pressure during the descending portion of the trajectory is used for scaling. For another 10 km below the minimum altitude a constant value equal to the dynamic pressure at the minimum altitude is used; this segment exists because some dispersed trajectories will fly below the minimum altitude of the reference. Finally, a small but nonzero value (0.01 in this case) is used for scaling at more than 10 km below the minimum altitude of the reference trajectory. These values for q are then further modified according to Eq. (3.14) to obtain the $\boldsymbol{k}_q$ scaling vector for aerocapture. The density profiles predicted by the resulting models are summarized in Fig. 3.7, and the corresponding peak heat flux results for Monte Carlo analyses of the aerocapture trajectory are shown in Fig. 3.8.



(a) KLE vs. Mars-GRAM

(b) VAE vs. Mars-GRAM

Figure 3.7: Models constructed from normalized density perturbations scaled by aerocapture dynamic pressure profile; thick dashed lines are ±3σ bounds, thin solid lines are sample profiles

Overall these results are similar to the corresponding results for SHIELD direct-entry, in that the models capture density variation most efficiently near the altitude of peak dynamic pressure and the KLE/VAE-q models perform best when predicting peak heat flux statistics. The altitude range where the models accurately match the GRAM 3σ bounds is shifted up by about 10 km compared to the SHIELD case due to peak dynamic pressure occurring at a higher altitude for the

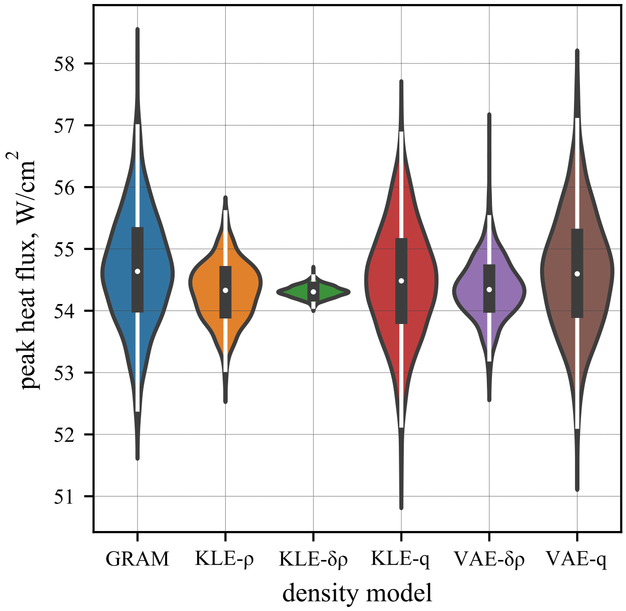


Figure 3.8: Peak heat flux statistics for aerocapture trajectories

aerocapture trajectory.

These results demonstrate that the models scaled based on reference dynamic pressure are the most compact representations of the possibilities considered here, as measured by the ability to predict statistics of peak heat flux. A relatively small number of terms ($d_K = 15$) is used for each KLE in order to highlight these differences and illustrate that some approaches are more compact than others. However, note that any of the KLE models should perform well if the number of included terms is sufficiently high, because the KLE representation of a GRF is exact for an infinite number of terms. Note that the patterns that have been discussed here are somewhat tied to the choice of random variable; because peak heat flux occurs at mid-altitudes near peak dynamic pressure, the KLE/VAE-q models will be particularly efficient in capturing those statistics. The most compact modeling approach, and the minimum dimensionality, thus somewhat depend on the particular quantities of interest.

Having shown good performance by both KLE and VAE models of uncertainty in a columnar atmosphere, a direct comparison of the two modeling approaches merits discussion. The VAE-

q models achieve slightly better performance than the KLE-q models, despite each VAE model having only four dimensions compared to 15 dimensions for each KLE model. The nonlinear generative modeling of the VAE appears to, in this case, enable a more compact model than the linear KLE modeling despite the approximately Gaussian nature of the training data. However, the setup process for the VAE modeling approach is significantly more involved. Obtaining good VAE results depends on careful tuning of neural network training parameters, which in general is only possible through trial and error, whereas there only exists one KLE model for a given set of input data and given expansion length. Moreover, as demonstrated in Section 3.6, updating a KLE model based on noisy measurements of density is much more straightforward than an equivalent measurement update would be for a VAE model. The benefits of the VAE modeling approach might be expected to outweigh those of the KLE method if the random field of interest were significantly non-Gaussian and sufficient samples of that field were available. Although it is still possible to construct a KLE model for non-Gaussian data, the expansion coefficients can become complex for generative sampling because the random variables Y_i are no longer i.i.d. [171]. In contrast, the VAE effectively uses the nonlinear transfer function defind by the decoder to absorb this complexity, keeping the distributions of the latent variables simple. For data-rich non-Gaussian fields, VAEs may thus offer advantages over KLEs, such as smaller latent variable dimensionality and a more accurate representation of the quantity of interest. However, for this particular application, in which the random process is approximately Gaussian, the KLE modeling approach has been shown to perform adequately well, and has the appealing quality of a one-to-one relationship between training data and model. Therefore, only KLE models are considered in subsequent sections, with equivalent contributions for VAE models left for future work.

3.5 Multi-Dimensional KLE Model

Although the columnar assumption is typical for onboard models of density as previously discussed, in some cases it may be of interest to represent density as a random function of multiple independent variables. The KLE approximation demonstrated in Section 3.4 can be straightforwardly extended to model longitudinal and latitudinal variations in density as well as in altitude. Thus, in this section the necessary steps for constructing a multi-dimensional KLE are presented, models are compared following the approach taken in Section 3.4, and finally there is a brief discussion of the potential utility of these models for onboard use.

Recall that the first step in forming a KLE approximation from some discrete dataset is computing the sample covariance matrix as shown in Eq. (3.8). The data matrix $\boldsymbol{\Psi}_c$ is formed such that each column is one observation vector with the sample mean subtracted. In the columnar KLE model, the observation vectors are ordered such that they correspond with a reference altitude vector. For the more general case, however, the indexing of the data matrix $\boldsymbol{\Psi}_c$ need not refer to a single independent variable. Rather, the index corresponds to a specific variable being observed, whether that be defined as density at 100 km or as density at 100 km, 20 ° E, and 40 ° N. Any arbitrary set of points in a multi-dimensional domain can be uniquely identified via sequential indexing, and then observations at these points can be reshaped into a column vector following that ordering; this process is conceptually illustrated in Fig. 3.9. The process of computing the covariance matrix and constructing and evaluating the KLE is unchanged. The original reshaping is then reversed to reshape the column vectors produced by realizations of the KLE to a set of values for each point in the multi-dimensional domain.

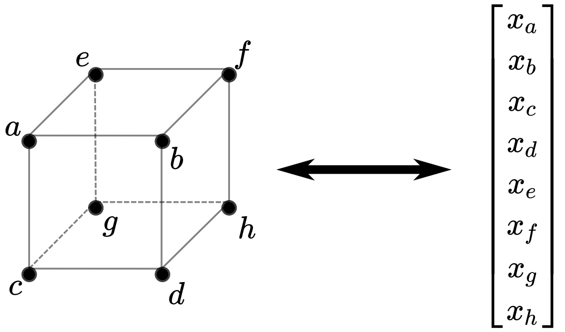
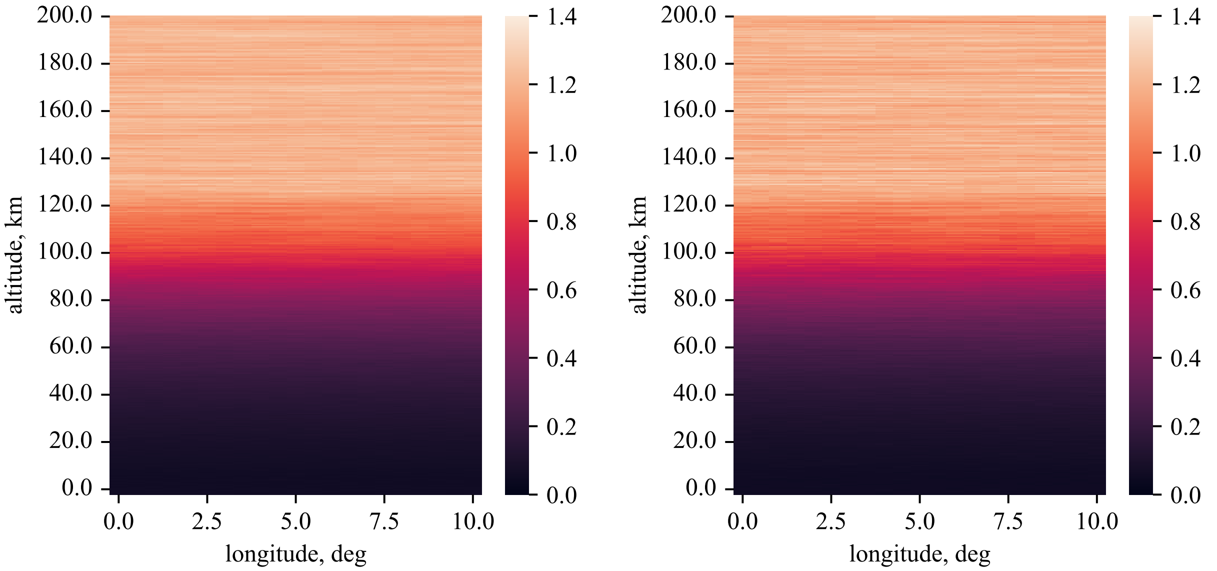


Figure 3.9: Illustration of reshaping between an arbitrary set of points in a multi-dimensional domain and an observation vector

As an example, MarsGRAM is used to generate 1000 density values at each point in an evenly-spaced 2D grid going from 0 to 200 km in altitude, from 0 to 10 ° in longitude, and at 0 ° latitude. Figure 3.10a visualizes the resulting data as a heatmap of the $+3\sigma$ value of $\delta\rho$; in other words, the heatmap values correspond to the right dashed line in figures like Fig. 3.2.



(a) Mars-GRAM

(b) KLE

Figure 3.10: 3σ values of normalized density perturbation for 2D density models

Figure 3.10b shows the equivalent statistics for a KLE approximation of the 2D MarsGRAM data; in contrast to Section 3.4, in this case a value of $\alpha = 0.99$ is used to truncate the KLE to $d_K = 884$ terms. From visual inspection, the results shown in Fig. 3.10 are virtually indistinguishable from each other.

As before, the real test of the KLE approximation is its ability to accurately predict trajectory dispersions. To this end, Figs. 3.11 and 3.12 show the peak heat flux statistics and a portion of the density profiles, respectively, resulting from 1000-trial Monte Carlo analyses of the same SHIELD direct-entry trajectory previously considered. In each case except GRAM 1D, bivariate spline approximation is used to compute density at the altitude and longitude of the vehicle based on a grid of density values. Recall that the reference SHIELD trajectory is ballistic and enters due-East, so the trajectory remains in the equatorial plane and thus, for this scenario, this approach is equivalent to computing density based on the 3D position of the vehicle. The GRAM 2D case interpolates from a set of density samples output by MarsGRAM directly, whereas the KLE $\alpha = 0.99$ case interpolates from values produced by a realization of an 884-term KLE approximation. The KLE $d_K = 50$ case also uses a KLE approximation, but in this case the expansion is limited to 50 terms. Finally, the GRAM 1D case interpolates from the same MarsGRAM data but always assumes a longitude of $0\,^\circ$, corresponding to a columnar assumption. This case should be exactly equivalent to the GRAM results shown for SHIELD in Fig. 3.6, but is slightly different. This occurs due to a quirk in how MarsGRAM density perturbations are computed. Thus, in this section the full 2D dataset is used but assuming a constant longitude of $0\,^\circ$ in order to create an apples-to-apples comparison.

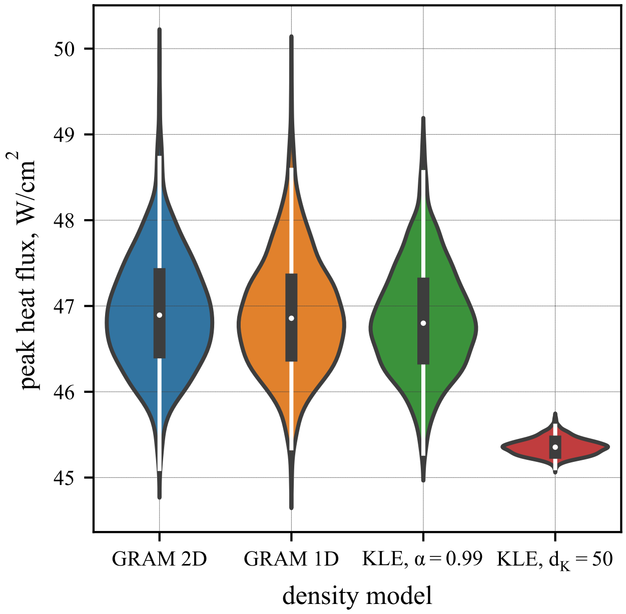


Figure 3.11: Peak heat flux statistics for SHIELD trajectories in 2D atmosphere models

From Fig. 3.11, it is clear that the peak heat flux statistics predicted by the 2D GRAM and 2D KLE ($\alpha = 0.99$) models are very similar, and Fig. 3.12 shows a characteristic similarity between the density profiles predicted by these two models. These results and the direct comparison of density values in Figs. 3.10a and 3.10b demonstrate the successful use of a multi-dimensional

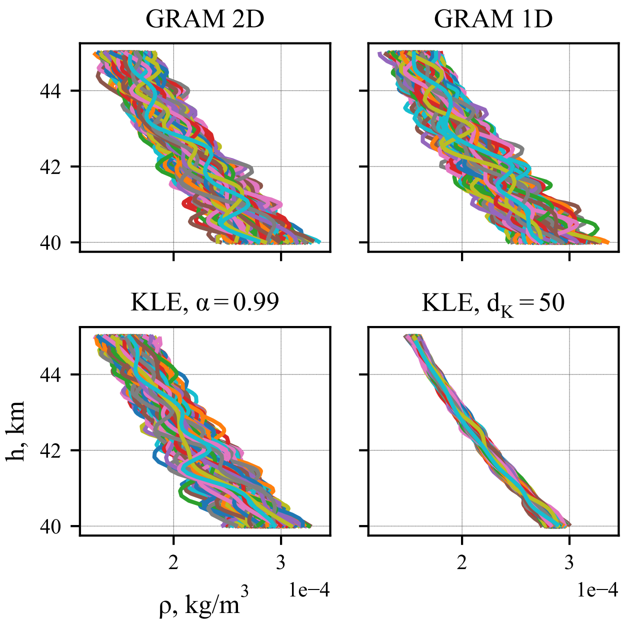


Figure 3.12: Density profiles on SHIELD trajectories for 2D density models

KLE to approximate density as a function of both altitude and longitude. In contrast, the 50-term KLE approximation performs very poorly, significantly under-predicting both the mean and uncertainty of peak heat flux. The expansion fails to capture much of the variability in density, as is clear from Fig. 3.12. The KLE $d_K = 50$ case performs worse than the KLE $\alpha = 0.99$ case because it has a much lower number of terms (50 vs. 884), and the expansion is truncated before sufficiently capturing the modes of variability present in the multivariate data.

These comparisons merit a broader discussion of the columnar atmosphere approximation for onboard density modeling. Figure 3.11 shows that the GRAM 1D case, which is equivalent to a columnar atmosphere assumption, almost exactly matches the 2D GRAM case in predictions of peak heat flux, and from Fig. 3.12 the sample density profiles themselves also appear to be very similar. This is not surprising when considering Fig. 3.10a, which shows no significant horizontal gradient to indicate changes in density variability with longitude. Note that, despite this uniformity in longitude, the KLE requires roughly 10x as many terms to accurately predict dispersed trajectories when constructed from the 2D density data as opposed to the columnar atmosphere case. This would require an increase in both memory and computational expense for onboard use. Furthermore, to sample across the entire 2D grid in altitude and longitude requires 8505 datapoints for the discretization used here, as compared to 405 datapoints for a columnar profile, further exacerbating the onboard computational burden. These results suggest that, based on the dataset used here, a columnar atmosphere model is likely a good enough approximation for onboard use, and is significantly less demanding of both memory and computational effort than a multi-dimensional model.

This is decidedly *not* to say that regional variations in density can be neglected. Density gradients occur due to a range of factors including gravity waves, time of day, and winds, and are relevant for both vehicle performance prediction and trajectory reconstruction [172, 173, 9]. MarsGRAM data is used in this study as an example only, and is not necessarily well-suited to capturing these types of regional density variation. Any hypersonic vehicle using closed-loop guidance would need to be simulated in a wide range of possible atmospheric conditions, regardless of the assumptions used for the onboard density model. The resulting vehicle performance, taken together with the relevant computational limitations, is ultimately what determines whether or not the onboard density model meets requirements.

Note also that, for a scenario where density is expected to change significantly along the groundtrack of an entry trajectory, a columnar model could be constructed using data generated along the reference trajectory. In other words, the raw data is generated along a 3D trajectory, but is then treated as a function of only altitude in the KLE approximation. This approach begins to fail if altitude is not monotonically decreasing, such as in the case of aerocapture. However, the procedure for onboard measurement updates presented in the next section would potentially result in different density predictions for the descending and ascending portions of the trajectory, and this could partly mitigate the limitations of a columnar model.

Finally, note that a VAE model may provide better dimensionality reduction than a KLE for the case of a density function varying across multiple dimensions. VAEs are well-suited for applications to complex, multi-faceted data including images and music [174, 175], and may do a superior job of recognizing the strong correlations between density profiles at different latitudes/longitudes/times, and then compressing the data based on these relationships. Applying a multi-dimensional VAE density modeling in scenarios where variation across dimensions other than altitude are important to trajectory prediction remains an interesting topic for future work.

3.6 Kalman Measurement Updates

During atmospheric flight, observations of estimated density $\rho^*(h_k)$ are typically available by taking estimated sensed accelerations measurements from an accelerometer or inertial measurement unit (IMU) and rearranging the equation for aerodynamic acceleration,

$$a(h_k) = \frac{v^2(h_k)}{2\beta}\rho(h_k) \rightarrow \rho^*(h_k) = \frac{2\beta a^*(h_k)}{v^2(h_k)}, \tag{3.15}$$

where estimates of the ballistic coefficient β and current velocity magnitude $v(h_k)$ are known. Thus, for any onboard density model to be useful in practice, it should accommodate some method of updating the model in real-time with noisy measurements. It is well-demonstrated in literature and in practice that appropriate onboard density estimation can significantly improve targeting performance [84].

The novel benefit of a KLE density model is the representation of both a nominal density profile and the associated uncertainty. Therefore, it is desirable to formulate an approach that updates both the mean and covariance represented by the KLE. Furthermore, this should be done in a way that respects the correlation structure assumed in the pre-update model, as opposed to replacing a single diagonal element of the covariance matrix. For clarity, this section returns to the columnar atmosphere assumption.

In this work a Bayesian approach for sequential estimation is applied, such that the mean and covariance of density from the previous update (or the initial model) form the prior, and these are updated with the noisy density measurement to form the posterior mean and covariance of density. The density estimates are assumed to be corrupted by additive white Gaussian noise, based on the assumption that some pre-processing removes artifacts such as IMU drift; note that this also implies accurate estimates for ballistic coefficient and velocity magnitude. The state uncertainty is also Gaussian based on the earlier assumption treating density as a Gaussian random process. Finally, density estimates are assumed to arrive at altitude points included in the original *a priori* density model, either by judiciously timing measurement updates or by interpolating multiple measurements.

Based on the above assumptions, density can be optimally estimated by the Kalman measurement update via the following formulation [176]. Take the series of density values at each altitude to be the state vector. The dynamic equation is trivial, because the density profile is assumed not to vary in time, so the state propagation step from the Kalman filter is unnecessary. The measurement equation is simply a direct observation of a single state component and is thus linear. Therefore, the optimal estimate of the vector of atmospheric density at each altitude $\hat{\boldsymbol{\rho}}^{+} \in \mathbb{R}^{n}$ and its covariance $\boldsymbol{P}_{k}^{+} \in \mathbb{R}^{n\times n}$ can be computed according to a scalar noisy density measurement $\rho_k^* \in \mathbb{R}$ according to the following equations:

$$\hat{\boldsymbol{\rho}}^{+} = \hat{\boldsymbol{\rho}}^{-} + \boldsymbol{K}(\rho_k^* - \boldsymbol{H}_k\hat{\boldsymbol{\rho}}^{-}), \tag{3.16}$$

$$\boldsymbol{P}^{+} = \boldsymbol{P}^{-} - \boldsymbol{K}\boldsymbol{H}_k\boldsymbol{P}^{-}, \tag{3.17}$$

$$\boldsymbol{K} = \boldsymbol{P}^{-}\boldsymbol{H}_k^{\intercal}\left(\boldsymbol{H}_k\boldsymbol{P}^{-}\boldsymbol{H}_k^{\intercal} + \boldsymbol{R}\right)^{-1}, \tag{3.18}$$

$$\boldsymbol{H}_k = [\delta_{1k}, \delta_{2k}, ..., \delta_{nk}], \tag{3.19}$$

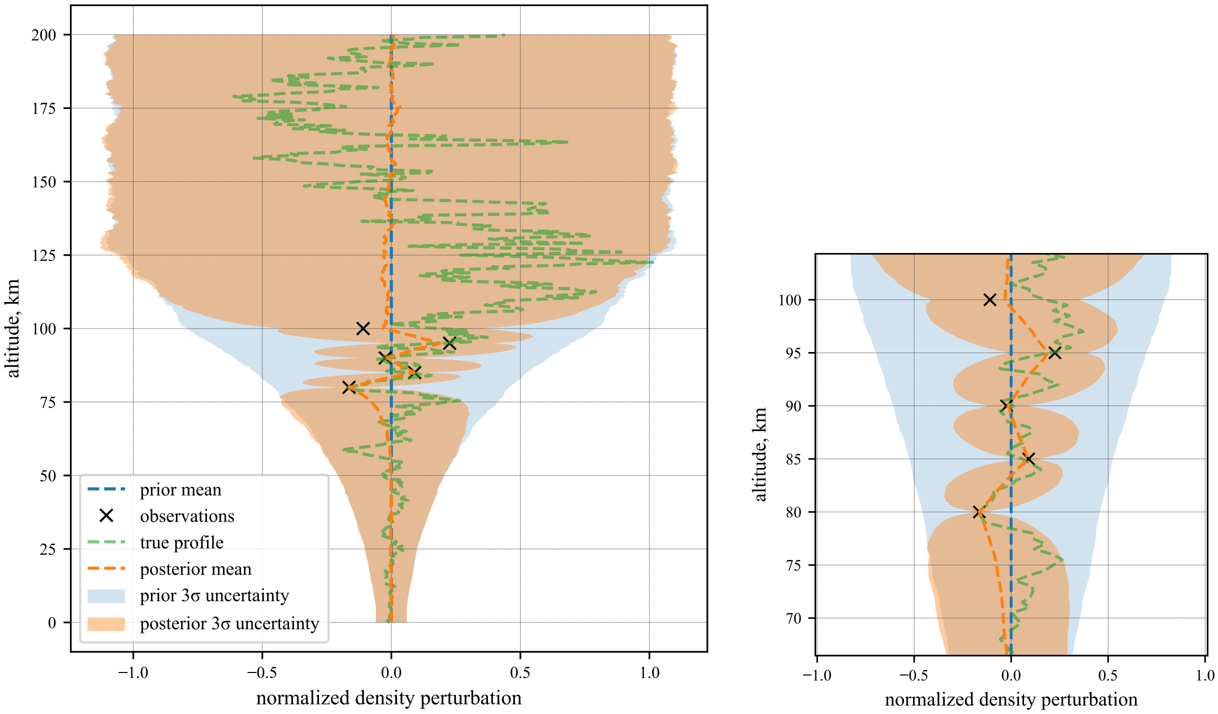
where $\boldsymbol{K} \in \mathbb{R}^{n\times 1}$ is the Kalman gain matrix, $\boldsymbol{H}_k \in \mathbb{R}^{1\times n}$ is the measurement matrix, $\boldsymbol{R} \in \mathbb{R}^{1\times 1}$ is the measurement noise covariance (generically a matrix, in this case a scalar), δ_{ij} is the Kronecker delta, n is the number of discrete altitudes considered, and k is the index of the altitude at which density is currently being observed. Notably, because only one density is measured at a time the bracketed term in Eq. (3.18) is a scalar, so taking its inverse is computationally inexpensive.

For notational clarity, consider an example where the discretization of density values is from 100 to 0 km in altitude steps of 0.5 km, in descending order, resulting in $n = 201$. Then $\hat{\boldsymbol{\rho}}^{-}$ and $\hat{\boldsymbol{\rho}}^{+}$ are the prior and posterior 201-vectors, respectively, containing density values at each altitude. Assume the scalar density measurement ρ_k^* is at an altitude of 80 km, such that $k = 41$ (indexing from 1 in this notation). Then, $\boldsymbol{H}_k$ becomes a row matrix with all elements equal to zero except in the 41st column, which is equal to one.

Equations (3.16) – (3.19) can be applied to sequentially ingest noisy density measurements and update the onboard model of the density profile and its covariance. By re-solving for the eigenvalues and eigenvectors of the $\boldsymbol{P}^+$, the KLE representation can be updated accordingly. This process is demonstrated in Figs. 3.13 and 3.13b; here, the prior mean and covariance are formed from a dataset of 3000 density profiles from MarsGRAM, where density perturbations are normalized by the sample mean and thus the normalized prior mean falls exactly along 0. The true profile to be estimated is also computed by MarsGRAM, but is not included in the prior dataset. Five density values are observed, corrupted by measurement noise with a standard deviation of $1 \times 10^{-9}\,\mathrm{kg/m^3}$, a value selected purely for illustrative purposes. In this example the assumed measurement noise R is equal to the true noise value, but note that this can instead be treated as a tuning parameter in practice and need not be the same value at each altitude.

Note that the posterior mean passes nearly through each observation (with one exception), but reverts to the mean for altitudes above and below the observation altitudes. The posterior uncertainty bounds are also only weakly affected at these higher and lower altitudes. This occurs because the correlation structure in the prior covariance dictates the degree to which new information at one altitude affects the estimated density at another altitude. Because in this model density perturbation at 80 km is only weakly correlated with density perturbation at 50 km, the posterior mean has reverted to nominal by that lower altitude. This can also be achieved by onboard estimation of a corrective scale factor that is then exponentially decayed back to unity for altitudes not near the measurement. However, the approach presented here has two advantages. First, the correlation length is inferred from the prior model (MarsGRAM in this case) rather than defined by the user, and second, the correlation length is not necessarily constant with altitude.

The reason that the posterior mean passes more closely through the lower three measurements than the first two is related to how measurement noise was defined. Measurement noise is applied to the density values directly and is constant across all altitudes, but the data is then converted to



(a) Full profile

(b) Close-up from Fig. 3.13a

Figure 3.13: Mean and 3σ bounds for prior and posterior density profiles, given five sequential noisy observations

normalized density perturbations for estimation and visualization. Thus, at higher altitudes where nominal density is significantly lower, the measurement noise has a more significant effect, and the filter tends to trust the prior. This is also observable by the much wider posterior uncertainty bounds for the higher-altitude measurements. At lower altitudes the same measurement noise has relatively less effect and the situation is reversed; because the actual measurement noise and the value used by the filter are the same, this also means the lower-altitude measurements fall closer to the true values. It should be reiterated that the assumption of a measurement noise constant with altitude is made here for demonstration purposes and is not required.

The application of a Kalman measurement update demonstrated here provides a way of updating the mean and covariance for atmospheric density based on noisy measurements, which could inform onboard predictions of state uncertainty for the purpose of closed-loop guidance. A significant drawback of this approach, however, is the requirement to re-solve the eigenvalues and eigenvectors after each measurement update in order to obtain the updated KLE representation. This adds significant computational expense to the update process, potentially to the point of infeasibility for onboard computation, depending on the resolution of the density profile and the choice of flight hardware. This motivates an approach that updates the eigenvectors and eigenvalues directly in a way that approximates the result of the Kalman measurement update at a lower computational expense. Such a method could take advantage of the fact that there is approximately zero covariance between altitudes more than a certain distance apart. Approaches such as low-rank partial Hessian approximations or sequential updates to singular value decompositions of a matrix provide potential pathways to significant computational efficiency improvement [177, 178]; this remains an area for future work. Another avenue for future work would be a method for onboard upating of a VAE density model based on noisy measurements, without requiring onboard retraining of the model. In this case, the model would be trained on the ground and then conditioned on noisy measurements in flight, permitting computationally-efficient updates to the VAE.

3.7 Linear Covariance Analysis

This subsection summarizes another application of the KLE density model, linear covariance analysis of guided aerocapture and entry trajectories in an uncertain atmosphere. Note that this work was originally presented in reference [146], for which S. W. Albert was second author. Linear covariance analysis approximates the uncertainty evolution of a nonlinear system by propagating the mean and covariance of the linearized system. By including KLE terms as uncertain parameters, the evolution of state covariance is approximated and shown to closely match the estimate provided by a Monte Carlo analysis.

Consider a nonlinear dynamical system with state $x \in \mathbb{R}^n$ acting under the influence of q uncertain parameters $p_0 \in \mathbb{R}^q$ according to the dynamics

$$\dot{x} = f(t, x, u(t, x), p_0) = f_{\mathrm{cl}}(t, x, p_0), \quad x(t_0) = x_0, \tag{3.20}$$

where $u(t, x)$ is a closed-loop control. For generality, let the initial state x_0 be included as an uncertain parameter and define the new $\ell = n + q$ dimensional parameter vector p as the concatenation

$$p = \begin{bmatrix} x_0 \\ p_0 \end{bmatrix} \in \mathbb{R}^{\ell}. \tag{3.21}$$

The following analysis, which is adapted from Ref. [179] Ch. 3, is concerned with approximating variations in trajectories of the system (3.20) as linear functions of variations of the parameter vector p.

Let $x(t, p)$ be the solution to (3.20) for a particular realization of the parameter vector p, which is given as

$$x(t, p) = x_0 + \int_{t_0}^{t} f_{\text{cl}}\big(\tau, x(\tau, p), p_0\big)\, \mathrm{d}\tau. \tag{3.22}$$

Taking the partial derivative of the trajectory $x(t, p)$ with respect to the parameter p, obtain

$$\frac{\partial x}{\partial p}(t, p) = \begin{bmatrix} I_n & 0_{n\times q} \end{bmatrix} + \int_{t_0}^{t} \left\{ \frac{\partial f_{\text{cl}}}{\partial x}\big(\tau, x(\tau, p), p_0\big) \frac{\partial x}{\partial p}(\tau, p) + \frac{\partial f_{\text{cl}}}{\partial p}\big(\tau, x(\tau, p), p_0\big) \right\} \mathrm{d}\tau. \tag{3.23}$$

Next, approximate the expression (3.23) about a given nominal parameter value $\bar{p} = (\bar{x}_0, \bar{p}_0)$. Define the matrix-valued functions of time

$$S(t) = \frac{\partial x}{\partial p}(t, \bar{p}), \quad A_{\text{cl}}(t) = \frac{\partial f_{\text{cl}}}{\partial x}\big(t, x(t, \bar{p}), \bar{p}_0\big), \quad C(t) = \frac{\partial f_{\text{cl}}}{\partial p}\big(t, x(t, \bar{p}), \bar{p}_0\big) = \begin{bmatrix} 0_n & \dfrac{\partial f_{\text{cl}}}{\partial p_0}\big(t, x(t, \bar{p}), \bar{p}_0\big) \end{bmatrix}. \tag{3.24}$$

The matrix $S(t)$ is known as the *sensitivity function*, since the trajectory $x(t, p)$ can be approximated to first order as

$$x(t, p) \approx x(t, \bar{p}) + S(t)(p - \bar{p}). \tag{3.25}$$

Furthermore, from (3.23), the sensitivity function is obtained as the solution to the ODE

$$\dot{S}(t) = A_{\text{cl}}(t) S(t) + C(t), \quad S(t_0) = \begin{bmatrix} I_n & 0_{n\times q} \end{bmatrix}. \tag{3.26}$$

Suppose that the parameter vector p_0 is Gaussian distributed as $p_0 \sim \mathcal{N}(\bar{p}_0, P_0)$. If the initial state x_0 is uncorrelated with the parameters p_0 and is also Gaussian distributed with covariance matrix X_0, then the parameter p is also Gaussian distributed as

$$p \sim \mathcal{N}(\bar{p}, P), \quad \text{where} \quad \bar{p} = \begin{bmatrix} \bar{p}_0 \\ \bar{x}_0 \end{bmatrix}, \quad P = \begin{bmatrix} X_0 & \\ & P_0 \end{bmatrix}, \tag{3.27}$$

It then follows from the sensitivity equation (3.25) that the state $x(t, p)$ is approximately Gaussian distributed with mean $\bar{x}(t) = x(t, \bar{p})$ and covariance

$$X(t) = S(t) P S^{\intercal}(t). \tag{3.28}$$

In summary, the state distribution can be approximated to first order about a nominal trajectory $\bar{x}(t)$ by the following procedure: Integrate the nominal trajectory $\bar{x}(t)$ from (3.20) with $p_0 = \bar{p}_0$; compute the matrices $A_{\text{cl}}(t)$ and $C(t)$ as functions of $\bar{x}(t)$ as in (3.24); integrate the matrix-valued ODE (3.26); and, finally, compute the state covariance from (3.28).

Now take a dynamical system that depends on a GRF Ψ, which is approximated by a q-term KLE Ψ_q:

$$\dot{x} = f_\Psi(t, x, \Psi(z(x))) \approx f_\Psi(t, x, \Psi_q(z(x))) = f_{\text{cl}}(t, x, p_0), \tag{3.29}$$

where the argument z of the field Ψ depends on the state x, and where $p_0 = (w_1, \ldots, w_q) = (Y_1\sqrt{\lambda_1}, \ldots, Y_q\sqrt{\lambda_q})$ are the KLE coefficients in Eq. (3.9). The partial derivatives of the dynamical system with respect to the uncertain parameters thus depend on the basis functions φ_i as:

$$\frac{\partial f_{\text{cl}}}{\partial p_0} = \frac{\partial f_\Psi}{\partial \Psi_q} \frac{\partial \Psi_q}{\partial p_0}, \tag{3.30}$$

where the partials

$$\frac{\partial \Psi_q(z)}{\partial p_0} = \begin{bmatrix} \dfrac{\partial \Psi_q(z)}{\partial w_1} & \cdots & \dfrac{\partial \Psi_q(z)}{\partial w_q} \end{bmatrix} \quad \text{and} \quad \frac{\partial \Psi_q(z)}{\partial w_i} = \varphi_i(z), \tag{3.31}$$

are evaluated at the nominal values $z = z(\bar{x}(t))$.

The method of linear covariance approximation in a GRF is summarized as in Algorithm 1 [146]. In order to compute the sensitivity matrix and perform linear covariance analysis for the closed-loop dynamical system, the matrices $A_{cl}(t)$ and $C(t)$ must be derived according to (3.24).

Algorithm 1 Linear covariance approximation procedure

```
Compute a $q$-term KLE model of $\Psi$ by solving for the eigenvalues and eigenfunctions of Eq. (3.4).
Obtain a nominal trajectory $\bar{x}(t)$ for $t \in [t_0, t_f]$.
Compute the matrices $A(t)$, $B(t)$, and $C(t)$ as in Eq. (3.24).
Integrate the sensitivity equation (Eq. (3.26)) from $t_0$ to $t_f$.
Obtain state covariance $X(t)$ from Eq. (3.28).
```

The control input u, which is taken to be the cosine of the bank angle $u = \cos\sigma$, is assumed to follow the linear feedback law

$$u(t, x) = \bar{u}(t) + K(t)\big(x - \bar{x}(t)\big), \tag{3.32}$$

for a given feedback gain matrix $K(t)$. The closed-loop matrix $A_{cl}(t)$ can then be expressed as

$$A_{\text{cl}}(t) = \frac{\partial f_{\text{cl}}}{\partial x} = \frac{\partial f}{\partial x} + \frac{\partial f}{\partial u}\frac{\partial u}{\partial x} = A(t) + B(t)K(t), \tag{3.33}$$

where the matrices $A(t)$ and $B(t)$ are evaluated along the reference trajectory $\bar{x}(t)$ and $\bar{u}(t)$. The matrices $A(t)$, $B(t)$, and $C(t)$ are provided for this dynamical system in Appendix B. This procedure is applied to two numerical examples, Mars direct entry and Mars aerocapture, with results summarized in the following subsections.

3.7.1 Guided Mars Entry

3.7.1.1 Problem Definition

Consider a Mars Science Laboratory (MSL)-like vehicle performing a guided entry at Mars. The vehicle lift-to-drag ratio is $E = 0.24$, the ballistic coefficient is $\beta = 130$ kg/m^2, and Mars is assumed to have gravitational parameter $\mu_{\text{grav}} = 4.2828 \times 10^{13}$ m^3/s^2 and surface radius $r_p = 3397$ km; these parameters are listed in Table 3.1. At the initial time $t_0 = 0$ the vehicle is nominally at an altitude of 125 km with planet-relative velocity 5.8 km/s and flight path angle $-15.5°$. The vehicle state error from these nominal values is Gaussian distributed such that the 3σ errors of velocity, flight path angle, and downrange distance are 20 m/s, 0.5°, and 5 km, respectively; the initial altitude is assumed to be exactly 125 km, by definition of the initialization condition at entry interface. Thus the initial state is Gaussian distributed as

$$x_0 \sim \mathcal{N}(\bar{x}_0, P_0), \quad \text{where} \quad \bar{x}_0 = \begin{bmatrix} 125\,\mathrm{km} + r_p \\ 5.8\,\mathrm{km/s} \\ -15.5^\circ \\ 0 \end{bmatrix}, \quad P_0 = \begin{bmatrix} 0 & & & \\ & (20\,\mathrm{m/s}/3)^2 & & \\ & & (0.5^\circ/3)^2 & \\ & & & (5\,\mathrm{km}/3)^2 \end{bmatrix}. \tag{3.34}$$

Both the nominal and samples of the dispersed atmospheric density are provided by Mars-GRAM 2010. The nominal bank angle is set as a piecewise-linear function of velocity, with the nodes

$$\begin{array}{c|cccccc} \cos^{-1}(\bar{u}) & 70^\circ & 70^\circ & 45^\circ & 45^\circ & 10^\circ & 10^\circ \\ \hline \bar{V} & 6\,\mathrm{km/s} & 5.5\,\mathrm{km/s} & 2.5\,\mathrm{km/s} & 1.1\,\mathrm{km/s} & 1\,\mathrm{km/s} & 0\,\mathrm{km/s} \end{array}. \tag{3.35}$$

The resulting nominal entry trajectory is shown in Figure 3.14. Closed-loop range control is provided by the Apollo final phase guidance algorithm [40, 180], which is described in the following.

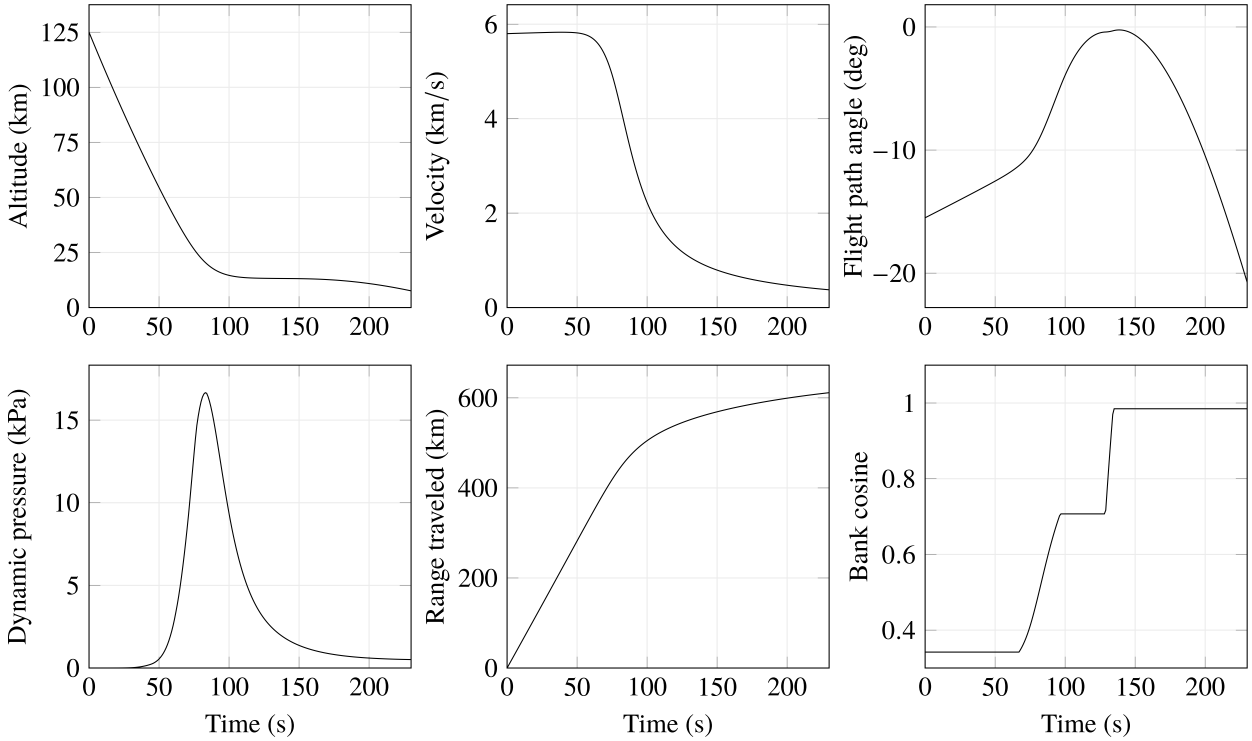


Figure 3.14: Nominal entry trajectory

Table 3.1: Vehicle and planetary parameters for entry and aerocapture examples

Parameter	Value
Lift-to-Drag ratio, E	0.24
Ballistic Coefficient, β	130 kg/m^2
Gravitational Parameter, μ_{grav}	4.2828×10^{13} m^3/s^2
Surface Radius, r_p	3397 km

3.7.1.2 Apollo Final Phase Guidance

Let $f(t, x, u, p_0)$ be the right hand side of the equation (**??**), with control $u = \cos\sigma$, and define the system matrices

$$A(t) = \frac{\partial f}{\partial x}\big(\bar{x}(t), \bar{u}(t), 0\big), \qquad B(t) = \frac{\partial f}{\partial u}\big(\bar{x}(t), \bar{u}(t), 0\big), \tag{3.36}$$

evaluated along the nominal trajectory $\bar{x}(t)$, nominal control $\bar{u}(t)$, and with nominal density $\bar{\rho}(h)$ (i.e., $\bar{p}_0 = 0$). The adjoint state (λ, λ_u) to the system (**??**) is defined as the solution to the backwards ODE

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix}\lambda(t)\\ \lambda_u(t)\end{bmatrix} = -\begin{bmatrix}A^{\mathsf{T}}(t) & 0\\ B^{\mathsf{T}}(t) & 0\end{bmatrix}\begin{bmatrix}\lambda(t)\\ \lambda_u(t)\end{bmatrix}, \qquad \begin{bmatrix}\lambda(t_f)\\ \lambda_u(t_f)\end{bmatrix} = \begin{bmatrix}\lambda_f\\ 0\end{bmatrix}, \tag{3.37}$$

where the boundary value λ_f is a user-defined vector determining the relative effects of the final states on the final range error. For the Apollo final phase algorithm, this boundary value is set to

$$\lambda_f = \begin{bmatrix}-\cot\bar{\gamma}(t_f) & 0 & 0 & 1\end{bmatrix}^{\mathsf{T}}, \tag{3.38}$$

and the state feedback gain is defined in terms of the adjoint values as

$$K(t) = -K_{\text{oc}}\frac{\lambda^{\mathsf{T}}(t)}{\lambda_u(t)}, \tag{3.39}$$

where K_{oc} is a user-defined overcontrol gain. In this example, we set $K_{\text{oc}} = 4$. Furthermore, we assume that the (range) control effect is zero during the heading alignment phase, which begins when the vehicle velocity decreases below 1.1 km/s. Thus the control matrix is set to $B(t) = 0$ when $\bar{V}(t) \leq 1.1$ km/s.

The closed-loop bank angle cosine is thus given by the linear feedback law (3.32). In practice, the nominal control $\bar{u}$, feedback gain K, and reference trajectory $\bar{x}$ are all set as functions of velocity. But, for the purposes of linear covariance analysis, we assume these reference values are set as functions of time. The closed-loop, linearized system is thus described by the state matrix in (3.33).

3.7.1.3 Results

The closed-loop entry trajectory dispersions, due to both the initial state uncertainty and the MarsGRAM-generated density variations, are computed using two methods: Monte Carlo, for which 5,000 sample trajectories are integrated, each with a fixed MarsGRAM density profile sample; and by linear covariance (LC) analysis, using a $q = 50$ dimensional KL representation of the density profile. Sample Monte Carlo trajectories together with 3σ bounds as computed by both the Monte Carlo and from LC are shown in Figure 3.15. The 3σ bounds from LC approximation is almost exactly equal to the bounds computed from Monte Carlo.

3.7.2 Mars Aerocapture

The same MSL-like vehicle performs an aerocapture trajectory at Mars. For this scenario, the desired final orbit is circular at $2,000$ km altitude.

For the aerocapture scenario the vehicle parameters, Mars properties, and atmospheric flight dynamics are all identical to the entry scenario. The nominal initial altitude, planet-relative velocity, and downrange distance are also identical to the entry case, with a shallower entry flight path angle of -9.8°. Smaller dispersions on the initial state are used for the aerocapture case, such that they are Gaussian distributed about the nominal values with 3σ errors of 10 m/s and 0.2° for velocity and flight path angle, respectively. Downrange distance is not particularly relevant to longitudinal aerocapture dynamics so is not dispersed, and initial altitude is again assumed to be exactly 125

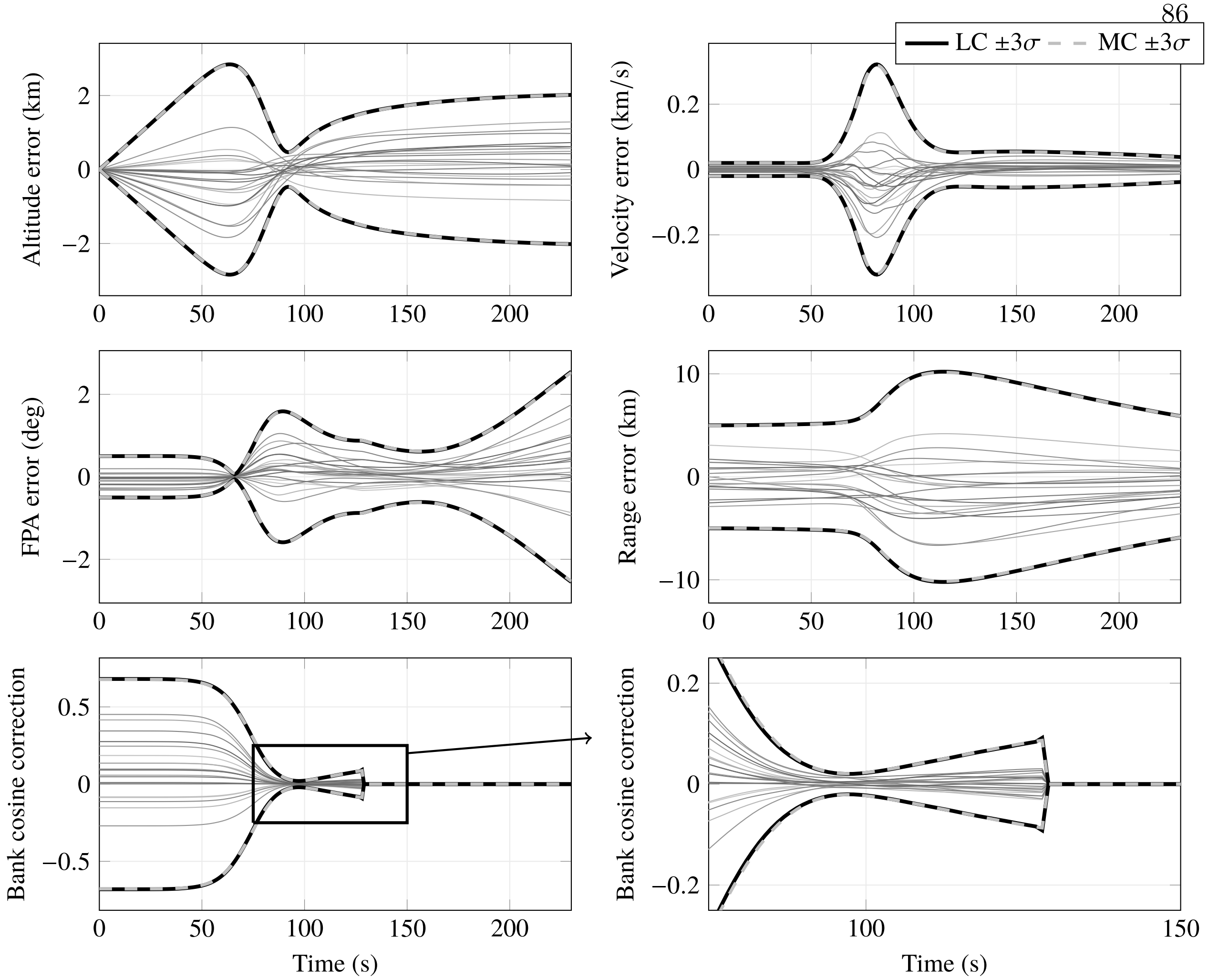


Figure 3.15: Sample closed-loop entry trajectories with 3σ bounds computed from both 5,000 trial Monte Carlo (MC) and linear covariance (LC).

km. Thus for aerocapture the initial state is Gaussian distributed as

$$x_0 \sim \mathcal{N}(\bar{x}_0, P_0), \quad \text{where} \quad \bar{x}_0 = \begin{bmatrix} 125\,\text{km} + r_p \\ 5.8\,\text{km/s} \\ -9.8^\circ \\ 0 \end{bmatrix}, \quad P_0 = \begin{bmatrix} 0 & & & \\ & (10\,\text{m/s}\,/3)^2 & & \\ & & (0.2^\circ/3)^2 & \\ & & & 0 \end{bmatrix}. \quad (3.40)$$

As a point of reference, MSL required entry flight path angle delivery within $3\sigma = 0.2^\circ$ and entry velocity knowledge of $3\sigma = 2.0$ m/s [181]. Mars-GRAM 2010 was again used for the nominal and dispersed atmospheric density profiles.

The nominal bank angle profile is assumed to have a bang-bang form with a single transition from lift-up to lift-down during the flight. To provide margin for feedback, the vehicle has an initial bank angle of $\sigma = 85^\circ$ from entry until some switching time t_s, then linearly increases the bank angle over a duration of 30 sec until reaching a final bank angle of $\sigma = 115^\circ$, and finally the bangle angle $\sigma = 115^\circ$ is held until atmospheric exit. The switching time t_s is solved by a foot-finding procedure so that the apoapsis after atmospheric exit equals a desired value. For this problem, the switching time was found to be $t_s = 114.9$ sec to meet a target apoapsis of $2,000$ km, and the resulting nominal trajectory is described by Figure 3.16.

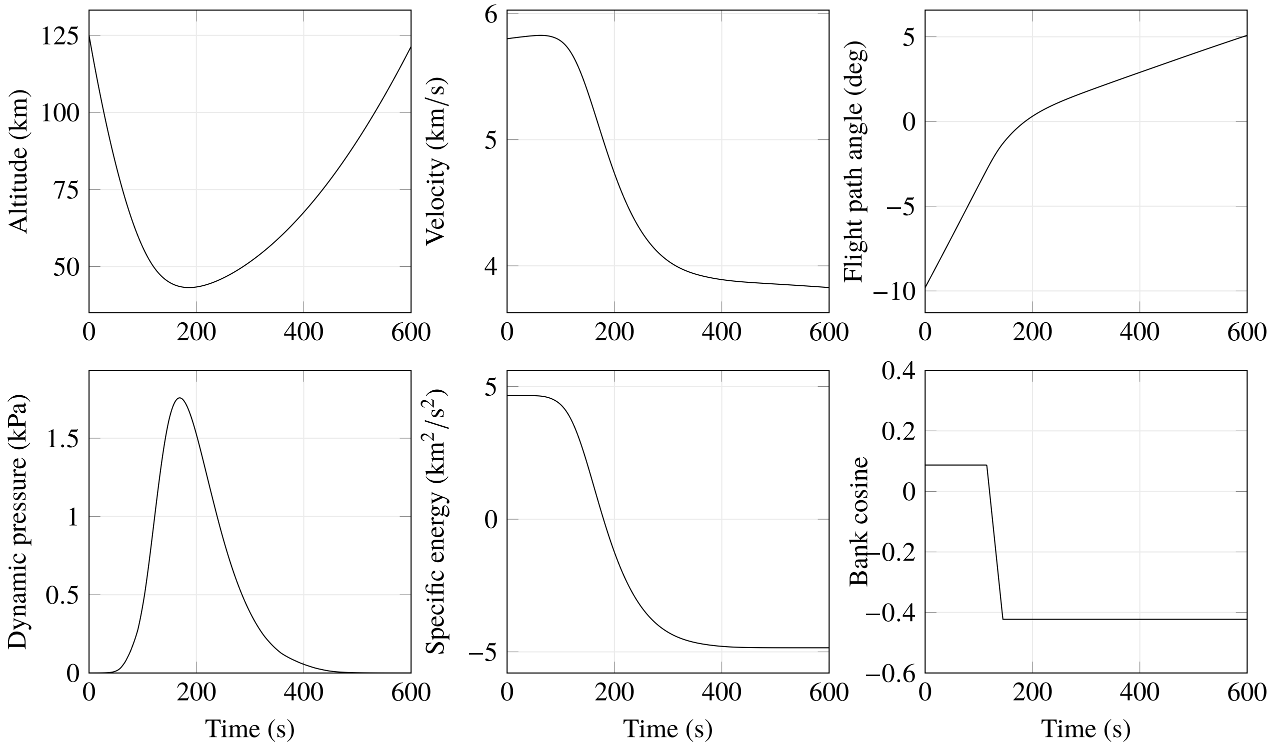


Figure 3.16: Nominal areocapture trajectory

In many ways aerocapture is the same as guided entry but with a different final objective, namely, targeting a Keplerian orbital state at atmospheric exit rather than a final range. Thus, we adapt the Apollo final phase guidance algorithm for aerocapture. This method of terminal point controller guidance for aerocapture is well-studied [72, 73]; the particular implementation used in this work is briefly reviewed here.

For this study, the closed-loop guidance during atmospheric flight is designed to target the desired apoapsis after atmospheric exit r_a, which is given as a function of the vehicle state $x_f = (r_f, V_f, \gamma_f, R_f)$ at atmospheric exit by

$$r_a = \frac{|\boldsymbol{h}_f|}{V_a}, \tag{3.41}$$

where $\boldsymbol{h}_f$ is the specific angular momentum and V_a is the velocity at apoapsis, which are given by

$$V_a = \frac{\mu_{\text{grav}} - \sqrt{\mu_{\text{grav}}^2 + 2\varepsilon_f |\boldsymbol{h}_f|^2}}{|\boldsymbol{h}_f|}, \qquad |\boldsymbol{h}_f| = r_f V_f \cos\gamma_f, \tag{3.42}$$

where,

$$\varepsilon_f = \frac{V_f^2}{2} - \frac{\mu_{\text{grav}}}{r_f} \tag{3.43}$$

is the specific energy. Note that in (3.41)-(3.43) the states are inertial, not planet-relative; when using the simplified longitudinal dynamics in (**??**), which assume a nonrotating spherical planet, the inertial and planet-relative states become identical.

After the atmosphere pass, two maneuvers are required to ensure the spacecraft reaches the desired final orbit. First, a periapsis raise maneuver is performed at first apoapsis along the velocity direction and with magnitude ΔV_1; this maneuver has some nonzero nominal value because initially the periapsis will be below the atmospheric interface altitude. Second, an apoapsis correction maneuver is performed at periapsis (at its new altitude) in either the posigrade (to raise apoaisis) or retrograde (to lower apoapsis) direction and with magnitude ΔV_2. Nominally $\Delta V_2 = 0$, but the value of ΔV_2 is uncertain as this maneuver corrects for any apoapsis error following the atmospheric pass. Lateral dynamics, guidance, and a plane correction maneuver are all neglected for the purpose of this study. The magnitudes of these maneuvers can be computed as shown, where in this study the target orbit is assumed to be circular at some radius r_c (the equations are readily modified to eliminate this assumption). The magnitudes of the velocity at apoapsis after the first maneuver V_1 and the velocity at periapsis before the second maneuver V_2 are given by

$$V_1 = \sqrt{\frac{2\mu_{\text{grav}} r_c}{r_a(r_a + r_c)}} \qquad V_2 = \sqrt{\frac{2\mu_{\text{grav}} r_a}{r_c(r_a + r_c)}}, \tag{3.44}$$

where V_c is the circular velocity at the radius r_c, given by

$$V_c = \sqrt{\frac{\mu_{\text{grav}}}{r_c}}. \tag{3.45}$$

Finally, the total maneuver cost ΔV is computed as the sum

$$\Delta V = \Delta V_1 + \Delta V_2 = (V_1 - V_a) + |V_c - V_2|. \tag{3.46}$$

Because of the absolute value sign in the expression for ΔV_2, the partial derivatives become undefined at the nominal value $\Delta V_2 = 0$. Therefore, in this work only ΔV_1 is linearly predicted.

The aerocapture guidance algorithm consists of integrating the same dynamics for the adjoint state (λ, λ_u) using the same open-loop system matrices $A(t)$ and $B(t)$ evaluated along the nominal aerocapture trajectory. The state feedback gain matrix $K(t)$ is also computed the same way and user-defined overcontrol gain is again used, this time with a value $K_{\text{oc}} = 3$. The first of two differences in the guidance is that the control is active until $t = 240$ sec, at which point the feedback control is set to zero, i.e. $B(t) = 0$ when $t > 240$ sec. This time was selected to correspond approximately to when the energy stops decreasing in the reference trajectory, and was set so that the Apollo guidance would remain well-behaved with minimal modifications.

The second difference between the aerocapture and entry guidance implementations is the way the boundary value λ_f is computed. Following the terminal control theory, the terminal condition is set equal to the partial derivative of a performance index $\Theta(t)$ with respect to the state, evaluated at the final time [182]. In Ref. [[72]], total ΔV is used as the performance index; in this study we use radius of apoapsis error, where the target apoapsis radius r_c falls out of the partial derivative given by

$$\lambda_f^{\mathsf{T}} = \frac{\delta\Theta(t_f)}{\delta x(t_f)} = \left(\frac{\delta r_a}{\delta x}\right)_{t=t_f}. \tag{3.47}$$

The construction of this control law implicitly assumes a constant bank angle [72] (even though the reference bank profile is not necessarily constant), and therefore apoapsis targeting is a nearly-equivalent proxy for ΔV optimization; a difference in the two solutions would only be expected for steep entry flight path angles [65]. The partial derivatives of apoapsis radius, apoapsis velocity, and total ΔV, each with respect to the state, are provided in the appendix. The aerocapture closed-loop guidance algorithm is implemented as in (3.36), (3.37), and (3.39)-(3.33), replacing the boundary value in (3.38) with the value for apoapsis targeting in (3.47).

3.7.2.1 Results

As with the guided entry example in Sec. 3.7.1, dispersions are estimated using both a 5,000-trial Monte Carlo analysis and a linear covariance analysis using a $q = 50$ dimensional KL representation of density variability. The trajectory dispersions are compared in Figure 3.17. Additionally, histograms of the Monte Carlo results for apoapsis altitude, velocity at apoapsis, and total ΔV are shown in Figure 3.18 with a Gaussian-fit probability density function estimated from the linear covariance analysis superimposed.

3.7.3 Discussion

The numerical examples show a close match between the Monte Carlo estimates and linear covariance approximations, as seen by the plots of standard deviation over time in Figures 3.15 and 3.17. This suggests that the implemented control laws keep the dispersed trajectories close enough to the reference for the linearization to remain accurate, and the linear feedback nature of these control laws enables estimating the full closed-loop system. It also suggests that the KL expansion of density models the MarsGRAM density variability well enough to make accurate predictions of this dynamical system. The aerocapture numerical example demonstrates how these predictions can be translated into performance metrics, such as a histogram of apoapsis targeting or 99th-percentile value of total ΔV, ΔV_{99}.

The main purpose of these two numerical examples was to show that the Monte Carlo and linear covariance analysis predictions matched, and this has been achieved. A next step would be removing some of the simplifying assumptions regarding the dynamics and guidance algorithms to implement this linear prediction in a more realistic simulation. Planetary rotation and nonspherical gravity terms were neglected and, for aerocapture in particular, these can be important effects. Lat-

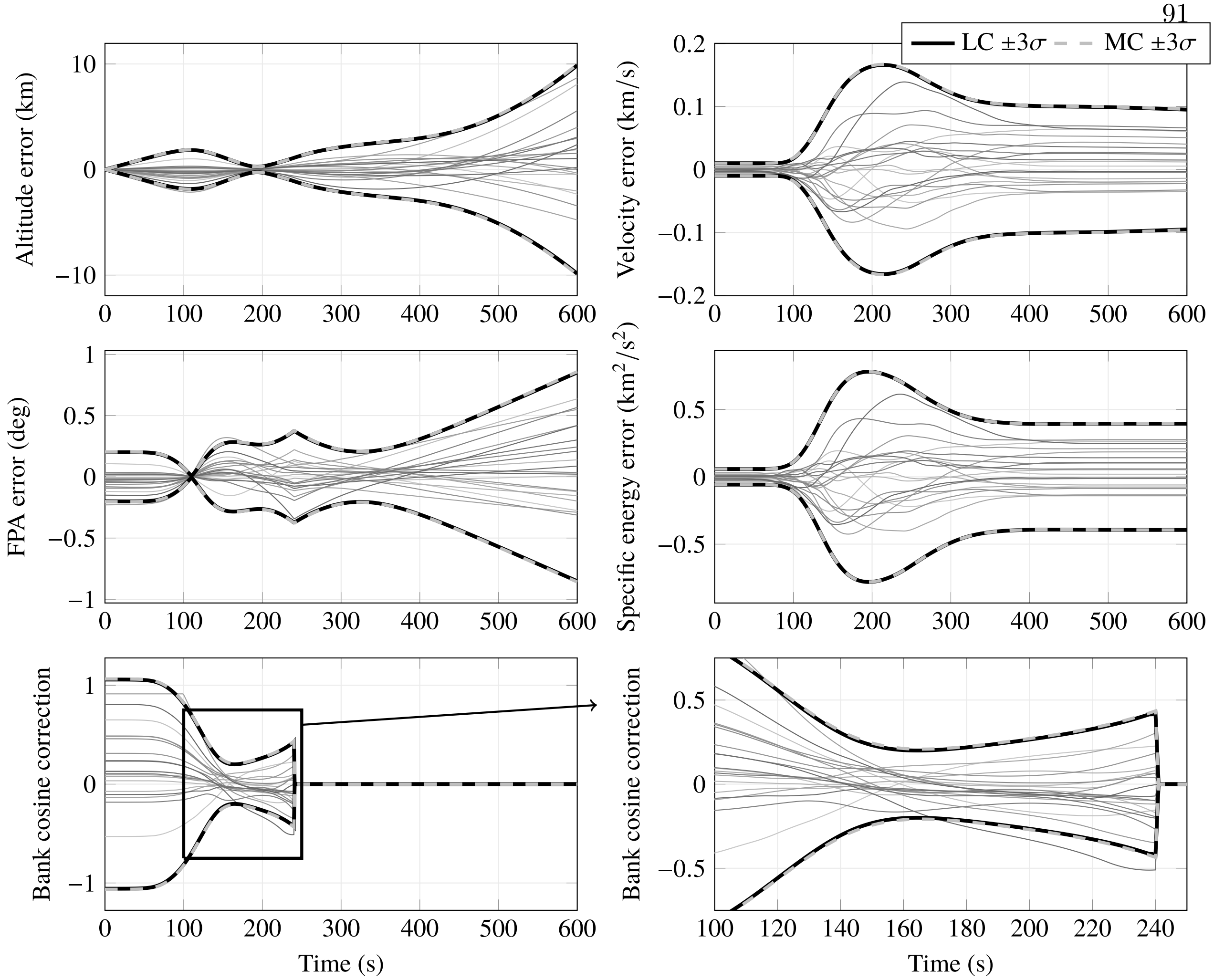


Figure 3.17: Sample closed-loop aerocapture trajectories with 3σ bounds computed from both 5,000 trial Monte Carlo (MC) and linear covariance (LC).

eral dynamics and guidance were not accounted for, and though these are often handled somewhat independently using bank reversals, the finite time spent reversing bank introduces a coupling between longitudinal and lateral guidance that is not considered here. The reference trajectory could also be further optimized for both of these examples to improve targeting performance, and the overcontrol could vary over the reference trajectory as a function of time, velocity, or energy. For the aerocapture guidance, traditionally terminal point control implements the feedback table as a function of energy instead of time as performed here.

One avenue for potential future work is to incorporate such predictions into an onboard

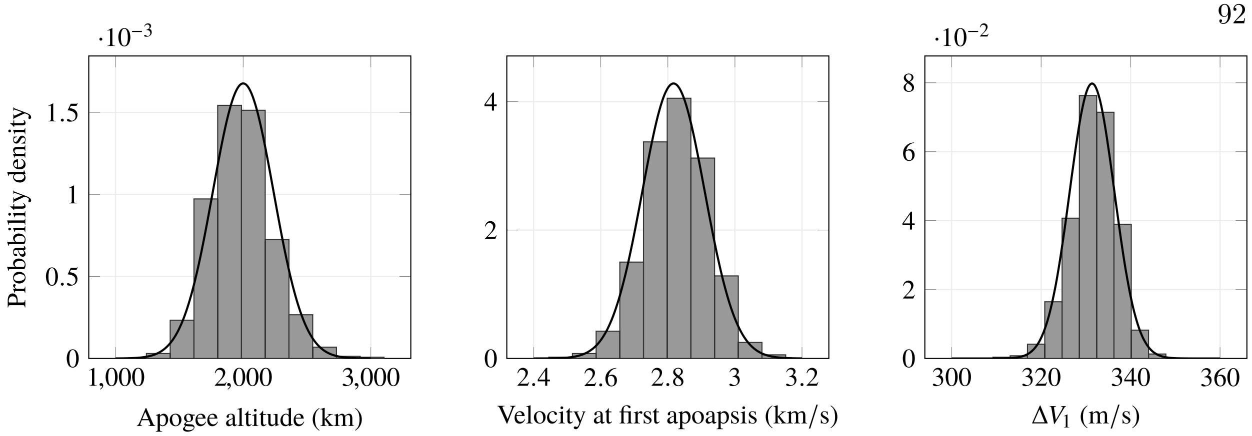


Figure 3.18: Statistics of post-aerocapture orbit from both 5,000 trial Monte Carlo (histograms) and from the linear covariance (Gaussian fit).

guidance scheme. For example, a numerical predictor-corrector could be wrapped around the linear feedback control, propagating the linear covariance and using ΔV_{99} as the error function instead of propagating single deterministic trajectories. The linear predictions could also be used to optimize the nominal trajectory and overcontrol value(s) to minimize ΔV_{99} with constraints on control saturation. Overcontrol could become a function of time and apply differently to different state errors as part of this process.

3.8 Conclusions

This work presents the mathematical foundation and practical implementation for modeling density using either a KLE or a VAE. This approach to compact modeling of an uncertain environment could have value in a wide range of other applications, including rocket ascent and drone flight planning. For the direct-entry and aerocapture scenarios considered here, a model constructed by scaling normalized density perturbations by the reference dynamic pressure is shown to be the best predictor of peak heat flux. Directly forming the model from density or normalized density perturbations is less compact but also gives accurate predictions, and could be the more straightforward approach if the necessary number of terms is allowable based on computational limitations. Because the data in this study are approximately Gaussian, the KLE modeling approach is shown to be adequate and has the advantage over VAE models of being simple to construct from the training data.

A KLE formed over a multi-dimensional domain is demonstrated, but for the MarsGRAM data considered here the gains compared to a columnar model are unlikely to outweigh the additional computational expense. Additionally, a Kalman measurement update is used to update the density covariance matrix for a KLE model based on new density measurements, and the example results show promising behavior. However, further work is necessary to improve the computational efficiency of this approach for onboard implementation. Implementing a VAE density model over a multi-dimensional domain and developing a method of conditioning a VAE model on noisy density measurements are both promising avenues for potential future work. In the case of the former, a VAE may outperform KLE models for dimensionality reduction of multi-dimensional density data, but might require a modified network architecture or training approach. In the case of the latter, retraining of the VAE onboard during flight would present an infeasible computational burden, so the key innovation would be a method of conditioning the VAE on noisy data without requiring further training.

Chapter 4

Co-Delivery of Direct-Entry Probe and Aerocapture Orbiter

4.1 Introduction

Co-delivery of a probe and an orbiter is a powerful architecture for a variety of interplanetary missions. The Galileo and Cassini-Huygens missions are two famous examples, among many others, of this approach. Given the infrequency of major planetary science missions, it is desirable to maximize scientific return by gathering data from orbit as well as *in-situ* measurements from the atmosphere or surface. Though interplanetary probe and orbiter missions have already been accomplished a number of times, two technologies could be combined to enable a new type of co-delivery architecture for planetary science missions.

The first technology is low-cost small satellites (smallsats), especially CubeSats, which have accounted for an increasingly large share of satellites launched each year since around 2012 [33]. Technological innovations including the miniaturization of electronics and availability of commercial-off-the-shelf hardware has led to a steady increase in the capabilities possible in these small form-factors, and CubeSat missions have now moved beyond serving a primarily educational role to make numerous notable scientific contributions [34]. A 2014 study sponsored by the Keck Institute for Space Studies presented space science mission concepts “uniquely enabled by the small satellite platform,” and recommended including small spacecraft as secondaries on all missions beyond low Earth orbit [35]. NASA has also studied a variety of mission concepts through its Planetary Science Deep Space SmallSat Studies program [36]. In November 2018 MarCO-A and MarCO-B, the twin CubeSat communications relays accompanying the InSight Mars lander, successfully demonstrated the merit of smallsats in deep space applications [37]. Ongoing research is applying smallsat innovation to entry, descent, and landing (EDL) by developing technologies including deployable aeroshells and multifunctional EDL structures [183, 164]. Smallsat secondary spacecraft enhance planetary science missions only if the secondary mission can minimize the additional mass, risk, cost, and complexity to the primary mission.

The second technology is aerocapture, the often-studied technique of flying through a planet's atmosphere to reduce the spacecraft's energy and capture into orbit, as shown in Figure 4.1. This technique has been studied for decades, but not implemented in flight. In recent years, significant work has contributed to the development of aerocapture and related technologies, including development of advanced thermal protection systems [184], robust flight-control methods and guidance algorithms [85, 116, 65], uncertainty quantification [185, 186, 187, 138], deployable decelerator technology [188, 189, 183], and broad aerocapture technology studies [19, 26, 25] to list a few. A 2016 study at the NASA Jet Propulsion Laboratory concluded that while aerocapture technology readiness is destination-dependent, no prior flight demonstration would be needed to implement aerocapture at Titan, Mars, and possibly Venus [19]. Some of the renewed interest in aerocapture can be attributed to recent concepts for missions to the ice giants (Uranus and Neptune) in preparation for the Planetary Science Decadal Survey [30], because it is these destinations where aerocapture can offer the most benefit compared to propulsive orbit insertion [18].

The concept that combines secondary smallsats and aerocapture is to design a probe and an orbiter to reach their desired final states from a single approach trajectory and entry state, illustrated in Fig. 4.2. The two vehicles travel together during cruise and separate shortly before atmospheric entry, then diverge during atmospheric flight due to differences in their aerodynamic properties and control strategies. The orbiter stays higher in the atmosphere, dissipating just enough energy to perform aerocapture, while the probe continues deeper into the atmosphere until reaching its desired target state, such as parachute deployment or surface impact. By designing the probe and orbiter to target a single atmospheric entry state, the need for a critical divert maneuver performed shortly before entry is avoided. For example, a satellite using lift-modulated aerocapture

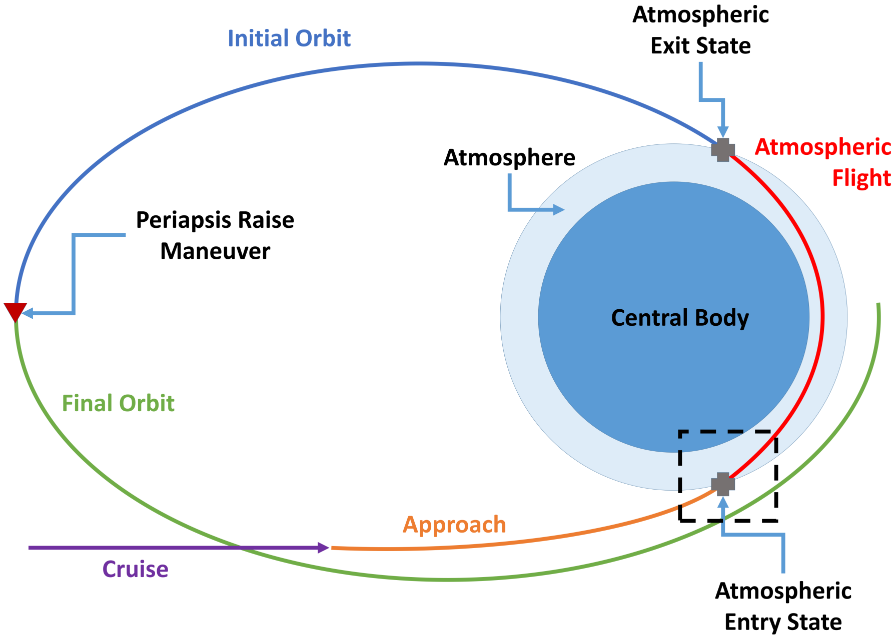


Figure 4.1: Diagram of the aerocapture process

to reach Mars orbit could release several small probes that follow ballistic trajectories down to the surface. A jettison event is still required to physically separate the orbiter and probe and prevent recontact in the atmosphere, akin to the mechanical separation of Mars Science Laboratory (MSL) aeroshell from its cruise stage 10 minutes prior to atmospheric entry [181]. In general this co-delivery approach can apply to missions with multiple probes or orbiters, but for simplicity this study proceeds assuming only one of each. Some key terms as used in this study:

- “Co-delivery” refers to any two or more spacecraft that reach a shared destination via a single interplanetary trajectory, such as the delivery of five separate probes by the Pioneer Venus Multiprobe bus [190].

- “Probe” is used as a catch-all term including landers, impactors, deep atmospheric probes, etc.

- “Secondary” refers to a smaller, ride-along addition to a larger, more expensive “primary” craft, e.g. MarCO was a secondary mission for the InSight primary spacecraft. In the context of the proposed co-delivery method, a primary orbiter could have a secondary

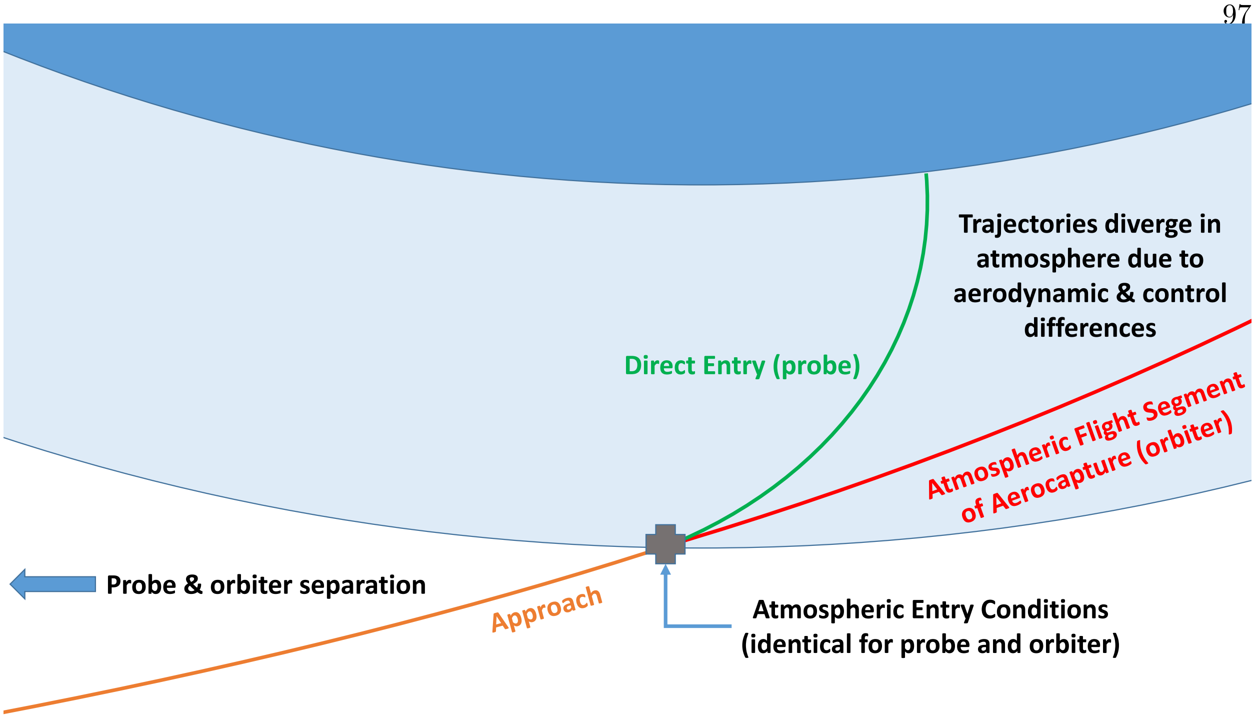


Figure 4.2: Conceptual diagram of co-delivery from a single entry state, shown as a close-up view of the region in the dashed-line box in Fig. 4.1. Features exaggerated.

probe or the other way around, hence these terms are defined separately.

The primary motivation for targeting a single entry state for both orbiter and probe is to avoid requiring a divert maneuver, and managing its associated error, shortly before entry. If this maneuver is performed early, the probe would either require a propulsion subsystem and navigation capability, or would be coasting without course-correction capability from separation until entry. The later the orbiter performs the divert maneuver, the larger this maneuver becomes and the less time there is to quantify and potentially mitigate maneuver execution error. This is not to say that these other co-delivery architectures are not feasible; indeed, Galileo and Cassini-Huygens successfully had probes coast passively for nearly 150 days and 20 days, respectively [99, 100]. Rather, targeting a single entry condition is a solution to this tradeoff that reduces maneuver complexity on approach and eliminates a source of navigation error. Furthermore, by co-delivering the probe and orbiter, the secondary craft is able to benefit from the primary spacecraft's resources such as power, propulsion, and communications until shortly before atmospheric entry.

This may significantly simplify the design of the secondary ride-along craft compared to separate post-launch operations; for example, independent operations and navigation during cruise proved to be a significant challenge for the MarCO CubeSats [191].

Reference [52] qualitatively discusses the challenges of the proposed co-delivery method in detail. These challenges include the timing and dynamics of the separation event, post-separation collision concerns, timing and observation geometry between the orbiter and probe, and the feasibility of trajectories that deliver an orbiter for aerocapture and a probe for direct-entry from a single entry state. The last of these is the focus of this study, with the remaining challenges left for future work.

This study focuses on the feasibility of the flight mechanics associated with this co-delivery strategy. A broad trade space is explored to understand the regions of feasibility for co-delivery from a single entry state while quantifying relevant mission constraints. Earth, Mars, Venus, Titan, and Neptune applications are considered. A single representative scenario is developed that implements closed-loop guidance for both vehicles and also includes a passive ballistic probe, and illustrates the performance of these vehicles under relevant uncertainties via Monte Carlo simulation.

4.2 Trade Study

The purpose of this section is to understand, at a high level, the combinations of trajectories and vehicles for which co-delivery from a single entry state is a possibility. A wide range of entry trajectories are simulated and classified by their final states, and a number of key constraining parameters are computed. The goal of this study is to demonstrate the fundamental flight mechanics feasibility of this co-delivery method at each destination and provide a starting point for further investigation of any specific mission concept.

4.2.1 Methodology

To simulate these trajectories, three degree-of-freedom equations of motion are numerically integrated assuming a point-mass gravity with lift and drag forces acting on each vehicle[1] [78]. Consistent with the flight of a blunt body in hypersonic continuum flow, constant aerodynamic coefficients, constant mass, and zero thrust are assumed, as well as zero wind. The vehicle state is propagated using a variable-step Runge-Kutta numerical integration method of order 5(4) [192]. The vehicle is initialized at the atmospheric interface altitude h_{atm}. For each target destination, a representative planet-relative entry velocity, $V_{R,0}$, is defined based on entry velocities of previous planetary entry missions or aerocapture mission studies [193, 56, 58]. Entry flight path angle (EFPA) and ballistic coefficient are varied as part of the trade study. EFPA is the angle between the vehicle's planet-relative velocity vector and the local horizontal. The ballistic coefficient β is effectively a ratio of inertial to aerodynamics forces on the vehicle and is defined in Eq. (4.1), where m is vehicle mass, C_D is hypersonic drag coefficient, and A is reference area. Note that while the particular results described herein will vary as a function of entry velocity, the purpose of this work is to demonstrate the conceptual feasibility of this co-delivery technique. The parameters used in this analysis are listed in Table 4.1.

$$\beta = \frac{m}{C_D A} \tag{4.1}$$

Table 4.1: Relevant Planetary Constants

Central Body	h_{atm}, km	$V_{R,0}$, km/s	k, kg$^{0.5}$/m	atm. composition by volume
Earth	125 [194][2]	11	1.748×10^{-4}	78.1% N_2, 20.9% O_2 [162]
Mars	125 [165]	6	1.904×10^{-4}	2.59% N_2, 95.1% CO_2, 1.94% Ar [163]
Venus	135 [162]	11.5	1.897×10^{-4}	3.50% N_2, 96.5% CO_2 [162]
Titan	800 [162]	6	1.758×10^{-4}	97.7% N_2, 2.30% CH_4 [162]
Neptune	1000 [195]	27	7.361×10^{-5}	1.50% CH_4, 79.6% H_2, 18.9% He [196][3]

Profiles of atmospheric density are taken from the nominal output of the Global Reference Atmospheric Model (GRAM) for that planet/moon [197, 151, 198, 199, 200], where each GRAM

---

[1] `https://github.com/salbert21/petunia`

[2] Orion uses $h_{atm} = 400,000$ ft., which here is rounded up to 125 km to match convention

[3] Particular values chosen to match NASA Space Science Data Coordinated Archive fact sheet for Neptune

provides an engineering-level model that can produce both mean and dispersed atmospheric data. The density profile is then linearly interpolated with altitude; although density varies approximately exponentially with altitude, GRAM data is output every 0.1 km, so linear interpolation between datapoints is sufficiently accurate for this application. To approximately characterize the effect of density variability, results are shown for density profiles at plus or minus three standard deviations from nominal, where these $\pm 3\sigma$ profiles are directly output by GRAM.

Several potentially constraining quantities are calculated for each trajectory, one of which is peak heat flux. Specifically, peak convective heat flux at the stagnation point for a fully catalytic surface is estimated using the Sutton-Graves method [201]. The expression is shown in Eq. (4.2), where $\dot{q}_s$ is total convective heat rate at the stagnation point, p_s and h_s are the total stagnation point pressure and enthalpy respectively, R_n is the effective nose radius, h_w is the surface enthalpy, and K_{SG} is a coefficient. This expression is then converted to the more useful form shown in Eq. (4.3) using a few assumptions for hypersonic flow. In hypersonic flow the surface enthalpy h_w is a negligible contribution to the total value, which can then be approximated as $h_s \approx V_w^2/2$ [202]. Using a Newtonian flow approximation, the pressure coefficient at the stagnation point is $C_{p,s} = 2$, and freestream pressure makes a negligible contribution, so stagnation point pressure becomes $p_s = 1/2\ C_{p,s} \rho V_w^2 + p_\infty = \rho V_w^2$ [202]. The modified Sutton-Graves coefficient is then $k = K_{SG}/(2\sqrt{101325})^4$. The values used in this study for k, as well as the atmospheric compositions used to compute them, are listed in Table 4.1.

$$\dot{q}_s = K_{SG}\sqrt{\frac{p_s}{R_n}}(h_s - h_w) \tag{4.2}$$

$$\dot{q}_s = k\sqrt{\frac{\rho}{R_n}}V_w^3 \tag{4.3}$$

Radiative heating is not included in this analysis. Total integrated heat load is computed by numerically integrating the stagnation point convective heat flux over time. An effective nose radius of $R_n = 1$ m is assumed, which allows easy scaling of these heating results for other nose

---

[4] The $1/\sqrt{101325}$ factor comes out of a unit conversion from atm to Pa.

radius values. The maximum sensed acceleration in terms of Earth g's is also calculated.

$$\dot{q}_s = k\sqrt{\frac{\rho}{R_n}}V_R^3 \tag{4.4}$$

Each trajectory is categorized based on its exit state: if the trajectory intersects the surface (or some minimum altitude) it is a probe, if the vehicle exits the atmosphere on an elliptical orbit it is an orbiter, and if the vehicle exits the atmosphere on a hyperbolic orbit the trajectory is categorized as escape. For the orbiter trajectories, apoapsis altitude is computed using the vehicle's post-atmospheric Keplerian state.

Three types of trajectories are described in the open-loop analysis presented in this study: ballistic, full-lift-up and full-lift-down. These descriptors do not imply that the vehicle has no additional control authority; rather, they represent nominal trajectories for which no lift- or drag-modulation is required. A lift-to-drag ratio of $L/D = 0.25$ is selected based on the approximate hypersonic trim L/D of MSL and Mars 2020 and the known capabilities of a 70° sphere cone aeroshell [165]. While the results of this study provide insight into the consequences of increasing or decreasing L/D from this value, quantitative analysis for vehicles with significantly different L/D is left for future work. By showing these three cases, the set of trajectories approximately accessible with a 70° sphere cone aeroshell is characterized for each scenario.

4.2.2 Results

The results at each planetary destination are summarized in Figs. 4.3 - 4.7. For each of the three trajectory types, trajectories are simulated across a grid of varying EFPA and ballistic coefficient. For each grid, at any given β there will be some EFPA value that delineates between orbiters and probes. These EFPA values form the black line on each plot. Similarly, if the inertial entry velocity exceeds escape velocity, there will be an EFPA value delineating between orbiters and escape trajectories, and this is shown as the purple line. The shaded regions for each line are bounded by the values of that line when the $\pm 3\sigma$ profiles are used for density. Therefore, any gridpoints left of the black line are probe trajectories, any gridpoints between the lines are orbiters (aerocapture), and to the right of the purple line are escape trajectories. Contours of apoapsis altitude, peak g-load, peak heat rate, and total heat load are then overlaid for each plot. Note that the contour values are not necessarily evenly incremented, and that the x-axis scale varies significantly between destinations.

The interpretation of these plots is illustrated through the following example. By definition the proposed co-delivery method is feasible where a probe trajectory and orbiter trajectory both exist at the same EFPA for realistic ballistic coefficients. Because the vehicles share an entry condition, co-delivery scenarios are identified in these plots with vertical cross-sections along a single EFPA. As a simple example, a vertical line at $-5.5\,^{\circ}$ (not shown) for the Earth-ballistic plot would pass through the middle of the black line. Here, ballistic coefficients less than $75\,\mathrm{kg\,m^{-2}}$ are probes, and greater than $110\,\mathrm{kg\,m^{-2}}$ are orbiters. Thus, for 11 km/s entry at Earth with an EFPA of $-5.5\,^{\circ}$, co-delivery from a single entry state is possible using only ballistic trajectories, just by tuning the β values of the two vehicles.

The application of lift broadens this feasible range significantly. In Fig. 4.3 a light blue vertical line is added at a nominal EFPA of $6.25\,^{\circ}$. On the ballistic plot the line is entirely behind the orbiters/probes cutoff, meaning all ballistic coefficients in the range considered (10 - $200\,\mathrm{kg\,m^{-2}}$) result in probe trajectories. On the full-lift-up plot the line is entirely in front of the cutoff line, so all β values result in orbiter trajectories. The initial apoapsis altitudes for these trajectories vary with ballistic coefficient, and are shown in the dashed blue contour lines.

For the proposed co-delivery method to be plausible, the architecture should be robust to a number of uncertainties, including navigation uncertainty. This can be described as an entry corridor, a range of possible EFPA values. In Fig. 4.3 the dashed light blue vertical lines represent an entry corridor of $-6.25^{\circ} \pm 0.5^{\circ}$. As a result of this uncertainty the dashed lines now intersect the black cutoff lines for ballistic and lift-up trajectories, and these intersection points give the ballistic coefficient requirements for this scenario. For feasibility even with this large EFPA uncertainty the orbiter β would need to be at least $40\,\mathrm{kg\,m^{-2}}$, and the probe coefficient no greater than $160\,\mathrm{kg\,m^{-2}}$. The value of $\pm 0.5^{\circ}$ used here is only an example; the same process can be followed for any width entry corridor using the results in Figs. 4.3 - 4.7.

These ballistic coefficients might be further constrained by other requirements. Continuing the example annotated in Fig. 4.3, to achieve an initial apoapsis altitude of at least 150 km, the orbiter β should be at least $75\,\mathrm{kg\,m^{-2}}$. To keep the total heat load at the stagnation point below $20\,\mathrm{kJ\,cm^{-2}}$, the probe β should be no greater than $120\,\mathrm{kg\,m^{-2}}$. Additionally, note that the EFPA range still does not intersect the cutoff line on the full-lift-down plot, so any ballistic coefficient in range would result in a probe trajectory, although the peak g-loads are significantly higher for lift-down trajectories. In addition to these flight mechanics constraints, packaging and vehicle geometry considerations make some ballistic coefficients more feasible than others.

Alternatively, these plots can be used in the other direction to determine the tolerable amount of uncertainty before the co-delivery architecture design fails to close entirely. Consider co-delivery at Neptune for an orbiter flying full-lift-up and a deep atmospheric probe flying ballistically, each with a ballistic coefficient of $150\,\mathrm{kg\,m^{-2}}$. Imagine, arbitrarily, that the orbiter is required to achieve an initial apoapsis below 50.000 km, without other constraints on either vehicle. The theoretical corridor width for the orbiter can then be determined by the intersection of the $150\,\mathrm{kg\,m^{-2}}$ horizontal line with the shaded black region on the left and the dashed blue contour for 50.000 km apoapsis on the right. This results, approximately, in a theoretical entry corridor of $-13.875° \pm 0.375°$. Here, theoretical corridor width represents a combination of navigation and aerodynamic uncertainties, but atmospheric density uncertainty is already taken into account in the plot.

The examples above demonstrate how a mission designer can choose constraints on nominal EFPA, entry corridor, apoapsis, etc. and then directly assess the feasibility of probe and orbiter co-delivery from a single entry state for that mission scenario from the plots in Figs. 4.3 - 4.7.

4.2.3 Discussion

The feasibility assessment at each destination depends on the specific scenario and constraints, making it challenging to compare the destinations in a general way. One heuristic approach is to consider the EFPA range spanned by the probes/orbiters cutoff line, i.e. the difference between

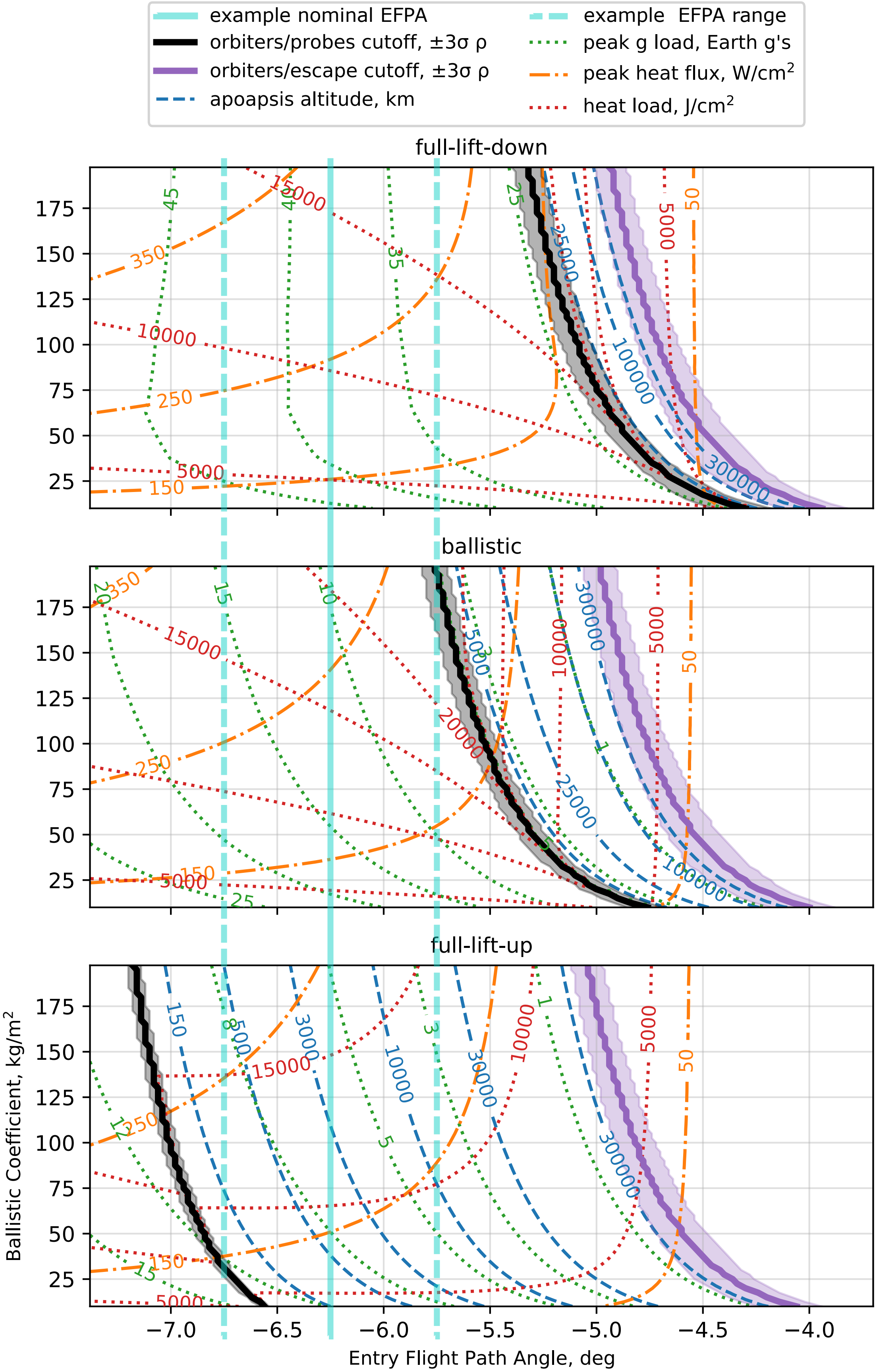


Figure 4.3: Feasibility space for Earth, $11\,\mathrm{km\,s^{-1}}$ relative entry velocity, shown with example annotation

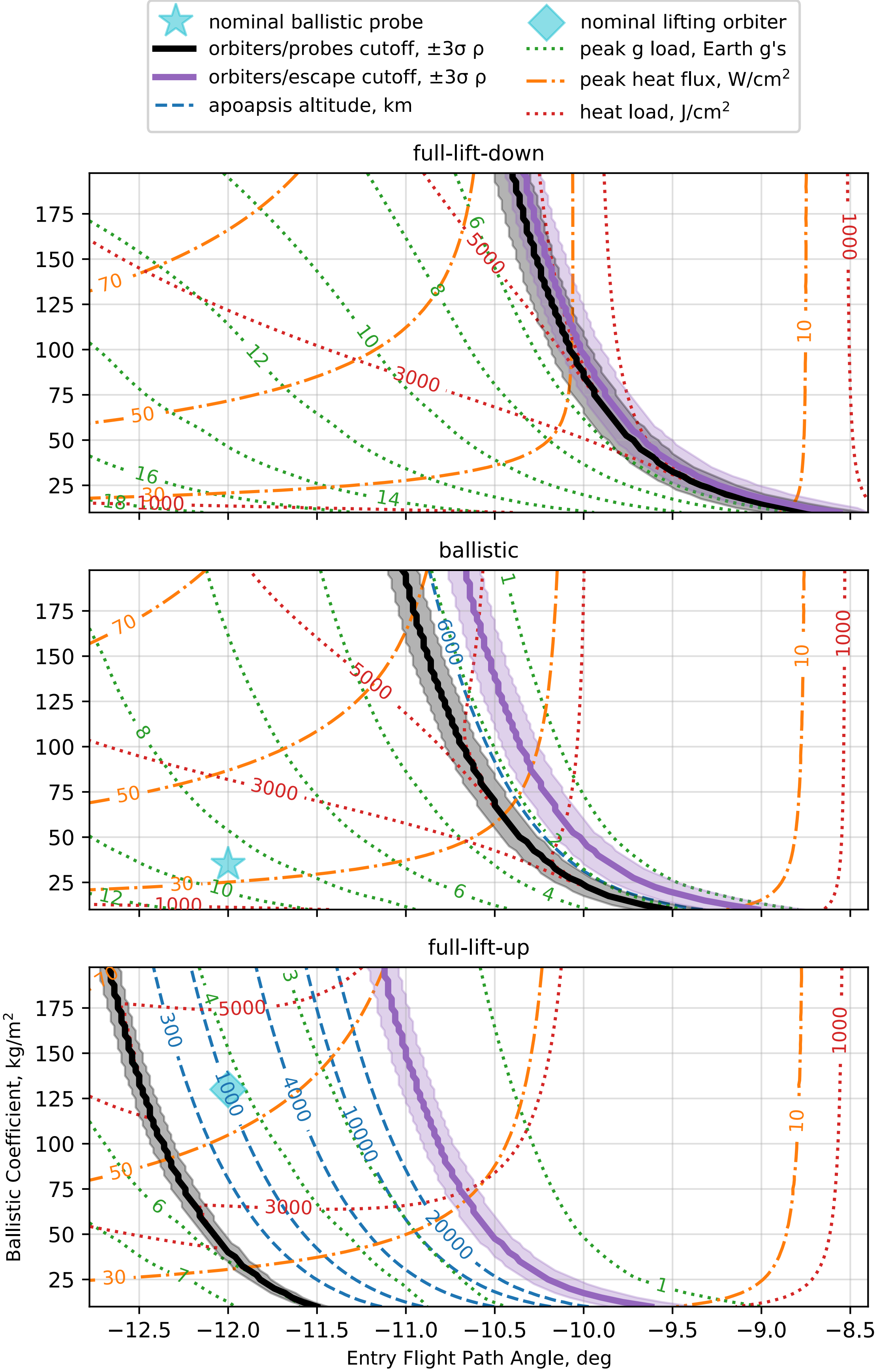


Figure 4.4: Feasibility space for Mars, $6\,\mathrm{km\,s^{-1}}$ relative entry velocity, and nominal scenario used in Section 4.3 marked in blue

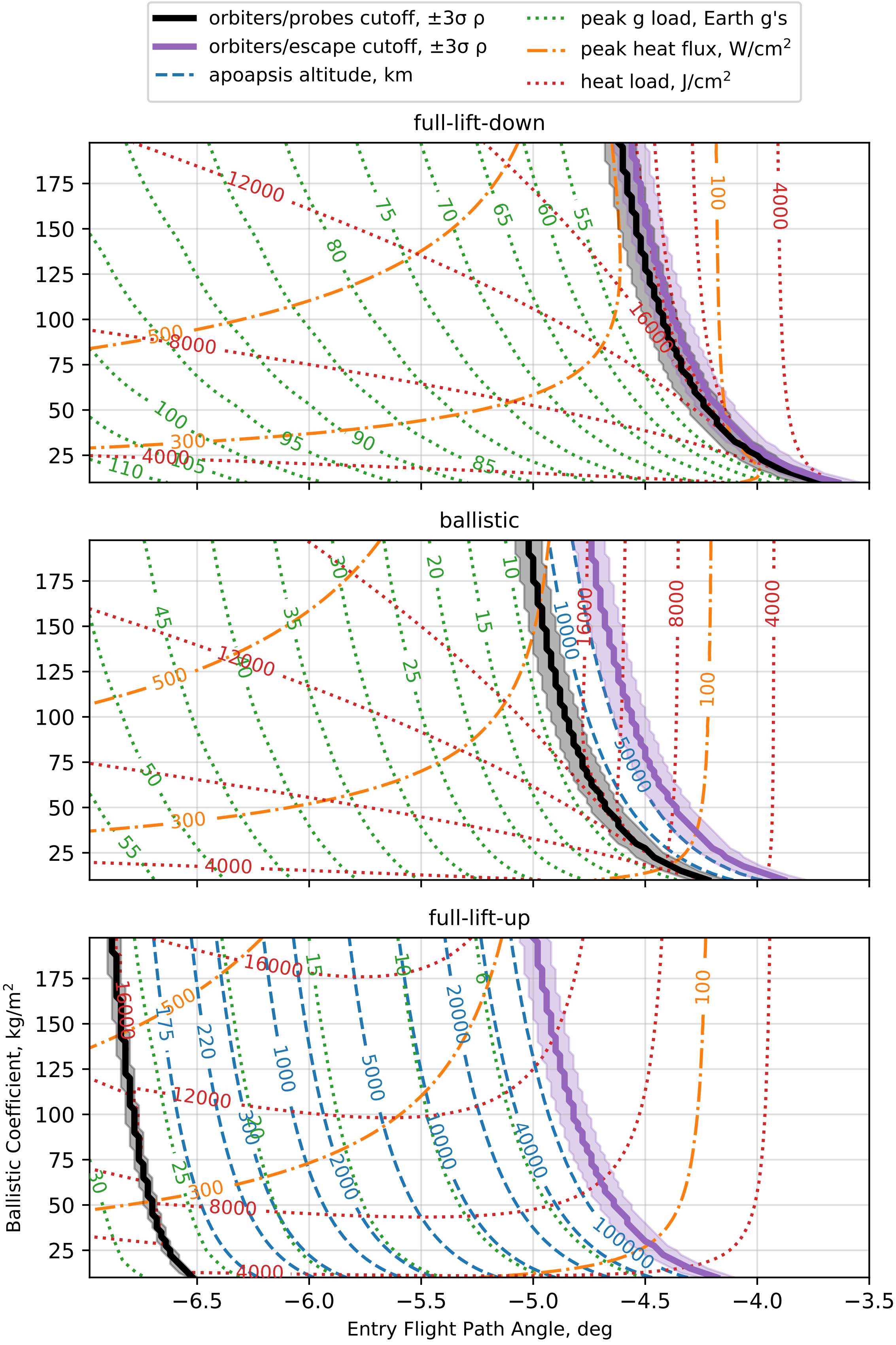


Figure 4.5: Feasibility space for Venus, 11.5 km s^{-1} relative entry velocity

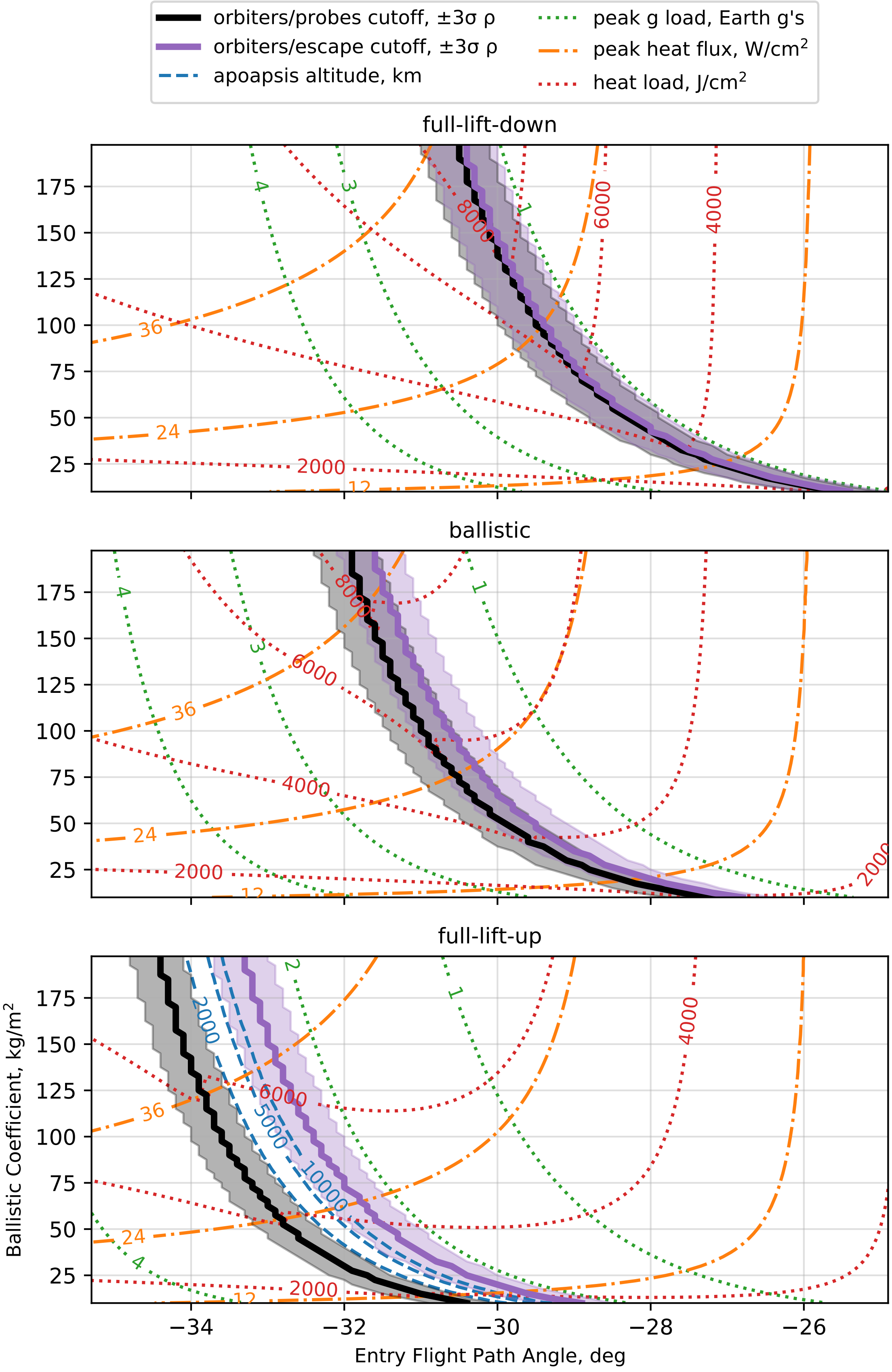


Figure 4.6: Feasibility space for Titan, $6\,\mathrm{km\,s^{-1}}$ relative entry velocity

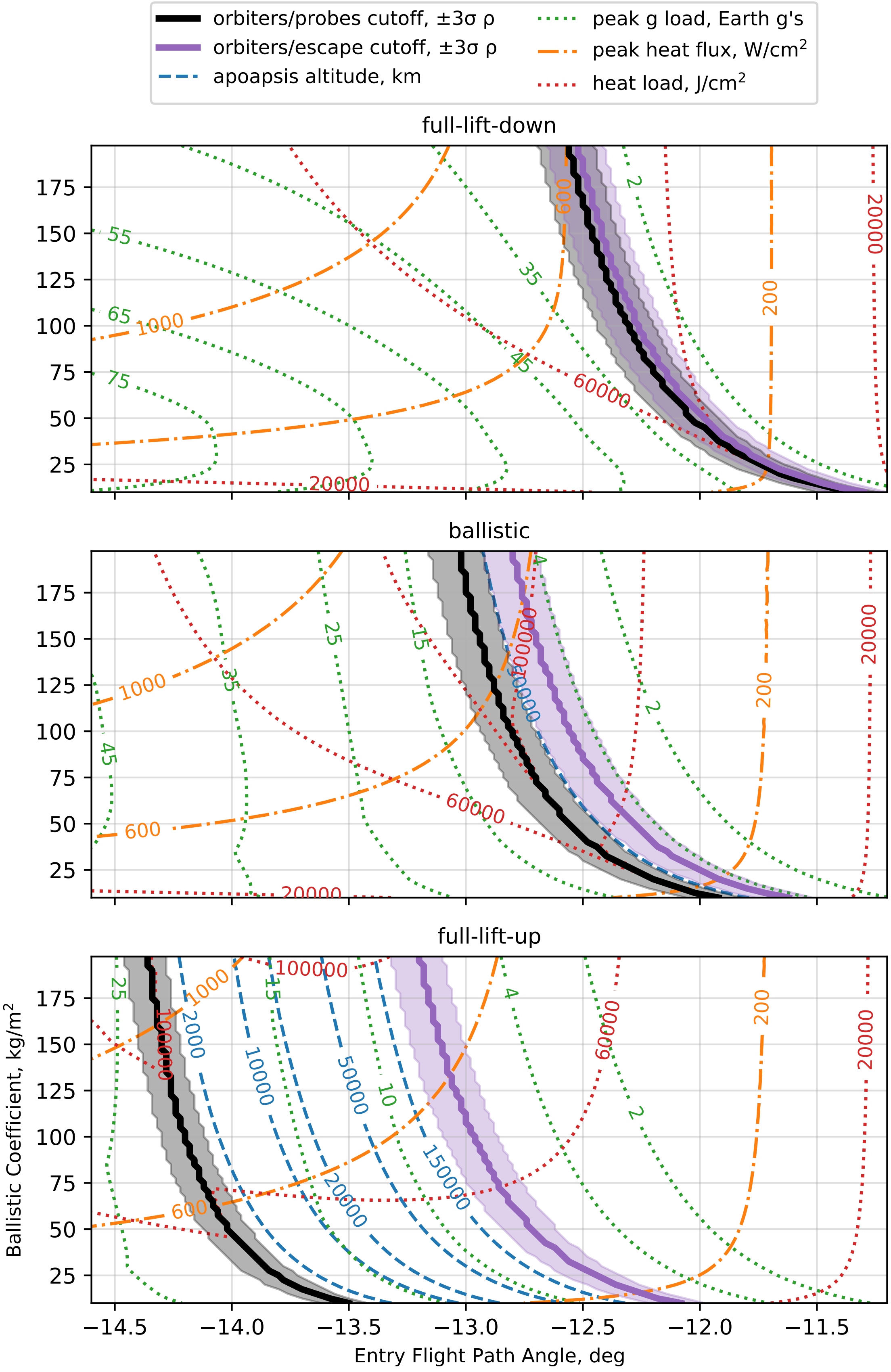


Figure 4.7: Feasibility space for Neptune, 27 km s^{-1} relative entry velocity

the probe/orbiter transition EFPAs at $\beta = 200\,\mathrm{kg\,m^{-2}}$ and $\beta = 10\,\mathrm{kg\,m^{-2}}$ for ballistic trajectories, including the narrowing effect of the atmospheric uncertainty bars. In ascending order, this value is approximately: 0.66 ° at Venus, 0.84 ° at Earth, 0.88 ° at Neptune, 1.22 ° at Mars, and 3.8 ° at Titan. These values reflect the the ranges of usable EFPAs for two ballistic vehicles, meaning Titan is by far the most flexible if no nominal lift is required. A similar heuristic parameter is the EFPA range gained from a full-lift-up trajectory, defined as the difference between the probe/orbiter transition EFPAs at $\beta = 100\,\mathrm{kg\,m^{-2}}$ for full-lift-up and ballistic trajectories including the narrowing effect of the atmospheric uncertainty bars. Again in ascending order, this value is approximately: 1.18 ° at Neptune, 1.36 ° at Earth, 1.52 ° at Mars, 1.84 ° at Venus, and 1.9 ° at Titan. Titan again has the widest range by this measure, though by a smaller margin. Notably, Venus had the smallest range for ballistic-only trajectories but has the second-widest range by this measure of lift-effectiveness. The small scale height of the Venusian atmosphere at aerocapture altitudes corresponds to rapid density variations with altitude [56], resulting in narrow corridor widths but a large control authority for lifting vehicles. Furthermore, all else being equal, high entry velocities lead to larger theoretical corridor widths for lift-modulation aerocapture vehicles [16], and the representative entry velocity chosen for Venus in this study is high relative to the planet's mass. It is important to note that these benefits are directly traded-off by high g-loads, heat rates, and heat loads at Venus; the high entry velocity at Neptune, dictated by its large gravity well and the constraint of reasonable times of flight from Earth, has similar drawbacks. Note that the particular values of these EFPA ranges are tied to the choices of atmospheric interface altitudes listed in Table 4.1.

The results shown in Figs. 4.3 - 4.7 and discussed above are primarily in terms of ballistic coefficient, which is a ratio and provides no information on the actual mass and volume of the vehicle. The mechanical and aerodynamic design of specific aeroshells to meet a target ballistic coefficient, fit within secondary smallsat mass and volume constraints, and accommodate a science payload is beyond the scope of this study. That said, there is precedent for entry capsules in a smallsat form factor. Most notably, the Mars Microprobes provide flight-heritage at Mars and each probe had a ballistic coefficient of $35.6\,\mathrm{kg\,m^{-2}}$, an entry mass of 3.6 kg, and would have fit within a $35 \times 35 \times 27.5\,\mathrm{cm}$ box [203], well within the mass and volume constraints for a secondary payload on an Evolved Expendable Launch Vehicle Secondary Payload Adapter (ESPA) ring [204]. The Adaptive Deployable Entry and Placement Technology (ADEPT) deployable aeroshell enables aeroshell diameters of up to $1.7\,\mathrm{m}$ to stow within an ESPA secondary payload volume and has an expected ballistic coefficient in the range of 25-50 $\mathrm{kg\,m^{-2}}$ and entry mass of 75-150 kg for delivery of a 12U CubeSat payload volume [183]. The notional design for the Small High Impact Energy Landing Device (SHIELD) has a ballistic coefficient as low as $10\,\mathrm{kg\,m^{-2}}$ for an entry mass of 50 kg and 6 kg payload mass [164, 205]. These example designs are included here to illustrate the fact that, while detailed design is out of scope, the flight mechanics of the co-delivery method with smallsat-class vehicles are feasible.

There are some key limitations to the approach taken in this study. For the sake of space, only one entry velocity is considered for each destination. In general, increases in entry velocity on the order of $1\,\mathrm{km\,s^{-1}}$ lead to an increase in lift-modulation control authority, increased g-loads, and a compression of the available apoapsis radii in terms of EFPA, but the overall feasibility of co-delivery is not dramatically affected. This is shown in Reference [206] through a comparison of feasibility with 10, 11, and $12.5\,\mathrm{km\,s^{-1}}$ entry velocities at Earth. Another limitation is the bounding case approach to lift-modulation. While it is possible to use 100% of available lift to bias the nominal trajectory – Viking flew a full-lift-up trajectory with no guidance – in general some control authority must be allocated to compensate for uncertainties in EFPA, atmospheric density, vehicle parameters, etc. MSL, for example, used about 70% of its available lift to bias its nominal trajectory, reserving 30% for control authority margin [165]. The ability of the results shown here to capture these types of trajectories is limited. For example, for a lift-up trajectory at Earth, $11\,\mathrm{km\,s^{-1}}$, and an EFPA of $-6\,^{\circ}$, a ballistic coefficient of $50\,\mathrm{kg\,m^{-2}}$ results in an apoapsis altitude of about 3000 km. Intuitively, a similar trajectory that instead uses only 70% of its lift for the nominal trajectory would result in aerocapture with a lower apoapsis, but the results shown here do not quantify this idea. Nonetheless, these results give bounding cases within which a vehicle could reserve some control margin for uncertainties by targeting a lower apoapsis or increasing control authority by increasing L/D.

Another aerocapture trade reflected in these results is that, in general, more is gained from the lift-up trajectories than from lift-down. From inspection of the example in Fig. 4.3, it is clear that one appealing configuration is a lift-modulated orbiter with a ballistic probe trajectory. The ballistic probe trajectory could be truly passive, such as for a simple penetrator probe mission, or it could apply lift- or drag-modulation to the ballistic trajectory for the purpose of accommodating uncertainties. Passive impactor or penetrator probes are already excellent candidates for co-delivery due to their simplicity and small size, so this configuration stands out as a promising mission architecture for multiple reasons.

4.3 Representative Scenario

The previous section explores a large trade space for probe and orbiter co-delivery by considering numerous point designs. Each of these trajectories is only passively controlled (full-lift-up or -down, or ballistic) and has no accounting for random uncertainties. In order to further demonstrate the fundamental feasibility of the proposed co-delivery method from a flight mechanics standpoint, this section more closely examines a single representative mission scenario. A nominal scenario is defined that makes use of bank-angle modulation lift control for the orbiter and guided probe while also considering a passive ballistic probe. Representative uncertainties are then applied, and their effect quantified through a Monte Carlo analysis.

4.3.1 Methodology

A general co-delivery scenario involves two vehicles, an aerocapture orbiter and a direct-entry probe, each of which may implement some closed-loop guidance to control their atmospheric flight. As noted in Subsection 4.2.3, a particular scenario of interest would involve a passive ballistic probe; because this vehicle would be significantly simpler than an entry vehicle using active guidance and control, it may be a better fit for ride-along probe missions. Thus, three vehicles are considered in this section: a guided orbiter that performs aerocapture, a guided probe following a direct-entry trajectory, and a passive ballistic probe that follows a similar direct-entry trajectory.

Mars is chosen as the destination for this representative example scenario. The trajectories are simulated using the same equations of motion as in Subsection 4.2.1, with the addition of the J_2 zonal term in the spherical harmonics gravity model, where $J_2 = 0.001964$ [161]. For the orbiter and guided probe, bank-angle modulation is used as the method of control, which updates the orientation of the lift vector about the velocity vector without changing the angle of attack. This method is selected here for its flight heritage on the Mars Science Laboratory and Mars 2020 missions [165, 207], but note that other control approaches such as drag-skirt jettison or direct-force control would also be applicable.

Mode 1 of the Fully-Numerical Predictor-corrector Aerocapture Guidance (FNPAG) scheme developed by Lu et. al [65] is implemented for the orbiter. This guidance algorithm assumes a bang-bang structure to lift-modulation, wherein the vehicle uses Brent's method [208] to numerically predict a switching time from a lift-up angle $0 \leq \sigma_i < 90\,^{\circ}$[5] to a lift-down angle $90\,^{\circ} < \sigma_d \leq 180\,^{\circ}$. This bang-bang structure targets the desired apoapsis while minimizing the total ΔV required for the periapsis raise and apoapsis correction maneuvers. A number of simplifying assumptions are made for the purpose of this demonstration.

(1) Only longitudinal guidance is implemented, meaning a final apoapsis radius is targeted while ignoring the final inclination or wedge angle. Lateral guidance is normally achieved separately from the modulation of the bank-angle magnitude through periodic bank reversals [65]. Assuming no plane change is desired during aerocapture, the feasibility of achieving the desired apoapsis under uncertainties can be approximately assessed without considering the lateral guidance component, even though for any real aerocapture mission the lateral logic is an important part of the guidance scheme.

(2) The initial bank-angle is assumed to be $\sigma_i = 0°$, and the initial guess for the final bank-angle (which is updated during Phase 2 of FNPAG) is assumed to be $\sigma_d = 150°$. Note that

---

[5] called σ_0 in [65], renamed here to distinguish from the FNPEG variable of the same name

with $\sigma_i = 0°$ there is neither lateral force nor lateral control authority during Phase 1, but bank reversals could be performed during Phase 2.

(3) The guidance is run at a rate of 1 Hz and the bank-angle is updated instantaneously, neglecting the effect of a finite roll rate and acceleration for the vehicle.

(4) No additional trajectory constraints are imposed, such as limits on peak heat rate or g-load, because the value of those limits would be strongly mission-dependent.

(5) No atmospheric estimation model is included in the guidance implementation as this was found to be unnecessary to demonstrate fundamental feasibility for this scenario.

These simplifying assumptions are appropriate here because this section presents a proof-of-concept demonstration; a dedicated mission analysis would iteratively tune σ_d, potentially assume a larger value for σ_i, implement bank reversals and a finite roll rate and acceleration, and so on.

A similar approach is taken for the guided probe, which implements the Fully-Numerical Predictor-corrector Entry Guidance (FNPEG) developed by Lu [143]. FNPEG assumes the bank-angle magnitude profile is a linear function of e as shown in Eq. (4.5), where e is the negative of the specific orbital energy as given in Eq. (4.6). The value of σ_0 is then updated with each guidance call in order to target a desired range and energy, where the desired energy is computed by applying Eq. (4.6) to the desired radial distance and inertial velocity at the final time. Note that because the target values for radius and velocity are combined into a single constraint, FNPEG can result in small altitude and velocity errors, but in many applications (such as targeting range at parachute deploy) this is acceptable [143]. At each guidance call, FNPEG uses the Golden-Section method to minimize the error function Eq. (4.7) [208], where $s(e_f)$ is the predicted great-circle range at the target energy and s_f^* is the target final range. The value of $s(e_f)$ is predicted numerically by including range s (in radians) in the equations of motion as $\dot{s} = V\cos(\gamma)/r$ and propagating until the target energy is reached [143], where V and γ are both planet-relative values. Similar assumptions are made here as for FNPAG. Lateral guidance is again neglected for the same reasons, instantaneous bank-angle updates are assumed with guidance run at a rate of 1 Hz, and no additional trajectory constraints are imposed. A value of $\sigma_f = 60°$ is used for this study. As with FNPAG, these assumptions are made for the sake of a proof-of-concept demonstration, and a more detailed mission analysis would tune σ_f, implement finite roll rate and acceleration, etc.

$$|\sigma(e)| = \sigma_0 + \frac{e - e_0}{e_f - e_0}(\sigma_f - \sigma_0) \tag{4.5}$$

$$e = \frac{\mu}{r} - \frac{v^2}{2} \tag{4.6}$$

$$f(\sigma_0) = \frac{1}{2}\left[s(e_f) - s_f^*\right]^2 \tag{4.7}$$

4.3.2 Nominal Scenario

A nominal scenario is constructed starting from the results shown in Fig. 4.4. As seen from the full-lift-up plot in Fig. 4.4, a lifting vehicle with $\beta = 130\,\mathrm{kg\,m^{-2}}$ can achieve aerocapture with a low apoapsis from an EFPA of $-12\,°$. As seen in the ballistic plot, a probe with $\beta = 35\,\mathrm{kg\,m^{-2}}$ can follow a direct-entry trajectory from the same EFPA, either as a ballistic probe or a lifting vehicle dedicating some or all of its control authority to mitigating uncertainties. The orbiter ballistic coefficient was chosen to be similar to that of MSL [165], and the probe ballistic coefficient to be similar to that of the Mars Microprobe capsules [203]. As in the trade study, the guided vehicles have a lift-to-drag ratio of $L/D = 0.25$, whereas the passive probe is ballistic ($L/D = 0$). As before, the entry state is defined at the atmospheric interface altitude with a planet-relative entry velocity or $V_{R,0} = 6\,\mathrm{km\,s^{-1}}$, with a due-East initial heading angle at $18.38\,°$ latitude. The nominal values of key parameters are listed in Table 4.2 in the Mean column.

For the orbiter, the target final orbit is defined to be a 250 km altitude circular orbit. By running an FNPAG trajectory once with no dispersions (a perfect predictor), the switching time required to reach this apoapsis from the entry state described above is found to be approximately 152.6 s. This nominal trajectory results in a nominal total ΔV cost of $74\,\mathrm{m\,s^{-1}}$, as shown in Table 4.3. This total ΔV is computed as the sum of the ΔV for a periapsis raise maneuver performed at

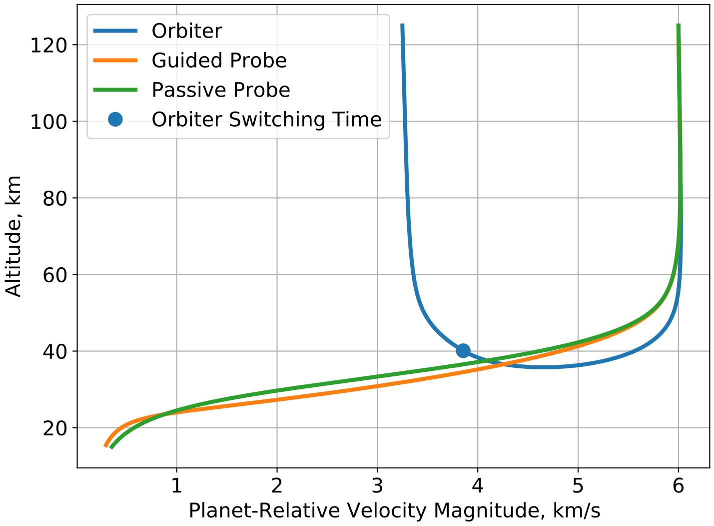


Figure 4.8: Nominal trajectories for the orbiter, guided lifting probe, and passive ballistic probe

the initial apoapsis (ΔV_1) and the ΔV for a subsequent apoapsis correction maneuver performed at the new periapsis (ΔV_2). This cost is computed as in Eq. (4.8), where r_a and r_p are the apoapsis and periapsis radii of the post-atmospheric state, respectively, and r_a^* and r_p^* are the desired apoapsis and periapsis radii, respectively.

$$\begin{aligned}\Delta V = \Delta V_1 + \Delta V_2 = &\sqrt{2\mu}\left|\left[\sqrt{\frac{1}{r_a} - \frac{1}{r_a + r_p^*}} - \sqrt{\frac{1}{r_a} - \frac{1}{r_a + r_p}}\right]\right| \\ &+ \sqrt{2\mu}\left|\left[\sqrt{\frac{1}{r_p^*} - \frac{1}{r_a^* + r_p^*}} - \sqrt{\frac{1}{r_p^*} - \frac{1}{r_a + r_p^*}}\right]\right|\end{aligned} \tag{4.8}$$

For the guided probe, the target altitude and velocity are set to 15 km and 300 m s^{-1}, respectively, and the target range is approximately 700.8 km. This target state corresponds to a Mach number of 1.3 and a dynamic pressure of 175 Pa, where the speed of sound at Mars is found by interpolating from the table provided in [162]. Depending on the specific mission design, the final state targeted by FNPEG could represent parachute deployment, retrorocket ignition, or simply a shift to some other guidance method as the entry capsule continues down to the surface. The main purpose here is to give FNPEG something to aim for so that the effect of uncertainties can be understood, rather than to design a full EDL sequence. With these target values and a perfect predictor, FNPEG computes an initial bank-angle magnitude of approximately $\sigma_0 = 139.3\,°$. This

Table 4.2: Input dispersions

Parameter	Mean	Dispersions
EFPA	$-12\,°$	$3\sigma = 0.2\,°$
Entry Velocity	$6\,\mathrm{km\,s^{-1}}$	$3\sigma = 10\,\mathrm{m\,s^{-1}}$
Orbiter β	$130\,\mathrm{kg\,m^{-2}}$	$\pm 5\,\%$
Probe β	$35\,\mathrm{kg\,m^{-2}}$	$\pm 5\,\%$
Orbiter L/D	0.25	$\pm 5\,\%$
Guided Probe L/D	0.25	$\pm 5\,\%$
Density	Mars-GRAM 2010	Mars-GRAM 2010

nominal trajectory results in zero range error, but has altitude and velocity errors of 441 m and $-5.5\,\mathrm{m\,s^{-1}}$, respectively. As mentioned in Subsection 4.3.1, here FNPEG undershoots the target velocity and overshoots the target altitude in such a way that the final energy is still correct, but these errors are relatively small and could also be further reduced by optimization of the reference trajectory. The discrepancy is mainly notable because a bias can be expected in the results under uncertainty due to these nonzero errors for the nominal trajectory.

Lastly, the passive ballistic probe has no target state because it has no variable control authority during atmospheric flight. In order to compare results with the guided probe, the passive probe's trajectory is always terminated at 15 km altitude, and the errors are defined as differences compared to the nominal velocity and range values at this altitude: $353.1\,\mathrm{m\,s^{-1}}$ and 735.0 km, respectively. Note that this means there are no performance results for altitude for the passive probe. This corresponds to a Mach number of 1.6 and a dynamic pressure of 242 Pa.

The nominal trajectories for these three vehicles are shown in Fig. 4.8, where the blue dot shows the point along the orbiter's trajectory where it switches from lift-up (σ_i) to lift-down (σ_d). The orbiter trajectory and either of the two probe trajectories constitute one representative scenario at Mars; a similar process could be followed for any of the feasible regions of the trade space identified in Section 4.2.

4.3.3 Performance Under Uncertainty

Having designed nominal trajectories for a guided orbiter, guided probe, and passive probe, the performance of these vehicles is assessed under representative uncertainties in the entry state, vehicle aerodynamics, and atmospheric density. A 1500-trial Monte Carlo analysis is performed. The mean and dispersions for each randomized input are listed in Table 4.2, where $3\sigma = X$ indicates a Gaussian distribution with zero mean and standard deviation σ, and $\pm Y\%$ indicates the bounds for a uniform distribution relative to the mean. The input parameter is computed by adding the mean and a dispersion value randomly generated from the associated probability distribution.

Variation in the entry state is simulated by independently normally dispersing the planet-relative EFPA and planet-relative entry velocity magnitude. Because the orbiter and probe are assumed to share a delivery state, the same randomly-selected entry state is used for all three vehicles for each trial. The numerical predictor component of FNPAG and FNPEG is given perfect state knowledge, including of the entry state, so the EFPA and velocity dispersions represent guidance performance under a range of initial conditions, as opposed to performance with an imperfect predictor or error between the navigated and true states. The EFPA dispersion is set equal to the delivery requirement for MSL, and the entry velocity dispersion is set to 5x the MSL requirement for the navigation knowledge accuracy used for EDL guidance system initialization [181]. The larger entry velocity dispersion is used in this study to generate a wider range of potential entry states for illustrative purposes.

Uncertainty in the vehicles' aerodynamic properties is modeled by independently uniformly dispersing ballistic coefficient and L/D. Because the orbiter and probe are separate vehicles, their aerodynamic properties are dispersed separately. However, because the passive probe is included for direct comparison to the guided probe, its ballistic coefficient is always set equal to the actual value of the guided probe's ballistic coefficient; there is no dispersion on the passive probe's L/D because it always equals zero. Unlike the entry state, the numerical predictor guidance always uses the nominal values for β and L/D, whereas the true state is propagated using the dispersed values for each trial, resulting in an imperfect predictor. The uniform ±5% dispersion for these vehicle parameters represents modeling uncertainty associated with computational fluid dynamics analysis and ballistic range testing, and is based on values used in previous studies [38].

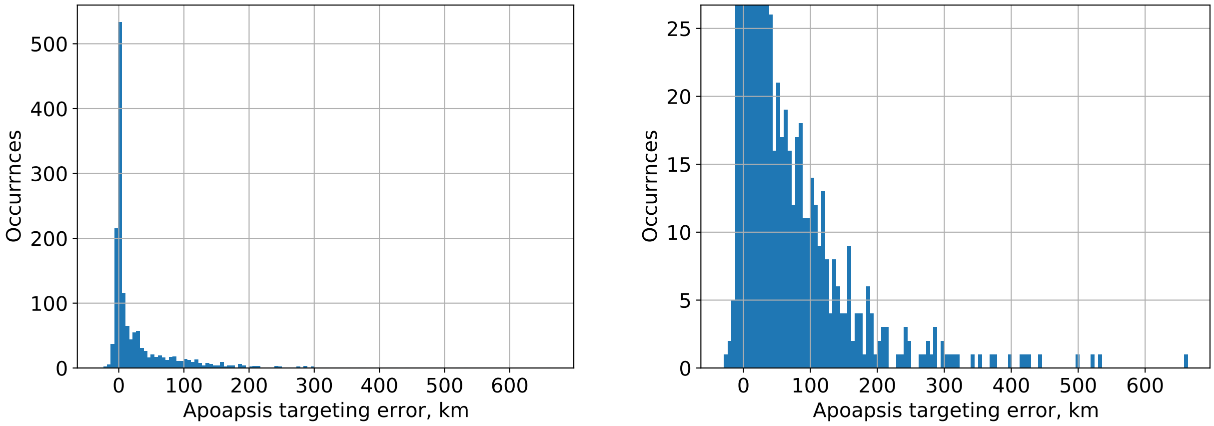
Finally, atmospheric density variability at Mars is modeled using Mars-GRAM 2010, which has a built-in capability to output randomly-perturbed correlated density profiles in a Monte Carlo sense [151]. Because the orbiter and probe would arrive simultaneously and experience the same atmosphere, the same dispersed density profile is used for all three vehicles in each trial. Differences in the density encountered at a given altitude due to different paths through the atmosphere are assumed negligible for this study. As with the aerodynamics dispersions, the guidance algorithm always uses the nominal density profile in its predictions, whereas the true state is propagated according to the dispersed density profile. The Mars-GRAM 2010 settings are generally kept at their default values, including a perturbation scale of 1 and solar radio flux at 10.7 cm of 68 sfu [151], using a date of Feb. 18 2021.

Table 4.3: Performance Results Under Uncertainty

Parameter	Nominal	Mean	Standard Deviation
Orbiter Apoapsis Error	0 km	30.87 km	65.05 km
Orbiter Total ΔV Cost	73.73 $\mathrm{m\,s^{-1}}$	86.75 $\mathrm{m\,s^{-1}}$	16.85 $\mathrm{m\,s^{-1}}$
Guided Probe Altitude Error	441.1 m	466.9 m	338.4 m
Guided Probe Range Error	0 km	−0.1821 km	1.076 km
Guided Probe Velocity Error	−5.486 $\mathrm{m\,s^{-1}}$	−5.840 $\mathrm{m\,s^{-1}}$	4.250 $\mathrm{m\,s^{-1}}$
Passive Probe Range Error	0 km	0.1483 km	9.984 km
Passive Probe Velocity Error	0 $\mathrm{m\,s^{-1}}$	0.1006 $\mathrm{m\,s^{-1}}$	11.45 $\mathrm{m\,s^{-1}}$

The results of this 1500-trial Monte Carlo analysis are summarized in Table 4.3, and histograms of error and cost parameters are shown in Figs. 4.9 – 4.12. The purpose of this analysis is to demonstrate feasibility for this mission scenario and to compare the performance of the guided and passive probes, not to precisely estimate the performance metrics of the vehicles. By numerically examining the convergence as the number of trials was increased, the quantities of interest listed in Table 4.3 are found to converge to within roughly ±5%. The mean range error for both probes and mean velocity error for the passive probe are exceptions to this statement, because as quantities with nominal values of zero and mean values near zero their percent errors are poorly behaved. The guided probe range error converges to within roughly 0.05 km, the passive probe range error to roughly 0.2 km, and the passive probe velocity error to roughly $0.5\,\mathrm{m\,s^{-1}}$.

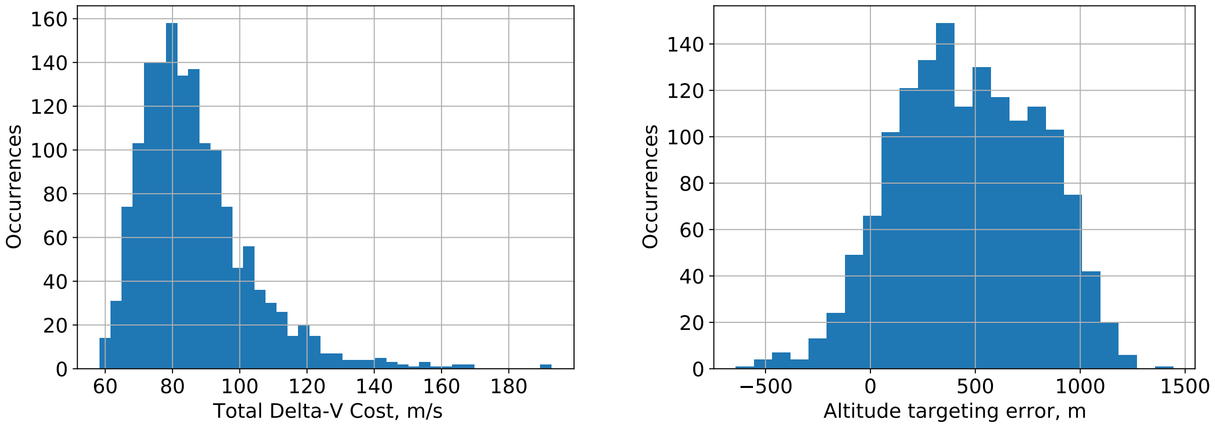
The orbiter achieves aerocapture with a 100% success rate despite a relatively large range in entry states, although some cases do significantly overshoot the desired apoapsis as seen in Fig. 4.9. Note that Fig. 4.9b is a close-up view of Fig. 4.9a in order to better see those high-error cases. These errors also lead to a positive skew in the total ΔV results shown in Fig. 4.10a, with the worst cases exceeding twice the nominal cost. Note that the total ΔV results are centered nearly one standard deviation above the nonzero nominal value. Although these errors are significant, they are not unexpected considering the relatively large entry state dispersions and the use of an imperfect predictor in the guidance algorithm. The large overshoot cases are often a result of saturation in Phase 2 of FNPAG, meaning the vehicle flies full-lift-down but is still unable to sufficiently reduce its energy, resulting in an apoapsis that is higher than desired. With better tuning of the σ_d parameter, robustness could be improved and a reduction in the high-error cases may be achieved. The performance could also be improved by adding some adaptive atmospheric estimation capability to the guidance implementation [65, 88].



(a) Histogram of apoapsis error for orbiter

(b) Zoomed-in view of Fig. 4.9a

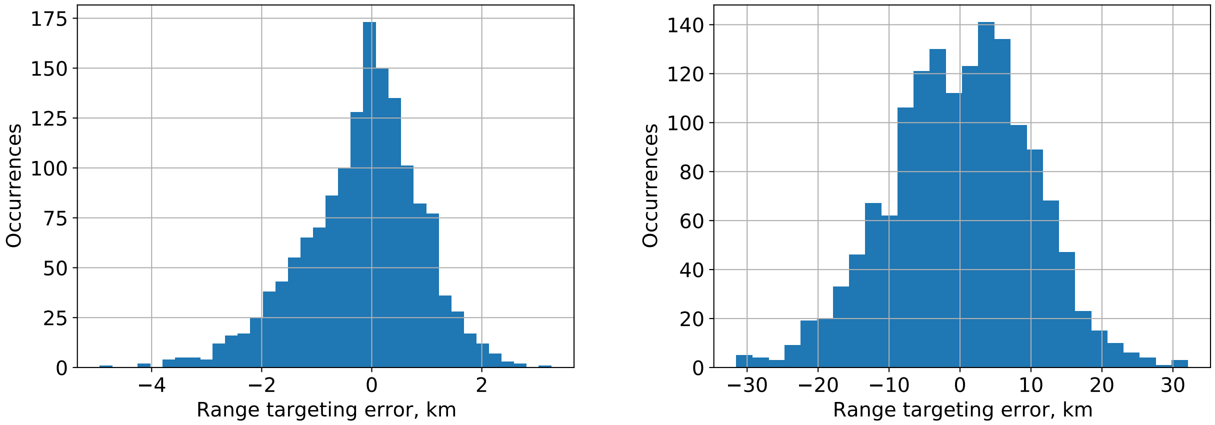
Figure 4.9: Apoapsis results for orbiter



(a) Histogram of total ΔV cost for orbiter

(b) Histogram of altitude error for guided probe

Figure 4.10: ΔV cost for orbiter and altitude error for guided probe

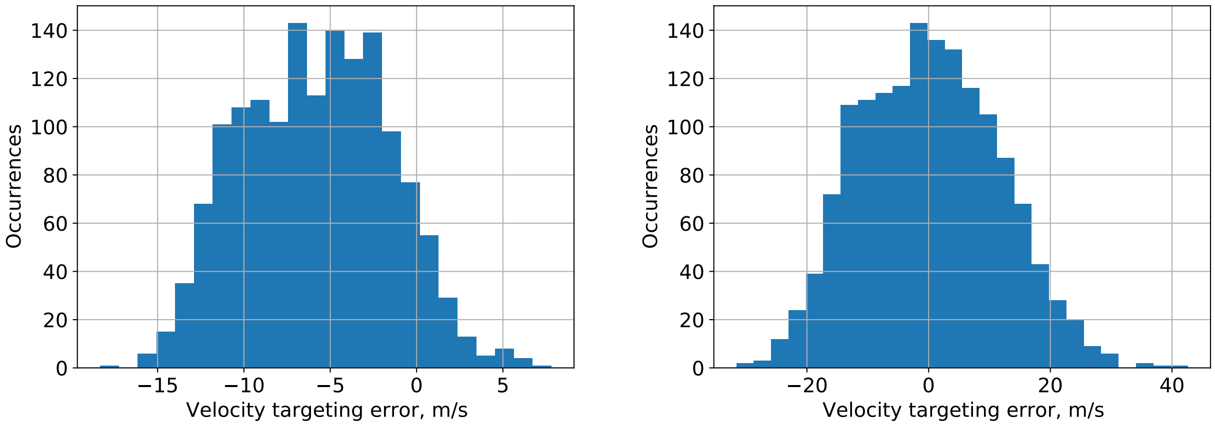


(a) Guided probe

(b) Passive probe

Figure 4.11: Target range error histograms for guided and passive probes

The altitude performance for the guided probe is shown in Fig. 4.10b, and the range and velocity performance is compared between the guided and passive probes in Figs. 4.11 and 4.12, respectively. As expected, the altitude and velocity errors for the guided probe are centered near the nonzero nominal error values. While the specific requirements for this delivery accuracy would be mission-dependent, in general FNPEG shows good performance. Particularly notable for this study is the comparison of range and velocity errors between the guided and passive probes. As expected,



(a) Guided probe

(b) Passive probe

Figure 4.12: Target velocity error histograms for guided and passive probes

the passive probe does perform much worse than the guided probe; in terms of standard deviation, the passive probe has roughly double the velocity error and roughly an order of magnitude more range error compared to the guided probe. That said, a delivery uncertainty on the order of $\pm 50\,\mathrm{km}$ range and $\pm 25\,\mathrm{m\,s^{-1}}$ velocity at $15\,\mathrm{km}$ altitude may well be acceptable for some applications. For example, if the probe were a small secondary ride-along payload targeting a broad surface region either by parachute or as an impactor, perhaps these error ranges would be sufficient.

The Monte Carlo analysis results are included to demonstrate two main conclusions. First, aerocapture and direct-entry trajectories from the same entry state are feasible even under significant navigation, vehicle, and atmospheric dispersions if the orbiter is provided some control authority and closed-loop guidance capability. Second, the probe can target a specific final state if also provided control authority and closed-loop guidance, though the final state dispersions for a passive ballistic probe may already be sufficient for some applications. The performance results presented in this section are intended to serve as a proof-of-concept for one representative scenario at Mars under uncertainty.

4.4 Conclusion

The proposed co-delivery method is an architecture for smallsat ride-along missions to interplanetary destinations. This co-delivery architecture is shown to be feasible for wide ranges of vehicle and trajectory parameters at Earth, Mars, Venus, Titan, and Neptune, subject to mission-specific heating and g-load constraints that are quantified across this trade space. An example scenario is developed using FNPAG and FNPEG closed-loop guidance for the orbiter and probe, respectively, and the vehicles' performance under uncertainty is shown to be adequate through a Monte Carlo analysis. Based on the trade space analysis and the uncertainty quantification results, passive ballistic impactor or penetrator probes as a secondary mission on an orbiter delivered by a lift-modulated aerocapture trajectory is shown to be a particularly promising configuration. A number of challenges remain for implementation, including separation design, timing and observation geometry, packaging, and tight volume and mass constraints.

Chapter 5

Co-Delivery of a Martian Probe Network

5.1 Introduction

Entry, descent, and landing (EDL) systems for Mars missions are complex, and typically involve multiple mission-critical subsystems that must operate autonomously in harsh conditions [20]. Bringing the risks associated with these subsystems down to acceptable levels is a significant engineering challenge, and this is one reason why, as the size and complexity of payloads to the Martian surface have increased over time, mission costs have also increased [209]. The top priority for Mars surface missions in this decade is Mars Sample Return (MSR), a multimission campaign with high cost and requiring significant technology development [210]. It is in this context that a community of planetary scientists and engineers is seeking lower-cost mission concepts and delivery vehicles to enable a sustained program of Mars surface exploration during and after MSR, as outlined in a recent report from the Keck Institute for Space Studies (KISS) [50].

One mission category examined by the KISS study as a potential pathway to reduced cost is networks of small, fixed landers without requirements for surface mobility and with tolerance for relatively high g-loads at landing [50]. These network missions are of growing interest for a variety of investigations, including atmospheric science and seismology [211, 50, 212, 213]. In some cases, relevant instruments can be built at small size (5-15 kg) and high g-load tolerance (1,000-2,000 Earth g's) [214, 215]. In general, for these mission concepts the probes must be delivered to a surface arrangement with roughly the right size and shape but precision landing is unimportant. Notionally, a probe network would consist of 4-8 probes delivered to Mars by a single carrier spacecraft, and networks of regional (10's of km), mid-range (100's of km), and global sizes are all potentially of interest. A wide variety of network missions for Mars have been proposed [216, 217, 102, 218, 219, 220, 221], but none have come to fruition. In most cases, these missions were cancelled early in development due to high-level budgetary and programmatic issues, influenced in part by the failures of the Mars Observer and Mars Polar Lander missions [50]; see Appendix A.3 of Ref. [50] for a brief history of Mars network mission concepts. The Mars '96 and Mars Microprobes technical failures are notable exceptions. A significant reduction in the cost and complexity of a Martian probe network could therefore improve the likelihood of selection and successful development of such a mission.

Probe network missions characterized by miniaturized instruments, high g-load tolerance, and the lack of a requirement for precision landing enable the use of small, simplified landing platforms with minimal flight-control requirements. The Small High Impact Energy Landing Device (SHIELD), illustrated in Fig. 5.1 [222], is a vehicle concept under development at the NASA Jet Propulsion Laboratory (JPL) that would meet these needs [164]. The purpose of SHIELD would be to deliver payloads of about 5 kg to the Martian surface at greatly reduced cost and complexity [164]. These reductions would be achieved by eliminating EDL subsystems wherever possible, relying entirely on a passive aeroshell-only entry system followed by a hard landing attenuated by crushable material, notionally resulting in landing decelerations on the order of 1,000 Earth g's [164]. As a point of comparison, the expected landing g-load for the Mars Microprobes, a pair of small probes designed to penetrate the Martian surface upon impact, was 30,000 g's [203].

Mission complexity may be further reduced if all of the probes could be co-delivered by a single carrier spacecraft onto their uncontrolled entry trajectories without requiring intervening translational maneuvers between probe deployments. The carrier spacecraft provides necessary resources to the probes during cruise and eliminates the need for attitude control or propulsion subsystems on the probes. The timing, magnitude, and direction of each probe's separation from the carrier spacecraft is an aspect of mission design faced with competing requirements. In the case of late probe separation, the impact of probe jettison execution error is reduced, and less battery

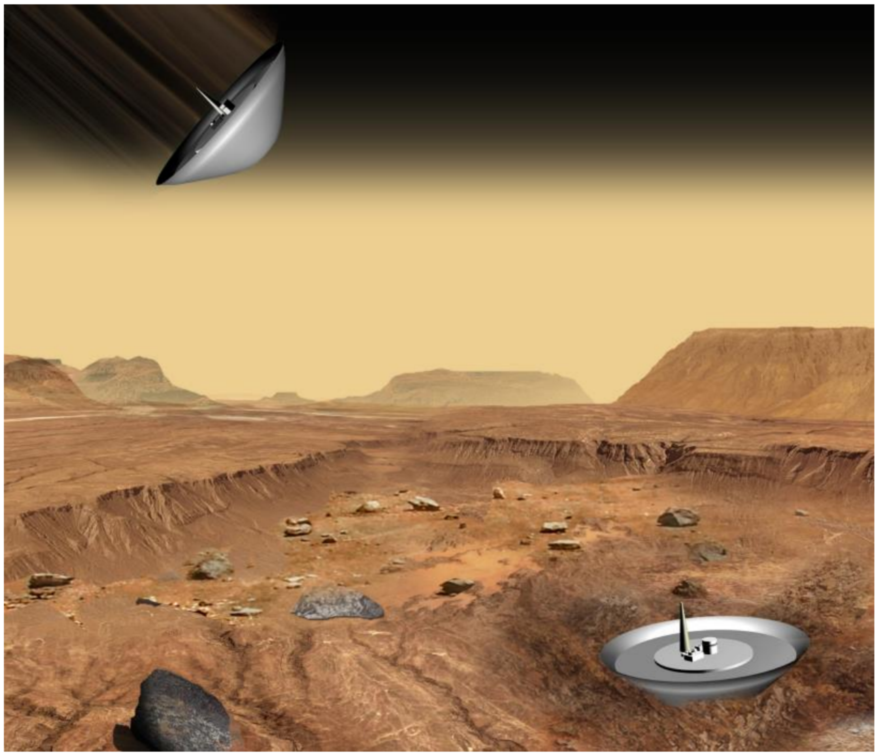


Figure 5.1: SHIELD concept image [222]

life is required for the probe to survive between separation and landing. In contrast, for an early probe separation, the required jettison speed is smaller and there is more time to estimate and correct any execution error.

Delivery of a passive probe to entry from a carrier spacecraft on a hyperbolic approach trajectory is not inherently a new architecture. The Galileo and Cassini-Huygens missions both successfully delivered probes to entry trajectories before performing orbit insertion [99, 100]. The sample return missions of Genesis [103], Stardust [104], Hayabusa [105], Hayabusa-2 [106], and OSIRIS-REx[1] all successfully delivered sample return capsules to Earth entry from a hyperbolic carrier [107], as will the Earth Entry System component of the Mars Sample Return campaign [108]. The upcoming DAVINCI mission will also include a passive probe delivered by a carrier spacecraft [223]. What all of these examples have in common, however, is that only a single probe is delivered in each case.

---

[1] NASA press release regarding successful return of OSIRIS-REx sample capsule on 9/24/23

Recent work for the Aeolus mission concept presents a design that co-delivers 20 probes to a global network on Mars from a single hyperbolic carrier, but assumes that the carrier maneuvers after each probe deployment [112]; this assumption is typical to previous studies of Mars network missions. A 2013 study demonstrates a unique method of co-delivery wherein two Phoenix-class landers enter the atmosphere together and then separate, one lander with a drag skirt and the other without [111]. This creates a discrete change in ballistic coefficient for both vehicles and is shown to achieve a 3000 km separation on the surface [111]. However, this method requires separation between two flight vehicles during hypersonic flight, a potentially risky event, and assumes a significantly larger landed mass than will be considered in this study. Broadening scope from planetary probes to include defense applications, missiles armed with multiple independently-targetable reentry vehicles are capable of delivering multiple warheads to separate locations from a single carrier vehicle on a suborbital trajectory [224]. Due to limited publicly-available information and significant differences in mission scenarios, defense applications are not further considered here.

In terms of planetary entry missions, the Pioneer Venus Multiprobe provides the most relevant example. One large probe and three small probes were delivered from a single spacecraft bus during hyperbolic approach, with the small probes accurately targeted to pre-determined entry locations separated by 8,800 to 10,400 km [109]. The Multiprobe bus first released the large probe, then performed a small maneuver, reoriented, and increased its rate of spin to 48.5 rpm [110]. The three small probes were then released simultaneously, achieving their desired separation due to the tangential velocity provided by the spinning bus [110]. This represents a unique approach to probe co-delivery without intervening maneuvering, and provides a degree of flight heritage for the concept. However, the Pioneer Venus mission design does not amount to a systematic study of co-delivery trajectory design. Recent work does provide a systematic study of co-delivery trajectories [225], but considers co-delivery of a probe with an orbiter performing aerocapture rather than multiple probes forming a network.

The purpose of this study is to broadly investigate the probe network co-delivery problem, assuming no intervening translational maneuvers and using SHIELD as an example probe design.

While literature related to specific probe networks does exist, as summarized earlier in this introduction, this article provides a more systematic study of the co-delivery problem under a set of assumptions relevant to missions of current-day interest. This work begins by presenting a flight-mechanics analysis for the SHIELD probe, considering event timing, landing accuracy, and the effect of varying entry flight-path angles. The problem of co-delivering probes to form a surface network is then considered, first for regional networks within 100 km of a central point followed by results for larger-scale networks. A linearized targeting method, inspired by B-plane targeting, is introduced for the design of regional networks and its limitations are quantified. Monte Carlo analyses are performed for both regional and large-scale networks to capture the impact of relevant uncertainties, including probe jettison execution error, on the feasibility of the computed co-delivery trajectories.

5.2 Models and Assumptions

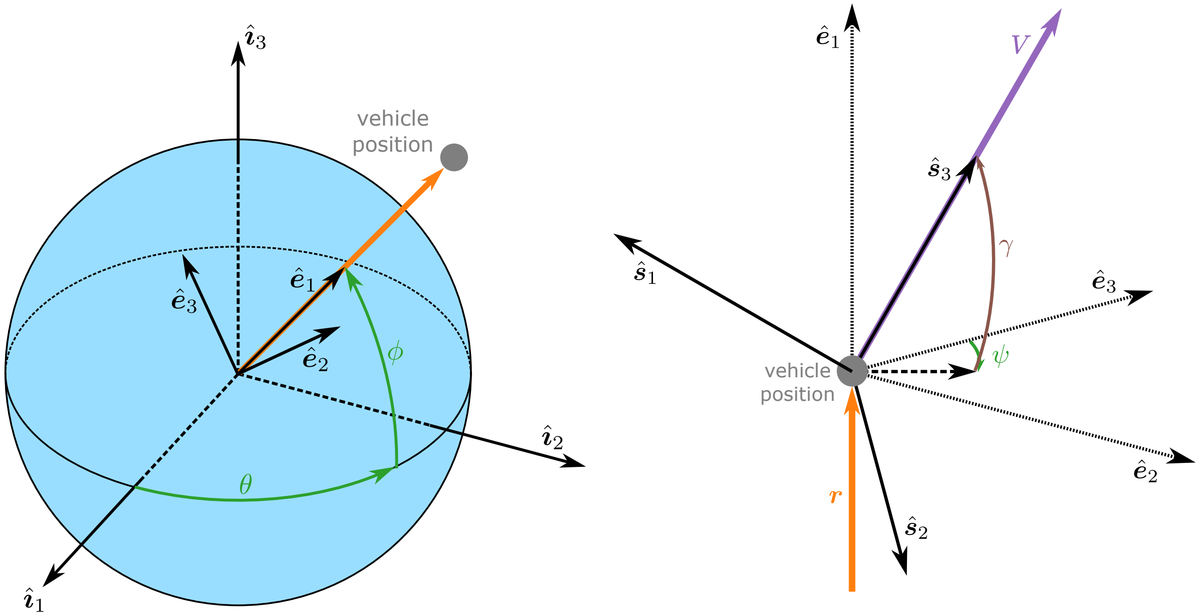
An assumed design goal in this work is that precision landing is not required, but the network should approximate a desired distribution and location on the surface. Additional assumptions regarding probe co-delivery include:

- Each probe is a ballistic rough lander, and is passive other than drag skirt deployment and heatshield jettison.
- The probes approach Mars on a single carrier spacecraft on an entry trajectory, and the separation events do not change the carrier's trajectory and no other maneuvers are performed. However, changes in carrier attitude between separation events are allowed.
- The probes separate from the carrier mechanically.
- Probe jettisons occur between 0.25 and 20 days before atmospheric entry.
- The carrier spacecraft has an approach trajectory such that the magnitude of the planet-relative velocity at the atmospheric entry interface altitude of 125 km is 6 km/s.

Approach trajectories in this study are defined by their state at atmospheric entry interface, that is, the position and velocity of the carrier spacecraft at 125 km altitude. This state is defined by altitude h, longitude θ, latitude ϕ, planet-relative velocity V, flight-path angle γ, and heading angle ψ; flight-path angle is the angle between the velocity vector and local horizon, and heading angle is the angle between the horizontal projection of the velocity vector and a due-North vector in that same plane (e.g. a 90° heading angle is due-east). Figure 5.2 provides a visualization for these definitions, where the $\hat{\boldsymbol{\imath}}$ basis vectors form a planet-fixed frame, $\hat{\boldsymbol{e}}_1$ is aligned with the position vector of the spacecraft, and $\hat{\boldsymbol{s}}_3$ is aligned with the planet-relative velocity vector. The central landing site is then the point on the surface where a SHIELD probe would nominally land after continuing on this trajectory. Two things should be noted about this convention. First, because of the probe jettison velocities, each probe will actually enter the atmosphere with different states, potentially resulting in significantly different entry flight-path angle and entry velocity values. Second, the carrier spacecraft would not itself be a SHIELD probe and could perform a divert maneuver or intentionally burn up in the atmosphere; the carrier's entry state and central landing site are simply convenient ways to define the approach trajectory and a reference point on the surface, respectively.

Separation events are assumed to impart an impulsive change in velocity to the probe, where the jettison velocity $\boldsymbol{V}_j$ is defined as the velocity of the probe relative to the carrier the moment after separation, and jettison speed is defined as the magnitude $V_j = |\boldsymbol{V}_j|$. This notation is used to distinguish from impulsive $\Delta\boldsymbol{V}$ because, while they are theoretically equivalent events, this study assumes jettisons occur mechanically (e.g. a spring jettison) rather than propulsively.

Though mostly passive, SHIELD does go through three different configurations from atmospheric interface to surface. First, in its *entry* configuration, SHIELD is entirely within its protected aeroshell, and this configuration is maintained through the hypersonic and high-heating portion of the flight. Next, SHIELD enters the *descent* configuration soon after beginning subsonic flight by deploying a drag skirt, the purpose of which is to reduce the terminal velocity of the vehicle. Shortly thereafter, the *landing* configuration is initiated with jettison of the heatshield. SHIELD



(a) Planet-fixed and position frames

(b) Position and velocity frames

Figure 5.2: Frame definitions

is assumed to fly at a trim zero angle of attack and has an axisymmetric shape with balanced center of mass, resulting in a lift-to-drag ratio of $L/D = 0$. The drag properties of SHIELD are linearly interpolated based on Mach number from tabular data provided by the JPL SHIELD team, resulting in ballistic coefficients ranging from about $20\,\mathrm{kg\,m^{-2}}$ in the entry configuration down to around $5\,\mathrm{kg\,m^{-2}}$ in the landing configuration. Ballistic coefficient describes the ratio of inertial to aerodynamic forces and is defined as $\beta = m/(C_D A)$, where m is mass, C_D is drag coefficient, and A is aerodynamic reference area. SHIELD is assumed to have a nose radius of $R_n = 0.85\,\mathrm{m}$.

Table 5.1 summarizes the relevant uncertainties applied throughout this study. Uncertainty in the approach trajectory of the carrier spacecraft is modeled by dispersing the state at atmospheric entry interface for each trial, then back-propagating the dispersed state to the time of first jettison. Interplanetary navigation to Mars has advanced to the extent that its contribution to the landing error is small compared to the impacts of atmospheric variability and aerodynamic modeling errors,[2] even for unguided entry. For example, the navigation-only errors for the Mars Exploration Rovers

---

[2] Note, however, that this statement assumes significant Deep Space Network coverage during approach, which

Table 5.1: Monte Carlo analysis input dispersions

Parameter	Dispersion
entry interface flight-path angle γ_0	$3\sigma = 0.2°$
entry interface velocity magnitude V_0	$3\sigma = 2\text{m/s}$
atmospheric density ρ	MarsGRAM 2010
probe ballistic coefficient β	±5%
jettison speed V_j	±10%

(MER) were 3.3 km for *Spirit* and 9.7 km for *Opportunity* [226]. Furthermore, the large majority of landing error is in the downrange direction; the final landing ellipses predicted for MER due to all error sources had crossrange components below 5 km, compared to approximately 60 km in downrange [226], indicating high accuracy in heading angle at entry. The driving requirement for approach navigation is precise targeting of the entry interface flight-path angle [226, 181, 165], γ_0, because even small variations can have a significant effect on the altitude-velocity entry profile [78]. The velocity magnitude at entry interface, V_0, is also relevant because of its impact on key quantities like peak heat flux and peak deceleration. These two entry state components are dispersed independently according to Gaussian distributions centered at the nominal value and with some standard deviation, σ. For this study, the 3σ value for γ_0 is set equal to the requirement on delivery error for MSL, and the 3σ value for V_0 is set equal to the required knowledge accuracy at EDL guidance system initialization for MSL [181]. In contrast, minor errors in entry position and heading angle have very little impact on the altitude-velocity profiles of the probes, and thus primarily contribute a small center error for the network without adding significant shape error. Recall that it is an assumption of this study that small errors in network center are unimportant compared to the distribution of the probes. Therefore, the longitude, latitude, and heading angle at entry interface are not dispersed in this study.

Variability of atmospheric density is modeled by using random profiles of density vs. altitude that are generated using the 2010 version of the Mars Global Reference Atmospheric Model (Mars-GRAM 2010) [151]. For a given trial, the dispersions on atmosphere, γ_0, and V_0 are applied once,

---

may potentially be a limiting factor for small missions.

such that all probes experience the same atmosphere and carrier spacecraft trajectory.

The ballistic coefficient of each probe is dispersed along a uniform distribution with bounds at ±5% of the nominal value. This value is representative of the confidence level provided by computational fluid dynamics and ballistic range testing, and is chosen based on previous studies [38, 225]. The lift-to-drag ratio always remains at its nominal value of zero, assuming that axisymmetric spin removes the effect of any small, unintended lift force. Finally, the magnitude of the jettison event is dispersed along a uniform distribution with bounds at ±10% of the nominal value under the assumption that a separation mechanism could be designed to within this uncertainty level; through discussions with mission engineers, this was judged to be a conservative assumption. The direction of the jettison velocities are assumed to be nominal for the purpose of this study. These two dispersions are applied independently to each probe for each trial. Finally, note that although carrier entry longitude, latitude, and heading angle are not dispersed, individual probes may have off-nominal values of these parameters due to jettison speed dispersions, and these effects are accounted for.

Trajectories are computed via numerical propagation of the three degree-of-freedom equations of motion for a rotating ellipsoidal planet using explicit Runge-Kutta integration of order 4(5) with relative and absolute tolerances equal to 1×10^{-11} [78]. Mars is assumed to have gravitational parameter $\mu = 4.305 \times 10^4\ \mathrm{km}^3\,\mathrm{s}^{-2}$, equatorial radius $R = 3397.2\,\mathrm{km}$, oblateness spherical harmonic coefficient $J_2 = 0.001964$, and a planetary rotation period of $T = 1.02595675$ days [161]. The speed of sound for the Martian atmosphere, which is used to compute Mach number, is interpolated from a nominal tabular model[162]. Heat flux is modeled by computing convective heat flux $\dot{q}$ at the stagnation point assuming a fully catalytic surface using the Sutton-Graves expression shown in Eq. (5.1) [201], where ρ is density and a value of the heating coefficient $k = 1.904 \times 10^{-4}\,\mathrm{kg}^{0.5}/\mathrm{m}$ is used based on nominal atmospheric composition at Mars [163].

$$\dot{q} = k\sqrt{\frac{\rho}{R_n}}V^3 \tag{5.1}$$

Finally, sensed deceleration (or g-load) is computed as $g = \sqrt{L^2 + D^2}/g_0$ where L and D are the accelerations due to lift and drag, respectively, and g_0 is the standard acceleration due to gravity at the Earth's surface.

5.3 SHIELD Flight Mechanics

Before investigating co-delivery of networks, a flight mechanics analysis is presented for the atmospheric flight of a single SHIELD probe in order to determine feasible nominal values for drag skirt deployment and heat shield jettison. This analysis also assesses the robustness to uncertainty of time-triggered configuration changes and the possibility of using drag skirt deployment timing as a method of control. Analysis is performed at several representative entry interface flight-path angles: $-12°$, $-18°$, and $-24°$.

First, EDL event timing is considered. Drag skirt deployment and heatshield jettison are constrained by three parameters: maximum Mach number at drag skirt deployment, maximum impact velocity, and minimum time between deployment and jettison. The assumed values[3] for these requirements are summarized in Table 5.2. The combined result of these parameters defines an acceptable range for the timing of each event for any entry trajectory, and the nominal event times can then be selected from within this range. The resulting bounds on event timing were computed for $\gamma_0 = -18°$, and were found to be 105.7 seconds after entry (denoted E+105.7 s) for earliest deployment and E+170.9 s for latest jettison, where in this context entry is defined as the point at which sensed deceleration first exceeds one Earth g. Nominal event times of E+140 s and E+150 s were then selected on the basis of being well-within this acceptable range, and the resulting trajectory is shown in Fig. 5.3. Similar analysis was performed for the other values of γ_0, with results summarized in Table 5.3.

EDL events are often triggered by processed sensor data, such as commanding parachute deployment using either a velocity trigger or range trigger [227]. For SHIELD, however, the goal of eliminating subsystems wherever possible motivates the following question: would a simple onboard timer be sufficient to trigger drag skirt deployment and heatshield jettison without violating the

---

[3] based on discussions with the SHIELD team at JPL

Table 5.2: Summary of SHIELD EDL requirements

Parameter	Requirement
Mach number at drag skirt deployment	≤ 0.9
Time between drag skirt deployment and heatshield jettison	$\geq 4\,\mathrm{s}$
Impact velocity	$\leq 50\,\mathrm{m/s}$

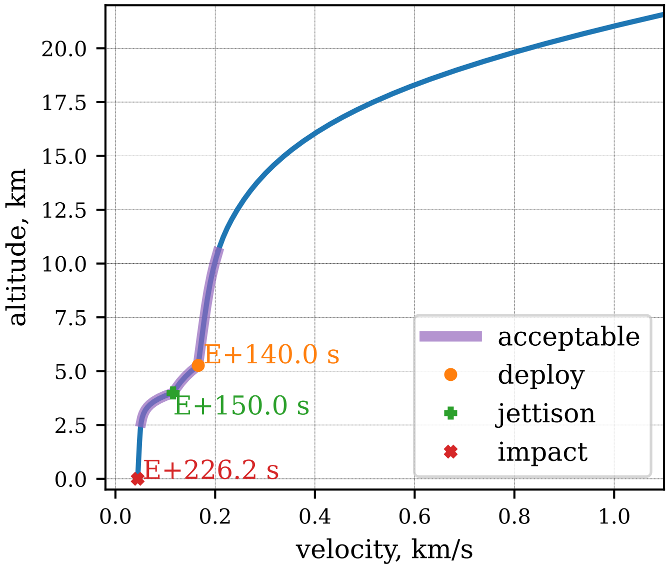


Figure 5.3: Nominal trajectory, with event timing annotated, for a SHIELD entry at −18°.

assumed requirements when relevant uncertainties are applied? If so, this could simplify EDL for SHIELD even further.

A 1000-trial Monte Carlo analysis is performed at each of the γ_0 values of interest to capture the impact of relevant uncertainties on SHIELD flight-mechanics. In each trial, drag skirt deployment and heatshield jettison are triggered once the nominal time after entry is reached, but the conditions at those points along the trajectory vary due to the input dispersions. Figure 5.4 shows the resulting Mach numbers at deployment; as can be seen from the histogram, none of the cases for any of the γ_0 values exceeded the 0.9 maximum. The requirement on impact velocity was also met, with the maximum value for any of the 3000 total trials being 45.9 m/s; in fact, impact velocity varied so little that the histograms become unhelpful visualizations and are thus not shown.

Table 5.3: EDL event timing, in terms of seconds after entry

γ_0	Earliest Deployment	Latest Jettison	Nominal Deployment	Nominal Jettison
$-12°$	193.9 s	258.2 s	225 s	235 s
$-18°$	105.7 s	170.9 s	140 s	150 s
$-24°$	76.9 s	132.8 s	105 s	115 s

This is because the probes always proceeded through the EDL stages on time to reach terminal velocity, which only varied slightly. Thus, a simple timer is sufficient to trigger EDL events while meeting requirements under the assumptions on this study, but further work would be necessary to determine if this result holds under larger uncertainties and more realistic Mars surface modeling.

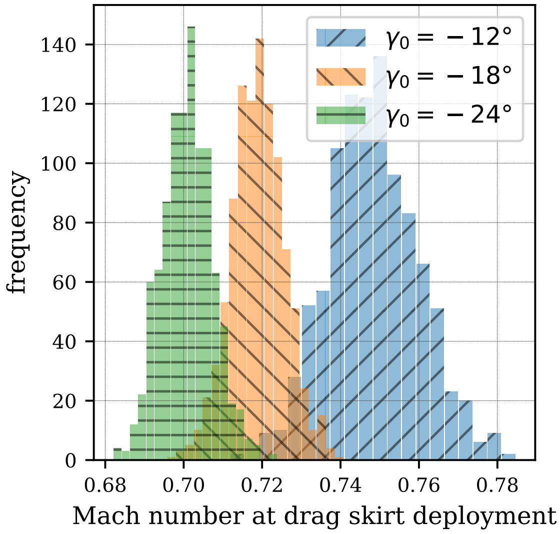
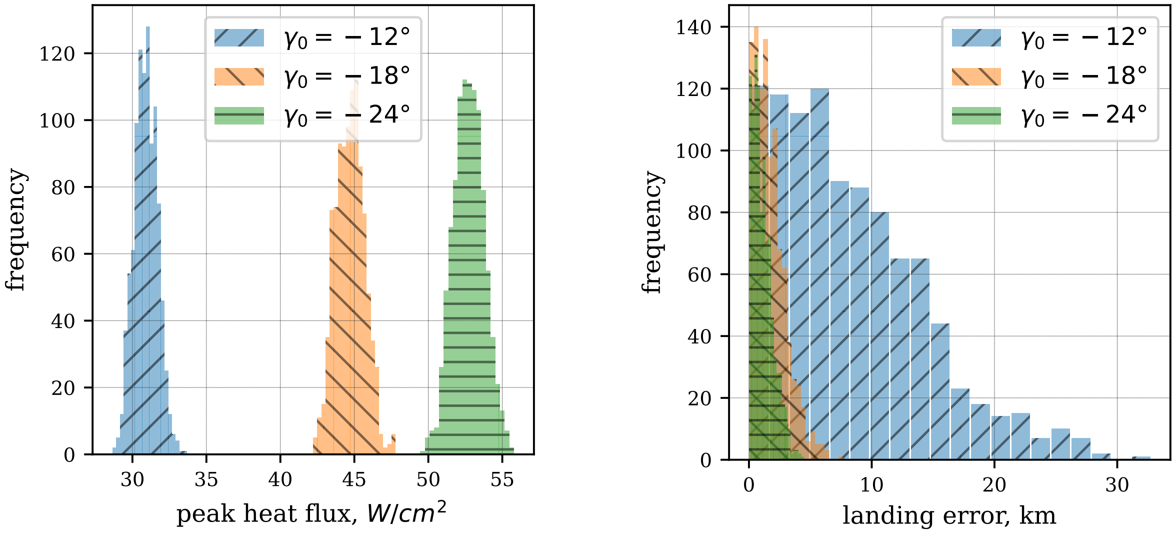


Figure 5.4: Monte Carlo results for Mach number at drag skirt deployment at varying γ_0 values.

Another relevant constraint, peak heat flux, is reported in Fig. 5.5a. As expected, steeper γ_0 values result in higher heating as deceleration occurs more rapidly. No requirement on peak heat flux is assumed for SHIELD in this study, but these values are well-within the capabilities of modern thermal protection systems such as SLA-561V, which has heritage for Mars entry missions and has demonstrated good performance at heat fluxes up to 240 W/cm^2 [228]. Additionally, histograms of landing error are shown in Fig. 5.5b, where error is defined as the distance between the nominal and actual landing sites. Note that there is a major decrease in landing error as the entry flight-path angle gets steeper from $-12°$ to $-18°$, and that while there is a further decrease for $\gamma_0 = -24°$ the returns are diminishing after some point. Table 5.4 summarizes the results of these Monte Carlo analyses, where σ is standard deviation.



(a) Peak heat flux

(b) Landing error

Figure 5.5: Monte Carlo results at varying γ_0 values

Finally, this study also examined the possibility of using drag skirt deployment and heatshield jettison as a method of range control. By carefully timing these events based on the difference between the nominal and estimated states for the current time, the vehicle could in theory control its in-plane terminal range during entry. In practice, however, the requirement that the drag skirt deploy in subsonic conditions severely limits the total achievable control authority, to the extent that this approach has little or no merit for this application. This is because by the time the vehicle reaches subsonic speeds it has already dissipated almost all of its energy and is at a low altitude (about 10 km in this case), leaving little time or energy for the change in ballistic coefficient to significantly impact the landing site. Specifically, for two trajectories at $\gamma_0 = -18°$, one with the earliest permissible deployment and jettison times and the other with the latest permissible

Table 5.4: Summary of Monte Carlo results for EDL of a single SHIELD probe

	Deployment Mach		Impact Velocity, m/s		Peak Heat Flux, W/cm^2		Error, km	
γ_0	Mean	σ	Mean	σ	Mean	σ	Mean	σ
$-12°$	0.748	0.0115	44.6	0.801	30.9	0.777	8.29	6.06
$-18°$	0.719	0.00672	44.7	0.792	44.7	0.962	1.70	1.27
$-24°$	0.701	0.00636	44.6	0.783	52.7	1.04	1.12	0.829

times, the two trajectories land only about 3 km apart. This represents the maximum possible control authority yielded by this method, and because this is well below the expected landing site dispersions the approach is discarded as a method of flight-control. If the drag skirt could be deployed at supersonic or hypersonic speeds the control authority would increase substantially and this method would merit reexamination. Indeed, jettison of a drag skirt during hypersonic flight has been shown in the literature to be an effective method of range control for entry at Mars [229, 230]. However, the drag skirt for the current SHIELD concept would not structurally or thermally withstand such conditions[4] .

5.4 Regional Probe Networks

This section investigates probe jettison velocity design and performs uncertainty quantification for regional networks, which are loosely defined as having all probes within 100 km of the central landing site. Because the changes in trajectory to achieve these separations are relatively small, the relationship between separation time and separation distance, as well as that between jettison speed and separation distance, is roughly linear[231]. A linearized numerical targeting method is therefore developed and employed to design probe jettisons for a reference network. These trajectories are then subjected to relevant uncertainties to quantify the impact of these dispersions on probe landing locations.

---

[4] based on discussions with the SHIELD team at JPL

5.4.1 Linearized Targeting Method

The linearized targeting method for regional networks is summarized as follows. Assume $\boldsymbol{x}_{\theta\phi} = [\theta, \phi]^T$ to be the landing site coordinates and $\boldsymbol{V} = \boldsymbol{V}(t)$ to be the velocity of the probe at some time prior to landing. Apply a Taylor series expansion to $\boldsymbol{x}_{\theta\phi}$ about the trajectory of the carrier spacecraft, $\boldsymbol{x}^*_{\theta\phi}$, as a function of velocity, then neglect terms of second order or higher:

$$\boldsymbol{x}_{\theta\phi} = \boldsymbol{x}^*_{\theta\phi} + \left.\frac{\partial \boldsymbol{x}_{\theta\phi}}{\partial \boldsymbol{V}}\right|_* (\boldsymbol{V} - \boldsymbol{V}^*) + \text{H.O.T.s} \tag{5.2}$$

$$\Delta\boldsymbol{x}_{\theta\phi} \approx \left.\frac{\partial \boldsymbol{x}_{\theta\phi}}{\partial \boldsymbol{V}}\right|_* \boldsymbol{V}_j = [\boldsymbol{J}]\boldsymbol{V}_j \tag{5.3}$$

$$[\boldsymbol{J}] = \begin{bmatrix} \frac{\partial\theta}{\partial V_x} & \frac{\partial\theta}{\partial V_y} & \frac{\partial\theta}{\partial V_z} \\ \frac{\partial\phi}{\partial V_x} & \frac{\partial\phi}{\partial V_y} & \frac{\partial\phi}{\partial V_z} \end{bmatrix}_* \tag{5.4}$$

where the jettison velocity is the velocity of the probe minus the velocity of the carrier spacecraft at the moment after jettison, $\boldsymbol{V}_j = \boldsymbol{V} - \boldsymbol{V}^*$. The Jacobian matrix $[\boldsymbol{J}]$ can then be evaluated for any value of jettison time to represent the sensitivity of landing site coordinates to velocity at that time. By inverting the Jacobian, the $\boldsymbol{V}_j$ vector required to achieve a desired change in landing location, $\Delta\boldsymbol{x}_{\theta\phi}$, can be linearly approximated. Because the Jacobian in this case is not square, the least-norm solution is selected to minimize the V_j magnitude.

$$\boldsymbol{V}_j = [\boldsymbol{J}]^T([\boldsymbol{J}][\boldsymbol{J}]^T)^{-1}\Delta\boldsymbol{x}_{\theta\phi} \tag{5.5}$$

For the purpose of this study, $[\boldsymbol{J}]$ is numerically approximated using first-order forward finite differencing; Eq. (5.6) gives an example for the first element of the matrix,

$$\frac{\partial\theta}{\partial V_x} = \frac{\theta_p - \theta^*}{\Delta V_x}, \tag{5.6}$$

where ΔV_x is a small velocity perturbation in the x-axis direction and θ_p is the landing site longitude that results from applying a jettison velocity of $[\Delta V_x, 0, 0]^T$ then propagating to surface impact. In this study, a perturbation value of $\Delta V_x = \Delta V_y = \Delta V_z = 1 \times 10^{-4}\,\mathrm{m\,s^{-1}}$ was selected based on trial and error. Numerically computing the Jacobian $[\boldsymbol{J}]$ according to Eqs. 5.4 and 5.6 allows one to linearly approximate the jettison velocity vector $\boldsymbol{V}_j$ required to achieve a shift in longitude and latitude equal to $\Delta\boldsymbol{x}_{\theta\phi} = [\Delta\theta, \Delta\phi]^T$ for a given separation time. To consider a different jettison time, the Jacobian is simply re-evaluated by applying perturbations at that time.

5.4.2 Reference Network Design

This linearized targeting method is employed to design probe jettisons for a regional network. As a motivating example, the reference science mission is a seismology network deployed to Cerberus Fossae, a region of known seismicity on Mars [232, 233]. A regional network in such an area can obtain useful geophysical measurements using significantly lower-sensitivity seismometers than a global network would require by relying on its proximity to seismic events, thereby bringing the required payload mass down to the range of 2–3 kg per lander[5] [234]. Shock-tolerant seismology payloads have been developed that can survive 15,000 g's at impact [235], and precision landing of probes is significantly less important than achieving a network geometry that permits observability.

To target Cerberus Fossae, the central state at atmospheric entry interface is defined to have a longitude of 150° East, latitude of 7.25° North, flight-path angle of $\gamma_0 = -12°$, and heading angle of 80° (slightly northward of due-east). The network consists of three pairs of probes such that each pair is targeted with equal and opposite jettison velocities $\pm\boldsymbol{V}_j$, resulting in a symmetrical network of six probes. The nominal configuration is shown to-scale in Fig. 5.6[6] . Note that the distance corresponding to the angular separation, shown on the top and right axes of Fig. 5.6, is computed as the great-circle distance according to Eq. (5.10) and assumes a spherical planetary surface. Because Mars is not perfectly spherical, this distance differs from the true distance along the surface and should be treated as approximate for large separations; specifically, for separation angles larger than 15° the difference could exceed 5 km.

The direction of the probe jettison velocities is not constrained to be equal, and thus changes in attitude may be required between probe jettisons. Because the separations are assumed to occur mechanically, designing for a uniform jettison speed is likely more desirable than an outright

---

[5] This assumption on total payload mass for a seismometer of the required sensitivity is based on the short-period instrument aboard the InSight lander and private communications with Mark Panning, Dec. 2021.

[6] The surface image is a mosaic created from data acquired from the Context Camera aboard NASA's Mars Reconnaissance Orbiter and generated using MarsTrek, trek.nasa.gov/mars

minimization of jettison speed. In order to deploy the network using a single jettison speed for all three separations, targeting is performed along a range of separation times to provide trends of required V_j; these results are shown in Fig. 5.7. As would be expected, required jettison speed increases dramatically as time between separation and entry approaches zero. A jettison speed of $10\,\mathrm{cm\,s^{-1}}$ is selected as a relatively low value that intersects all three curves between 0.5 and 5 days before entry; note that iterating between these results and the design of the network allows for flexibility in selection of the nominal jettison speed. A root solver is used to compute the precise jettison time for each probe that targets the desired landing location and results in a jettison speed equal to the desired value, with the approximate intersections of the curves with the dashed line in Fig. 5.7 providing good initial guesses for the solver. Specifically, Brent's method[7] is applied to within numerical precision with a maximum of 10 iterations[236, 237]. The resulting separation times are 0.7613, 1.6529, and 3.1697 days before entry.

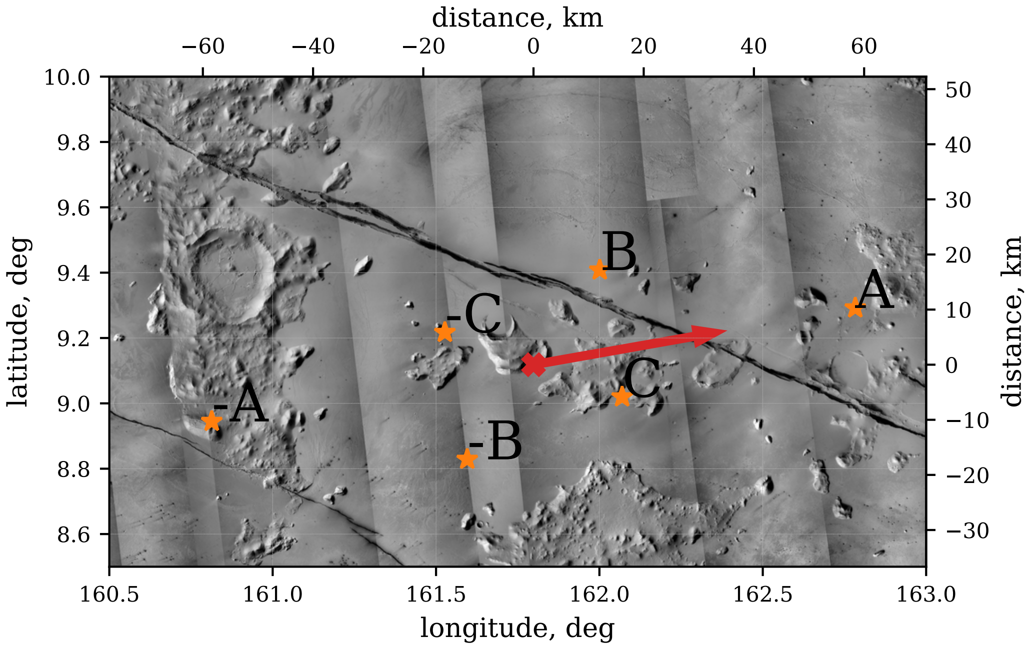


Figure 5.6: Nominal landing locations for example network, with downrange direction shown by red arrow and central point shown by red X

---

[7] scipy.optimize.brentq

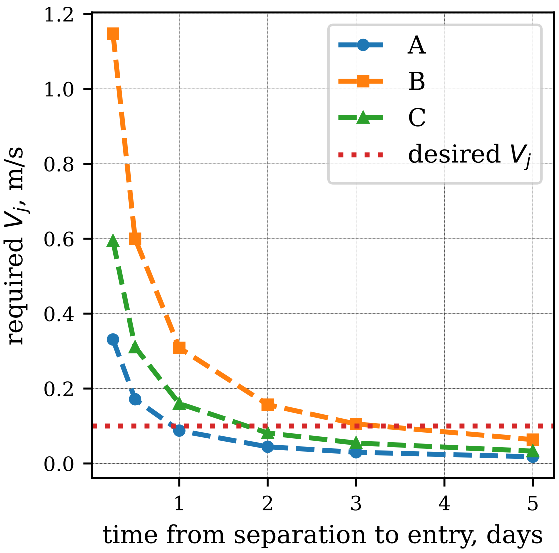


Figure 5.7: Required V_j vs. separation time for the three chosen nominal landing sites

5.4.3 Error Parameters

The assumed mission goal is to deploy a network in approximately the correct geometry and location, rather than to precisely target each probe. Thus, the separate statistics of landing error for each probe do not directly relate to the performance requirements. To better characterize network delivery performance, two error parameters are defined, center error ε_c and shape error ε_s. Center error describes off-nominal location of the center of the network, and shape error describes off-nominal distribution of probes around that center. Define center error for any given trial as follows:

$$\varepsilon_c = \sqrt{(\bar{\theta}^* - \bar{\theta})^2 + (\bar{\phi}^* - \bar{\phi})^2} \tag{5.7}$$

where $\bar{\theta}^*$ and $\bar{\phi}^*$ are the average longitude and latitude, respectively, across all probe locations for the *nominal* network design, and $\bar{\theta}$ and $\bar{\phi}$ are the average longitude and latitude of the actual probe landing sites. This error is computed in radians and can be converted to distance by multiplying by the planet's radius. To calculate the shape error, compute the great circle distance between every unique pair of landing sites, yielding $d = N(N-1)/2$ distances for N probes, and label these values

δ_i^* and δ_i for the nominal and actual landing sites, respectively. The shape error is then defined as the root sum squared of the differences between the nominal distance and actual distance for each unique pair of landing sites, divided by the total number of probes N:

$$\varepsilon_s = \frac{\sqrt{(\delta_1^* - \delta_1)^2 + \ldots + (\delta_d^* - \delta_d)^2}}{N}, \qquad d = \frac{N(N-1)}{2} \tag{5.8}$$

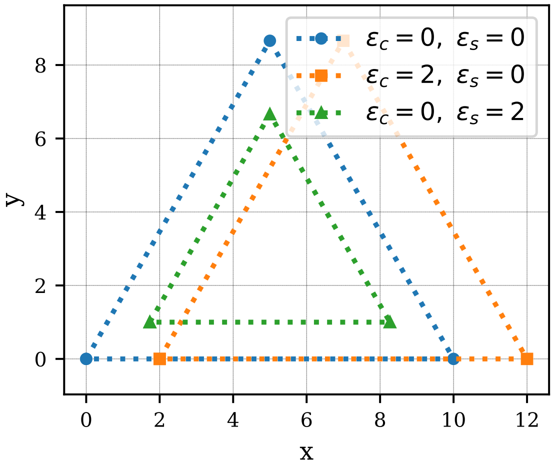


Figure 5.8: Three example networks, illustrating center error ε_c vs. shape error ε_s

A more intuitive representation of these parameters is provided by Fig. 5.8, which shows a generic nominal network in blue circles. The orange squares have the correct network shape but all points are shifted to the right, resulting in $\varepsilon_c = 2$ and $\varepsilon_s = 0$. The green triangles are centered correctly but the entire geometry has been reduced in size, resulting in $\varepsilon_c = 0$ and $\varepsilon_s = 2$.

5.4.4 Dispersion Analysis

The results of applying relevant uncertainties to this reference scenario in a 1000-trial Monte Carlo analysis are shown in Fig. 5.9. As expected based on intuition and the earlier flight mechanics analysis, the probes experience large dispersions in landing site, primarily in the downrange direction. However, it turns out that these dispersions are highly-correlated between probes for any given trial, because all dispersions except jettison speed and ballistic coefficient apply to the

Table 5.5: Statistics of error parameters

Parameter	Mean	3σ
center error ε_c, km	7.391	16.573
shape error ε_s, km	3.335	3.555
min. separation, km	22.029	2.569
max. separation, km	119.370	22.554
avg. separation, km	52.491	8.126

trial as a whole and affect all of the probes in more or less the same way. For this scenario, dispersions on jettison speed have relatively little effect because the nominal speeds are low enough to be within the regime of roughly linear sensitivity for these trajectories. As a result of all this, the network shifts back-and-forth in downrange but its shape deforms relatively little. This is reflected in the statistics of center and shape error, summarized in Table 5.5; the minimum, maximum, and average distances between every unique pair of landing sites are computed for each trial and the statistics of these values are also shown in Table 5.5. The landing locations for the trial with the largest shape error are shown in the red pentagons in Fig. 5.9. It can be seen by inspection that the network shape in this trial is qualitatively similar to the nominal shape, but with an offset in the positive downrange direction. The key takeaway is that, for the example regional network considered here, the probes can be delivered to roughly the desired arrangement on the surface despite large dispersions for each individual probe, so long as roughly ±25 km downrange shifts of the entire network can be tolerated.

5.4.5 Limits of Linearization

The linear approach to targeting applied in this section is a good approximation only within some local region of the reference trajectory, that is, near the approach trajectory of the carrier spacecraft leading to the central entry point. Thus, it is important to quantify the limits of applicability for the linearization. To do so, the linearized targeting method is applied to compute a probe jettison velocity targeting progressively greater offsets from the central point in both the downrange and crossrange directions, assuming separation one day before entry along the

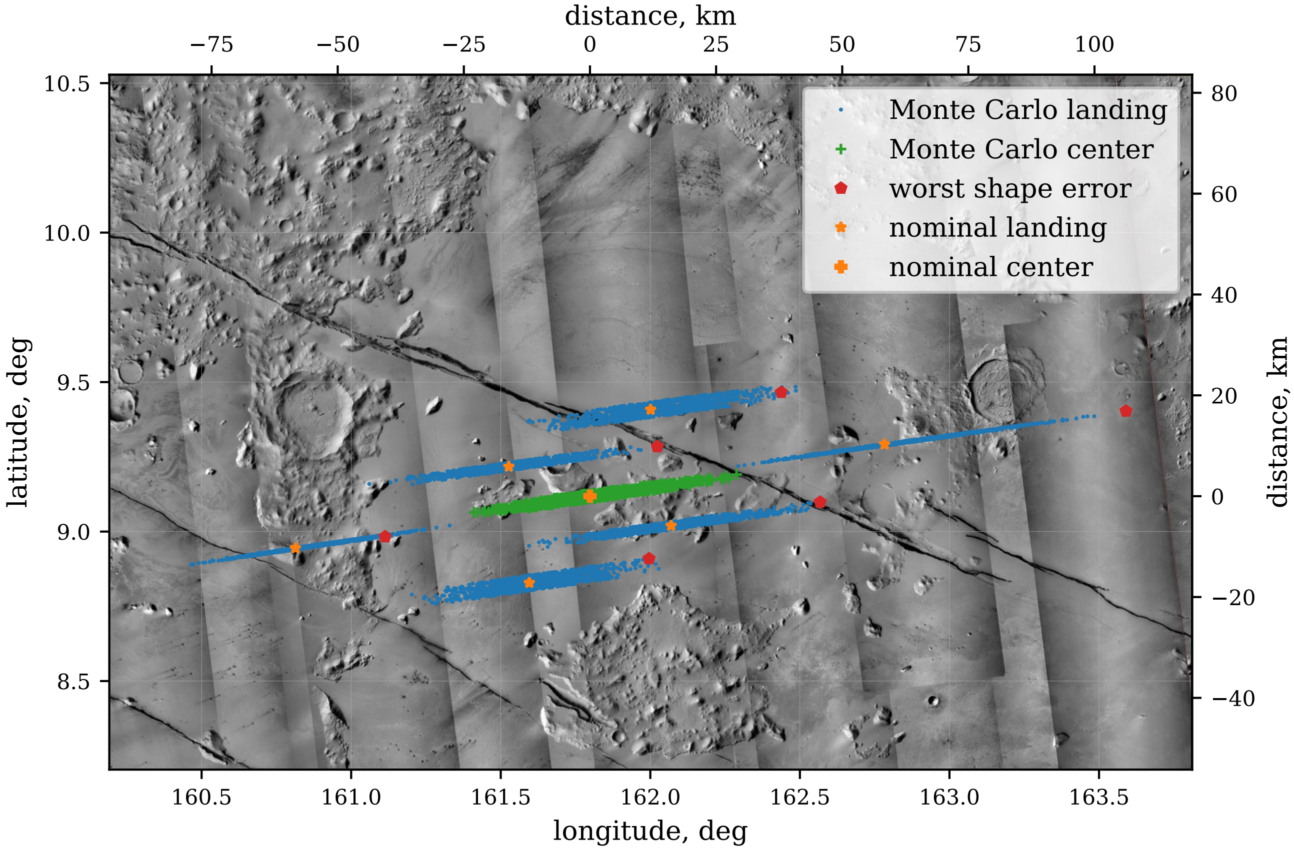


Figure 5.9: Nominal and random trial landing locations shown against to-scale Martian surface.

approach trajectory defined in Subsection 5.4.2. A trajectory is then simulated for each probe jettison, and the great-circle distance between the achieved landing site and the targeted offset defines the error. Figure 5.10 shows this error along with the jettison speed computed by the linearized targeting. The x-axis shows the targeted offset in terms of the separation distance (top), which is essentially the great-circle distance, and the corresponding separation angle (bottom), e.g. separation distance = separation angle × planetary radius. From these results, it is clear that after about 100 km of desired separation distance, the approximation error due to linearization begins to increase rapidly. By about 300 km of desired separation, the targeting error is of similar magnitude to the desired separation and thus, under these assumptions, the linearized targeting method has no utility beyond this point.

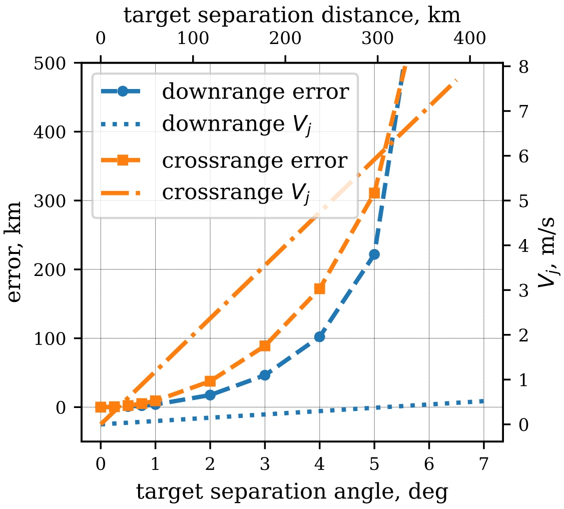


Figure 5.10: Error and required V_j for linearized targeting for varying downrange and crossrange spacing. After the desired change in angle exceeds 6.5°, both cases begin to miss the planet entirely.

5.5 Large-Scale Probe Networks

The linearization method presented in the previous section fails for networks that extend beyond about 100 km from the central point, requiring a different approach. In this section a numerical nonlinear optimization tool is applied to design probe jettisons for large-scale networks of co-delivered probes, and a similar uncertainty quantification analysis is performed.

5.5.1 Nonlinear Optimization Procedure

The quasi-Newton method of Broyden, Fletcher, Goldfarb, and Shanno (BFGS[8]) is used to numerically perform unconstrained optimization of a scalar cost function by iteratively approximating the Hessian matrix [236, 238], where the design variables are the three components of the jettison velocity. Each trajectory is propagated until either reaching the surface or reaching a final time, where the final time is defined to be much later than the nominal time at target. The cost function being minimized is nominally the great-circle distance d along the surface between the

---

[8] scipy.optimize.BFGS

target and achieved landing locations, assuming a spherical planet. In certain cases, this can result in erroneous convergence to a flyby or skip-out trajectory that never reaches the surface, but is at the correct latitude and longitude at the final time. To account for this possibility, the altitude of the spacecraft at the final time is added to the cost function. Equation 5.9 gives the resulting cost function, where (θ_t, ϕ_t) and (θ_f, ϕ_f) are the target and achieved landing coordinates, respectively, and r_f is the radial distance of the spacecraft at the final time. The tolerance is 1 km, meaning that the computed probe jettison velocity delivers the probe to within 1 km, or approximately 0.0169°, of the target landing location.

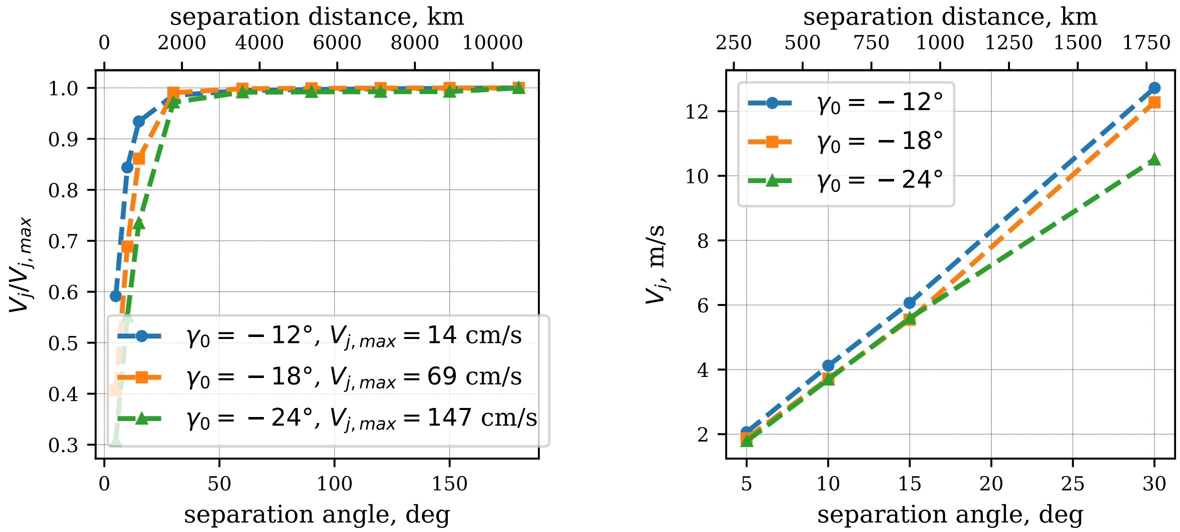
$$J(\boldsymbol{r}_f, \boldsymbol{v}_f) = d + r_f - R \tag{5.9}$$

$$d = R\cos^{-1}\left(\sin\phi_f \sin\phi_t + \cos\phi_f \cos\phi_t \cos(|\theta_t - \theta_f|)\right) \tag{5.10}$$

5.5.2 Targeting Results

For these scenarios a generic entry interface state of 0° longitude, 0° latitude, and 90° (due-east) heading angle is assumed, such that downrange and crossrange are directed east-west and north-south, respectively. downrange and crossrange separations are treated separately in this analysis based on the significant difference in required jettison speed, as shown in Fig. 5.10; this is also intuitive from orbital mechanics, wherein changing the plane of motion in general takes greater effort than changes of similar magnitude within the plane of motion.

Figure 5.11 explores the relationships between desired separation, required jettison speed, and entry flight-path angle for both downrange and crossrange separations, with separation performed three days before entry. As expected, larger separations tend to require larger jettison speeds. This relationship takes a roughly linear form for crossrange separations, as shown in Fig. 5.11b, despite the breakdown of the linearization method based on finite-differencing from small perturbations. Furthermore, γ_0 of the approach trajectory has very little effect on the required jettison speed. In sharp contrast, the jettison speed required for downrange separations asymptotically approaches a fixed value beyond 30° of separation, and is strongly affected by approach trajectory γ_0. Note that the y-axis of Fig. 5.11a is normalized with respect to the required jettison speed of the largest separation, highlighting the similarity in shape between the different γ_0 cases despite their offset values, whereas Fig. 5.11b shows non-normalized speeds.

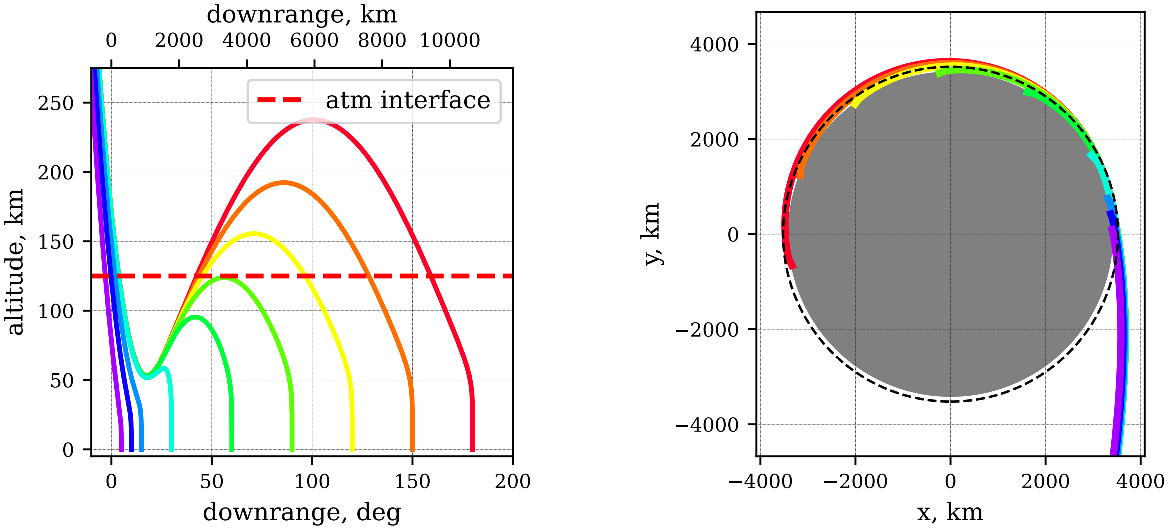


(a) downrange separation

(b) crossrange separation

Figure 5.11: Required jettison speed vs. desired landing separation, for separation 3 days before entry and varying γ_0 values

The reason for the plateau in jettison speed shown in Fig. 5.11a is that downrange separations larger than 15° are achieved via either long coast phases in the atmosphere or skip-out trajectories, in which the vehicle exits the atmosphere on a suborbital arc and then re-enters farther downrange. This can be seen in Fig. 5.12, which shows trajectories targeting downrange separation for the $\gamma_0 = -18°$ case, where Fig. 5.12a plots planet-relative motion in the altitude vs. downrange plane and Fig. 5.12b shows trajectories in the planet-centered inertial frame. The three cases with smallest separations can be seen to follow similarly-shaped trajectories down to the surface, separated due to offsets in their exoatmospheric trajectories and incremental changes in their entry interface states. The rest of the trajectories, however, enter the atmosphere on nearly the same trajectory and then achieve separation during atmospheric flight, with each subsequent trajectory coasting for longer in the atmosphere until eventually skip-out trajectories are realized. Sensitivity of the landing separation with respect to the state at entry interface increases dramatically for these long coast or skip-out trajectories. The use of increasingly long atmospheric flight phases to achieve downrange separation is also the reason that the entry flight-path angle of the approach trajectory significantly impacts the required jettison speed. In contrast, crossrange separation is achieved primarily by modifying the exoatmospheric trajectory, and is thus insensitive to approach trajectory γ_0.

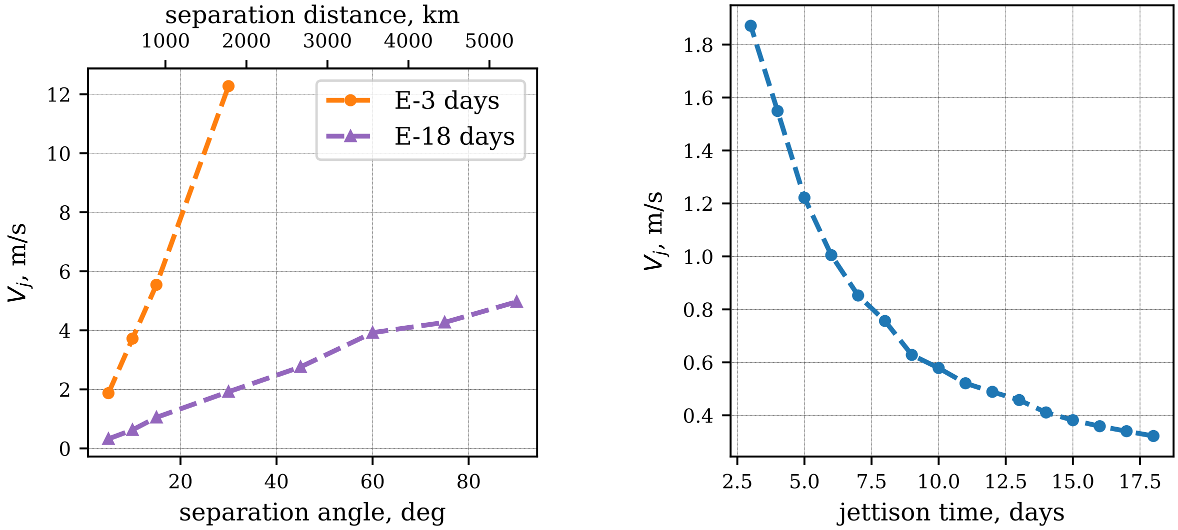


(a) Altitude vs. downrange

(b) Viewed down from North pole, inertial

Figure 5.12: Probe trajectories for downrange separations ranging from 5° (purple) to 180° (red)

All of the large-scale network results thus far assume a separation time of three days before entry. Therefore, it is instructive to consider the relationship between separation time and required jettison speed, particularly for crossrange separation since the required speeds are larger in those cases. To this end, Fig. 5.13a compares the required jettison speed vs. desired separation for separation events 3 and 18 days before entry. The required speeds for 18 days before entry are not only lower, but also increase at a slower rate compared to separation 3 days beforehand. Figure 5.13b shows how required jettison speed changes with varying separation timing for a 5° crossrange separation; one can imagine this as representing the continuum between the leftmost points of the two lines in Fig. 5.13a. The required jettison speed decreases monotonically and nonlinearly as the time between separation and entry increases, as is the case for regional networks as shown in Fig. 5.7.



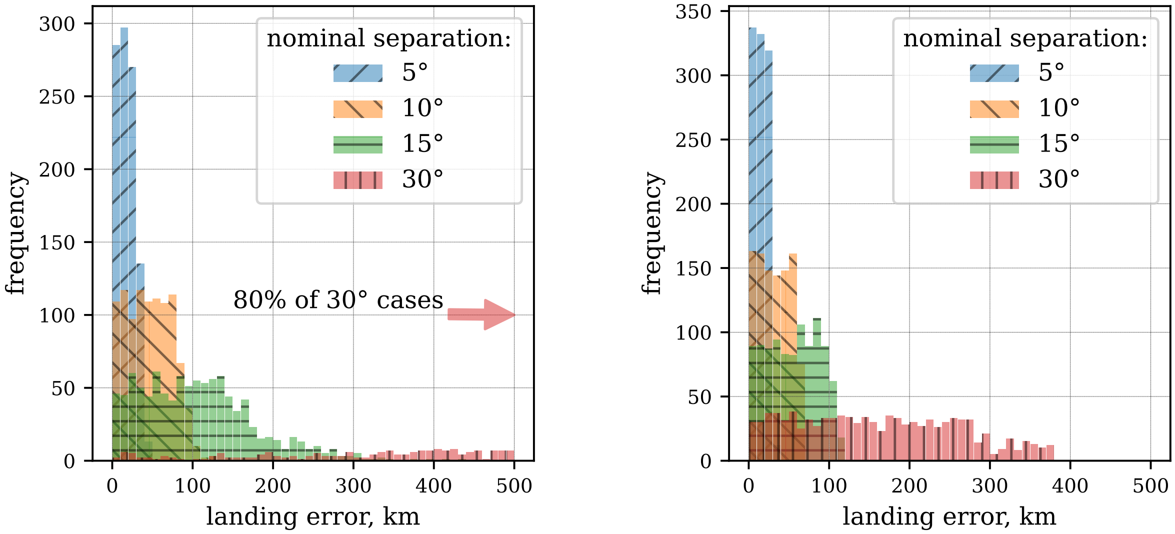
(a) Varying crossrange separation

(b) 5° crossrange separation

Figure 5.13: Comparisons of required jettison speed at varying separation times, for $\gamma_0 = -18°$

5.5.3 Dispersion Analysis

The preceding results demonstrate the ability to use nonlinear numerical optimization to design probe jettisons to co-deliver large-scale probe networks. To understand the practicality of these trajectories, however, the impact of relevant uncertainties must be considered. A 1000-trial Monte Carlo analysis is performed for this purpose, assuming $\gamma_0 = -18°$ and using the same input dispersions as in the previous section. In this case, eight total probes are considered. Four probes target downrange separations of 5°, 10°, 15°, and 30° and separate from the carrier 3 days before entry. The other four probes target crossrange separations of the same magnitudes and separate from the carrier 18 days before entry; the later separation time is selected to reduce the jettison speeds required to reach crossrange separations. The resulting nominal γ_0 values are, in order of increasing separation, $-15.15°$, $-12.79°$, $-11.27°$, and $-9.91°$ for the downrange separations and $-17.82°$, $-17.92°$, $-18.54°$, and $-19.04°$ for the crossrange separations. Figure 5.14 shows the resulting landing error for these 8 probes, and Fig. 5.15 shows the resulting dispersed γ_0 values. It should be reiterated that γ_0 of the central approach trajectory, which is the same for all cases, is distinct from the actual flight-path angle of each probe upon entering the atmosphere.

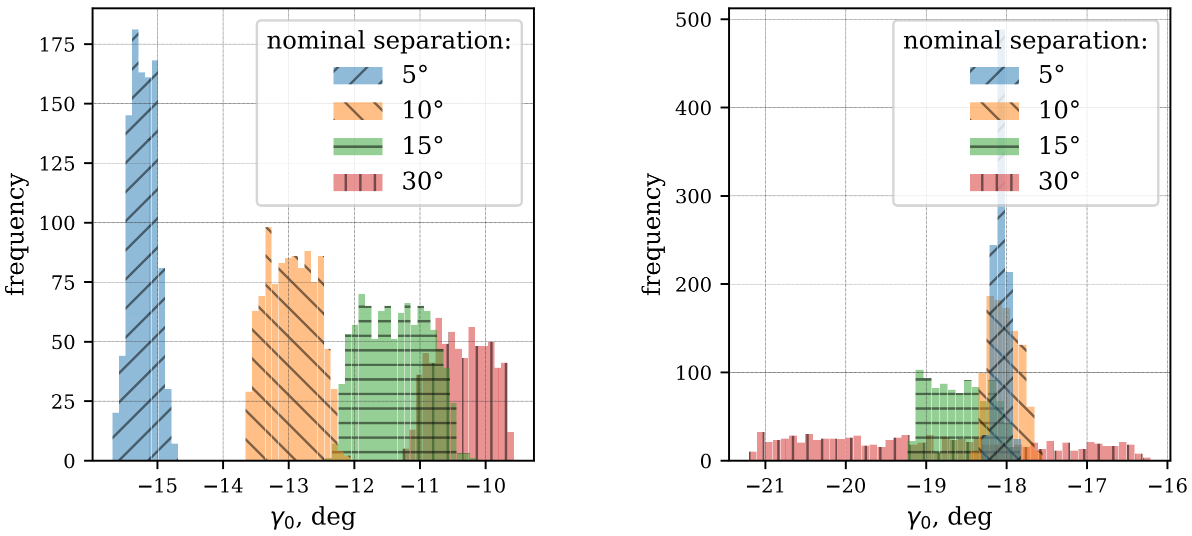


(a) downrange separations

(b) crossrange separations

Figure 5.14: Monte Carlo results for large-scale network

From Fig. 5.14a, it is clear that landing error increases substantially with each increase in nominal downrange separation. The landing dispersions for the 10 and 15 degree cases are large but bounded, such that they would conceivably still suffice if the probes were targeting a broad region on the surface. In contrast, the 30° case has such large landing error that 80% of the trials have greater than 500 km error, and in 31% of trials, the spacecraft skips out of the atmosphere while still on a hyperbolic trajectory. This large jump in error statistics is the result of the plateau in required jettison speed observed in Fig. 5.11a. Because very small changes in jettison speed result in large changes in landing separation, the ±10% jettison speed dispersion is sufficient to radically degrade



(a) downrange separation

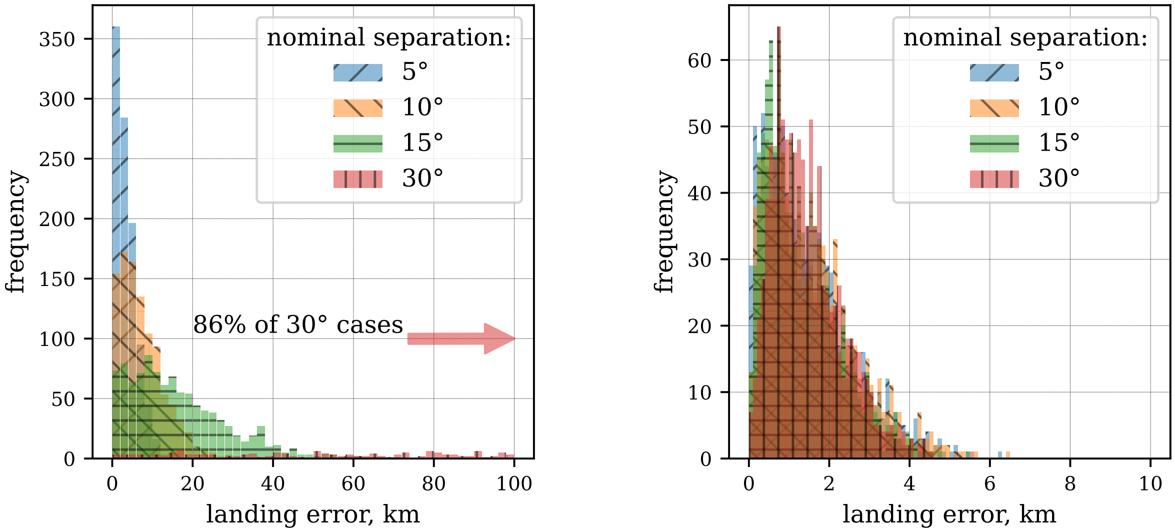
(b) crossrange separation

Figure 5.15: Actual entry flight-path angles for probe trajectories

targeting. Clearly, under the assumed scenario and dispersions, the 30° downrange separation is not a viable trajectory, nor are the trajectories with greater downrange separation.

To get a sense of to what extent the error results from jettison speed dispersions, Fig. 5.16 shows the results of an equivalent Monte Carlo analysis but without jettison speed dispersions. All probe trajectories show marked improvement, including the 30° case, and no trajectories remain hyperbolic. However, the 30° case still has dramatically greater landing error than the other three cases, with the large majority of trials exceeding 100 km of error and 31% of trials exceeding 500 km of error. This is because the extended coast phase, as seen in Fig. 5.12, results in a trajectory that is fundamentally more sensitive to variations. That is, even when perfect probe jettison execution is assumed, small variations have a major impact due to the shallow entry flight-path angle and close proximity to other trajectories in the solution family that extend much further in downrange.

Turning to crossrange separations, the landing error with dispersed jettison speed also increases substantially as desired separation increases, with the 30° case again performing much worse than the other three probes, but this time without any errors exceeding 500 km. However, in con-



(a) downrange separations

(b) crossrange separations

Figure 5.16: Monte Carlo results for large-scale network without jettison speed dispersions

trast to the downrange cases, the landing error results for the Monte Carlo analysis without jettison speed dispersions are relatively small and seemingly insensitive to desired separation. The differing behavior comes down to nominal entry flight-path angle for each probe. As desired downrange separation increases, the nominal γ_0 becomes shallower[9] and, as shown in Fig. 5.15a, the variation in γ_0 increases. In contrast, crossrange trajectories all have a nominal γ_0 near $-18°$ similar to the central approach trajectory, but the dispersions on γ_0 still increase with desired separation, as seen in Fig. 5.15b. As desired separation increases, so does required jettison speed, and thus the jettison speed dispersions have greater effect. Thus, when probe jettison execution error is removed from the assumed dispersions, all crossrange trajectories enter at about $-18°$ and experience landing errors consistent with the single-probe flight mechanics analysis shown in Fig. 5.5b. The larger γ_0 dispersions resulting from probe jettison execution error for the 30° crossrange case are what account for the much greater landing site dispersions seen in Fig. 5.14b; although this case is benign compared to the largest downrange separation case, it still is most likely too much error for

---

[9] Note, though, that this trend plateaus as separations greater than 30° are targeted, because the differences in entry interface states for these trajectories are very small.

practical application.

5.6 Conclusions

Networks of co-delivered probes on the Martian surface would be scientifically valuable at a range of scales, and this study examines some of the relevant flight mechanics and mission design considerations. It is shown that regional networks within about 100 km of the central point can be co-delivered with a small mechanical jettison within five days of atmospheric entry. Larger networks are considered and trajectories are successfully identified using numerical nonlinear optimization. However, the long coast and skip-out trajectories identified in this study are too sensitive to error to be practical beyond a certain separation distance, even under reduced targeting accuracy requirements. For the scenario considered here, this cutoff occurs beyond roughly 15° of desired separation, corresponding to a maximum separation of approximately 890 km along the surface in either direction. Several alternative approaches could potentially enable larger separations. For instance, the strict co-delivery assumption could be relaxed to allow the carrier spacecraft to perform multiple maneuvers during approach, including between separation events. Also, the probe jettisons could be performed much earlier, enabling different targeting geometries with much lower required jettison speeds compared to similar geometries for separation within 20 days of entry. The targeting optimization process itself could be constrained to solutions with a desired entry flight-path angle, potentially resulting in larger probe jettison speeds but avoiding highly sensitive trajectories. An even more constrained approach would be to first define states at atmospheric entry interface that meet requirements and result in the desired landing locations, and then to design maneuvers targeting those entry states, effectively decoupling the exoatmospheric and atmospheric trajectory design processes. This decoupled approach would enable guaranteeing desirable properties of the nominal atmospheric flight trajectories, at the expense of potentially further increasing required jettison speed.

Chapter 6

Relative Motion in the Velocity Frame for Atmospheric Entry Trajectories

6.1 Introduction

In studies of spacecraft formation flying it is common to represent the relevant dynamics using relative motion models centered on one spacecraft [239]. This central spacecraft is labelled as the chief and all other neighboring spacecraft are labelled deputies. These models can take the form of exact or linearized relative equations of motion (EOMs), which may admit analytical solutions, and a wide variety of solutions have been studied [113]. Such relative motion models provide a degree of analytical insight, reduce the computational complexity for simulation, and supply a dynamics representation more amenable to onboard control and estimation methods. Notably, the choice of state representation (Cartesian coordinate frame, relative orbit elements, etc.) has a significant impact on the utility of these models [240].

Existing formation flying literature is primarily concerned with motion about circular or slightly-eccentric elliptical orbits, such as the well-known works by Hill [241], Clohessy and Wiltshire [242], and Tschauner and Hempel [243]. In contrast, relative motion about highly-eccentric elliptical or hyperbolic chief orbits has received little dedicated attention. Carter presents a state transition matrix (STM) applicable for Keplerian orbits with any eccentricity in terms of Cartesian coordinates in a rotating frame with true anomaly as the independent variable [244], and a time-explicit STM is given by Dang [245]. A direct solution of the STM for any non-parabolic Keplerian orbit is given by Reynolds in terms of inertial states [246]. Dang and Zhang present linearized relative equations of motion in terms of orbit element differences that are valid about a hyperbolic orbit

[247]; the work by Willis *et. al* gives a second-order solution in terms of time and true anomaly of the chief [248], and Melton shows that this model holds true for hyperbolic orbits [249]. While the aforementioned approaches provide accurate models of relative motion about a highly-eccentric chief, they do not necessarily present an intuitive representation in the way that the Clohessy-Wiltshire-Hill equations do in a Cartesian rotating frame for motion about a circular chief. This is because all prior work expresses the Cartesian relative motion coordinates in the rotating orbit (or Hill) frame of the chief. This frame is not as convenient for highly-eccentric chief orbits, as even the simplest formation, the lead-follower formation in which there is only a difference in true anomaly, results in a two-dimensional trajectory in the orbit frame. The relative motion in a lead-follower formation is primarily in the velocity direction, which is not along an orbit frame unit vector for non-circular orbits. This is illustrated in Fig. 6.1, which shows relative motion for a lead-follower formation about a hyperbolic chief in both the Hill and velocity frames, where the y-axis of the latter is defined as the velocity direction of the chief; see the following section for detailed frame definitions. This paper explores relative motion expressed in the rotating chief velocity frame rather than the orbit frame. Although the velocity frame is commonly used in astrodynamics, a direct formulation of relative motion in this frame is a novel contribution.

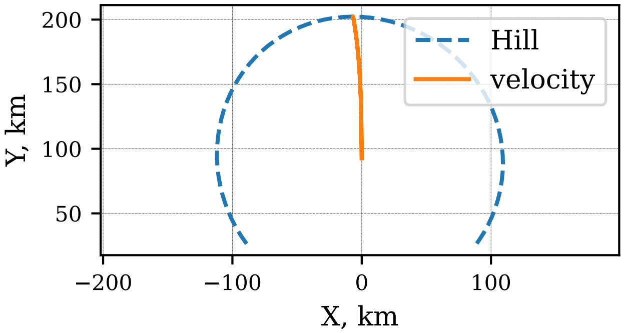


Figure 6.1: Relative motion about hyperbolic chief shown in Hill and velocity frame components

Velocity frame relative motion is of particular interest in scenarios where force due to atmospheric drag is significant, because drag is purely along the anti-velocity direction. Examples range from the small perturbing acceleration due to drag in low-Earth orbit, to aerobraking in which the spacecraft repeatedly passes through the upper atmosphere to reduce orbital energy, to aerocapture and direct-entry scenarios in which the vehicle flies deep into the atmosphere and drag becomes the dominant force. An intuitive and accurate model of relative motion is relevant to all of these scenarios. For example, propulsionless satellites can achieve orbit phasing and adjustment by changing attitude to modulate the amount of drag on each spacecraft, as has been demonstrated on orbit [250]. A similar method of control can be used during aerobraking [251]. In a mission deploying two or more satellites into orbit via aerobraking, understanding the relative motion as affected by atmospheric drag would be critical. Relative motion is also relevant to various aerocapture and entry scenarios. The Galileo and Cassini-Huygens missions each delivered a probe from an orbiter [99, 100], and the Pioneer Venus mission deployed multiple probes from a single bus on an entry trajectory [109]. Future missions could involve delivering an entry probe from a mothercraft that then enters the atmosphere to perform aerocapture [225], simultaneous aerobraking of a satellite constellation [252], or deployment of multiple probes to a regional surface network from a single entry vehicle [231]. Intuitive relative motion models could aid in early reference trajectory design for such missions, and could also enhance onboard control and state estimation between the multiple spacecraft. Moreover, aerobraking, aerocapture, and entry trajectories are typically highly-elliptical or hyperbolic, further motivating representation in the velocity frame. Thus, atmospheric entry trajectories, including aerobraking, are considered as a set of motivating examples in this study.

The contributions of this work are an exploration of relative motion models in the velocity frame and the application of these models to Keplerian formation flying, differential drag during aerobraking, and atmospheric entry trajectories. The relative equations of motion in the velocity frame are presented, the linearized approximation is developed, and the non-dimensional form is also provided. In addition, descriptions of velocity frame relative motion in terms of orbit element differences are derived for both elliptical and hyperbolic chief orbits. Expressions for including differential drag as a perturbing acceleration are developed. Relative motion during atmospheric flight is also considered by linearizing the Allen-Eggers solution for ballistic entry[253]. The Keplerian and atmospheric flight models are then combined to create a model of relative motion about an entry vehicle, and the approximate models are validated against numerical propagation of the full dynamics for several representative example scenarios. Relative motion models as described above would benefit an analyst designing trajectories for multiple co-delivered entry vehicles for applications such as a planetary probe network[231] or probe delivery by a carrier spacecraft on an entry trajectory[225], as well as for multiple independently-targetable reentry vehicles[224]. These models also could be incorporated into onboard guidance, navigation, and control algorithms.

6.2 Keplerian Motion in the Velocity Frame

6.2.1 Reference Frame Definitions

Let $\mathcal{N} : \{\hat{\boldsymbol{n}}_1, \hat{\boldsymbol{n}}_2, \hat{\boldsymbol{n}}_3\}$ be a generic inertial frame. The orbit frame, also known as the Hill or LVLH frame, is defined through the base vectors $\mathcal{O} : \{\hat{\boldsymbol{o}}_r, \hat{\boldsymbol{o}}_\theta, \hat{\boldsymbol{o}}_h\}$. Here $\hat{\boldsymbol{o}}_r$ is along the orbit radial direction and $\hat{\boldsymbol{o}}_h$ is along the angular momentum vector of the spacecraft $\boldsymbol{h} = \boldsymbol{r} \times \boldsymbol{v}$, where $\boldsymbol{r}$ and $\boldsymbol{v}$ are the position and inertial velocity vectors for the spacecraft, respectively. Lastly, $\hat{\boldsymbol{o}}_\theta$ completes the right-handed set and is referred to as the along-track direction. The velocity frame is defined through the base vectors $\mathcal{V} : \{\hat{\boldsymbol{v}}_n, \hat{\boldsymbol{v}}_v, \hat{\boldsymbol{v}}_h\}$ where $\hat{\boldsymbol{v}}_v$ is directed along the inertial velocity, $\hat{\boldsymbol{v}}_h = \hat{\boldsymbol{o}}_h$, and $\hat{\boldsymbol{v}}_n$ completes the right-handed set. Flight-path angle γ is defined as the angle from the along-track direction $\hat{\boldsymbol{o}}_\theta$ to the velocity direction $\hat{\boldsymbol{v}}_v$. Finally, true anomaly f is the angle between the position vector and the eccentricity vector, the latter of which is inertially fixed for Keplerian motion, such that the angular velocity between the Hill and inertial frames is $\boldsymbol{\omega}_{\mathcal{O}/\mathcal{N}} = \dot{f}\hat{\boldsymbol{o}}_h$. Figure 6.2 summarizes these frame definitions, where $\hat{\boldsymbol{o}}_h$ is directed out of the page.

6.2.2 Exact Relative Equations of Motion

The chief spacecraft position vector is defined as

$$\boldsymbol{r}_c = r_c\hat{\boldsymbol{o}}_r = x_c\hat{\boldsymbol{v}}_n + y_c\hat{\boldsymbol{v}}_v, \tag{6.1}$$

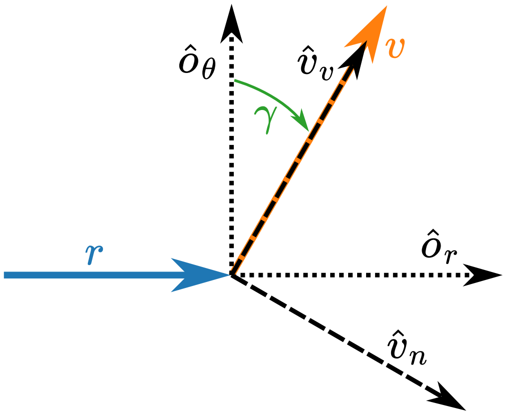


Figure 6.2: Hill and velocity frames

where r_c is the current orbit radius of the chief spacecraft, and noting that, under the assumption of Keplerian motion, the chief has no position component in the orbit-normal direction. As the orbit and velocity frame only differ by a rotation about $\hat{\boldsymbol{o}}_h$, the out-of-plane motion description is identical in both the orbit and velocity frame. This allows the following development to focus on the in-plane relative motion.

The deputy spacecraft position vector is then written in terms of the relative orbit position vector $\boldsymbol{\rho}$ as

$$\boldsymbol{r}_d = \boldsymbol{r}_c + \boldsymbol{\rho} = (x + x_c)\hat{\boldsymbol{v}}_n + (y + y_c)\hat{\boldsymbol{v}}_v + z\hat{\boldsymbol{v}}_h, \tag{6.2}$$

noting that here x, y, and z are defined as velocity frame components, a break from the common use of these variables as Hill frame components.

The velocity frame rotates with respect to the inertial frame with an angular velocity of $\boldsymbol{\omega}_{\mathcal{V}/\mathcal{N}}$, which expands as

$$\boldsymbol{\omega}_{\mathcal{V}/\mathcal{N}} = \boldsymbol{\omega}_{\mathcal{V}/\mathcal{O}} + \boldsymbol{\omega}_{\mathcal{O}/\mathcal{N}} = (\dot{f} - \dot{\gamma})\hat{\boldsymbol{v}}_h. \tag{6.3}$$

The time derivative of this vector with respect to the inertial frame, $\dot{\boldsymbol{\omega}}_{\mathcal{V}/\mathcal{N}}$, is similarly written as

$$\dot{\boldsymbol{\omega}}_{\mathcal{V}/\mathcal{N}} = \dot{\boldsymbol{\omega}}_{\mathcal{V}/\mathcal{O}} + \dot{\boldsymbol{\omega}}_{\mathcal{O}/\mathcal{N}} = (\ddot{f} - \ddot{\gamma})\hat{\boldsymbol{v}}_h. \tag{6.4}$$

Applying transport theorem[239] twice to Eq. (6.2) to find the second time derivative with respect to the inertial frame yields the following kinematic expression for the deputy spacecraft acceleration vector:

$$\begin{aligned}\ddot{\boldsymbol{r}}_d = &\Big(\ddot{x} + \ddot{x}_c - 2(\dot{f} - \dot{\gamma})(\dot{y} + \dot{y}_c) - (\ddot{f} - \ddot{\gamma})(y + y_c) - (\dot{f} - \dot{\gamma})^2(x + x_c)\Big)\,\hat{\boldsymbol{v}}_n \\ &+ \Big(\ddot{y} + \ddot{y}_c + 2(\dot{f} - \dot{\gamma})(\dot{x} + \dot{x}_c) + (\ddot{f} - \ddot{\gamma})(x + x_c) - (\dot{f} - \dot{\gamma})^2(y + y_c)\Big)\,\hat{\boldsymbol{v}}_v + \ddot{z}\hat{\boldsymbol{v}}_h.\end{aligned} \tag{6.5}$$

An expression for the chief spacecraft acceleration vector $\ddot{\boldsymbol{r}}_c$ is similarly derived, and in this case is equal to the Keplerian acceleration vector $-(\mu/r_c^3)\boldsymbol{r}_c$, where μ is the gravitational parameter of the central body. Equating the vector components in the resulting expression for $\ddot{\boldsymbol{r}}_c = -(\mu/r_c^3)\boldsymbol{r}_c$ yields the following equations:

$$\ddot{x}_c - 2(\dot{f} - \dot{\gamma})\dot{y}_c - (\ddot{f} - \ddot{\gamma})y_c - (\dot{f} - \dot{\gamma})^2 x_c = -\frac{\mu}{r_c^3}x_c \tag{6.6a}$$

$$\ddot{y}_c + 2(\dot{f} - \dot{\gamma})\dot{x}_c + (\ddot{f} - \ddot{\gamma})x_c - (\dot{f} - \dot{\gamma})^2 y_c = -\frac{\mu}{r_c^3}y_c \tag{6.6b}$$

The vectors $\boldsymbol{\omega}_{\mathcal{V}/\mathcal{N}}$ and $\dot{\boldsymbol{\omega}}_{\mathcal{V}/\mathcal{N}}$ are conveniently expressed as [239]:

$$\boldsymbol{\omega}_{\mathcal{V}/\mathcal{N}} = (\dot{f} - \dot{\gamma})\hat{\boldsymbol{v}}_h = \frac{\alpha}{\zeta}\dot{f}\hat{\boldsymbol{v}}_h, \tag{6.7}$$

$$\dot{\boldsymbol{\omega}}_{\mathcal{V}/\mathcal{N}} = (\ddot{f} - \ddot{\gamma})\hat{\boldsymbol{v}}_h = \left(\frac{\alpha}{\zeta}\ddot{f} - \frac{e\left(e^2 - 1\right)\sin f}{\zeta^2}\dot{f}^2\right)\hat{\boldsymbol{v}}_h, \tag{6.8}$$

where the dimensionless quantities α and ζ are defined for ease of notation:

$$\alpha = (e\cos f + 1), \tag{6.9}$$

$$\zeta = (e^2 + 2e\cos f + 1). \tag{6.10}$$

Auxiliary variables defined for concise notation are collected in Appendix 6.7 for easy reference. The chief orbit angular momentum magnitude h is constant for Keplerian motion, and setting its time derivative equal to zero yields an expression for true anomaly acceleration [239]:

$$h = r_c^2\dot{f} \tag{6.11}$$

$$\dot{h} = 0 = 2r_c\dot{r}_c\dot{f} + r_c^2\ddot{f} \tag{6.12}$$

Finally, the acceleration acting on the deputy spacecraft is written as the sum of Keplerian acceleration plus an arbitrary perturbing acceleration vector $\boldsymbol{u} = u_x\hat{\boldsymbol{v}}_n + u_y\hat{\boldsymbol{v}}_v + u_z\hat{\boldsymbol{v}}_h$,

$$\ddot{\boldsymbol{r}}_d = -\frac{\mu}{r_d^3}\boldsymbol{r}_d + \boldsymbol{u}, \tag{6.13}$$

where $r_d = \sqrt{(x+x_c)^2 + (y+y_c)^2 + z_c^2}$ is the orbit radius of the deputy spacecraft.

Substituting Eq. (6.6)–(6.8) and (6.12) into Eq. (6.5) gives a kinematic expression for acceleration of the deputy spacecraft; equating this with the kinetic acceleration defined in Eq. (6.13) and simplifying yields the exact nonlinear relative equations of motion in terms of velocity frame components:

$$\ddot{x} + \dot{f}\frac{\alpha}{\zeta}\left[y\left(2\frac{\dot{r}_c}{r_c} - \frac{\dot{f}e(1-e^2)\sin f}{\alpha\zeta}\right) - 2\dot{y} - x\frac{\dot{f}\alpha}{\zeta}\right] - \frac{\mu x_c}{r_c^3} = -\frac{\mu}{r_d^3}(x_c + x) + u_x \tag{6.14a}$$

$$\ddot{y} - \dot{f}\frac{\alpha}{\zeta}\left[x\left(2\frac{\dot{r}_c}{r_c} - \frac{\dot{f}e(1-e^2)\sin f}{\alpha\zeta}\right) - 2\dot{x} + y\frac{\dot{f}\alpha}{\zeta}\right] - \frac{\mu y_c}{r_c^3} = -\frac{\mu}{r_d^3}(y_c + y) + u_y \tag{6.14b}$$

$$\ddot{z} = -\frac{\mu}{r_d^3} z + u_z \tag{6.14c}$$

Note that the flight-path angle of the chief spacecraft is written as [239]

$$\tan\gamma = \frac{e\sin f}{1 + e\cos f} = \frac{e\sin f}{\alpha}. \tag{6.15}$$

Therefore, Eqs. (6.14a) and (6.14b) can also be written as:

$$\ddot{x} + \dot{f}\frac{\alpha}{\zeta}\left[y\left(2\frac{\dot{r}_c}{r_c} - \frac{\dot{f}(1-e^2)}{\zeta}\tan\gamma\right) - 2\dot{y} - x\frac{\dot{f}\alpha}{\zeta}\right] - \frac{\mu x_c}{r_c^3} = -\frac{\mu}{r_d^3}(x_c + x) + u_x \tag{6.16a}$$

$$\ddot{y} - \dot{f}\frac{\alpha}{\zeta}\left[x\left(2\frac{\dot{r}_c}{r_c} - \frac{\dot{f}(1-e^2)}{\zeta}\tan\gamma\right) - 2\dot{x} + y\frac{\dot{f}\alpha}{\zeta}\right] - \frac{\mu y_c}{r_c^3} = -\frac{\mu}{r_d^3}(y_c + y) + u_y \tag{6.16b}$$

It is worth briefly noting how the relative equations of motion are correctly initialized for propagation. A typical scenario is that the position and inertial velocity vectors of the chief and deputy spacecraft are known at the initial time, and the relative state must be computed. The relative position vector, $\boldsymbol{\rho}$, is computed according to Eq. (6.2) and rotated into the velocity frame, providing initial values for x, y, and z. To complete the full state, relative velocity components $\dot{x}$, $\dot{y}$, and $\dot{z}$ are also required, but these comprise a vector defined as the time derivative of the relative position as seen by the velocity frame. Using the transport theorem yields

$$^{\mathcal{V}}\begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{z}\end{bmatrix} \equiv \frac{^{\mathcal{V}}\mathrm{d}}{\mathrm{d}t}(\boldsymbol{\rho}) = \dot{\boldsymbol{\rho}} - \boldsymbol{\omega}_{\mathcal{V}/\mathcal{N}} \times \boldsymbol{\rho}, \tag{6.17}$$

where $\dot{\boldsymbol{\rho}} = \dot{\boldsymbol{r}}_d - \dot{\boldsymbol{r}}_c$ is the difference between the inertial velocities of the deputy and chief spacecraft.

6.2.3 Linearized Relative Equations of Motion

In order to linearize Eqs. (6.14a)–(6.14c), assume that the distance between the chief and deputy spacecraft is small compared to the chief orbit radius, $(x, y, z) \ll r_c$. By taking a first-order Taylor series expansion about $x = y = z = 0$, r_d is approximated as

$$\frac{\mu}{r_d^3} \approx \frac{\mu}{r_c^3}(1 - 3\kappa) \tag{6.18}$$

where

$$\kappa = \frac{x_c x + y_c y}{r_c^2}. \tag{6.19}$$

Substituting Eq. (6.18) into the vector expression on the right-hand side of Eq. (6.13) and neglecting terms that are quadratic in terms of x, y, or z results in a further simplification:

$$-\frac{\mu}{r_d^3}\,{}^{\mathcal{V}}\begin{bmatrix}x + x_c\\ y + y_c\\ z\end{bmatrix} \approx -\frac{\mu}{r_c^3}\,{}^{\mathcal{V}}\begin{bmatrix}x + x_c - 3\kappa x_c\\ y + y_c - 3\kappa y_c\\ z\end{bmatrix} \tag{6.20}$$

Additionally, note that μ/r_c^3 can be expressed as the following identities [239]:

$$\frac{\mu}{r_c^3} = \frac{r_c}{p}\dot{f}^2 = \frac{\dot{f}^2}{\alpha}. \tag{6.21}$$

Substituting Eqs. (6.20) and (6.21) into Eqs. (6.14a)–(6.14c) gives the linearized relative equations of motion in terms of velocity frame components:

$$\ddot{x} + \dot{f}\frac{\alpha}{\zeta}\left[y\left(2\frac{\dot{r}_c}{r_c} - \frac{\dot{f}e(1-e^2)\sin f}{\alpha\zeta}\right) - 2\dot{y} - x\frac{\dot{f}\alpha}{\zeta}\right] + \frac{\dot{f}^2}{\alpha}(x - 3\kappa x_c) = u_x \tag{6.22a}$$

$$\ddot{y} - \dot{f}\frac{\alpha}{\zeta}\left[x\left(2\frac{\dot{r}_c}{r_c} - \frac{\dot{f}e(1-e^2)\sin f}{\alpha\zeta}\right) - 2\dot{x} + y\frac{\dot{f}\alpha}{\zeta}\right] + \frac{\dot{f}^2}{\alpha}(y - 3\kappa y_c) = u_y \tag{6.22b}$$

$$\ddot{z} + \frac{\dot{f}^2}{\alpha}z = u_z \tag{6.22c}$$

6.2.4 Non-Dimensional Relative Equations of Motion

In the case of relative motion in the Hill frame, the linearized equations of motion take on an elegant form when non-dimensionalized by the chief orbit radius r_c and differentiated with respect to the chief orbit true anomaly f instead of time [239]. These are known as the Tschauner-Hempel equations [243], and a variety of solution approaches exist in the literature [244]. For completeness, the equivalent non-dimensional forms of the linearized relative EOMs in terms of velocity frame components are presented here.

Define the non-dimensional relative orbit coordinates (u, v, w) as

$$u = \frac{x}{r_c} \qquad v = \frac{y}{r_c} \qquad w = \frac{z}{r_c} \tag{6.23}$$

Unlike the Tschauner-Hempel equations in the Hill frame, the velocity frame equations require similarly defining non-dimensional coordinates for the chief spacecraft:

$$u_c = \frac{x_c}{r_c} \qquad v_c = \frac{y_c}{r_c} \qquad w_c = \frac{z_c}{r_c} \tag{6.24}$$

Denote the derivative with respect to chief orbit true anomaly as

$$()' \equiv \frac{\mathrm{d}()}{\mathrm{d}f} \tag{6.25}$$

The following identities relate time derivatives of (x, y, z) to derivatives of (u, v, z) with respect to true anomaly [239]:

$$\frac{\dot{x}}{r_c} = u'\dot{f} + u\frac{\dot{r}_c}{r_c} \qquad \frac{\ddot{x}}{r_c} = u''\dot{f}^2 + u\dot{f}^2\left(1 - \frac{r_c}{p}\right) \tag{6.26a}$$

$$\frac{\dot{y}}{r_c} = v'\dot{f} + v\frac{\dot{r}_c}{r_c} \qquad \frac{\ddot{y}}{r_c} = v''\dot{f}^2 + v\dot{f}^2\left(1 - \frac{r_c}{p}\right) \tag{6.26b}$$

$$\frac{\dot{z}}{r_c} = w'\dot{f} + w\frac{\dot{r}_c}{r_c} \qquad \frac{\ddot{z}}{r_c} = w''\dot{f}^2 + w\dot{f}^2\left(1 - \frac{r_c}{p}\right) \tag{6.26c}$$

Dividing Eq. (6.22) by r_c and substituting Eq. (6.26) gives the following non-dimensional linearized relative equations of motion in terms of velocity frame components:

$$u'' + \left[1 - \frac{\alpha^2}{\zeta^2} - 3\frac{r_c}{p}u_c^2\right]u - 2\frac{\alpha}{\zeta}v' - \left[\frac{e(1-e^2)\sin f}{\zeta^2} + 3\frac{r_c}{p}u_c v_c\right]v = \frac{u_x}{r_c\dot{f}^2} \tag{6.27a}$$

$$v'' + \left[1 - \frac{\alpha^2}{\zeta^2} - 3\frac{r_c}{p}v_c^2\right]v + 2\frac{\alpha}{\zeta}u' + \left[\frac{e(1-e^2)\sin f}{\zeta^2} - 3\frac{r_c}{p}u_c v_c\right]u = \frac{u_y}{r_c\dot{f}^2} \tag{6.27b}$$

$$w'' + w = \frac{u_z}{r_c\dot{f}^2} \tag{6.27c}$$

6.2.5 Relative Orbit Element Description

A disadvantage of the relative equations of motion discussed thus far is that, for a general orbit, describing the relative motion requires solving the differential equations. As an alternative approach, a direct mapping between orbit element differences and the Cartesian relative position vector $\boldsymbol{\rho}$ would provide analytical insight into the relative orbit geometry. This is provided in Ref. [254] in terms of Hill frame components. An equivalent mapping between orbit element differences and velocity frame components can be found by pre-multiplying that result by the direction cosine matrix (DCM) relating the two frames, ${}^{\mathcal{V}}\boldsymbol{\rho} = [VO]{}^{\mathcal{O}}\boldsymbol{\rho}$, where [VO] is [239]:

$$[VO] = \begin{bmatrix} \frac{\alpha}{\sqrt{\zeta}} & -\frac{e\sin f}{\sqrt{\zeta}} & 0 \\ \frac{e\sin f}{\sqrt{\zeta}} & \frac{\alpha}{\sqrt{\zeta}} & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{6.28}$$

A brief derivation of the equations relating orbit element differences and Cartesian velocity frame position components is given here, with additional detail provided in Appendix C closely following sections 14.4.1 and 14.6.1 of Ref. [239].

Define the orbit element vector as $\boldsymbol{\oe} = (a, e, i, \Omega, \omega, M)^T$, consisting of semi-major axis, eccentricity, inclination, right ascension of the ascending node, argument of periapsis, and mean anomaly, respectively. The orbit element difference vector is then defined as deputy orbit element vector minus the chief orbit element vector,

$$\delta\boldsymbol{\oe} = \boldsymbol{\oe}_d - \boldsymbol{\oe}_c = (\delta a, \delta e, \delta i, \delta\Omega, \delta\omega, \delta M)^T \tag{6.29}$$

A linearized mapping between orbit element differences and Cartesian relative position is provided by Eq. (6.30):

$$^{\mathcal{O}}\boldsymbol{\rho} = \begin{bmatrix} \delta r \\ r(\delta\theta + \cos i \delta\Omega) \\ r(\sin\theta\delta i - \cos\theta \sin i \delta\Omega \end{bmatrix} \tag{6.30}$$

To obtain a more intuitive description that does not rely on $\delta\theta$, Eq. (6.30) is reformulated to instead rely on differences in mean anomaly M (or, in the case of a hyperbolic chief, mean hyperbolic anomaly N), which will remain constant if $\delta a = 0$ for Keplerian motion [239]. The derivations of each first-order variation are provided in Ref. [254] and included in Appendix C with the exception of difference in mean hyperbolic anomaly δN, which is added here to include expressions that are valid for a hyperbolic chief. The derivations of δM and δN only differ slightly and are therefore shown here in parallel.

Take the definitions of mean anomalies M and N in terms of eccentric anomaly E and hyperbolic anomaly H:

$$M = E - e \sin E \tag{6.31a}$$

$$N = e \sinh H - H \tag{6.31b}$$

and take the first variations of these expressions:

$$\delta M = (1 - e \cos E)\delta E - \sin E \delta e \tag{6.32a}$$

$$\delta N = (e \cosh H - 1)\delta H + \delta e \sinh H \tag{6.32b}$$

Note the orbit identities relating E and H with f [239],

$$\tan\frac{f}{2} = \sqrt{\frac{1+e}{1-e}}\tan\frac{E}{2} \tag{6.33a}$$

$$\tan\frac{f}{2} = \sqrt{\frac{e+1}{e-1}}\tanh\frac{H}{2}, \tag{6.33b}$$

and take the first variations:

$$\delta E = \frac{\eta}{\alpha}\delta f - \frac{\sin f}{\alpha\eta}\delta e \tag{6.34a}$$

$$\delta H = \frac{\eta_h}{\alpha}\delta f + \frac{\sin f}{\alpha\eta_h}\delta e \tag{6.34b}$$

where $\eta = \sqrt{1-e^2}$ and $\eta_h = \sqrt{e^2-1}$.

Additionally, note the following orbit identities [239]:

$$\sin E = \frac{\eta\sin f}{\alpha} \qquad \cos E = \frac{e+\cos f}{\alpha} \tag{6.35a}$$

$$\sinh H = \frac{\eta_h\sin f}{\alpha} \qquad \cosh H = \frac{e+\cos f}{\alpha} \tag{6.35b}$$

Substituting Eqs. (6.34) and (6.35) into Eq. (6.32), simplifying, and rearranging provides expressions for δf in terms of δM and δN:

$$\delta f = \frac{\alpha^2}{\eta^3}\delta M + \frac{\sin f(2+e\cos f)}{1-e^2}\delta e \tag{6.36a}$$

$$\delta f = \frac{\alpha^2}{\eta_h^3}\delta N - \frac{\sin f(2+e\cos f)}{e^2-1}\delta e \tag{6.36b}$$

Substituting the orbit identities given in Eqs. (C.7) – (C.15) and (6.36) into Eqs. (6.30), premultipliying by $[VO]$, and simplifying yields the desired mapping, where Eq. (6.37) and Eq. (6.38) correspond to elliptical and hyperbolic chief orbits, respectively.

$$x = \frac{(1-e^2)}{\sqrt{\zeta}}\delta a - \frac{a\left(\left(e^2+1\right)\cos f + 2e\right)}{\alpha\sqrt{\zeta}}\delta e - \frac{re\sin f}{\sqrt{\zeta}}(\delta\omega + \delta\Omega\cos i) \tag{6.37a}$$

$$y = \frac{re\sin f}{a\sqrt{\zeta}}\delta a + \frac{2a\sin f}{\sqrt{\zeta}}\delta e + \frac{p}{\sqrt{\zeta}}(\delta\omega + \delta\Omega\cos i) + \frac{a\sqrt{\zeta}}{\eta}\delta M \tag{6.37b}$$

$$z = r\left(\sin\theta\delta i - \sin i\cos\theta\delta\Omega\right) \tag{6.37c}$$

$$x = \frac{(1-e^2)}{\sqrt{\zeta}}\delta a - \frac{a\left(\left(e^2+1\right)\cos f + 2e\right)}{\alpha\sqrt{\zeta}}\delta e - \frac{re\sin f}{\sqrt{\zeta}}(\delta\omega + \delta\Omega\cos i) \tag{6.38a}$$

$$y = \frac{re\sin f}{a\sqrt{\zeta}}\delta a + \frac{2a\sin f}{\sqrt{\zeta}}\delta e + \frac{a(1-e^2)}{\sqrt{\zeta}}(\delta\omega + \delta\Omega\cos i) - \frac{a\sqrt{\zeta}}{\eta_h}\delta N \tag{6.38b}$$

$$z = r\left(\sin\theta\delta i - \sin i\cos\theta\delta\Omega\right) \tag{6.38c}$$

By sweeping chief true anomaly values through a single full revolution, the corresponding relative orbit geometry can be analytically computed according to Eq. (6.37). In the case of an invariant orbit, where $\delta a = 0$ and the dynamics are fully Keplerian, this describes the complete relative orbit geometry. It is important to note that in the case of a hyperbolic chief, the equations should only be evaluated for physically-reachable values of true anomaly as defined by the range $(-f_\infty, f_\infty)$ where f_∞ is the asymptotic true anomaly:

$$\cos f_\infty = -\frac{1}{e} \tag{6.39}$$

Note that it is possible for α to equal 0 and cause a singularity in Eq. (6.38) for hyperbolic orbits if this constraint is ignored, but that this singularity is never encountered as long as the physical limitation of $-f_\infty < f < f_\infty$ is respected.

In order to fully describe the spacecraft state, velocity is also necessary, and this can be obtained by differentiating Eqs. (6.37) and (6.38) with respect to time. The only time-varying quantities appearing in these equations are true anomaly of the chief f and either difference in mean anomaly δM or difference in mean hyperbolic anomaly δN, respectively. The conservation of angular momentum of the chief spacecraft, given by Eq. (6.11), can be rearranged to give an expression for $\dot{f}$,

$$\dot{f} = \frac{h}{r_c^2}. \tag{6.40}$$

To find expressions for $\delta\dot{M}$ and $\delta\dot{N}$, take the first variation of Kepler's equation,

$$\delta M = \delta M_0 - \frac{3}{2}\frac{\delta a}{a}(M - M_0) \tag{6.41a}$$

$$\delta N = \delta N_0 - \frac{3}{2}\frac{\delta a}{a}(N - N_0). \tag{6.41b}$$

Taking the time derivative of Eq. (6.41) gives the desired expressions,

$$\delta\dot{M} = \delta\dot{N} = -\frac{3}{2}\frac{\delta a}{a}n, \tag{6.42}$$

where $n = \sqrt{\mu/a^3}$ is the mean motion.

Eqs. (6.43) and (6.44) give the resulting mappings, for elliptical and hyperbolic chief orbits respectively, from orbit element differences to components of the time derivative of the relative position vector with respect to the velocity frame. Note that the inertial velocity of the deputy can be recovered from the relative velocity components by rearranging Eq. (6.17).

$$\begin{aligned}\dot{x} = {} & \frac{eh\alpha\sin f}{ar\zeta^{3/2}}\delta a + \frac{h\sin f\left[\left(e^2+1\right)\left(\zeta\alpha - e\cos f(\alpha+\zeta)\right) - 2e^2(\alpha+\zeta)\right]}{p\left(1-e^2\right)\zeta^{3/2}}\delta e \\ & - \frac{eh\left(\zeta\alpha\cos f + e\sin^2 f(\alpha+\zeta)\right)}{p\zeta^{3/2}}(\delta\omega + \delta\Omega\cos i)\end{aligned} \tag{6.43a}$$

$$\begin{aligned}\dot{y} = {} & \left(\frac{eh\left(\zeta\alpha\cos f + e\sin^2 f(\alpha+\zeta)\right)}{ap\zeta^{3/2}} - \frac{3n\sqrt{\zeta}}{2\eta}\right)\delta a + \frac{2h\alpha\left(\zeta\cos f + e\sin^2 f\right)}{r\left(1-e^2\right)\zeta^{3/2}}\delta e \\ & + \frac{eh\alpha\sin f}{r\zeta^{3/2}}(\delta\omega + \delta\Omega\cos i) - \frac{eh\alpha\sin f}{r\eta^3\sqrt{\zeta}}\delta M\end{aligned} \tag{6.43b}$$

$$\dot{z} = \frac{h}{p}(e\cos\omega + \cos\theta)\delta i + \frac{h}{p}\sin i(e\sin\omega + \sin\theta)\delta\Omega \tag{6.43c}$$

$$\begin{aligned}\dot{x} = {} & \frac{eh\alpha\sin f}{ar\zeta^{3/2}}\delta a + \frac{h\sin f\left[\left(e^2+1\right)\left(\zeta\alpha - e\cos f(\alpha+\zeta)\right) - 2e^2(\alpha+\zeta)\right]}{p\left(1-e^2\right)\zeta^{3/2}}\delta e \\ & - \frac{eh\left(\zeta\alpha\cos f + e\sin^2 f(\alpha+\zeta)\right)}{p\zeta^{3/2}}(\delta\omega + \delta\Omega\cos i)\end{aligned} \tag{6.44a}$$

$$\begin{aligned}\dot{y} = {} & \left(\frac{eh\left(\zeta\alpha\cos f + e\sin^2 f(\alpha+\zeta)\right)}{ap\zeta^{3/2}} + \frac{3n\sqrt{\zeta}}{2\eta_h}\right)\delta a + \frac{2h\alpha\left(\zeta\cos f + e\sin^2 f\right)}{r\left(1-e^2\right)\zeta^{3/2}}\delta e \\ & + \frac{eh\alpha\sin f}{r\zeta^{3/2}}(\delta\omega + \delta\Omega\cos i) - \frac{eh\alpha\sin f}{r\eta_h^3\sqrt{\zeta}}\delta N\end{aligned} \tag{6.44b}$$

$$\dot{z} = \frac{h}{p}(e\cos\omega + \cos\theta)\delta i + \frac{h}{p}\sin i(e\sin\omega + \sin\theta)\delta\Omega \tag{6.44c}$$

6.3 Application to Differential Drag

The relative motion models developed thus far assume a chief spacecraft governed only by Keplerian dynamics. For low-altitude orbits, one of the most important perturbing forces is aerodynamic drag. While in some cases drag is an undesirable consequence of operating at low altitude, drag can also be utilized as a method of orbit adjustment. Examples include passive end-of-life deorbiting for satellites [255], constellation phasing [250], and aerobraking [8]. The exact and linearized relative equations of motion can be straightforwardly extended to simulate the effect of differential drag using the perturbing acceleration vector $\boldsymbol{u}$. The chief spacecraft is simulated according to the full dynamics model, including the acceleration due to drag directed opposite the atmosphere-relative velocity vector. The perturbation term on the deputy then approximates the differential drag between the deputy and the chief; that is, if the chief and deputy spacecraft are identical in mass, aerodynamic properties, and attitude, this term should go to zero. Following this reasoning, the magnitude of the perturbation $u = |\boldsymbol{u}|$ is constructed as

$$u = \frac{1}{2}\rho_c \left|\dot{\boldsymbol{r}}_c\right|^2 \left(\frac{1}{\beta_d} - \frac{1}{\beta_c}\right), \tag{6.45}$$

where ρ_c is atmospheric density at the altitude of the chief and β is ballistic coefficient,

$$\beta = \frac{m}{C_D A}, \tag{6.46}$$

where m, C_D, and A are the mass, drag coefficient, and aerodynamic reference area of the vehicle, respectively. Ballistic coefficient can be understood as the ratio of inertial to aerodynamic forces on the vehicle, and will be treated as constant, which is generally a good approximation for hypersonic flight [256].

The drag force on the deputy spacecraft is directed opposite the atmosphere-relative velocity vector of the deputy. However, if the chief and deputy are close together, this direction is approximately aligned with the atmosphere-relative velocity of the chief. Furthermore, for most applications the rotational period of the central body is slow enough that the inertial velocity direction is a good approximation of the atmosphere-relative velocity. Making these two approximations, the perturbing acceleration due to differential drag can be treated as entirely in the $-y$ direction of the chief spacecraft velocity frame. Thus, the full and linearized relative motion EOMs (Eqs. (6.14) and (6.22), respectively) are implemented for differential drag by defining the velocity-frame components of the perturbing acceleration vector $\boldsymbol{u}$ as:

$$u_x = 0, \qquad u_y = -\frac{1}{2}\rho_c \left|\dot{\boldsymbol{r}}_c\right|^2 \left(\frac{1}{\beta_d} - \frac{1}{\beta_c}\right), \qquad u_z = 0 \tag{6.47}$$

It is important to note that in this case, because of the approximations summarized above, the full relative EOMs are no longer an exact representation of the dynamics. However, they avoid making the additional approximations of the linearized EOMs and should thus be expected to be more accurate.

6.4 Application to Atmospheric Entry Trajectories

For trajectories that pass deep within the atmosphere, such as for landing or aerocapture, aerodynamic drag goes from being a small perturbation to being the dominant force acting on the vehicle. In this section, Keplerian relative motion models are combined with an analytical approximation of hypersonic flight mechanics to make predictions of relative motion about atmospheric entry trajectories. The exoatmospheric portions of entry trajectories are typically either hyperbolic (in the case of sample return or planetary exploration) or highly elliptical (in the case of suborbital defense or rapid transport applications), and therefore the velocity frame descriptions of relative motion are well-suited for these applications. Relevant example missions include probe delivery by a carrier spacecraft on an entry trajectory [225], co-delivery of a probe network [231], or multiple independently-targetable reentry vehicles [224].

6.4.1 Enhanced Allen-Eggers Equations

The Allen-Eggers equations were developed in the 1950's and provide an analytical, closed-form description of ballistic (nonlifting) entry under certain assumptions relevant to the missile applications for which they were originally derived [253, 257]. Namely, these assumptions include:

- Ballistic entry, meaning a lift-to-drag ratio of $L/D = 0$
- Constant flight-path angle, $\dot{\gamma} = 0$

- Gravity is negligible compared to drag force, $D \gg g \sin\gamma$
- Zero thrust and constant mass, $T = \dot{m} = 0$
- Nonrotating planet, such that inertial and planet-relative velocity and flight-path angle are identical

Additionally, atmospheric density ρ is assumed to be an exponential function of altitude h,

$$\rho(h) = \rho_{\mathrm{ref}} \exp\left(\frac{h_{\mathrm{ref}} - h}{\mathcal{H}}\right), \tag{6.48}$$

where ρ_{ref} and h_{ref} are reference density and altitude (typically defined at sea level), respectively, and $\mathcal{H}$ is atmospheric scale height. Note that throughout this section the subscript x_0 refers to the value at entry, defined as reaching the atmospheric interface altitude $h_0 = r_0 - R$, whereas the subscript x_i refers to the value at some earlier exoatmospheric initial state.

In the original development of the Allen-Eggers equations, flight-path angle is assumed to be constant at its value at entry, $\gamma^* = \gamma_0$ [253]. This is a good approximation for steep entries, but for shallow entry trajectories an alternate value can improve accuracy [258]. The closed-form expression given in Ref. [259], described below, is used to compute γ^* in this study and was found to improve prediction accuracy in the examples shown later in this section. Let V_0, γ_0, and ρ_0 be the velocity, flight-path angle, and density at entry, respectively. Additionally, let $V_C = \sqrt{gR}$ be circular velocity where g is acceleration due to gravity at the surface and R is planetary radius. Then, γ^* is computed as

$$\sin\gamma^* = \sin\gamma_0 (2F^* - 1) \tag{6.49a}$$

$$F^* = \sqrt{1 + \frac{\mathcal{H}}{R \tan^2\gamma_0}\left\{C\frac{V_C^2}{V_0^2} + \left(\frac{V_C^2}{V_0^2} - 1\right)\ln\left(1 - \frac{\beta \sin\gamma_0}{\mathcal{H}\rho_0}\right)\right\}} \tag{6.49b}$$

$$C = \mathrm{Ei}(1) - \Gamma \approx 1.3179 \tag{6.49c}$$

where $\mathrm{Ei}(x)$ is the exponential integral

$$\mathrm{Ei}(x) = -\int_{-x}^{\infty} \frac{e^{-y}}{y} \mathrm{d}y \tag{6.50}$$

and $\Gamma \approx 0.57722$ is the Euler-Mascheroni constant.

The original Allen-Eggers equations do not include a closed-form expression for range, meaning distance along the planetary surface from the point of atmospheric entry to landing. However, Putnam and Braun develop such an expression in an extension and enhancement of the Allen-Eggers equations by directly integrating the simplified equations of motion and without making any additional assumptions [259]. Range s between the entry radius r_0 and current radius r can thus be estimated as:

$$s = \frac{\ln(r) - \ln(r_0)}{\tan\gamma^*} R \tag{6.51}$$

An expression for the offset in range between the chief and deputy landing locations (where $r = R$) can be derived by taking the first variation of Eq. (6.51) with respect to the entry radius and constant flight-path angle:

$$\delta s(r = R) = -R \left(\frac{\delta r_0}{r_0 \tan\gamma^*} + \frac{\ln(R) - \ln(r_0)}{\sin^2\gamma^*} \delta\gamma^* \right) \tag{6.52}$$

Note that in Eq. (6.52) $\delta r_0 = r_d - r_c$ and $\delta\gamma^* = \gamma_d^* - \gamma_c^*$ are both computed at the moment when the chief vehicle reaches atmospheric interface, $r_c = r_0$.

6.4.2 Methodology

By combining the relative orbit element expressions with Eq. (6.52), the range offset between landing locations due to a maneuver during exoatmospheric approach can be predicted analytically. This subsection gives an overview of the step-by-step procedure combining these relative motion models.

First, define the state of the chief and deputy vehicles at an initial time prior to atmospheric entry, and compute the relative orbit elements $\delta\boldsymbol{\alpha}$. In this work the chief state is computed by defining a state at atmospheric interface, computing Keplerian orbital elements, then changing the mean anomaly to a value of $M = -90°$ to obtain a state on that same orbit earlier in time. The Cartesian chief state is then computed and rotated into the velocity frame, and the deputy state is defined by adding a maneuver defined in the velocity frame. That is, the chief and deputy have identical position and different velocity vectors at the initial time. The deputy state is then converted to orbit elements and used to compute $\delta \boldsymbol{œ}$.

Second, the true anomaly of the chief vehicle at atmospheric interface altitude is computed via Eq. (C.11):

$$f_0 = \cos^{-1}\left(\frac{a(1-e^2)}{r_0 e} - \frac{1}{e}\right) \tag{6.53}$$

Third, the relative orbit element equations (Eqs. (6.37) and (6.43) for an elliptical chief or Eqs. (6.38) and (6.44) for a hyperbolic chief) are applied to compute the relative state of the deputy vehicle in the velocity frame at the epoch when the chief is at atmospheric entry.

Fourth, compute the radial position r, velocity magnitude V, and flight-path angle γ of both the chief and deputy. This requires converting the chief Keplerian state to inertial Cartesian vectors, as well as converting the velocity frame relative deputy state to an inertial absolute state. Fifth, compute δr_0 and $\delta\gamma^*$. Note that when evaluating Eq. (6.49) for the deputy the values used for γ_0, V_0, and ρ_0 are those at the time of *chief* entry, which for the general case is not identical to the state of the deputy at entry. Sixth, compute δs from Eq. (6.52); this is the range offset at landing predicted due to differences in entry states of the two vehicles. The predicted bearing of this offset is assumed to equal the heading angle of the chief at entry, $\psi_B = \psi_{0,c}$, where heading angle ψ is the angle between the projection of the velocity vector onto the plane normal to the radius vector and a due-North vector in that same plane (e.g. a 90° heading angle is due-East).

The procedure could stop here, but tends to be more accurate with an additional step. Due to the assumptions of the Allen-Eggers relations, Eq. (6.52) is poor at modeling cases such as a lead-follower, where δr_0 and $\delta\gamma^*$ are nonzero at the time of chief entry but the actual range offset will be very small, due only to the rotation of the planet between chief and deputy entries. Furthermore, Allen-Eggers assumes planar motion and is therefore unable to capture range offset due to out-of-plane relative motion between the chief and deputy. Thus, the seventh and final step is to compute range offset at time of chief entry δs_0 and geometrically combine δs_0 with δs to find the final prediction for range offset on the surface, δs_f. To do so, use the latitude and longitude of the deputy at time of chief entry along with the predicted range offset magnitude and bearing to compute an offset pair of coordinates. Then, compute the range and bearing angle from the coordinates of the chief at entry to the pair of coordinates just computed; this provides the final estimate of range and bearing between the chief and deputy landing locations. For convenience, Appendix D lists the equations required for this final step.

6.5 Numerical Results

In this section, the velocity frame relative motion models are applied to three relevant scenarios: two-body problem (Keplerian) dynamics about Earth, aerobraking at Mars, and ballistic entry at Earth. In each case, numerical simulation is used to compare the approximate models against the expected behavior, where the latter is determined via standard, individual simulation of the spacecraft dynamics; see Ref. [78] for the relevant equations of motion.

6.5.1 Simulation Methodology

The three degree-of-freedom equations of motion are numerically propagated using the Runge-Kutta method of order 5(4) via the open-source scipy.integrate.solve_ivp tool [192, 236], with relative and absolute error tolerances of 1×10^{-12}. The values assumed for physical constants are summarized in Tables 6.1 and 6.2, where T_p is the rotation period of the central body. Point-mass gravity is assumed in all cases. When applicable, atmospheric density in the truth simulation is calculated by linearly interpolating from a table output by the 2010 Global Reference Atmospheric Models for Earth and Mars [197, 151]; the data are sufficiently dense that linear interpolation is accurate despite the approximately exponential nature of density. In every case, the truth model makes the same assumptions about the underlying dynamics as the relative motion models (e.g., the effect of J_2 is also neglected in the truth models), but makes no additional approximations.

Table 6.1: Physical constants for Earth

Parameter	Value
μ	$3.986 \times 10^5\,\mathrm{km^3/s^2}$ [161]
R	6378.14 km [161]
g	9.81 m/s [161]
T_p	0.9973 days [161]
$\mathcal{H}$	8.5 km [260]
ρ_{ref}	$1.215\,\mathrm{kg/m^3}$ [260]
h_{ref}	0 km [260]

Table 6.2: Physical constants for Mars

Parameter	Value
μ	$4.305\,\mathrm{km^3/s^2}$ [161]
R	3397.2 km [161]

6.5.2 Keplerian Relative Motion

The results in this subsection demonstrate relative motion behavior in the velocity frame for purely Keplerian dynamics via two examples. Table 6.3 summarizes the chief orbit parameters and orbit element differences for each scenario, and Figs. 6.3 and 6.4 show the results for Scenarios A and B, respectively. In these figures "absolute" refers to separate simulation of the Keplerian dynamics, "relative" refers to propagation of the exact relative EOMs, "linearized" refers to propagation of the linearized relative EOMs, and "oe differences" refers to sweeping through the relative orbit element equations for all relevant true anomaly values.

Table 6.3: Orbital parameters for example scenarios

Scenario	a	e	i	Ω	ω	δa	δe	δi	$\delta\Omega$	$\delta\omega$	δM_0
A	−7000 km	1.2	0	0	0	0	0	0	0	0	0.5°
B	−7000 km	1.2	0	0	0	0	0.005	0	0	0	0

Scenario A is a lead-follower formation. As mentioned in the introduction, almost all of the relative motion is along the velocity direction, with only a small component along $\hat{\boldsymbol{v}}_n$. The linearization ignores this $\hat{\boldsymbol{v}}_n$ component and traverses down and back up along $\hat{\boldsymbol{v}}_v$. These results

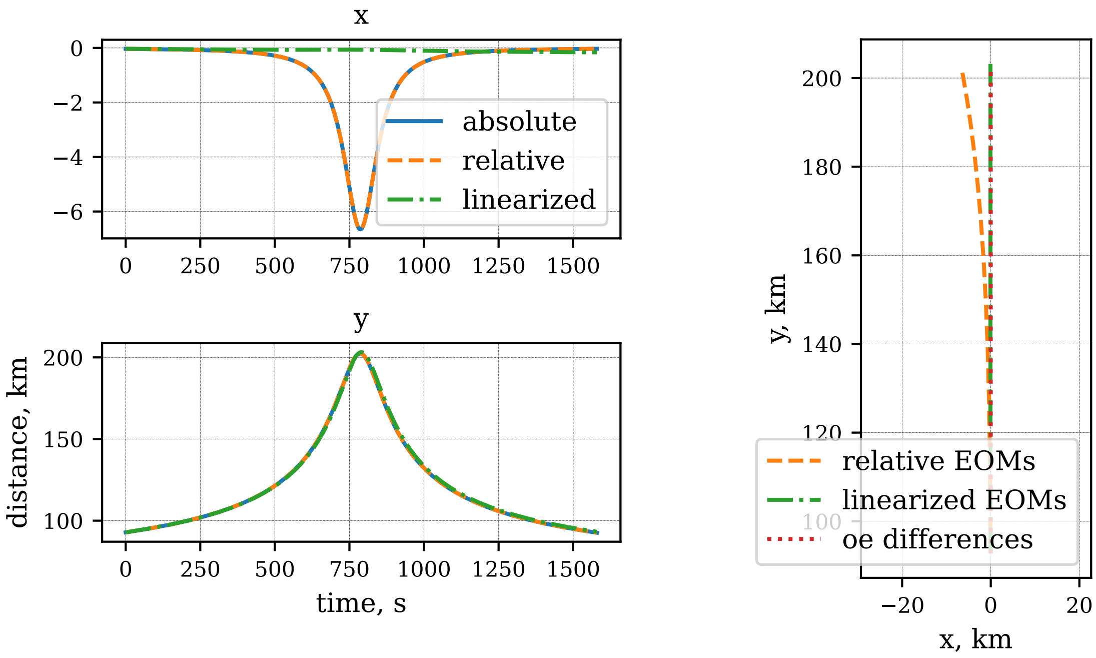


Figure 6.3: Relative motion for Scenario A

also show perfect agreement between the absolute and relative EOMs, as is expected in the absence of any approximations or non-Keplerian accelerations. Scenario B captures the behavior of a deputy spacecraft offset only in eccentricity. The lower-right plot intuitively shows how the deputy begins ahead of the chief, comes closer as the two spacecraft approach periapsis, and is behind the chief after periapsis.

6.5.3 Aerobraking

Aerobraking is the process of repeatedly passing through the upper atmosphere in order to reduce orbital energy and lower apoapsis. By utilizing atmospheric drag in place of propulsive maneuvers, this process enables mass-efficient transfer from an initial high-energy, highly-eccentric orbit to a lower-energy near-circular science orbit. Thus, aerobraking provides a relevant scenario for which drag acts as a perturbing acceleration and the chief orbit is highly-eccentric. In this subsection, a chief orbit is defined based loosely on the aerobraking campaign for the 2001 Mars Odyssey mission [8], then the behavior of a deputy that is initially offset only in phase and attitude

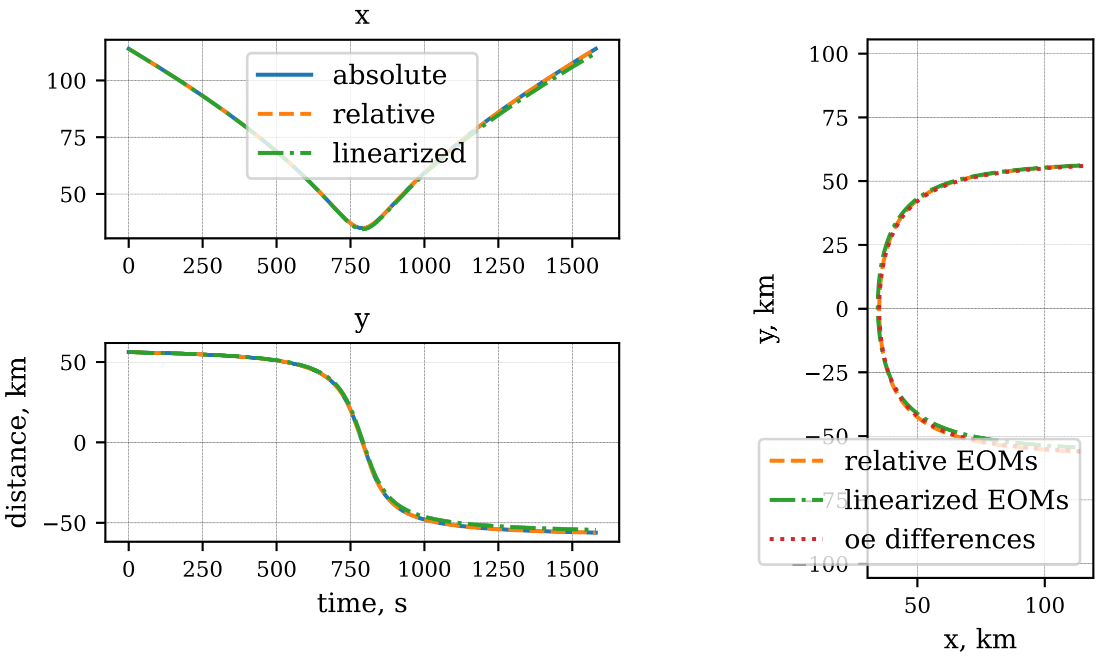


Figure 6.4: Relative motion for Scenario B

is simulated using the expressions for $\boldsymbol{u}$ developed in Sec. 6.3.

The initial chief orbit is defined by apses with altitudes of 100 km and 10 000 km, and the deputy is initially offset by $\delta M = 0.5°$. Both spacecraft are assumed to have mass of 426 kg and aerodynamic reference area of 11 m^2, similar to Mars Odyssey [8]. During aerobraking, the attitude of the solar panels with respect to the velocity vector can be adjusted in order to dissipate more or less kinetic energy each pass. For the purpose of this example, the deputy spacecraft is assumed to be in a slightly higher-drag configuration than the chief, resulting in drag coefficients of 2.2 and 2 for the deputy and chief, respectively; these values are based on past aerodynamic analysis for a similar scenario [251]. Lift and side force are neglected for both vehicles. Figure 6.5 shows the results from simulating this relative motion scenario for a duration equal to five periods of the initial chief orbit.

As mentioned in Sec. 6.3, in this case the relative EOMs already include some level of approximation, and indeed Fig. 6.5 shows that there is small but significant disagreement, especially near periapsis. The linearized relative EOMs perform well even across five orbits, although during

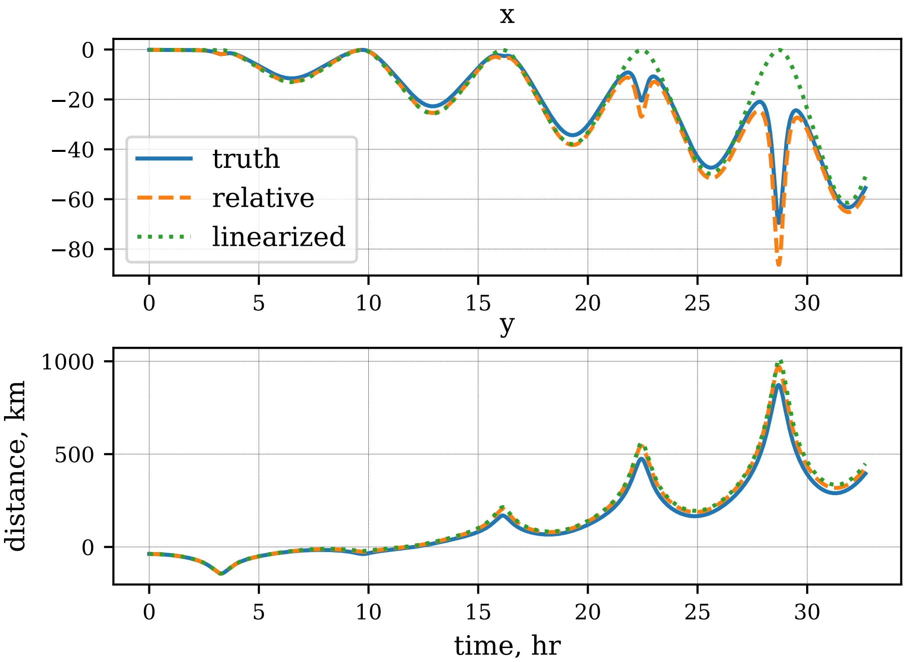


Figure 6.5: Velocity frame relative motion about aerobraking trajectory

the fourth and fifth passes through periapsis (near 22 and 28 hours, respectively) the linearization misses important behavior in the $x-$axis direction. Note, however, the significantly different magnitudes between motion in the $x-$ and $y-$directions.

6.5.4 Entry Trajectories

The procedure for analytically estimating range offset is applied to three ballistic entry vehicles and trajectories, chosen to serve as representative examples and to align with the examples selected in Ref. [259]. The first scenario is based on the sample return capsule for the NASA Stardust mission [261, 262], which entered Earth's atmosphere on a hyperbolic return trajectory. A second scenario is constructed as a modified version of the Stardust scenario with a steeper entry flight-path angle. The third scenario is a “high ballistic coefficient vehicle on a steep, high-energy suborbital trajectory[259]”; this case is referred to as strategic and is representative of a ballistic missile re-entry trajectory [263]. The parameters for each scenario are summarized in Table 6.4. In every case, the radius at entry is defined as the atmospheric interface altitude $r_0 = 125\,\text{km}$, with entry longitude θ_0 and latitude ϕ_0 set to 0° and an entry heading angle of $\psi_0 = 70°$. The chief orbit is fully defined by the entry state, but the semi-major axis, eccentricity, and final range are also included for reference.

Table 6.4: Entry trajectory chief orbit descriptions

Scenario	V_0, km/s	γ_0, deg.	β, kg/m^2	a, km	e	s_c, km
Stardust	12.8	-8.2	60	-7554.	1.848	805.064
Steep Stardust	12.8	-15	60	-7593.	1.815	375.745
Strategic	7.2	-30	10000	6136.	0.477	213.991

For each chief orbit scenario, three different deputy orbits are considered. In each case the chief is initialized with a mean anomaly of $M = -90°$, the deputy is initialized at the identical position, and the velocity vector of the deputy at the initial time is modified by a maneuver with $\Delta V = 10\,\text{m/s}$. The three deputy scenarios correspond to directing this maneuver along each of the unit vectors of the velocity frame of the chief spacecraft. Thus, nine total scenarios are considered in this section.

Figure 6.6 and Table 6.5 summarize the comparison between predicted and simulated range offset for each of the nine scenarios under consideration. Figure 6.6 shows the magnitude of the percent error, meaning normalized by the simulated range offset, whereas Table 6.5 reports the absolute values. Table 6.6 compares the predicted and simulated bearing of the offset between chief and deputy landing locations in each scenario, and shows that the predicted bearing was approximately correct in all cases.

Table 6.5: Simulated and analytically predicted range offsets, km

	$\hat{\boldsymbol{v}}_n$		$\hat{\boldsymbol{v}}_v$		$\hat{\boldsymbol{v}}_h$	
Scenario	truth	pred.	truth	pred.	truth	pred.
Stardust	287.737	334.617	58.484	81.031	13.059	12.772
Steep Stardust	69.809	78.490	14.660	16.537	12.808	12.497
Strategic	5.780	5.565	1.880	1.903	2.934	2.547

In the cases of maneuvers along the $\hat{\boldsymbol{v}}_n$ and $\hat{\boldsymbol{v}}_v$ directions, the predictions are consistently

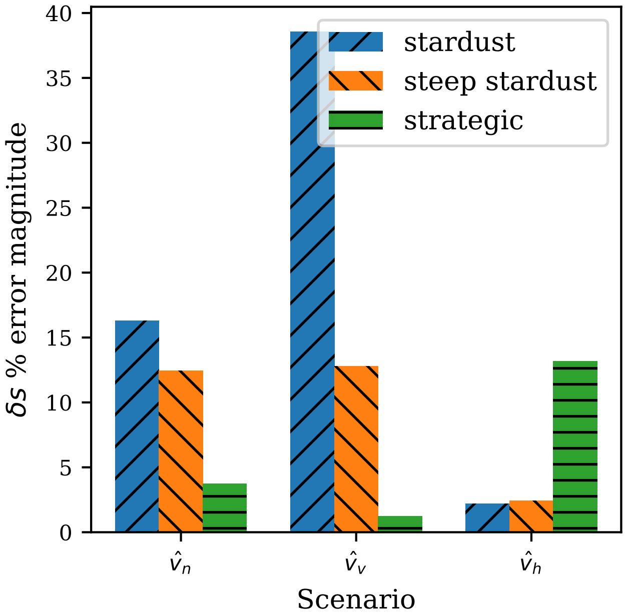


Figure 6.6: Absolute value of percent error of range offset error, where x-axis label denotes direction of 10 m/s maneuver during approach

Table 6.6: Simulated and analytically predicted offset bearing, deg

	$\hat{\boldsymbol{v}}_n$		$\hat{\boldsymbol{v}}_v$		$\hat{\boldsymbol{v}}_h$	
Scenario	truth	pred.	truth	pred.	truth	pred.
Stardust	70.163	69.985	70.357	70.124	-18.103	-16.553
Steep Stardust	70.005	69.964	70.766	70.613	-18.945	-18.603
Strategic	70.187	70.137	72.119	71.773	-18.575	-18.321

most accurate for the strategic scenario and least accurate for the Stardust scenario. This is as expected; the steep flight-path angle and high ballistic coefficient of the strategic scenario mean the Allen-Eggers assumptions are much more accurate than in the case of Stardust, despite the higher entry speed of the latter. Most of the error present in the range predictions for these cases is due to the disparity between the true entry trajectories and the Allen-Eggers approximations. The cases corresponding to a maneuver along $\hat{\boldsymbol{v}}_h$ appear to present an exception to this trend based on Fig. 6.6, but examination of Table 6.5 reveals that the actual error is similarly small (within 0.5 km) in all cases. Maneuvering along $\hat{\boldsymbol{v}}_h$ primarily serves to offset the orbital plane of the deputy. As a result, the difference in r_0, γ^*, and V_0 is negligibly small, but the deputy enters at a different location and with a different heading angle. Thus, in the $\hat{\boldsymbol{v}}_h$ cases almost all of the final range offset is due to existing offset at entry, as accounted for by step 7 of the prediction procedure. Finally, note that while the percent error values are relatively high in some cases, the errors are small compared to the total range covered by the chief (s_c in Table 6.4): less than 6% in all cases.

6.6 Conclusions

Describing relative motion in terms of velocity frame components is an intuitive model for motion about highly-eccentric chief spacecraft, and provides a complementary alternative to traditional descriptions in the Hill frame. The equations of motion and orbit element difference equations developed in this work give a direct approach that could be appropriate for onboard use, such as within a navigation filter or for the design of reference trajectories. Results for several simple scenarios about a hyperbolic chief show good agreement between the linearized and exact solutions and develop a more intuitive understanding of the types of relative motion possible about flyby, aerobraking, or atmospheric entry trajectories. The procedure developed in this work for analytically predicting the offset in final range for an atmospheric entry trajectory extends the range of application of these relative motion models to include steep ballistic entry vehicles, such as planetary probes. This method would enable rapid onboard estimation of the impact of a maneuver during approach on the entry, descent, and landing profile of ballistic probes.

6.7 Auxiliary Variables List

$$\zeta = (e^2 + 2e\cos f + 1) \tag{6.54}$$

$$\alpha = (e\cos f + 1) \tag{6.55}$$

$$\kappa = \frac{x_c x + y_c y}{r_c^2}, \tag{6.56}$$

$$\eta = \sqrt{1 - e^2} \tag{6.57}$$

$$\eta_h = \sqrt{e^2 - 1} \tag{6.58}$$

Chapter 7

Conclusions

7.1 Looking Ahead: Uranus Aerocapture with Fly-Ahead Entry Probe

In this final section, the results and methods contributed by this dissertation are applied to investigate the utility of a fly-ahead probe for aerocapture at Uranus. An in-depth study of pathfinder probes, including detailed mission design and full exploration of the trade space, is out of scope and remains an interesting topic for future work. Rather, this coda serves to illustrate how the range of tools developed in the preceding chapters can be used together to address a relevant problem while also providing novel insight into the fly-ahead probe concept.

7.1.1 Introduction

Missions to explore the ice giants Uranus and Neptune present the most promising, yet perhaps also the most challenging, potential applications for aerocapture [18]. The duration of interplanetary cruise and the mass available for science payload on orbit are severely limiting factors for these missions due to the large heliocentric distances of the two planets (roughly 20 AU and 30 AU, respectively). At Uranus, the highest-priority destination for new planetary science missions in this decade [29], aerocapture could reduce the total mass required for orbit insertion by an estimated 40% while reducing transit time by 2-5 years (15-30% of total transit time) [32, 25, 18]. However, aerocapture at the ice giants is also uniquely challenging. The large gravity well of Uranus leads to high minimum entry speeds, and this issue is exacerbated by the fact that trajectories with shorter trip times generally increase the hyperbolic excess velocity of arrival [19, 264]. Worse, very little is known about the atmosphere. During its flyby of Uranus in 1986 the *Voyager 2* probe performed radio occultation experiments and took measurements in infrared and ultraviolet wavelengths [27], and optical wavelength stellar occultations provide some information about the atmosphere at very high altitudes [19], but there is still high epistemic uncertainty regarding the Uranian atmosphere and its variability over time. In previous studies, this has typically been mitigated by requiring higher control authority in the atmosphere (that is, a mid-L/D of 0.4-0.5) than would be necessary for aerocapture at Titan or the inner planets [58, 19]. However, this would require the development and flight qualification of novel aeroshells, an expensive and time-consuming process.

A potential alternative strategy for mitigating this high epistemic uncertainty is to send an entry probe into the Uranian atmosphere some time in advance of the arrival of the orbiter for aerocapture. The probe would take measurements during its descent through the atmosphere, then relay data back to the orbiter, possibly via an accompanying relay smallsat. The orbiter could adjust parameters for onboard guidance and navigation accordingly, and potentially perform a corrective maneuver. This concept, referred to as a “fly-ahead probe” or “pathfinder probe,” has been proposed a number of times for ice giants aerocapture [19, 71]; however, there has not been a quantitative investigation of to what extent this would actually improve targeting performance for the orbiter. Whether a probe is included specifically for this purpose or the concept of operations for a scientific probe is modified in order for it to double as a pathfinder probe, the fly-ahead probe adds complexity and contributes to the overall cost and risk of the mission. Therefore, it would be critical for a proposal team or mission manager to address a few basic questions about the concept. Most importantly: to what extent is targeting performance, in terms of the ΔV required for correction maneuvers or the overall probability of successful aerocapture, improved by the addition of a fly-ahead probe? Numerous secondary questions also arise; for example, how much further benefit is gained by the addition of a second/third/fourth probe? Also, the greater the time elapsed between probe and orbiter entry, the more the planet rotates and the more the atmospheric density profile potentially changes. However, a significant communication delay and limited bandwidth would mean that, if the time between entries is below a certain threshold, then the orbiter would need to perform its update autonomously, without the benefit of ground-in-the-loop. Because we do not know the true aleatory variability of the Uranus atmosphere, any answers to these questions will be approximate, but without a quantitative estimate of the benefit conferred by a fly-ahead probe it is impossible to objectively consider whether or not its addition would be a net benefit.

7.1.2 Methodology

The aerocapture scenario described in Ref. [265] is used as the example reference mission for this study, with key parameters and their associated uncertainties repeated in Table 7.1. However, unlike in that study, here the probe is assumed to enter the atmosphere only three days before the orbiter. The target apoapsis for this scenario is set to 550 000 km, or approximately 21 Uranus radii. Bank angle modulation is assumed as the control mechanism for the orbiter, and so the fully-numerical predictor-corrector aerocapture guidance (FNPAG) algorithm in Mode 1 (apoapsis targeting), as described in Sec. 2.3, is implemented as the baseline solution. In order to leave sufficient margin for mitigating uncertainties, the initial and final bank angle parameters are initialized as 45°and 135°, respectively. Longitudinal guidance via deadbanding is not implemented for this example, and the bank angle is commanded instantaneously according to the guidance commands; both of these simplifications results in somewhat better apoapsis targeting performance than is realistic. The nominal switching time is 240 seconds after entry at the atmospheric interface altitude of 1000 km above the 1 bar pressure level, which has a radial distance of 25 559 km and is treated as the “surface” for the purpose of computing altitude as an input to the density model. In this implementation of FNPAG, the predictor uses nominal values of the vehicle aerodynamics and a nominal density profile, but does maintain perfect knowledge of the state and uses the same dynamics model as the truth simulation. A next step would be to add white noise to the value of the state used by the predictor, simulating navigation error.

UranusGRAM is used to produce a nominal profile of density for use by the FNPAG predictor, and profiles of perturbed density for the truth simulation Monte Carlo trials are generated by

Table 7.1: Aerocapture scenario parameters

Parameter	Mean	Dispersions
EFPA	$-10\,°$	$3\sigma = 0.3\,°$
Entry Velocity	$29\,\mathrm{km\,s^{-1}}$	$3\sigma = 3\,\mathrm{m\,s^{-1}}$
Orbiter β	$146\,\mathrm{kg\,m^{-2}}$	$\pm 5\,\%$
Orbiter L/D	0.24	$\pm 5\,\%$
Density	UranusGRAM	UranusGRAM

UranusGRAM using the same settings. In this example, FNPAG does not include an atmospheric estimation component. Thus, in the absence of an update, the guidance algorithm will repeatedly make incorrect predictions about the future state because of the modeling error between the nominal and actual density profiles.

To approximate the benefit gained by a fly-ahead probe, the nominal density profile used by FNPAG is updated using the Kalman measurement update equations as shown in Sec. 3.6. For these preliminary results, the true density profile experience by the probe is assumed to equal the true density profile experienced by the orbiter; this reflects the assumption that the probe enters the atmosphere only a few days before the orbiter and at a similar location. This short time between probe and orbiter entry requires that the update be performed autonomously onboard, but the key benefit is that the expected difference between the atmosphere experienced by the probe and that experienced by the orbiter is smaller. The probe is assumed to make measurements of density every 20 km, and Gaussian white noise with a standard deviation of 0.02 is applied to the normalized perturbations of the observed densities. The noise is applied to the normalized values rather than raw density values to create a consistent dispersion magnitude across the full altitude range. The Kalman measurement equations use this same value for the assumed measurement noise, reflecting an assumption of an accurately-calibrated accelerometer or inertial measurement unit.

7.1.3 Results

Figure 7.1 shows the prior and posterior mean (nominal) and uncertainty bounds obtained by following the approach described above, in terms of normalized density perturbations. As shown,

Table 7.2: Apoapsis altitude targeting results, $N = 100$

Scenario	Mean	Standard Deviation
No Update	681 556 km	2 202 310 km
Update	548 867 km	24 258 km

even with measurement noise and 20 km gaps between observations the posterior mean is far more accurate to the true profile than the prior model, which is normalized by itself and is thus simply zero at all altitudes. In the Monte Carlo analysis comparisons, the dashed blue line represents the nominal density profile used by the orbiter in the no-update case, and the dashed orange line represents the nominal profile after incorporating an update based on the probe measurements. Clearly, the latter model of the environment is significantly more accurate, and a corresponding improvement in guidance targeting performance should be expected.

The summary results for apoapsis altitude targeting, shown in Table 7.2, strongly confirm this intuition. In a comparison of Monte Carlo analyses, the mean apoapsis altitude for the updated scenario is incrementally better than that without the update, but the standard deviation is a full two orders of magnitude lower. This is due in part to the fact that, in the scenario without the update, there are some failure cases skewing the results whereas with the update no failures occur. This clearly shows the benefit of a fly-ahead probe, and by translating expected apoapsis targeting performance into expected correction maneuver ΔV costs the utility of adding a fly-ahead probe for aerocapture can be readily quantified with these results.

7.1.4 Discussion

The use of the Kalman update equations to improve the performance of the FNPAG algorithm demonstrates some of the key contributions of this dissertation. In doing so, intuition – that measurements obtained by a fly-ahead probe should improve aerocapture targeting performance – is confirmed and quantified. That said, an in-depth treatment of the fly-ahead probe concept is beyond the scope of this dissertation, but these preliminary results illuminate a number of interesting possible avenues.

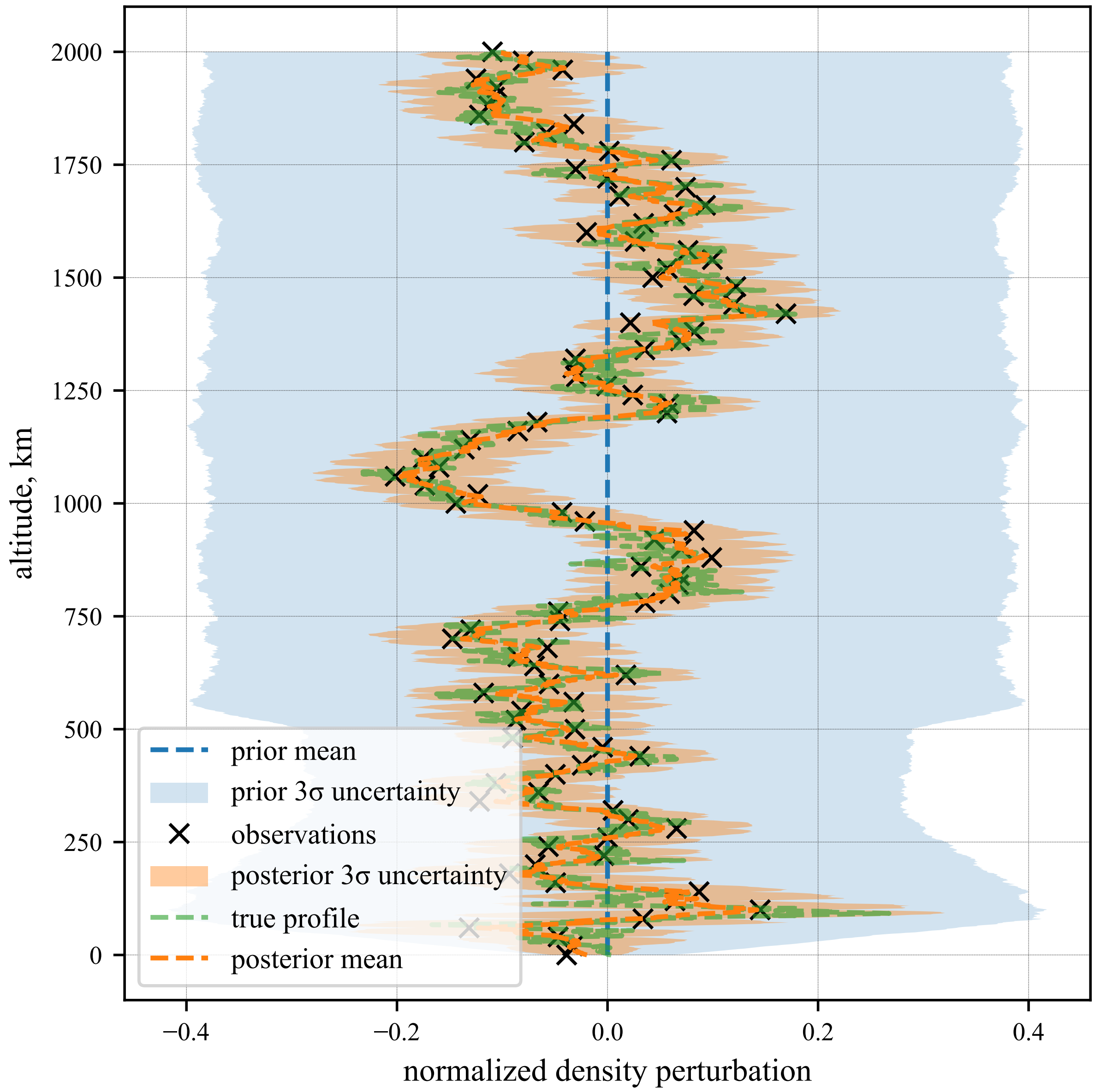


Figure 7.1: Prior nominal density updated by noisy probe measurements of true density via Kalman equations

For one, the ballistic coefficient of the probe could be constrained in order to use AeroDrop, as described in Chapter 4, to deliver the probe to entry with approximately the same entry flight-path angle and entry velocity as the orbiter uses for aerocapture, but earlier in time. This would potentially reduce the magnitude of, or even eliminate entirely, the divert maneuver required of the orbiter after releasing the pathfinder probe shortly before entry. The relative orbit element relationships derived in Chapter 6 could then be applied to compute the magnitude and direction of any required divert maneuver. With a fully-designed probe trajectory, a density profile corresponding to that time and location of entry could be used for probe observations rather than using the simplifying assumption that the density experienced by the probe is the same as the true density that the orbiter will encounter.

Atmospheric estimation within the guidance algorithm would be expected to significantly improve targeting performance, especially in the no-update scenario. A simple linear scale factor could be estimated as described in Sec. 2.3. However, note that the true profile of density shown in Fig. 7.1 oscillates between more or less dense than the nominal; this is typical of density perturbations as modeled by GRAM. Thus, just because the orbiter is encountering densities that are higher than expected does not mean that it is helpful to assume the rest of the profile is also denser than the nominal. To account for this, one could add an exponential correlation function to the scale factor, such that the correction decays back to one (i.e., regresses toward the prior mean) for densities at altitudes more than some correlation length away. An interesting extension would be to estimate this correlation length on board using the posterior covariance matrix after each Kalman update. The value would need to be estimated, rather than computed exactly, because the true correlation structure in the data output by GRAM does not follow an exponential kernel, and indeed is not stationary.

7.2 Summary of Contributions

The aspiration of this thesis is to advance the state of the art for aerocapture so that it may be incrementally closer to readiness for implementation by the missions that would most benefit from it. The preceding chapters summarize several different but interrelated contributions to this overarching aim. For single-event jettison drag-modulated aerocapture of small satellites, simpler is better. The guidance algorithm that could reasonably be considered state of the art for this application, a numerical predictor-corrector, is robust and accurate but is computationally challenging.

By shifting the main burden of computation to between final navigation update and atmospheric entry, and then by relying on arithmetic calculations on a large family of pre-computed reference trajectories, the guidance algorithm proposed in this work is able to match the performance of the state of the art while improving on the computational demand.

An emerging class of stochastic guidance algorithms, further discussed below, motivates the ability to rapidly and accurately predict future state uncertainty onboard a spacecraft. However, many of the existing approaches rely on an overly-simplified model of the atmosphere, ignoring one of the most important and complex sources of uncertainty for aerocapture. In this dissertation, both linear and nonlinear dimensionality reduction techniques are applied to this problem to produce compact models of an uncertain atmosphere. The linear model can readily be updated in flight based on new measurements and can be incorporated into linear covariance analysis to make accurate predictions of the future state uncertainty. This is enabling for rapid trade space exploration in addition to stochastic guidance.

This dissertation also provides the first systematic study of co-delivery as a possible architecture for multi-vehicle missions. The feasibility of AeroDrop, the method of designing a probe and aerocaptured-orbiter to seek the same entry conditions, is assessed at a range of destinations across the solar system. Linear and nonlinear methods are developed for designing co-delivery trajectories for an entire network of ballistic surface probes, and both are employed to design an example mission to form a seismic network in the Cerberus Fossae region of Mars. In support of these co-delivery studies, relative motion expressions are derived that provide better insight into the behavior of spacecraft formations than the standard orbit frame for highly-elliptical chief motion. These expressions are combined with first-order variations of the extended Allen-Eggers solutions for steep ballistic atmospheric entry, and the combined model is able to accurately analytically predict landing location offset from a maneuver ΔV applied during the approach phase.

Finally, fly-ahead probes for aerocapture are considered as a motivating case study. Quantitatively addressing this problem illustrates how the various contributions of this thesis can jointly enable straightforward design and modeling of interesting multi-vehicle scenarios involving aerocapture or entry.

7.3 Notes on Future Work

A number of avenues exist for potentially extending the contributions of this dissertation. In some cases, initial progress has already been made. The following subsections summarize some of these ideas with the aim of enabling the future researcher.

7.3.1 Stochastic Guidance for Aerocapture and Entry

As has been discussed at length, autonomous closed-loop guidance is a key capability for aerocapture and entry vehicles, and in order to be successful this guidance must adequately mitigate uncertainties due to the atmosphere, vehicle state at entry, etc. Flight-heritage and state-of-the-art guidance algorithms for aerocapture and entry, including those presented in Chapter 2, generally treat the problem as deterministic. The guidance implicitly controls uncertainty by repeatedly updating commands based on new estimates of the current state and environment, and the effectiveness of this approach is estimated in uncertainty quantification studies of the closed-loop dynamics, namely via Monte Carlo analyses. An alternative approach would be to *explicitly* control uncertainty by considering the effect of present and future control decisions on the trajectory uncertainty evolution; this is broadly referred to as stochastic guidance. A valuable aspect of stochastic guidance is that nominal performance (e.g. mean ΔV) and performance at the margin (e.g. 99th-percentile ΔV) can be quantitatively balanced. This is in contrast to the more typical approach of designing guidance to control the nominal performance then adding margin as necessary to account for performance at the margin, which can result in unnecessary conservatism.

At the risk of oversimplification, the necessary ingredients for stochastic guidance are: 1) models of relevant uncertainties, including the state and environment; 2) a method for predicting a future state and its associated uncertainty (e.g. mean and covariance); and 3) an algorithm for making updated control decisions based on these predictions. Importantly, all of these components must be efficient enough to plausibly execute repeatedly on a limited-capacity onboard computer.

Chapter 3 develops relatively compact, high-fidelity models of atmospheric density variability for onboard use. In Sec. 3.7, linear covariance analysis is demonstrated as a method of incorporating a KLE representation of density into accurate approximate predictions of the future mean and covariance of the vehicle state. Thus, this dissertation includes many of the relevant ingredients for applying stochastic guidance to aerocapture and entry problems, but stops short of developing a closed-loop stochastic guidance algorithm. The rest of this section reviews the relevant literature and summarizes recent work, including by the authors, related to this final algorithmic component.

The high-level task of a stochastic guidance algorithm is to solve a stochastic optimal control problem that can be summarized as follows: *given* the estimated current vehicle state and probability distribution functions that model the relevant uncertainties, compute a control policy that *minimize* some cost function *subject to* chance constraints. Chance constraints enforce an upper bound on the probability of some event occurring, and must replace deterministic constraints in cases where uncertainties are unbounded (e.g. Gaussian distributed).

The general stochastic optimal control problem (nonlinear dynamics, arbitrary probability distributions, arbitrary chance constraints, etc.) can be solved via either the Hamilton-Jacobi-Bellman partial differential equation (for continuous-time systems) or dynamic programming (for discrete time) [266, 267]. In either case, however, the general problem is intractable to solve numerically because all possible states must be visited and the state space grows exponentially with the number of dimensions, thus suffering from the curse of dimensionality [268]. Therefore, the typical approach and the one taken here is to turn to techniques that address the stochastic optimal control problem in more computationally-tractable ways, motivated by the general problem statement above.

A wide range of approaches to stochastic optimal control exist in the literature, with varying sets of assumptions or approximations[1] . One set of techniques, referred to as robust optimal control, handles dynamics under the influence of dispersions but requires a bounded description of system uncertainties [269]. However, many practical applications, including the aerocapture and entry

---

[1] This literature review benefited greatly from the introduction provided in [98].

problems, involve probabilistic descriptions of uncertainties such as Gaussian distributions, and are thus not naturally suited to a robust optimal control approach. Differential dynamic programming and stochastic differential dynamic programming offer significant computational improvements to standard dynamic programming [270, 271], though they require analytical second-order derivatives of the dynamics and cost function. Model predictive control and, more recently, stochastic model predictive control have been applied successfully to a range of problems [272, 273, 269]. These techniques rely on repeatedly solving an open-loop optimal control problem online, providing implicit feedback by solving from an updated state estimate. For the nonlinear stochastic systems common in aerospace, however, the optimal control problem may be too computationally expensive to solve repeatedly onboard, and the area of stochastic nonlinear model predictive control is very new [269].

Recent works have applied advanced uncertainty propagation methods to aerocapture, including analytical polynomial chaos expansion and other spectral methods [89, 90], the Perron-Frobenius operator [91], and the stochastic Liouville equation [92]. A stochastic terminal point control problem is solved in [93] for an uncertain atmosphere and entry state. The aforementioned works are all limited, however, by only considering an exponential model of atmospheric density. Several approaches to stochastic numerical predictor-corrector guidance are proposed in [94] using the unscented transform for uncertainty propagation, and a similar stochastic retargeting method is applied to aerocapture in [95] using a Gaussian mixture model for uncertainty propagation.

The approach that forms the focus of this section is to linearize the dynamics about a reference trajectory, then jointly optimize the nominal control (and thus reference trajectory) and feedback gains for a linear state feedback control law while minimizing a cost function and meeting chance constraints. This technique is known as covariance steering and was developed in the 1980s for the infinite-horizon problem [274]. It has seen renewed attention in recent years with successful application to the finite-horizon problem and with the insight that optimal covariance control for stochastic linear systems can be cast as a deterministic convex optimization problem [275, 276, 277, 278, 279]. Applications of covariance steering to aerospace problems, which are typically characterized by nonlinear dynamics and nonconvex constraints, rely on successive convex programming and other convexification techniques to reformulate the problem as necessary [280, 281, 96, 98, 97].

The linearization of the true dynamics about some reference trajectory, whether obtained analytically or numerically, is only valid within some local neighborhood of the reference. Thus, covariance steering for nonlinear systems is typically performed in an iterative sense, where the optimal solution of the current iteration becomes the reference trajectory for the subsequent iteration. The process repeats until some termination criteria are met, often that the current and previous solutions differ by less than some tolerance. The iterative covariance steering procedure is summarized in Algorithm 2 [96], and is detailed in Appendix E.

Algorithm 2 Iterative Covariance Steering Procedure

```
input: Initial state mean and covariance and initial guess for the reference nominal control
output: Nominal control and feedback gains for each time in the discretization
while Termination criteria are not met do
    Numerically propagate the (nonlinear) nominal trajectory (E.2)
    Linearize the system dynamics (e.g., evaluate analytical expressions for the necessary Jacobians) (E.7)
    Discretize the system (E.13)
    Solve the convex optimization problem (E.28)
    Set nominal control equal to the obtained optimal solution
```

A limitation of iterative covariance steering as it is typically implemented (and as presented in Appendix E is the treatment of the perturbation term as a Weiner process; effectively, the disturbances are modeled as Gaussian white noise. As detailed in Chapter 3, atmospheric density variability is better modeled by a Gaussian Random Field, including spatial correlation structure and uncertainty bounds that change with altitude. Recent work by Ridderhof et. al (2022) successfully applied iterative covariance steering to the aerocapture problem with density variability defined by a GRF [96]. Note that the derivation in Appendix E models disturbances as a function of only time, and that state-dependent uncertainty changes the assumptions and invalidates the covariance steering method. The work by Ridderhof et. al addresses this by evaluating the covariance function along the nominal trajectory, thus providing a mapping from the spatially-varying uncertainties to temporally-defined Gaussian disturbance values at each discretization step [96] as visualized in Fig. 7.2. However, the solution relies on state history feedback to convexify the problem as described by Eq. (E.16), potentially increasing the problem scale beyond the point of feasibility for repeated onboard computation for aerocapture or entry applications. A vehicle could instead directly implement the resulting feedback law as the onboard guidance, but this approach is potentially subject to poor performance if the vehicle deviates far enough from the reference trajectory that linearization is no longer a good approximation. Existing applications that do relax the covariance equation to an inequality as shown in Appendix E, such as the recent contribution by Benedikter et. al (2022)[2] [98], do not also incorporate more complex disturbance models such as a GRF, assuming instead that disturbances are independent from each other. Furthermore, none of the contributions mentioned here explicitly incorporate a method of updating the onboard uncertainty model using noisy measurements.

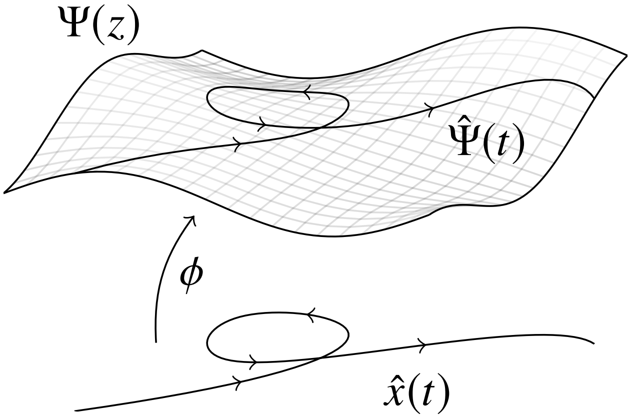


Figure 7.2: From [96]: Mapping a state-dependent random field Ψ to the time-dependent random process $\hat{\Psi}$ by evaluating along the nominal trajectory $\hat{x}$

Figure 7.3 summarizes a proposed novel guidance architecture. An iterative covariance steering procedure is solved via successive convex programming, incorporating a GRF model of uncertainty and using the change of variables to keep the computational effort low enough that this

---

[2] Note that since initial publication it has become clear that the relaxation used in Ref. [98], while numerically very accurate for the applications tested in that work, is not lossless in general; see the published comment in Ref. [282] and the authors' response in Ref. [283].

optimization could feasibly be run repeatedly onboard. This stochastic guidance runs at a relatively low rate and serves as a re-planner for a lower-level controller that simply applies the linear feedback control law, such that the nominal control and feedback gains for each time are updated every time the guidance returns a solution. Both the stochastic guidance and lower-level controller operate on the current navigation estimate of the state. Furthermore, the vehicle maintains an onboard belief state of the relevant environmental uncertainties (e.g. density), represented as a KLE and periodically updated with noisy measurements as described in Sec. 3.6. Thus, the solution to the covariance steering problem evolves as the vehicle state deviates from the expected reference *and* as new information is gathered about the random field describing environmental uncertainties. A wide range of practical problems could benefit from this kind of online stochastic guidance, including rocket ascent or drone path planning.

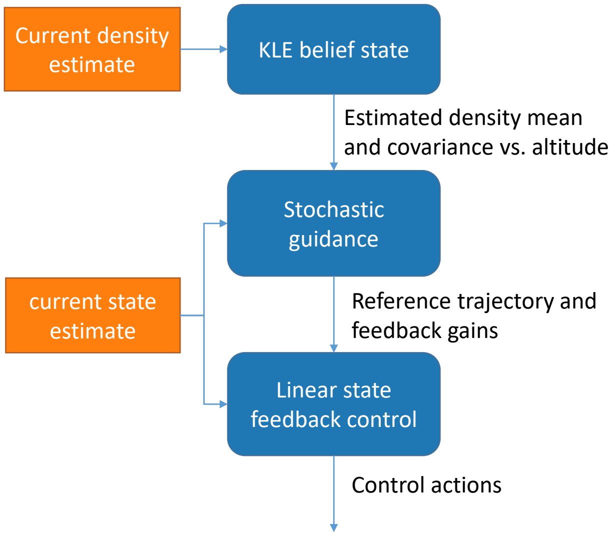


Figure 7.3: Summary of Proposed Guidance and Control Architecture

In order to ensure feasibility of the guidance solutions, additional problem-relevant constraints would need to be included, which in general will require identifying a suitable convexification approach. For example, aerocapture and entry trajectory optimizations should incorporate constraints on peak heat flux, total heat load, and peak sensed acceleration. Another useful constraint would be limiting the rate of change for the control, such that instantaneous changes in attitude are not assumed by the guidance algorithm; this has a noticeable effect if the crossrange targeting of the vehicle is simultaneously being controlled. The performance of the proposed guidance architecture would then need to be quantified and compared with existing solutions. The behavior of the vehicle when faced with significant modeling error, such that the a priori assumptions about uncertainty levels poorly match the truth, is of particular interest. Two versions of the stochastic guidance, one with active onboard replanning as summarized in Fig. 7.3 and the other simply flying the resulting feedback law after an online optimization as proposed in Ref. [96], would need to be directly compared in addition to testing against existing methods such as NPC guidance or terminal-point control.

Initial progress has been made toward this goal. Figure 7.4 shows results from a simple covariance steering example on a discrete linear time-invariant system, a one-dimensional double integrator. The solid lines labelled *cvx* show the mean and standard deviation as represented by the parameters of the convex program; the dashed/dotted lines labelled *MC* show the mean and covariance as estimated based on 10,000 Monte Carlo trials numerically propagating the true system with randomly generated disturbances. The mean state is constrained to begin and end at rest while going from a position of 20 to the origin, and the final state covariance is constrained to be equal to 0.1 times the initial state covariance. Following the methods outlined in Appendix E, in this example the disturbance term is assumed to be Brownian motion and the covariance propagation equation is relaxed to an inequality via a slack variable. The cost function is of the form shown in Eq. (E.28a) and penalizes state variance more than control variance. Chance constraints on both the state and control are implemented at each time node, and as seen in Fig. 7.4 these constraints are respected. The example is implemented in python[3] using the cvxpy package [284, 285], and could form the starting point for an iterative covariance steering module.

---

[3] thanks to G. Rapakoulias for significant help troubleshooting this new implementation

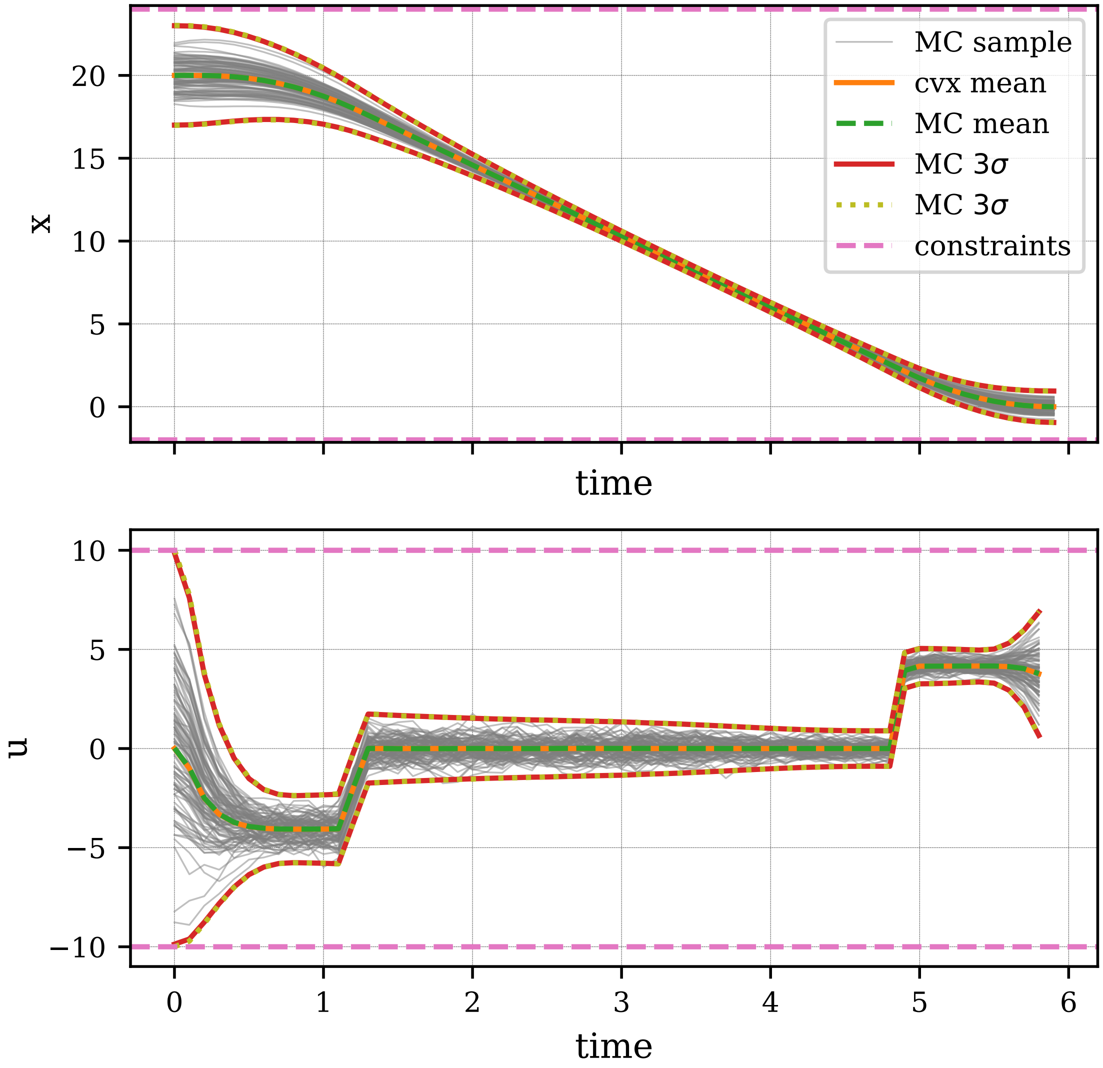


Figure 7.4: Covariance steering example for discrete double integrator

7.3.2 Rapid Uncertainty Propagation via Polynomial Chaos Expansion

Non-intrusive polynomial chaos expansion (PCE) offers a promising method for rapid uncertainty propagation for aerocapture and entry trajectories. In contrast to linear covariance analysis, which propagates Gaussian probabilities through linearized dynamics to predict only the first and second moment of the resulting distribution, PCE enables the creation of a surrogate model that can readily estimate higher-order moments as well as provide sensitivity analysis. In Ref. [138], PCE was applied to the aerocapture problem via compressed sensing and modeling atmospheric uncertainty using a Karhunen-Loève expansion (KLE); see Appendix F for a description of the method employed in that work. In Ref. [286], a two-stage approach is taken to capture the unique multi-model distributions that result for aerocapture when input uncertainties are high.

A limitation of this work to date, however, is that generally 15-20 stochastic dimensions are required to include a reasonably high-fidelity KLE plus dispersed initial state and vehicle aerodynamics. While greatly reduced from the original problem, this dimensionality is often still too high for PCE to significantly outperform Monte Carlo or other random sampling techniques such as Latin hypercube sampling. Thus, after the preliminary results presented in Ref. [138], the PCE approach was largely dropped from this research plan in favor of linear covariance analysis.

The variational autoencoder (VAE) results presented in Sec. 3.2.2, which were conltributed near the end of this degree plan, have potentially changed this situation. The VAE model enables comparable modeling accuracy with a stochastic dimensionality (i.e. number of latent variables) of only four. This may lower the total dimensionality of an entry or aerocapture trajectory simulation to the point that non-intrusive PCE can significantly outperform Monte Carlo. To be sure, this cannot be said of the very high-fidelity simulations written in simulation environments like POST2 or DSENDS and used by NASA JPL or NASA Langley on post-Phase A flight projects. However, the ability to rapidly propagate uncertainty for a low-to-medium-fidelity simulation is still valuable for early trade space exploration or onboard guidance and navigation.

In particular, an intriguing possibility would be to essentially develop a nonlinear parallel to the stochastic guidance framework described in the preceding section. Rather than using a KLE to model density uncertainty as Gaussian and propagating through linearized dynamics, a VAE density model and PCE uncertainty propagation could provide nonlinear, not-necessarily-Gaussian equivalents. In the linear Gaussian case, it is straightforward to use the Kalman update equations to incorporate new information, as shown in Sec. 3.6. The equivalent step of conditioning a VAE density model on noisy measurements onboard a spacecraft without requiring new training is something that would require dedicated investigation.

Bibliography

[1] J. R. Wertz, D. F. Everett, and J. J. Puschell, Space Mission Engineering: The New SMAD. Microcosm Press, 2011.

[2] J. Holt and T. Monk, “Propellant Mass Fraction Calculation Methodology for Launch Vehicles and Application to Ares Vehicles,” in AIAA SPACE 2009 Conference & Exposition.

[3] M. D. Rayman, P. Varghese, D. H. Lehman, and L. L. Livesay, “Results from the Deep Space 1 technology validation mission,” Acta Astronautica, vol. 47, no. 2, pp. 475–487, 2000. Space an Integral Part of the Information Age.

[4] B. Foing, G. Racca, A. Marini, E. Evrard, L. Stagnaro, M. Almeida, D. Koschny, D. Frew, J. Zender, J. Heather, M. Grande, J. Huovelin, H. Keller, A. Nathues, J. Josset, A. Malkki, W. Schmidt, G. Noci, R. Birkl, L. Iess, Z. Sodnik, and P. McManamon, “SMART-1 Mission To the Moon: Status, First Results and Goals,” Advances in Space Research, vol. 37, no. 1, pp. 6–13, 2006. The Moon and Near-Earth Objects.

[5] S. N. Williams and V. Coverstone-Carroll, “Benefits of Solar Electric Propulsion for the Next Generation of Planetary Exploration Missions,” The Journal of the Astronautical Sciences, vol. 45, pp. 143–159, June 1997.

[6] R. G. Jahn and E. Y. Choueiri, “Electric Propulsion,” in Encyclopedia of Physical Science and Technology (Third Edition) (R. A. Meyers, ed.), pp. 125–141, New York: Academic Press, third edition ed., 2003.

[7] D. O’Reilly, G. Herdrich, and D. F. Kavanagh, “Electric Propulsion Methods for Small Satellites: A Review,” Aerospace, vol. 8, no. 1, 2021.

[8] J. C. Smith and J. L. Bell, “2001 Mars Odyssey Aerobraking,” Journal of Spacecraft and Rockets, vol. 42, no. 3, pp. 406–415, 2005.

[9] S. Bougher, G. Keating, R. Zurek, J. Murphy, R. Haberle, J. Hollingsworth, and R. T. Clancy, “Mars Global Surveyor aerobraking: Atmospheric trends and model interpretation,” Advances in Space Research, vol. 23, no. 11, pp. 1887–1897, 1999.

[10] D. T. Lyons, R. Saunders, and D. G. Griffith, “The Magellan Venus Mapping Mission: Aerobraking Operations,” Acta Astronautica, vol. 35, no. 9, pp. 669–676, 1995. Challenges of Space for a Better World.

[11] R. W. Zurek and S. E. Smrekar, “An Overview of the Mars Reconnaissance Orbiter (MRO) Science Mission,” Journal of Geophysical Research: Planets, vol. 112, no. E5, 2007.

[12] D. A. Spencer and R. Tolson, “Aerobraking Cost and Risk Decisions,” Journal of Spacecraft and Rockets, vol. 44, no. 6, pp. 1285–1293, 2007.

[13] H. S. London, “Change of Satellite Orbit Plane by Aerodynamic Maneuvering,” Journal of the Aerospace Sciences, vol. 29, no. 3, pp. 323–332, 1962.

[14] G. D. Walberg, “A Survey of Aeroassisted Orbit Transfer,” Journal of Spacecraft and Rockets, vol. 22, no. 1, pp. 3–18, 1985.

[15] E. M. Repic, M. G. Boobar, and F. G. Chapel, “Aerobraking as a Potential Planetary Capture Mode,” Journal of Spacecraft and Rockets, vol. 5, no. 8, pp. 921–926, 1968.

[16] C. Heidrich, E. E. Roelke, S. W. Albert, and R. D. Braun, “Comparative Study of Lift- and Drag-Modulation Control Strategies for Aerocapture,” in 43rd Annual AAS Guidance, Navigation and Control Conference, Breckenridge, CO, pp. 1–16, February 2020.

[17] M. I. Cruz, “The Aerocapture Vehicle Mission Design Concept,” in Conference on Advanced Technology for Future Space Systems, AIAA, May 1979.

[18] J. L. Hall, A. N. Noca, and R. W. Bailey, “Cost-Benefit Analysis of the Aerocapture Mission Set,” Journal of Spacecraft and Rockets, vol. 42, no. 2, pp. 309–320, 2005.

[19] T. R. Spilker, M. Adler, N. Arora, P. M. Beauchamp, J. A. Cutts, M. M. Munk, R. W. Powell, R. D. Braun, and P. F. Wercinski, “Qualitative Assessment of Aerocapture and Applications to Future Missions,” Journal of Spacecraft and Rockets, vol. 56, pp. 536–545, Mar.-Apr. 2019.

[20] R. D. Braun and R. M. Manning, “Mars Exploration Entry, Descent, and Landing Challenges,” Journal of Spacecraft and Rockets, vol. 44, no. 2, pp. 310–323, 2007.

[21] G. W. Wells, J. M. Lafleur, A. Verges, K. Manyapu, J. A. Christian, C. Lewis, and R. D. Braun, “Entry, Descent and Landing Challenges of Human Mars Exploration,” in 29th AAS Guidance and Control Conference, no. AAS 06-072, 2006.

[22] G. Walberg, P. Siemers, R. Calloway, and J. Jones, “The Aeroassist Flight Experiment,” in Brighton International Astronautical Federation Congress, pp. 1–11, Oct. 1987.

[23] R. Powell, “Numerical Roll Reversal Predictor Corrector Aerocapture and Precision Landing Guidance Algorithms for the Mars Surveyor Program 2001 Missions,” in 23rd Atmospheric Flight Mechanics Conference, pp. 1–9, 2012.

[24] C. Cazaux, F. Naderi, C. Whetsel, D. Beaty, B. Gershman, R. Kornfeld, B. Mitcheltree, and B. Sackheim, “The NASA/CNES Mars sample return: A status report,” Acta Astronautica, vol. 54, pp. 601–617, Apr. 2004.

[25] S. Dutta, “White Paper: Aerocapture as an Enhancing Option for Ice Giants Missions,” tech. rep., NASA, 2020.

[26] A. Austin, "White Paper: Enabling and Enhancing Science Exploration Across the Solar System: Aerocapture Technology for SmallSat to Flagship Missions," tech. rep., NASA JPL/CalTech, 2020.

[27] B. A. Smith, L. A. Soderblom, R. Beebe, D. Bliss, J. M. Boyce, A. Brahic, G. A. Briggs, R. H. Brown, S. A. Collins, A. F. Cook, S. K. Croft, J. N. Cuzzi, G. E. Danielson, M. E. Davies, T. E. Dowling, D. Godfrey, C. J. Hansen, C. Harris, G. E. Hunt, A. P. Ingersoll, T. V. Johnson, R. J. Krauss, H. Masursky, D. Morrison, T. Owen, J. B. Plescia, J. B. Pollack, C. C. Porco, K. Rages, C. Sagan, E. M. Shoemaker, L. A. Sromovsky, C. Stoker, R. G. Strom, V. E. Suomi, S. P. Synnott, R. J. Terrile, P. Thomas, W. R. Thompson, and J. Veverka, "Voyager 2 in the Uranian System: Imaging Science Results," Science, vol. 233, no. 4759, pp. 43–64, 1986.

[28] B. A. Smith, L. A. Soderblom, D. Banfield, c. Barnet, A. T. Basilevsky, R. F. Beebe, K. Bollinger, J. M. Boyce, A. Brahic, G. A. Briggs, R. H. Brown, c. Chyba, s. A. Collins, T. Colvin, A. F. Cook, D. Crisp, S. K. Croft, D. Cruikshank, J. N. Cuzzi, G. E. Danielson, M. E. Davies, E. D. Jong, L. Dones, D. Godfrey, J. Goguen, I. Grenier, V. R. Haemmerle, H. Hammel, c. J. Hansen, c. P. Helfenstein, C. Howell, G. E. Hunt, A. P. Ingersoll, T. V. Johnson, J. Kargel, R. Kirk, D. I. Kuehn, S. Limaye, H. Masursky, A. McEwen, D. Morrison, T. Owen, W. Owen, J. B. Pollack, c. c. Porco, K. Rages, P. Rogers, D. Rudy, C. Sagan, J. Schwartz, E. M. Shoemaker, M. Showalter, B. Sicardy, D. Simonelli, J. Spencer, L. A. Sromovsky, C. Stoker, R. G. Strom, V. E. Suomi, S. P. Synott, R. J. Terrile, P. Thomas, W. R. Thompson, A. Verbiscer, and J. Veverka, "Voyager 2 at Neptune: Imaging Science Results," Science, vol. 246, no. 4936, pp. 1422–1449, 1989.

[29] Committee on the Planetary Science and Astrobiology Decadal Survey, Space Studies Board, Division on Engineering and Physical Sciences, and National Academies of Sciences, Engineering, and Medicine, Origins, Worlds, and Life: A Decadal Strategy for Planetary Science and Astrobiology 2023-2032. Washington, D.C.: National Academies Press, 2022. Pages: 26522.

[30] M. Hofstadter, A. Simon, K. Reh, J. Elliott, C. Niebur, and L. Colangeli, "Ice Giants Pre-Decadal Study Final Report," tech. rep., NASA, June 2017.

[31] S. J. Saikia, J. Millane, A. Mudek, A. Arora, P. Witsberger, E. Shibata, L. Podesta, Y. Lu, P. Edelman, J. Longuski, K. Hughes, A. Petropoulos, N. Arora, J. Cutts, J. Elliott, J. Sims, K. Reh, and T. Spilker, "Aerocapture Assessment at Uranus and Neptune for NASA's Ice Giant Studies," Tech. Rep. PU-AAC-2016-MC-0002, Purdue University, 2016.

[32] P. Agrawal, G. A. Allen, E. B. Sklyanskiy, H. H. Hwang, L. C. Huynh, K. McGuire, M. S. Marley, J. A. Garcia, J. F. Aliaga, and R. W. Moses, "Atmospheric entry studies for Uranus," in 2014 IEEE Aerospace Conference, (Big Sky, MT, USA), pp. 1–19, IEEE, Mar. 2014.

[33] T. Halt, A. Wieger, N. Boensch, A. Dolgopolov, P. Smith, and A. Hernandez, "Smallsats by the Numbers 2019," tech. rep., Bryce Space and Technology, 2019.

[34] A. Poghosyan and A. Golkar, "CubeSat Evolution: Analyzing CubeSat Capabilities for Conducting Science Missions," Progress in Aerospace Sciences, vol. 88, pp. 59–83, January 2017.

[35] C. D. Norton, S. Pellegrino, and M. Johnson, “Small Satellites: A Revolution in Space Science,” tech. rep., Keck Institute for Space Studies, California Institute of Technology, July 2014.

[36] C. R. Mercer, “Small Satellite Missions for Planetary Science,” in 33rd Annual AIAA/USU Conference on Small Satellites, pp. 1–6, 2019.

[37] S. W. Asmar and S. Matousek, Mars Cube One (MarCO): Shifting the Paradigm in Relay Deep Space Operation, ch. OCFE - Mission Ops Concept I, pp. 1–7. American Institute of Aeronautics and Astronautics, 2016.

[38] A. Austin, A. Nelessen, B. Strauss, J. Ravich, M. Jesick, E. Venkatapathy, R. Beck, P. Wercinski, M. Aftosmis, M. Wilder, G. Allen, R. Braun, M. Werner, and E. Roelke, “SmallSat Aerocapture to Enable a New Paradigm of Planetary Missions,” in 2019 IEEE Aerospace Conference, pp. 1–20, 2019.

[39] G. Falcone, J. W. Williams, and Z. R. Putnam, “Assessment of Aerocapture for Orbit Insertion of Small Satellites at Mars,” Journal of Spacecraft and Rockets, vol. 56, no. 6, pp. 1689–1703, 2019.

[40] P. E. Moseley, “The Apollo entry guidance: A review of the mathematical development and its operational characteristics.” TRW Note No. 69-FMT-791, December 1969.

[41] Z. R. Putnam, M. D. Neave, and G. H. Barton, “PredGuid Entry Guidance for Orion Return from Low Earth Orbit,” in 2010 IEEE Aerospace Conference, pp. 1–13, 2010.

[42] A. D. Steltzner, A. Miguel San Martin, T. P. Rivellini, A. Chen, and D. Kipp, “Mars Science Laboratory Entry, Descent, and Landing System Development Challenges,” Journal of Spacecraft and Rockets, vol. 51, no. 4, pp. 994–1003, 2014.

[43] A. Nelessen, C. Sackier, I. Clark, P. Brugarolas, G. Villar, A. Chen, A. Stehura, R. Otero, E. Stilley, D. Way, K. Edquist, S. Mohan, C. Giovingo, and M. Lefland, “Mars 2020 Entry, Descent, and Landing System Overview,” in 2019 IEEE Aerospace Conference, pp. 1–20, 2019.

[44] T. Percy, E. Bright, and A. Torres, Assessing the Relative Risk of Aerocapture Using Probabilistic Risk Assessment.

[45] F. J. Regan and S. M. Anandakrishnan, Dynamics of Atmospheric Re-Entry. AIAA, 1993.

[46] D. Xiu and J. Hesthaven, “High-Order Collocation Methods for Differential Equations with Random Inputs,” SIAM J. Sci. Comput., vol. 27, no. 3, pp. 1118–1139, 2005.

[47] F. Nobile, R. Tempone, and C. Webster, “A Sparse Grid Stochastic Collocation Method for Partial Differential Equations with Random Input Data,” SIAM J. Numer. Anal., vol. 46, no. 5, pp. 2309–2345, 2008.

[48] R. Ghanem and P. Spanos, Stochastic Finite Elements: A Spectral Approach. Dover, 2002.

[49] D. Xiu and G. Karniadakis, “The Wiener-Askey polynomial chaos for stochastic differential equations,” SIAM Journal on Scientific Computing, vol. 24, no. 2, pp. 619–644, 2002.

[50] C. J. Culbert, B. L. Ehlmann, and A. A. Fraemen, "Revolutionizing Access to the Martian Surface," final workshop report, W. M. Keck Institute for Space Studies, 2022.

[51] S. Curry, C. Edwards, D. Banfield, N. Barba, B. Ehlmann, S. Hubbard, L. Santos, F. Tan, and R. Zurek, "Low-Cost Science Mission Concepts for Mars Exploration Final Report," Tech. Rep. JPL D-109705, NASA Jet Propulsion Laboratory, 2022. `https://www.hou.usra.edu/meetings/lowcostmars2022/program/Low-Cost-Mars-Final-Report.pdf`.

[52] S. W. Albert, R. D. Braun, and H. Schaub, "AeroDrop: Prospects and Challenges for Co-Delivery of Probe and Orbiter via Aerocapture," in AAS/AIAA Astrodynamics Specialist Conference, Lake Tahoe, CA, pp. 1–19, 2020.

[53] A. Arora, S. J. Saikia, T. R. Spilker, R. A. Dillman, A. Bowes, S. J. Horan, D. H. Atkinson, and K. M. Sayanagi, "Multiprobe Mission Architecture Options for a Uranus Flagship Mission," Journal of Spacecraft and Rockets, vol. 58, pp. 697–707, May 2021.

[54] R. Deshmukh, System Analysis of a Numerical Predictor-Corrector Aerocapture Guidance Architecture. PhD thesis, Purdue University, 2021.

[55] H. S. Wright, D. Y. Oh, C. H. Westhelle, J. L. Fisher, R. E. Dyke, S. Aerospace, K. T. Edquist, J. L. Brown, H. L. Justh, and M. M. Munk, "Mars Aerocapture Systems Study," Tech. Rep. TM-2006-214522, NASA, 2006.

[56] M. K. Lockwood, B. R. Starr, J. W. Paulson Jr, D. A. Kontinos, Y. Chen, B. Laub, J. Olejniczak, M. J. Wright, N. Takashima, C. G. Justus, et al., "Systems Analysis for a Venus Aerocapture Mission," Tech. Rep. TM-2006-214291, NASA, 2006.

[57] M. K. Lockwood, E. M. Queen, D. W. Way, R. W. Powell, K. Edquist, B. W. Starr, B. R. Hollis, E. V. Zoby, G. A. Hrinda, R. W. Bailey, J. L. Hall, T. R. Spilker, M. Noca, R. Haw, C. G. Justus, A. Duvall, D. L. Johnson, J. Masciarelli, N. Takashima, K. Sutton, J. Olejniczak, B. Laub, M. J. Wright, D. Prabhu, E. Corporation, R. E. Dyke, and S. Aerospace, "Aerocapture Systems Analysis for a Titan Mission," Tech. Rep. TM-2006-214273, NASA, 2006.

[58] M. K. Lockwood, K. T. Edquist, B. R. Starr, B. R. Hollis, G. A. Hrinda, R. W. Bailey, J. L. Hall, T. R. Spilker, M. A. Noca, N. O'Kongo, et al., "Aerocapture Systems Analysis for a Neptune Mission," Tech. Rep. TM-2006-214300, NASA, 2006.

[59] A. Cassell, "White Paper: Deployable Entry Vehicles for Future Science and Exploration Missions," tech. rep., NASA Ames Research Center, 2020.

[60] A. Cassell, P. Wercinski, B. Smith, B. Yount, O. Nishioka, and C. Kruger, "ADEPT Sounding Rocket One Flight Test Overview," in AIAA Aviation 2019 Forum.

[61] A. Olds, R. Beck, D. M. Bose, J. White, K. T. Edquist, B. R. Hollis, M. Lindell, F. M. Cheatwood, V. Gsell, and E. L. Bowden, "IRVE-3 Post-Flight Reconstruction," in AIAA Aerodynamic Decelerator Systems (ADS) Conference.

[62] E. Venkatapathy, D. Ellerby, P. Gage, D. Prabhu, M. Gasch, C. Kazemba, C. Kellerman, S. Langston, B. Libben, M. Mahzari, F. Milos, A. Murphy, O. Nishioka, K. Peterson, C. Poteet, S. Splinter, M. Stackpoole, J. Williams, and Z. Young, "Entry System Technology Readiness for Ice-Giant Probe Missions," Space Science Reviews, vol. 216, p. 22, Mar. 2020.

[63] E. Roelke, Guidance Algorithm and Vehicle Architecture Enhancements for Improved Discrete-Event, Drag-Modulated Aerocapture. PhD thesis, University of Colorado Boulder, 2021.

[64] A. Alunni, S. N. D'Souza, B. Yount, W. Okolo, B. Nikaido, B. Margolis, B. J. Johnson, K. Hibbard, J. Barton, G. Lopez, and Z. B. Hays, Pterodactyl: Trade Study for an Integrated Control System Design of a Mechanically Deployed Entry Vehicle.

[65] P. Lu, C. J. Cerimele, M. A. Tigges, and D. A. Matz, "Optimal Aerocapture Guidance," Journal of Guidance, Control, and Dynamics, vol. 38, no. 4, pp. 553–565, 2015.

[66] C. R. Heidrich, Optimal Covariance Control on Singular Manifolds with Application to Aerospace Mission Design. PhD thesis, University of Colorado Boulder, 2021.

[67] J. Ridderhof, Applied Stochastic Optimal Control for Spacecraft Guidance. PhD thesis, Georgia Institute of Technology, 2021.

[68] A. P. Girija, A Systems Framework and Analysis Tool for Rapid Conceptual Design of Aerocapture Missions. PhD thesis, Purdue University, 2021.

[69] S. Bhaskaran, Autonomous Navigation for Deep Space Missions.

[70] C. R. Heidrich, S. Dutta, and R. D. Braun, "Modern Aerocapture Guidance to Enable Reduce-Lift Vehicles at Neptune," in AAS/AIAA Space Flight Mechanics Meeting, no. AAS 19-221, 2019.

[71] A. P. Girija, S. J. Saikia, J. M. Longuski, S. Bhaskaran, M. S. Smith, and J. A. Cutts, "Feasibility and Performance Analysis of Neptune Aerocapture Using Heritage Blunt-Body Aeroshells," Journal of Spacecraft and Rockets, vol. 57, pp. 1186–1203, Nov. 2020.

[72] T. Ro and E. Queen, "Study of Martian Aerocapture Terminal Point Guidance," in 23rd Atmospheric Flight Mechanics Conference, (Boston, MA), 1998.

[73] S. Rousseau, E. Perot, C. Graves, J. Masciarelli, and E. Queen, "Aerocapture Guidance Algorithm Comparison Campaign," in AIAA/AAS Astrodynamics Specialist Conference and Exhibit, pp. 1–11, 2012.

[74] J. Masciarelli and E. Queen, "Guidance Algorithms for Aerocapture at Titan," in 39th AIAA/ASME/SAE/ASEE Joint Propulsion Conference and Exhibit.

[75] G. F. Mendeck and L. Craig McGrew, "Entry Guidance Design and Postflight Performance for 2011 Mars Science Laboratory Mission," Journal of Spacecraft and Rockets, vol. 51, no. 4, pp. 1094–1105, 2014.

[76] J. Masciarelli, "Aerocapture Guidance Algorithm Development and Testing," in NASA Science Technology Conference, 2007. `https://esto.nasa.gov/conferences/nstc2007/papers/Masciarelli_Jim_D8P3_NSTC-07-0030.pdf`.

[77] C. CERIMELE and J. GAMBLE, "A Simplified Guidance Algorithm for Lifting Aeroassist Orbital Transfer Vehicles," in 23rd Aerospace Sciences Meeting.

[78] N. X. Vinh, A. Busemann, and R. D. Culp, Hypersonic and Planetary Entry Flight Mechanics, ch. 2, pp. 26–27. The University of Michigan Press, 1980.

[79] J. D. Gamble, C. J. Cerimele, T. E. Moore, and J. Higgins, "Atmospheric guidance concepts for an aeroassist flight experiment," Journal of the Astronautical Sciences, vol. 36, pp. 45–71, June 1988.

[80] R. D. Braun and R. W. Powell, "Predictor-Corrector Guidance Algorithm for Use in High-Energy Aerobraking System Studies," Journal of Guidance, Control, and Dynamics, vol. 15, no. 3, pp. 672–678, 1992.

[81] R. W. Powell and R. D. Braun, "Six-Degree-Of-Freedom Guidance and Control Analysis of Mars Aerocapture," Journal of Guidance, Control, and Dynamics, vol. 16, no. 6, pp. 1038–1044, 1993.

[82] R. Powell, "Numerical Roll Reversal Predictor Corrector Aerocapture and Precision Landing Guidance Algorithms for the Mars Surveyor Program 2001 Missions," in 23rd Atmospheric Flight Mechanics Conference.

[83] D. A. Matz, P. Lu, G. F. Mendeck, and R. R. Sostaric, Application of a Fully Numerical Guidance to Mars Aerocapture.

[84] E. Perot and S. Rousseau, "Importance of an On-board Estimation of the Density Scale Height for Various Aerocapture Guidance Algorithms," in AIAA/AAS Astrodynamics Specialist Conference and Exhibit, 2002.

[85] Z. R. Putnam and R. D. Braun, "Drag-Modulation Flight-Control System Options for Planetary Aerocapture," Journal of Spacecraft and Rockets, vol. 51, no. 1, pp. 139–150, 2014.

[86] J. Wagner, A. Wilhite, D. Stanley, and R. Powell, "An Adaptive Real Time Atmospheric Prediction Algorithm for Entry Vehicles," in 3rd AIAA Atmospheric Space Environments Conference, 2011.

[87] D. Amato, S. Hume, B. Grace, and J. McMahon, "Robustifying Mars Descent Guidance Through Neural Networks," in AAS Guidance, Navigation, and Control Conference, no. AAS 20-073, 2020.

[88] E. Roelke, P. Hattis, and R. Braun, "Improved Atmospheric Estimation For Aerocapture Guidance," in AAS/AIAA Astrodynamics Specialist Conference, pp. 1–16, 08 2019.

[89] X. Jiang and S. Li, "Mars Entry Trajectory Planning Using Robust Optimization and Uncertainty Quantification," Acta Astronautica, vol. 161, pp. 249–261, Aug. 2019.

[90] X. Jiang, Correction: Uncertainty Quantification for Mars Atmospheric Entry Using Polynomial Chaos and Spectral Decomposition.

[91] A. Halder and R. Bhattacharya, Beyond Monte Carlo: A Computational Framework for Uncertainty Propagation in Planetary Entry, Descent and Landing.

[92] A. Halder and R. Bhattacharya, "Dispersion Analysis in Hypersonic Flight During Planetary Entry Using Stochastic Liouville Equation," Journal of Guidance, Control, and Dynamics, vol. 34, no. 2, pp. 459–474, 2011.

[93] C. R. Heidrich and R. D. Braun, “Aerocapture Trajectory Design in Uncertain Entry Environments,” in AIAA Scitech 2020 Forum, 2020.

[94] J. W. McMahon, D. Amato, D. Kuettel, and M. J. Grace, Stochastic Predictor-Corrector Guidance.

[95] D. Amato, S. Hume, B. Grace, and E. Roelke, “Mars EDL and Aerocapture Guidance Under Dynamic Uncertainty,” in AIAA/AAS Astrodynamics Specialist Conference, no. AAS 20-595, 2020.

[96] J. Ridderhof and P. Tsiotras, “Chance-Constrained Covariance Steering in a Gaussian Random Field via Successive Convex Programming,” Journal of Guidance, Control, and Dynamics, vol. 45, pp. 599–610, Apr. 2022.

[97] K. Oguri and J. W. McMahon, “Robust Spacecraft Guidance Around Small Bodies Under Uncertainty: Stochastic Optimal Control Approach,” Journal of Guidance, Control, and Dynamics, vol. 44, pp. 1295–1313, July 2021.

[98] B. Benedikter, A. Zavoli, Z. Wang, S. Pizzurro, and E. Cavallini, “Covariance Control for Stochastic Low-Thrust Trajectory Optimization,” in AIAA SCITECH 2022 Forum, (San Diego, CA & Virtual), American Institute of Aeronautics and Astronautics, Jan. 2022.

[99] W. J. O’Neil, The Three Galileos: The Man, the Spacecraft, the Telescope, vol. 220, ch. The Galileo Spacecraft Architecture, pp. 75–94. Springer, Dordrecht, 1997.

[100] B. Kazeminejad, D. H. Atkinson, M. Pérez-Ayúcar, J.-P. Lebreton, and C. Sollazzo, “Huygens’ Entry and Descent Through Titan’s Atmosphere—Methodology and Results of the Trajectory Reconstruction,” Planetary and Space Science, vol. 55, no. 13, pp. 1845 – 1876, 2007.

[101] A. J. Ball, J. R. C. Garry, R. D. Lorenz, and V. V. Kerzhanovich, Planetary Landers and Entry Probes. Cambridge University Press, 2009.

[102] A.-M. Harri, V. Linkin, J. Polkko, M. Marov, J.-P. Pommereau, A. Lipatov, T. Siili, K. Manuilov, V. Lebedev, A. Lehto, R. Pellinen, R. Pirjola, T. Carpentier, C. Malique, V. Makarov, L. Khloustova, L. Esposito, J. Maki, G. Lawrence, and V. Lystsev, “Meteorological Observations on Martian Surface: Met-Packages of Mars-96 Small Stations and Penetrators,” Planetary and Space Science, vol. 46, no. 6, pp. 779–793, 1998.

[103] M. W. Lo, B. G. Williams, W. E. Bollman, D. Han, Y. Hahn, J. L. Bell, E. A. Hirst, R. A. Corwin, P. E. Hong, K. C. Howell, B. Barden, and R. Wilson, “Genesis Mission Design,” The Journal of the Astronautical Sciences, vol. 49, pp. 169–184, Mar. 2001.

[104] P. N. Desai, D. T. Lyons, J. Tooley, and J. Kangas, “Entry, Descent, and Landing Operations Analysis for the Stardust Entry Capsule,” Journal of Spacecraft and Rockets, vol. 45, no. 6, pp. 1262–1268, 2008.

[105] M. Yoshikawa, J. Kawaguchi, A. Fujiwara, and A. Tsuchiyama, “Chapter 6 - The Hayabusa mission,” in Sample Return Missions (A. Longobardo, ed.), pp. 123–146, Elsevier, 2021.

[106] Y. Tsuda, M. Yoshikawa, M. Abe, H. Minamino, and S. Nakazawa, "System design of the Hayabusa 2—Asteroid sample return mission to 1999 JU3," Acta Astronautica, vol. 91, pp. 356–362, 2013.

[107] B. Williams, P. Antreasian, E. Carranza, C. Jackman, J. Leonard, D. Nelson, B. Page, D. Stanbridge, D. Wibben, K. Williams, M. Moreau, K. Berry, K. Getzandanner, A. Liounis, A. Mashiku, D. Highsmith, B. Sutter, and D. S. Lauretta, "OSIRIS-REx Flight Dynamics and Navigation Design," Space Science Reviews, vol. 214, p. 69, Apr. 2018.

[108] B. K. Muirhead, A. K. Nicholas, J. Umland, O. Sutherland, and S. Vijendran, "Mars Sample Return Campaign Concept Status," Acta Astronautica, vol. 176, pp. 131–138, 2020.

[109] L. Colin, "The Pioneer Venus Program," Journal of Geophysical Research: Space Physics, vol. 85, no. A13, pp. 7575–7598, 1980.

[110] S. D. Dorfman and C. M. Meredith, "The Pioneer Venus spacecraft program," Acta Astronautica, vol. 7, no. 6, pp. 773–795, 1980.

[111] S. J. Saikia, B. Rogers, J. M. Longuski, H. Saranathan, and M. J. Grant, "Strategies for Mars Network Science Missions via Innovative Aerocapture and EDL Architectures," in 10th International Planetary Probe Workshop (IPPW-10), pp. 1–13.

[112] A. L. Genova, D. Morrison-Fogel, A. D. Perez, and A. M. Cassell, "Trajectory Design for a Spacecraft Capable of Deploying Probes to the Martian Surface En Route to Low Mars Orbit," in 2022 AAS/AIAA Astrodynamics Specialist Conference, pp. 1–20, 2022.

[113] J. Sullivan, S. Grimberg, and S. D'Amico, "Comprehensive Survey and Assessment of Spacecraft Relative Motion Dynamics Models," Journal of Guidance, Control, and Dynamics, vol. 40, no. 8, pp. 1837–1859, 2017.

[114] S. W. Albert and H. Schaub, "Relative Motion in the Velocity Frame for Atmospheric Entry Trajectories," Journal of Spacecraft and Rockets, vol. 60, no. 5, pp. 1614–1624, 2023.

[115] N. Vinh, W. Johnson, and J. Longuski, "Mars Aerocapture using Bank Modulation," in Astrodynamics Specialist Conference, p. 4424, 2000.

[116] R. G. Deshmukh, D. A. Spencer, and S. Dutta, "Investigation of Direct Force Control for Aerocapture at Neptune," Acta Astronautica, vol. 175, pp. 375–386, 2020.

[117] N. X. Vinh, J. R. Johannesen, K. D. Mease, and J. M. Hanson, "Explicit Guidance of Drag-Modulated Aeroassisted Transfer Between Elliptical Orbits," Journal of Guidance, Control, and Dynamics, vol. 9, no. 3, pp. 274–280, 1986.

[118] R. R. Rohrschneider and R. D. Braun, "Survey of Ballute Technology for Aerocapture," Journal of Spacecraft and Rockets, vol. 44, no. 1, pp. 10–23, 2007.

[119] E. Roelke, J. W. McMahon, R. D. Braun, and P. D. Hattis, "Multi-Event Jettison Guidance Approaches for Drag-Modulation Aerocapture," Journal of Spacecraft and Rockets, vol. 59, no. 1, pp. 190–202, 2022.

[120] R. Powell and R. Braun, "Six-Degree-of-Freedom Guidance and Control Analysis of Mars Aerocapture," Journal of Guidance, Control, and Dynamics, vol. 16, no. 6, pp. 1038–1044, 2012.

[121] G. Falcone, J. W. Williams, and Z. R. Putnam, "Assessment of Aerocapture for Orbit Insertion of Small Satellites at Mars," Journal of Spacecraft and Rockets, vol. 56, no. 6, pp. 1689–1703, 2019.

[122] W. Johnson and D. Lyons, "Titan Ballute Aerocapture Using a Perturbed TitanGRAM Model," in AIAA Atmospheric Flight Mechanics Conference and Exhibit, pp. 1–12, 2004.

[123] R. D. Braun, D. A. Spencer, P. H. Kallemeyn, and R. M. Vaughan, "Mars Pathfinder Atmospheric Entry Navigation Operations," Journal of Spacecraft and Rockets, vol. 36, no. 3, pp. 348–356, 1999.

[124] M. S. Werner and R. D. Braun, "Mission Design and Performance Analysis of a Smallsat Aerocapture Flight Test," Journal of Spacecraft and Rockets, vol. 56, no. 6, pp. 1704–1713, 2019.

[125] D. Gulick, J. Lewis, K. Miller, D. Lyons, J. Stein, and R. Wilmoth, "Trailing Ballute Aerocapture: System Definition," in 39th AIAA/ASME/SAE/ASEE Joint Propulsion Conference and Exhibit, pp. 1–12, 2003.

[126] L. Cheng, F. Jiang, Z. Wang, and J. Li, "Multiconstrained Real-Time Entry Guidance Using Deep Neural Networks," IEEE Transactions on Aerospace and Electronic Systems, vol. 57, no. 1, pp. 325–340, 2021.

[127] H. Wang and T. A. Elgohary, "A Simple and Accurate Apollo-Trained Neural Network Controller for Mars Atmospheric Entry," International Journal of Aerospace Engineering, vol. 2020, pp. 1–15, Sept. 2020. Publisher: Hindawi.

[128] Y. Shi and Z. Wang, "Onboard Generation of Optimal Trajectories for Hypersonic Vehicles Using Deep Learning," Journal of Spacecraft and Rockets, vol. 58, no. 2, pp. 400–414, 2021.

[129] G. Falcone and Z. R. Putnam, "Autonomous Decision-Making for Aerobraking via Parallel Randomized Deep Reinforcement Learning," IEEE Transactions on Aerospace and Electronic Systems, pp. 1–18, 2022.

[130] M. Werner, B. Woollard, A. Tadanki, S. R. Pujari, R. D. Braun, R. Lock, A. Nelessen, and R. Woolley, "Development of an Earth Smallsat Flight Test to Demonstrate Viability of Mars Aerocapture," in 55th AIAA Aerospace Sciences Meeting, pp. 1–16, 2017.

[131] A. Cassell, B. Smith, P. Wercinski, S. Ghassemieh, K. Hibbard, A. Nelessen, and J. Cutts, "ADEPT, A Mechanically Deployable Re-Entry Vehicle System, Enabling Interplanetary CubeSat and Small Satellite Missions," in Proceedings of the Small Satellite Conference, Session 3: Advanced Technologies I, pp. 1–8, 2018.

[132] P. Wegner, J. Ganley, and J. Maly, "EELV Secondary Payload Adapter (ESPA): Providing Increased Access to Space," in 2001 IEEE Aerospace Conference Proceedings (Cat. No.01TH8542), vol. 5, pp. 2563–2568 vol.5, 2001.

[133] J. M. Cameron, A. Jain, P. D. Burkhart, E. S. Bailey, B. Balaram, E. Bonfiglio, M. Ivanov, J. Benito, E. Sklyanskiy, and W. Strauss, "DSENDS: Multi-mission Flight Dynamics Simulator for NASA Missions," in AIAA SPACE 2016, pp. 1–18, 2016.

[134] F. W. Leslie and C. G. Justus, “The NASA Marshall Space Flight Center Earth Global Reference Atmospheric Model - 2010 Version,” Tech. Rep. NASA/TM-2011-216467, NASA, 2011.

[135] E. R. Burnett and H. Schaub, “Approximating Orbits in a Rotating Gravity Field With Oblateness and Ellipticity Perturbations,” Celestial Mechanics and Dynamical Astronomy, vol. 134, p. 5, Jan. 2022.

[136] D. Way, R. Powell, J. Masciarelli, B. Starr, and K. Edquist, “Aerocapture Simulation and Performance for the Titan Explorer Mission,” in 39th AIAA/ASME/SAE/ASEE Joint Propulsion Conference and Exhibit, 2003.

[137] W. D. Strauss, A. Austin, A. Nelessen, M. Lobbia, A. Schutte, J. Ravich, L. Luthman, B. Kennedy, D. Landau, and M. Jadusingh, “Aerocapture Trajectories for Earth Orbit Technology Demonstration and Orbiter Science Missions at Venus, Earth, Mars, and Neptune,” in AAS/AIAA Space Flight Mechanics Meeting, no. AAS 21-229, pp. 1–20, 2021.

[138] S. W. Albert, A. Doostan, and H. Schaub, “Finite-Dimensional Density Representation for Aerocapture Uncertainty Quantification,” in AIAA Scitech 2021 Forum, no. AIAA 2021-0932, (Virtual event), pp. 1–17, Jan. 2021.

[139] E. R. Burnett, S. W. Albert, and H. Schaub, “A New Guidance Technique for Discrete-Event Drag Modulation for Aerocapture Missions,” in AAS Guidance, Navigation, and Control Conference, no. AAS 22-102, pp. 1–15, 2022.

[140] S. Rousseau, “An Energy Controller Aerocapture Guidance Algorithm for the Mars Sample Return Orbiter,” pp. 67–85, 2001.

[141] A. Rizvi, K. F. Ortega, and Y. He, “Developing Lunar Flashlight and Near-Earth Asteroid Scout Flight Software Concurrently using Open-Source F Prime Flight Software Framework,” in Small Satellite Conference, 2022.

[142] T. Imken, J. Castillo-Rogez, Y. He, J. Baker, and A. Marinan, “CubeSat Flight System Development for Enabling Deep Space Science,” in 2017 IEEE Aerospace Conference, pp. 1–14, 2017.

[143] P. Lu, “Entry Guidance: A Unified Method,” Journal of Guidance, Control, and Dynamics, vol. 37, no. 3, pp. 713–728, 2014.

[144] M. J. Grace, E. R. Burnett, and J. W. McMahon, “Quasi-Initial Conditions as a State Representation for Aerocapture,” in AIAA SciTech Forum, pp. 1–16, 2022.

[145] S. N. D’Souza and N. Sarigul-Klijn, “Survey of Planetary Entry Guidance Algorithms,” Progress in Aerospace Sciences, vol. 68, pp. 64–74, July 2014.

[146] J. Ridderhof, S. W. Albert, P. Tsiotras, and H. Schaub, “Linear Covariance Analysis of Entry and Aerocapture Trajectories in an Uncertain Atmosphere,” in AIAA SCITECH 2022 Forum, (San Diego, CA), 2022.

[147] D. Woffinden, S. Robinson, J. Williams, and Z. R. Putnam, “Linear Covariance Analysis Techniques to Generate Navigation and Sensor Requirements for the Safe and Precise Landing Integrated Capabilities Evolution (SPLICE) Project,” in AIAA Scitech Forum, (San Diego, CA), p. 0662, Jan. 7 – 11 2019.

[148] A. Doostan and G. Iaccarino, "A Least-Squares Approximation of Partial Differential Equations With High-Dimensional Random Inputs," Journal of Computational Physics, vol. 228, no. 12, pp. 4332–4345, 2009.

[149] A. Doostan and H. Owhadi, "A non-adapted sparse approximation of PDEs with stochastic inputs," Journal of Computational Physics, vol. 230, no. 8, pp. 3015–3034, 2011.

[150] X. Jiang, "Uncertainty Quantification for Mars Atmospheric Entry using Polynomial Chaos and Spectral Decomposition," in 2018 AIAA Guidance, Navigation, and Control Conference, 2018.

[151] H. L. Justh, "Mars Global Reference Atmospheric Model 2010 Version: Users Guide," Tech. Rep. NASA/TM—2014–217499, NASA, 2014.

[152] J.-F. Le Gall, Brownian Motion, Martingales, and Stochastic Calculus. Graduate Texts in Mathematics, Springer International Publishing, 2016.

[153] C. E. Rasmussen and C. K. I. Williams, Gaussian Processes for Machine Learning. Adaptive Computation and Machine Learning, The MIT Press, 2005.

[154] M. Loeve, "Elementary Probability Theory," in Probability theory i, pp. 1–52, Springer, 1977.

[155] D. D. Kosambi, "Statistics in Function Space," Journal of the Indian Mathematical Society, vol. 7, pp. 76–88, 1943.

[156] H. Bourlard and Y. Kamp, "Auto-Association by Multilayer Perceptrons and Singular Value Decomposition," Biological Cybernetics, vol. 59, pp. 291–294, Sept. 1988.

[157] D. P. Kingma and M. Welling, "Auto-Encoding Variational Bayes," 2013.

[158] S. Kullback and R. A. Leibler, "On Information and Sufficiency," The Annals of Mathematical Statistics, vol. 22, no. 1, pp. 79 – 86, 1951.

[159] D. P. Kingma and M. Welling, "An Introduction to Variational Autoencoders," Foundations and Trends in Machine Learning, vol. 12, no. 4, pp. 307–392, 2019.

[160] L. Pinheiro Cinelli, M. Araújo Marins, E. A. Barros da Silva, and S. Lima Netto, Variational Autoencoder, pp. 111–149. Cham: Springer International Publishing, 2021.

[161] D. A. Vallado, Fundamentals of Astrodynamics and Applications, ch. Appendix D, pp. 1041–1042. Microcosm Press, 4th ed., 2013.

[162] C. G. Justus and R. D. Braun, "Atmospheric Environments for Entry, Descent and Landing (EDL)," in 5th International Planetary Probes Workshop and Short Course, pp. 1–37, 2007. `https://ntrs.nasa.gov/citations/20070032693`.

[163] M. G. Trainer, M. H. Wong, T. H. McConnochie, H. B. Franz, S. K. Atreya, P. G. Conrad, F. Lefèvre, P. R. Mahaffy, C. A. Malespin, H. L. Manning, J. Martín-Torres, G. M. Martínez, C. P. McKay, R. Navarro-González, A. Vicente-Retortillo, C. R. Webster, and M.-P. Zorzano, "Seasonal Variations in Atmospheric Composition as Measured in Gale Crater, Mars," Journal of Geophysical Research: Planets, vol. 124, no. 11, pp. 3000–3024, 2019.

[164] N. Barba, T. Komarek, R. Woolley, L. Giersch, V. Stamenković, M. Gallagher, and C. D. Edwards, "Mars Small Spacecraft Studies: Overview," in 2019 IEEE Aerospace Conference, pp. 1–10, 2019.

[165] D. Way, R. Powell, A. Chen, A. Steltzner, M. San Martin, P. Burkhart, and G. Mendeck, "Mars Science Laboratory: Entry, Descent, and Landing System Performance," in IEEE Aerospace Conference Proceedings, pp. 1–19, March 2006.

[166] D. Hendrycks and K. Gimpel, "Bridging Nonlinearities and Stochastic Regularizers with Gaussian Error Linear Units," CoRR, vol. abs/1606.08415, 2016.

[167] N. Cheng, O. A. Malik, S. Becker, and A. Doostan, "Bi-fidelity Variational Auto-encoder for Uncertainty Quantification," arXiv preprint arXiv:2305.16530, 2023.

[168] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "PyTorch: An Imperative Style, High-Performance Deep Learning Library," 2019.

[169] S. W. Albert, A. Doostan, and H. Schaub, "Onboard Density Modeling for Planetary Entry via Karhunen-Loève Expansion," in 2023 IEEE Aerospace Conference, pp. 1–12, 2023.

[170] J. Ridderhof and P. Tsiotras, "Chance-constrained covariance steering in a Gaussian random field via successive convex programming," 2021.

[171] R. G. Ghanem and A. Doostan, "On the Construction and Analysis of Stochastic Models: Characterization and Propagation of the Errors Associated With Limited Data," Journal of Computational Physics, vol. 217, no. 1, pp. 63–81, 2006.

[172] M. A. Mischna, G. Villar, D. M. Kass, S. Dutta, S. Rafkin, D. Tyler, J. Barnes, B. Cantor, S. R. Lewis, D. Hinson, J. Pla-García, A. Kleinböhl, and C. Karlgaard, "Pre- and Post-Entry, Descent and Landing Assessment of the Martian Atmosphere for the Mars 2020 Rover," The Planetary Science Journal, vol. 3, p. 147, jun 2022.

[173] P. Withers, "Mars Global Surveyor and Mars Odyssey Accelerometer Observations of the Martian Upper Atmosphere During Aerobraking," Geophysical Research Letters, vol. 33, no. 2, 2006.

[174] A. Razavi, A. van den Oord, and O. Vinyals, "Generating Diverse High-Fidelity Images with VQ-VAE-2," in Advances in Neural Information Processing Systems (H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, eds.), vol. 32, Curran Associates, Inc., 2019.

[175] P. Dhariwal, H. Jun, C. Payne, J. W. Kim, A. Radford, and I. Sutskever, "Jukebox: A Generative Model for Music," 2020.

[176] D. Simon, Optimal State Estimation. John Wiley & Sons, Ltd, 2006.

[177] H. P. Flath, L. C. Wilcox, V. Akçelik, J. Hill, B. van Bloemen Waanders, and O. Ghattas, "Fast Algorithms for Bayesian Uncertainty Quantification in Large-Scale Linear Inverse Problems Based on Low-Rank Partial Hessian Approximations," SIAM Journal on Scientific Computing, vol. 33, no. 1, pp. 407–432, 2011.

[178] M. Brand, "Fast Low-Rank Modifications of the Thin Singular Value Decomposition," Linear Algebra and its Applications, vol. 415, no. 1, pp. 20–30, 2006. Special Issue on Large Scale Linear and Nonlinear Eigenvalue Problems.

[179] H. Khalil, Nonlinear Systems, ch. 3, pp. 99–103. Prentice Hall, 2002.

[180] G. L. Carman, D. G. Ives, and D. K. Geller, "Apollo-Derived Precision Lander Guidance," in 23rd Atmospheric Flight Mechanics Conference, no. 98-4570, (Boston, MA), 1998.

[181] T. J. Martin-Mur, G. L. Kruizinga, P. D. Burkhart, F. Abilleira, M. C. Wong, and J. A. Kangas, "Mars Science Laboratory Interplanetary Navigation," Journal of Spacecraft and Rockets, vol. 51, no. 4, pp. 1014–1028, 2014.

[182] A. E. Bryson and Y.-C. Ho, Applied Optimal Control: Optimization, Estimation and Control, ch. 2, p. 48. Ginn and Company, 1975.

[183] A. M. Cassell, B. P. Smith, P. F. Wercinski, S. M. Ghassemieh, K. E. Hibbard, A. P. Nelessen, and J. A. Cutts, "ADEPT, A Mechanically Deployable Re-Entry Vehicle System, Enabling Interplanetary CubeSat and Small Satellite Missions," in 32nd Annual AIAA/USU Conference on Small Satellites, pp. 1–8, 2018.

[184] P. Gage, M. Mahzari, K. Peterson, D. Ellerby, and E. Venkatapathy, "Technology Readiness Assessment for HEEET TPS," in 16th International Planetary Probe Workshop, pp. 1–19, July 2019.

[185] C. R. Heidrich and R. D. Braun, "Aerocapture Trajectory Design in Uncertain Entry Environments," in AIAA Scitech 2020 Forum, pp. 1–21, 2020.

[186] A. Halder and R. Bhattacharya, "Dispersion Analysis in Hypersonic Flight During Planetary Entry Using Stochastic Liouville Equation," Journal of Guidance, Control, and Dynamics, vol. 34, no. 2, pp. 459–474, 2011.

[187] J. Ridderhof and P. Tsiotras, "Stochastic Atmosphere Modeling for Risk Adverse Aerocapture Guidance," in 2020 IEEE Aerospace Conference, pp. 1–7, 2020.

[188] B. P. Smith, C. L. Tanner, M. Mahzari, I. G. Clark, R. D. Braun, and F. M. Cheatwood, "A Historical Review of Inflatable Aerodynamic Decelerator Technology Development," in 2010 IEEE Aerospace Conference, pp. 1–18, 2010.

[189] S. Hughes, F. Cheatwood, R. Dillman, A. Calomino, H. Wright, J. DelCorso, and A. Calomino, "Hypersonic Inflatable Aerodynamic Decelerator (HIAD) Technology Development Overview," in 21st AIAA Aerodynamic Decelerator Systems Technology Conference and Seminar, pp. 1–24, 2011.

[190] L. Colin, "The Pioneer Venus Program," Journal of Geophysical Research: Space Physics, vol. 85, no. A13, pp. 7575–7598, 1980.

[191] A. T. Klesh, J. Baker, and J. Krajewski, "MarCO: Flight Review and Lessons Learned," in Small Satellite Conference, pp. 1–6, 2019.

[192] J. Dormand and P. Prince, “A Family of Embedded Runge-Kutta Formulae,” Journal of Computational and Applied Mathematics, vol. 6, no. 1, pp. 19–26, 1980.

[193] C. A. Graves and J. C. Harpold, “Apollo Experience Report - Mission Planning for Apollo Entry,” Technical Note D-6725, NASA, 1972.

[194] L. W. McNamara and R. D. Braun, “Definition of the Design Trajectory and Entry Flight Corridor for the NASA Orion Exploration Mission 1 Entry Trajectory Using an Integrated Approach and Optimization,” in 2014 American Astronautical Society Guidance and Control Conference, pp. 14–094, 2014.

[195] A. P. Girija, S. J. Saikia, J. M. Longuski, S. Bhaskaran, M. S. Smith, and J. A. Cutts, “Feasibility and Performance Analysis of Neptune Aerocapture Using Heritage Blunt-Body Aeroshells,” Journal of Spacecraft and Rockets, vol. 57, no. 6, pp. 1186–1203, 2020.

[196] G. F. Lindal, J. R. Lyons, D. N. Sweetnam, V. R. Eshleman, D. P. Hinson, and G. L. Tyler, “The Atmosphere of Neptune: Results of Radio Occultation Measurements With the Voyager 2 Spacecraft,” Geophysical Research Letters, vol. 17, no. 10, pp. 1733–1736, 1990.

[197] F. Leslie and C. Justus, “The NASA Marshall Space Flight Center Earth Global Reference Atmospheric Model—2010 Version,” Tech. Rep. NASA/TM—2011–216467, NASA, 2011.

[198] H. L. Justh, “Venus Global Reference Atmospheric Model,” tech. rep., NASA, 2017. gRAM Virtual Workshop Presentation `https://ntrs.nasa.gov/citations/20170012290`.

[199] H. L. Justh and J. Hoffman, “Titan Global Reference Atmospheric Model (Titan-GRAM): User Guide,” Tech. Rep. NASA/TM–20205006805, NASA, 2020.

[200] H. L. Justh and J. Hoffman, “Neptune Global Reference Atmospheric Model (Neptune-GRAM): User Guide,” Tech. Rep. NASA/TM–20205001193, NASA, 2020.

[201] K. Sutton and R. A. Graves, “A General Stagnation-Point Convective-Heating Equation for Arbitrary Gas Mixtures,” Tech. Rep. TR R-376, NASA, 1971.

[202] J. D. Anderson, Hypersonic and High-Temperature Gas Dynamcis, ch. Chapter 3: Local Surface Inclination Methods and Chapter 6: Viscous Flow: Basic Aspects, Boundary Layer Results, and Aerodynamic Heating, p. 55 and 347. AIAA, 2006.

[203] R. D. Braun, R. A. Mitcheltree, and F. M. Cheatwood, “Mars Microprobe Entry-to-Impact Analysis,” Journal of Spacecraft and Rockets, vol. 36, no. 3, pp. 412–420, 1999.

[204] J. Goodwin and P. Wegner, “Evolved Expendable Launch Vehicle Secondary Payload Adapter - Helping Technology Get to Space,” in AIAA Space 2001 Conference and Exposition, pp. 1–7, 2001.

[205] N. J. Barba, T. A. Komarek, V. Stamekovic, L. R. Giersh, R. C. Woolley, and C. D. Edwards, “Small Spacecraft Capabilities for Mars Exploration,” 2019. Ninth International Conference on Mars, `https://www.hou.usra.edu/meetings/ninthmars2019/eposter/6341.pdf`.

[206] S. W. Albert and R. D. Braun, “Conceptual Development of AeroDrop: Aerocapture and Direct Entry for Two Spacecraft on a Common Approach Trajectory,” in AIAA Scitech 2020 Forum, pp. 1–20, AIAA, 2020.

[207] P. Brugarolas, "Guidance, Navigation and Control for the Entry, Descent, and Landing of the Mars 2020 Mission," in 40th Annual AAS Guidance and Control Conference, Breckenridge, Colorado, pp. 1–12, February 2017.

[208] R. P. Brent, Algorithms for Minimization Without Derivatives. Courier Corporation, 2013.

[209] J. Callahan and C. Dreier, "Mars In Retrograde: A Pathway to Restoring NASA's Robotic Mars Exploration Program," tech. rep., The Planetary Society, 2017.

[210] A. Elfving, A. L. Elias, M. T. King, G. Lee, J. Pellicciotti, P. C. Theisinger, D. W. Thompson, M. Wadhwa, T. Young, and M. Zuber, "Mars Sample Return (MSR) Program: Final Report of the Independent Review Board (IRB)," tech. rep., NASA, 2020.

[211] D. Banfield, "Mars Science Goals, Objectives, Investigations, and Priorities: 2020 Version," tech. rep., Mars Exploration Program Analysis Group, 2020.

[212] S. Diniega, N. Barba, L. Giersch, B. Jackson, A. Soto, S. Rafkin, C. Swann, R. Sullivan, D. Banfield, and L. Fenton, "Optimally-sized Mission Concepts for Focused In-situ Studies of Planetary Surface-atmosphere Interactions." Oral Presentation, Low-Cost Science Mission Concepts for Mars Workshop, 2022.

[213] S. C. Stähler, M. P. Panning, D. Antonangeli, W. B. Banerdt, D. Banfield, M. Banks, S. Ceylan, C. Charalambous, J. Clinton, I. Daubar, B. Fernando, D. Giardini, M. Grott, A. Horleston, K. Hurst, T. Kawamura, D. Kim, M. Knapmeyer, R. Lorenz, L. Margerin, A. Marusiak, S. Menina, A. Mittelholz, N. Murdoch, C. Perrin, W. T. Pike, C. Schmelzbach, N. Schmerr, M. Schimmel, A. Spiga, A. Stott, J. Taylor, and R. Weber, "A Cerberus Fossae Seismic Network." Oral Presentation, Low-Cost Science Mission Concepts for Mars Workshop, 2022.

[214] S. Rafkin, "The Atmospheric Characterization for Exploration and Science (ACES) instrument suite for Mars," in 2015 IEEE Aerospace Conference, pp. 1–6, 2015.

[215] C. Nunn, W. T. Pike, I. M. Standley, S. B. Calcutt, S. Kedar, and M. P. Panning, "Standing on Apollo's Shoulders: a Microseismometer for the Moon," The Planetary Science Journal, vol. 2, no. 1, p. 36, 2021.

[216] G. E. N. Scoon and G. P. Whitcomb, "MARSNET: A Precursor to the Surface Exploration of Mars," Acta Astronautica, vol. 29, no. 10, pp. 823–831, 1993. World Space Congress.

[217] S. Squyres, "The Mars Environmental Survey (MESUR) Mission," Advances in Space Research, vol. 15, no. 4, pp. 179–188, 1995.

[218] B. Banderdt, A. F. Chicarro, M. Coradini, C. Federico, R. Greeley, M. Hechler, J. M. Knudsen, C. Leovy, P. Lognonne, L. Lowry, D. McCleese, C. McKay, R. Pellinen, R. Phillips, G. E. N. Scoon, T. Spohn, S. Squyres, F. Taylor, and H. Wanke, "INTERMARSNET: Phase-A Study Report," Tech. Rep. ESA Publication D/SCI(96)2, European Space Agency, April 1996.

[219] S. Smrekar, D. Catling, R. Lorenz, J. Magalhães, J. Moersch, P. Morgan, B. Murray, M. Presley, A. Yen, A. Zent, and D. Blaney, "Deep Space 2: The Mars Microprobe Mission," Journal of Geophysical Research: Planets, vol. 104, no. E11, pp. 27013–27030, 1999.

[220] A.-M. Harri, O. Marsal, P. Lognonne, G. Leppelmeier, T. Spohn, K.-H. Glassmeier, F. Angrilli, W. Banerdt, J. Barriot, J.-L. Bertaux, J. Berthelier, S. Calcutt, J. Cerisier, D. Crisp, V. Dehant, D. Giardini, R. Jaumann, Y. Langevin, M. Menvielle, G. Musmann, J. Pommereau, S. Di Pippo, D. Guerrier, K. Kumpulainen, S. Larsen, A. Mocquet, J. Polkko, J. Runavot, W. Schumacher, T. Siili, J. Simola, and J. Tillman, "Network Science Landers for Mars," Advances in Space Research, vol. 23, no. 11, pp. 1915–1924, 1999.

[221] A.-M. Harri, K. Pichkadze, L. Zeleny, L. Vazquez, W. Schmidt, S. Alexashkin, O. Korablev, H. Guerrero, J. Heilimo, M. Uspensky, V. Finchenko, V. Linkin, I. Arruego, M. Genzer, A. Lipatov, J. Polkko, M. Paton, H. Savijärvi, H. Haukka, T. Siili, V. Khovanskov, B. Ostesko, A. Poroshin, M. Diaz-Michelena, T. Siikonen, M. Palin, V. Vorontsov, A. Polyakov, F. Valero, O. Kemppinen, J. Leinonen, and P. Romero, "The MetNet Vehicle: A Lander to Deploy Environmental Stations for Local and Global Investigations of Mars," Geoscientific Instrumentation, Methods and Data Systems, vol. 6, no. 1, pp. 103–124, 2017.

[222] C. Edwards and L. Matthies, "New Capabilities for Accessing the Martian Surface," tech. rep., Jet Propulsion Laboratory, California Institute of Technology, 2021.

[223] J. B. Garvin, S. A. Getty, G. N. Arney, N. M. Johnson, E. Kohler, K. O. Schwer, M. Sekerak, A. Bartels, R. S. Saylor, V. E. Elliott, C. S. Goodloe, M. B. Garrison, V. Cottini, N. Izenberg, R. Lorenz, C. A. Malespin, M. Ravine, C. R. Webster, D. H. Atkinson, S. Aslam, S. Atreya, B. J. Bos, W. B. Brinckerhoff, B. Campbell, D. Crisp, J. R. Filiberto, F. Forget, M. Gilmore, N. Gorius, D. Grinspoon, A. E. Hofmann, S. R. Kane, W. Kiefer, S. Lebonnois, P. R. Mahaffy, A. Pavlov, M. Trainer, K. J. Zahnle, and M. Zolotov, "Revealing the Mysteries of Venus: The DAVINCI Mission," The Planetary Science Journal, vol. 3, p. 117, May 2022.

[224] H. F. York, "Multiple-Warhead Missiles," Scientific American, vol. 229, no. 5, pp. 18–27, 1973.

[225] S. W. Albert, H. Schaub, and R. D. Braun, "Flight Mechanics Feasibility Assessment for Co-Delivery of Direct-Entry Probe and Aerocapture Orbiter," AIAA Journal of Spacecraft and Rockets, vol. 59, no. 1, pp. 19–32, 2022.

[226] L. A. D'Amario, "Mars Exploration Rovers Navigation Results," The Journal of the Astronautical Sciences, vol. 54, pp. 129–173, June 2006.

[227] D. Way, "On The Use of a Range Trigger for the Mars Science Laboratory Entry, Descent, and Landing," in 2011 Aerospace Conference, pp. 1–8, 2011.

[228] W. H. Willcockson, "Mars Pathfinder Heatshield Design and Flight Experience," Journal of Spacecraft and Rockets, vol. 36, no. 3, pp. 374–379, 1999.

[229] Z. R. Putnam and R. D. Braun, "Precision Landing at Mars Using Discrete-Event Drag Modulation," Journal of Spacecraft and Rockets, vol. 51, no. 1, pp. 128–138, 2014.

[230] D. M. Fawley and Z. R. Putnam, "Performance Assessment of Discrete-Event Drag Modulation for Mars Entry with Real-Time Guidance," Journal of Spacecraft and Rockets, vol. 58, no. 4, pp. 1071–1083, 2021.

[231] S. W. Albert and H. Schaub, "Co-Delivery of Multiple Small Probes to the Martian Surface," in AIAA SciTech 2022 Forum, pp. 1–14, 2022.

[232] D. Giardini, P. Lognonné, W. B. Banerdt, W. T. Pike, U. Christensen, S. Ceylan, J. F. Clinton, M. van Driel, S. C. Stähler, M. Böse, R. F. Garcia, A. Khan, M. Panning, C. Perrin, D. Banfield, E. Beucler, C. Charalambous, F. Euchner, A. Horleston, A. Jacob, T. Kawamura, S. Kedar, G. Mainsant, J.-R. Scholz, S. E. Smrekar, A. Spiga, C. Agard, D. Antonangeli, S. Barkaoui, E. Barrett, P. Combes, V. Conejero, I. Daubar, M. Drilleau, C. Ferrier, T. Gabsi, T. Gudkova, K. Hurst, F. Karakostas, S. King, M. Knapmeyer, B. Knapmeyer-Endrun, R. Llorca-Cejudo, A. Lucas, L. Luno, L. Margerin, J. B. McClean, D. Mimoun, N. Murdoch, F. Nimmo, M. Nonon, C. Pardo, A. Rivoldini, J. A. R. Manfredi, H. Samuel, M. Schimmel, A. E. Stott, E. Stutzmann, N. Teanby, T. Warren, R. C. Weber, M. Wieczorek, and C. Yana, "The Seismicity of Mars," Nature Geoscience, vol. 13, pp. 205–212, Mar 2020.

[233] S. Kedar, M. P. Panning, S. E. Smrekar, S. C. Stähler, S. D. King, M. P. Golombek, M. Manga, B. R. Julian, B. Shiro, C. Perrin, J. A. Power, C. Michaut, S. Ceylan, D. Giardini, P. H. Lognonné, and W. B. Banerdt, "Analyzing Low Frequency Seismic Events at Cerberus Fossae as Long Period Volcanic Quakes," Journal of Geophysical Research: Planets, vol. 126, no. 4, p. e2020JE006518, 2021.

[234] P. Lognonné, W. B. Banerdt, D. Giardini, W. T. Pike, U. Christensen, P. Laudet, S. de Raucourt, P. Zweifel, S. Calcutt, M. Bierwirth, K. J. Hurst, F. Ijpelaan, J. W. Umland, R. Llorca-Cejudo, S. A. Larson, R. F. Garcia, S. Kedar, B. Knapmeyer-Endrun, D. Mimoun, A. Mocquet, M. P. Panning, R. C. Weber, A. Sylvestre-Baron, G. Pont, N. Verdier, L. Kerjean, L. J. Facto, V. Gharakanian, J. E. Feldman, T. L. Hoffman, D. B. Klein, K. Klein, N. P. Onufer, J. Paredes-Garcia, M. P. Petkov, J. R. Willis, S. E. Smrekar, M. Drilleau, T. Gabsi, T. Nebut, O. Robert, S. Tillier, C. Moreau, M. Parise, G. Aveni, S. Ben Charef, Y. Bennour, T. Camus, P. A. Dandonneau, C. Desfoux, B. Lecomte, O. Pot, P. Revuz, D. Mance, J. tenPierick, N. E. Bowles, C. Charalambous, A. K. Delahunty, J. Hurley, R. Irshad, H. Liu, A. G. Mukherjee, I. M. Standley, A. E. Stott, J. Temple, T. Warren, M. Eberhardt, A. Kramer, W. Kühne, E.-P. Miettinen, M. Monecke, C. Aicardi, M. André, J. Baroukh, A. Borrien, A. Bouisset, P. Boutte, K. Brethomé, C. Brysbaert, T. Carlier, M. Deleuze, J. M. Desmarres, D. Dilhan, C. Doucet, D. Faye, N. Faye-Refalo, R. Gonzalez, C. Imbert, C. Larigauderie, E. Locatelli, L. Luno, J.-R. Meyer, F. Mialhe, J. M. Mouret, M. Nonon, Y. Pahn, A. Paillet, P. Pasquier, G. Perez, R. Perez, L. Perrin, B. Pouilloux, A. Rosak, I. Savin de Larclause, J. Sicre, M. Sodki, N. Toulemont, B. Vella, C. Yana, F. Alibay, O. M. Avalos, M. A. Balzer, P. Bhandari, E. Blanco, B. D. Bone, J. C. Bousman, P. Bruneau, F. J. Calef, R. J. Calvet, S. A. D'Agostino, G. de los Santos, R. G. Deen, R. W. Denise, J. Ervin, N. W. Ferraro, H. E. Gengl, F. Grinblat, D. Hernandez, M. Hetzel, M. E. Johnson, L. Khachikyan, J. Y. Lin, S. M. Madzunkov, S. L. Marshall, I. G. Mikellides, E. A. Miller, W. Raff, J. E. Singer, C. M. Sunday, J. F. Villalvazo, M. C. Wallace, D. Banfield, J. A. Rodriguez-Manfredi, C. T. Russell, A. Trebi-Ollennu, J. N. Maki, E. Beucler, M. Böse, C. Bonjour, J. L. Berenguer, S. Ceylan, J. Clinton, V. Conejero, I. Daubar, V. Dehant, P. Delage, F. Euchner, I. Estève, L. Fayon, L. Ferraioli, C. L. Johnson, J. Gagnepain-Beyneix, M. Golombek, A. Khan, T. Kawamura, B. Kenda, P. Labrot, N. Murdoch, C. Pardo, C. Perrin, L. Pou, A. Sauron, D. Savoie, S. Stähler, E. Stutzmann, N. A. Teanby, J. Tromp, M. van Driel, M. Wieczorek, R. Widmer-Schnidrig, and J. Wookey, "SEIS: Insight's Seismic Experiment for Internal Structure of Mars," Space Science Reviews, vol. 215, p. 12, Jan 2019.

[235] T. Hopf, S. Kumar, W. Karl, and W. Pike, "Shock Protection of Penetrator-Based Instrumentation via a Sublimation Approach," Advances in Space Research, vol. 45, no. 3, pp. 460–467,

2010.

[236] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors, "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," Nature Methods, vol. 17, pp. 261–272, 2020.

[237] R. P. Brent, Algorithms for Minimization without Derivatives, ch. 3–4, pp. 19–61. Prentice-Hall, 1973.

[238] J. Nocedal and S. J. Wright, Numerical Optimization, ch. 6, pp. 136–143. Springer, 2006.

[239] H. Schaub and J. L. Junkins, Analytical Mechanics of Space Systems, pp. 536–538, 775–780, 873–876. AIAA, fourth ed., 2018.

[240] E. R. Burnett, Novel Dynamics and Control Formulations for Multi-Spacecraft Formation Flying, Rendezvous, and Proximity Operations. PhD thesis, University of Colorado Boulder, 2021.

[241] G. W. Hill, "Researches in the Lunar Theory," American Journal of Mathematics, vol. 1, no. 1, pp. 5–26, 1878.

[242] W. H. Clohessy and R. S. Wiltshire, "Terminal Guidance System for Satellite Rendezvous," Journal of the Aerospace Sciences, vol. 27, no. 9, pp. 653–658, 1960.

[243] J. Tschauner and P. Hempel, "Rendezvous zu einem in Elliptischer Bahn Umlaufenden Ziel," Acta Astronautica, vol. 11, no. 2, pp. 104–109, 1965.

[244] T. E. Carter, "State Transition Matrices for Terminal Rendezvous Studies: Brief Survey and New Example," Journal of Guidance, Control, and Dynamics, vol. 21, no. 1, pp. 148–155, 1998.

[245] Z. Dang, "New State Transition Matrix for Relative Motion on an Arbitrary Keplerian Orbit," Journal of Guidance, Control, and Dynamics, vol. 40, no. 11, pp. 2917–2927, 2017.

[246] R. G. Reynolds, "Direct Solution of the Keplerian State Transition Matrix," Journal of Guidance, Control, and Dynamics, vol. 45, no. 6, pp. 1162–1165, 2022.

[247] Z. Dang and H. Zhang, "Linearized relative motion equations through orbital element differences for general Keplerian orbits," Astrodynamics, vol. 2, pp. 201–215, Sept. 2018.

[248] M. Willis, K. T. Alfriend, and S. D'Amico, "Second-Order Solution for Relative Motion on Eccentric Orbits in Curvilinear Coordinates," in AAS/AIAA Astrodynamics Specialist Conference, 2019.

[249] R. G. Melton, "Relative Motion Between Hyperbolic Trajectories – A Technical Footnote," in 73rd International Astronautical Congress, Paris, France, 2022.

[250] C. Foster, J. Mason, V. Vittaldev, L. Leung, V. Beukelaers, L. Stepan, and R. Zimmerman, “Constellation Phasing with Differential Drag on Planet Labs Satellites,” Journal of Spacecraft and Rockets, vol. 55, no. 2, pp. 473–483, 2018.

[251] G. Falcone and Z. R. Putnam, “Energy Depletion Guidance for Aerobraking Atmospheric Passes,” Journal of Guidance, Control, and Dynamics, vol. 45, no. 4, pp. 651–668, 2022.

[252] D. Bell, R. Cesarone, T. Ely, C. Edwards, and S. Townes, “Mars Network: A Mars Orbiting Communications and Navigation Satellite Constellation,” in 2000 IEEE Aerospace Conference. Proceedings (Cat. No.00TH8484), vol. 7, pp. 75–88 vol.7, 2000.

[253] H. J. Allen and A. J. Eggers, “A Study of the Motion and Aerodynamic Heating of Ballistic Missiles Entering the Earth’s Atmosphere at High Supersonic Speeds,” Tech. Rep. 1381, NACA, 1958.

[254] H. Schaub, “Relative Orbit Geometry Through Classical Orbit Element Differences,” AIAA Journal of Guidance, Control, and Dynamics, vol. 27, pp. 839–848, Sept.–Oct. 2004.

[255] D. Stelzl, E. K. Pfeiffer, H. G. Hemme, P. Lindenmaier, A. Riemer, A. A. Cubillo, T. Spröwitz, P. Seefeldt, S. Meyer, M. Zander, K. D. Bunte, A. Miller, S. Weikert, S. Meier, M. Koch, S. Lender, C. Vogt, R. Ernst, M. Schmid, T. Cardone, and G. Smet, “ADEO: the European commercial passive de-orbit subsystem family enabling space debris mitigation,” CEAS Space Journal, vol. 13, pp. 591–598, Oct. 2021.

[256] J. D. Anderson, Hypersonic and High Temperature Gas Dynamics. AIAA, 2000.

[257] H. J. Allen and A. J. Eggers, “A Study of the Motion and Aerodynamics Heating of Missiles Entering the Earth’s Atmosphere at High Supersonic Speeds,” Tech. Rep. RM A53D28, NASA, 1953.

[258] S. J. Citron and T. C. Meir, “An Analytic Solution for Entry Into Planetary Atmospheres,” AIAA Journal, vol. 3, no. 3, pp. 470–475, 1965.

[259] Z. R. Putnam and R. D. Braun, “Extension and Enhancement of the Allen–Eggers Analytical Ballistic Entry Trajectory Solution,” Journal of Guidance, Control, and Dynamics, vol. 38, pp. 414–430, Mar. 2015.

[260] B. Fegley Jr., Properties and Composition of the Terrestrial Oceans and of the Atmospheres of the Earth and Other Planets, pp. 320–345. American Geophysical Union (AGU), 1995.

[261] P. N. Desai and G. D. Qualls, “Stardust Entry Reconstruction,” Journal of Spacecraft and Rockets, vol. 47, no. 5, pp. 736–740, 2010.

[262] D. A. Kontinos and M. J. Wright, “Introduction: Atmospheric Entry of the Stardust Sample Return Capsule,” Journal of Spacecraft and Rockets, vol. 47, no. 6, pp. 865–867, 2010.

[263] H. H. King, “Ballistic Missile Re-Entry Dispersion,” Journal of Spacecraft and Rockets, vol. 17, no. 3, pp. 240–247, 1980.

[264] A. Simon, F. Nimmo, and R. C. Anderson, "Uranus Orbiter and Probe: Journey to an Ice Giant System. Planetary Mission Concept Study for the 2023-2032 Decadal Survey," tech. rep., NASA, 2021. `https://drive.google.com/drive/folders/1Qlun6EF0v472eOMXXokHxa6B2tBLNkdv`.

[265] A. P. Girija, "A Flagship-class Uranus Orbiter and Probe mission concept using aerocapture," Acta Astronautica, vol. 202, pp. 104–118, 2023.

[266] J. Yong and X. Y. Zhou, Stochastic Controls. New York, NY: Springer New York, 1999.

[267] R. Bellman, Dynamic programming. Princeton, NJ: Princeton Univ. Pr, 1984.

[268] H. J. Kappen, "An Introduction to Stochastic Control Theory, Path Integrals and Reinforcement Learning," in AIP Conference Proceedings, vol. 887, (Granada (Spain)), pp. 149–181, AIP, 2007. ISSN: 0094243X.

[269] A. Mesbah, "Stochastic Model Predictive Control: An Overview and Perspectives for Future Research," IEEE Control Systems, vol. 36, pp. 30–44, Dec. 2016.

[270] D. H. Mayne and D. Q. Jacobson, Differential Dynamic Programming. New York: American Elsevier Pub. Co., 1970.

[271] E. Theodorou, Y. Tassa, and E. Todorov, "Stochastic Differential Dynamic Programming," in Proceedings of the 2010 American Control Conference, (Baltimore, MD), pp. 1125–1132, IEEE, June 2010.

[272] M. Morari and J. H. Lee, "Model Predictive Control: Past, Present and Future," Computers and Chemical Engineering, pp. 667–682, 1999.

[273] D. Q. Mayne, "Model predictive control: Recent developments and future promise," Automatica, vol. 50, pp. 2967–2986, Dec. 2014.

[274] A. Hotz and R. E. Skelton, "Covariance Control Theory," International Journal of Control, vol. 46, no. 1, pp. 13–32, 1987.

[275] Y. Chen, T. T. Georgiou, and M. Pavon, "Optimal Steering of a Linear Stochastic System to a Final Probability Distribution, Part I," IEEE Transactions on Automatic Control, vol. 61, pp. 1158–1169, May 2016.

[276] Y. Chen, T. T. Georgiou, and M. Pavon, "Optimal Steering of a Linear Stochastic System to a Final Probability Distribution, Part II," IEEE Transactions on Automatic Control, vol. 61, pp. 1170–1180, May 2016.

[277] E. Bakolas, "Optimal Covariance Control for Stochastic Linear Systems Subject to Integral Quadratic State Constraints," in 2016 American Control Conference (ACC), (Boston, MA, USA), pp. 7231–7236, IEEE, July 2016.

[278] E. Bakolas, "Optimal Covariance Control for Discrete-Time Stochastic Linear Systems Subject to Constraints," in 2016 IEEE 55th Conference on Decision and Control (CDC), (Las Vegas, NV), pp. 1153–1158, IEEE, Dec. 2016.

[279] K. Okamoto and P. Tsiotras, "Optimal Stochastic Vehicle Path Planning Using Covariance Steering," IEEE Robotics and Automation Letters, vol. 4, pp. 2276–2281, July 2019.

[280] J. Ridderhof, J. Pilipovsky, and P. Tsiotras, “Chance-Constrained Covariance Control for Low-Thrust Minimum-Fuel Trajectory Optimization,” in AAS/AIAA Astrodynamics Specialist Conference, 2020.

[281] J. Ridderhof and P. Tsiotras, “Minimum-Fuel Closed-Loop Powered Descent Guidance with Stochastically Derived Throttle Margins,” Journal of Guidance, Control, and Dynamics, vol. 44, pp. 537–547, Mar. 2021.

[282] G. Rapakoulias and P. Tsiotras, “Comment on “Convex Approach to Covariance Control with Application to Stochastic Low-Thrust Trajectory Optimization”,” Journal of Guidance, Control, and Dynamics, vol. 46, no. 5, pp. 1023–1024, 2023.

[283] B. Benedikter, A. Zavoli, Z. Wang, S. Pizzurro, and E. Cavallini, “Reply by the Authors to G. Rapakoulias and P. Tsiotras,” Journal of Guidance, Control, and Dynamics, vol. 46, no. 5, pp. 1025–1025, 2023.

[284] S. Diamond and S. Boyd, “CVXPY: A Python-Embedded Modeling Language for Convex Optimization,” Journal of Machine Learning Research, vol. 17, no. 83, pp. 1–5, 2016.

[285] A. Agrawal, R. Verschueren, S. Diamond, and S. Boyd, “A rewriting system for convex optimization problems,” Journal of Control and Decision, vol. 5, no. 1, pp. 42–60, 2018.

[286] M. J. Grace and J. W. McMahon, Two-Stage Polynomial Chaos Expansion: An Extension of Uncertainty Quantification Techniques For Multi-Modal Distributions in Aerocapture.

[287] J. Ridderhof, J. Pilipovsky, and P. Tsiotras, “Chance-Constrained Covariance Control for Low-Thrust Minimum-Fuel Trajectory Optimization,” in Advances in the Astronautical Sciences, Volume 175: Proceedings of the AAS/AIAA Astrodynamics Specialist Conference, pp. 4979–4999, 2020.

[288] G. Rapakoulias and P. Tsiotras, “Discrete-time Optimal Covariance Steering via Semidefinite Programming,” in IEEE Conference on Decision and Control (preprint), 2023.

[289] F. Liu, G. Rapakoulias, and P. Tsiotras, “Optimal Covariance Steering for Discrete-Time Linear Stochastic Systems,” 2023.

[290] E. Bakolas, “Finite-Horizon Covariance Control for Discrete-Time Stochastic Linear Systems Subject to Input Constraints,” Automatica, vol. 91, pp. 61–68, 2018.

[291] K. Okamoto, M. Goldshtein, and P. Tsiotras, “Optimal Covariance Control for Stochastic Systems Under Chance Constraints,” IEEE Control Systems Letters, vol. 2, no. 2, pp. 266–271, 2018.

[292] P. J. Goulart, E. C. Kerrigan, and J. M. Maciejowski, “Optimization Over State Feedback Policies for Robust Control With Constraints,” Automatica, vol. 42, pp. 523–533, Apr. 2006.

[293] F. Oldewurtel, C. N. Jones, and M. Morari, “A tractable approximation of chance constrained stochastic MPC based on affine disturbance feedback,” in 2008 47th IEEE Conference on Decision and Control, (Cancun, Mexico), pp. 4731–4736, IEEE, 2008.

[294] P. E. Kloeden and E. Platen, Stochastic Differential Equations, pp. 103–160. Berlin, Heidelberg: Springer Berlin Heidelberg, 1992.

[295] V. Renganathan, J. Pilipovsky, and P. Tsiotras, "Distributionally Robust Covariance Steering with Optimal Risk Allocation," 2022.

[296] M. Hadigol and A. Doostan, "Least squares polynomial chaos expansion: A review of sampling strategies," Computer Methods in Applied Mechanics and Engineering, vol. 332, pp. 382 – 407, 2018.

[297] A. Doostan and H. Owhadi, "A Non-Adapted Sparse Approximation of PDEs With Stochastic Inputs," Journal of Computational Physics, vol. 230, pp. 3015–3034, 2011.

[298] G. Blatman and B. Sudret, "Adaptive Sparse Polynomial Chaos Expansion Based on Least Angle Regression," Journal of Computational Physics, vol. 230, pp. 2345–2367, 2011.

[299] J. Hampton and A. Doostan, "Compressive Sampling of Polynomial Chaos Expansions: Convergence Analysis and Sampling Strategies," Journal of Computational Physics, vol. 280, pp. 363–386, 2015.

[300] J. Peng, J. Hampton, and A. Doostan, "A Weighted $\ell 1$-Minimization Approach for Sparse Polynomial Chaos Expansions," Journal of Computational Physics, vol. 267, pp. 92–111, 2014.

[301] A. Doll, M. Ravasi, and D. Relyea, "SPGL1: A Solver for Large-Scale One-Norm Regularized Least Squares," December 2020. `https://github.com/drrelyea/spgl1`.

[302] E. van den Berg and M. P. Friedlander, "SPGL1: A Solver for Large-Scale Sparse Reconstruction," December 2019. `https://friedlander.io/spgl1`.

[303] E. van den Berg and M. P. Friedlander, "Probing the Pareto Frontier for Basis Pursuit Solutions," SIAM Journal on Scientific Computing, vol. 31, no. 2, pp. 890–912, 2008.

[304] B. Sudret, "Global Sensitivity Analysis Using Polynomial Chaos Expansions," Reliability Engineering & System Safety, vol. 93, no. 7, pp. 964 – 979, 2008. Bayesian Networks in Dependability.

[305] B. Sudret and C. Mai, "Computing Derivative-Based Global Sensitivity Measures Using Polynomial Chaos Expansions," Reliability Engineering & System Safety, vol. 134, pp. 241 – 250, 2015.

Appendix A

Equations of Motion Derivation and Useful Coordinate Relations

A.1 Assumptions

This appendix derives equations of motion in the spherical coordinates common in EDL practice, and provides some useful coordinate relationships. Make the following assumptions about the dynamics of the problem:

- Gravity is modeled by point-mass plus the J_2 oblateness perturbation; other gravitational terms are neglected.
- Lift and drag are modeled assuming constant aerodynamic coefficients. This equates to assuming constant angle-of-attack and zero side-slip angle, and that the vehicle remains in hypersonic continuum flow. This is a reasonable approximation for many hypersonic entry problems. However, this assumption is not required to use these EOMs: if higher fidelity is desired, a table of aerodynamics coefficients (usually vs. Mach number and angle-of-attack) can be interpolated at each integration time step and then used in the same EOMs developed here.
- The vehicle has constant mass and zero thrust.
- There is negligible wind and no atmospheric rotation relative to the surface of the central body.
- Attitude dynamics are not modeled in these 3 DOF EOMs, but the orientation of the lift

vector is prescribed through the bank angle σ (defined later). This angle does not need to be constant, but the EOMs do not describe how it evolves. Since attitude dynamics are not modeled, aerodynamic moments are ignored.

A.2 State Representation

The state representation used in this derivation is defined as $\boldsymbol{x} = [r, \theta, \phi, U, \gamma, \psi]$ where r is radius (distance from the center of the planet to the vehicle), θ is planetocentric longitude, and ϕ is planetocentric latitude. The latter three components relate to the air-relate velocity of the vehicle, such that U is airspeed, γ is flight-path angle, and ψ is heading angle. These two angles are described in more detail below, but γ is the angle between the air-relative velocity vector and the local horizontal plane, and ψ is the angle between the horizontal projection of the air-relative velocity vector and a due-North vector in that same plane (i.e., a due-East entry has $\psi = 90°$). Note that, since the atmosphere is assumed to rotate with the surface of the planet, air-relative velocity, planet-relative velocity, and surface-relative velocity all refer to the same quantity $\boldsymbol{U}$.

A.3 Frames

Define the following reference frames:

A.3.1 Planet-Fixed Frame $\mathcal{I} : \{\hat{\boldsymbol{\imath}}_1, \hat{\boldsymbol{\imath}}_2, \hat{\boldsymbol{\imath}}_3\}$

- $\hat{\boldsymbol{\imath}}_1$ is in the equatorial plane and through the prime meridian, such that it has zero latitude and zero longitude
- $\hat{\boldsymbol{\imath}}_3$ is through the North pole, i.e. orthogonal to the equatorial plane
- $\hat{\boldsymbol{\imath}}_2$ completes the set

A.3.2 Inertial Frame $\mathcal{N} : \{\hat{\boldsymbol{n}}_1, \hat{\boldsymbol{n}}_2, \hat{\boldsymbol{n}}_3\}$

Define an inertial frame that is equal to the planet-fixed frame at initial time t_0. The planet-fixed frame then rotates relative to the inertial frame about their shared third axis at a constant rate ω_p:

$$\boldsymbol{\omega_{I/N}} = \omega_p \hat{\boldsymbol{\imath}}_3 \tag{A.1}$$

- $\{N\} = \{I\}$ at time $t = 0$
- $\boldsymbol{\omega_{I/N}} = \omega_p \hat{\boldsymbol{\imath}}_3 = \text{const.}$

A.3.3 Position Frame $\mathcal{E} : \{\hat{\boldsymbol{e}}_1, \hat{\boldsymbol{e}}_2, \hat{\boldsymbol{e}}_3\}$

Define the position frame such that the first basis vector aligns with the position vector of the spacecraft, $\hat{r} = \hat{\boldsymbol{e}}_1$. This can be defined as a sequence of two principal rotations from the planet-fixed frame: first, a rotation about the north pole $\hat{\boldsymbol{\imath}}_3$ by the longitude θ, and second, a negative rotation about the (rotated) $\hat{\boldsymbol{\imath}}_2$ by the latitude ϕ. Thus, the DCM to rotate from the planet-fixed frame to the position frame can be assembled as the product of two principal rotation matrices: $[EI] = [M_2(-\phi)][M_3(\theta)]$, defined below. The orientation of the position frame with respect to the planet-fixed and inertial frames is illustrated in Fig. A.1.

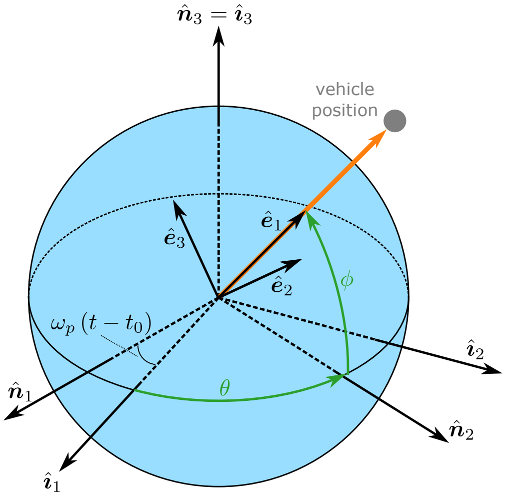


Figure A.1: Relating the inertial, planet-fixed, and position frames

All three principal rotation matrices are defined below:

$$[M_1(\theta)] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & \sin\theta \\ 0 & -\sin\theta & \cos\theta \end{bmatrix} \tag{A.2}$$

$$[M_2(\theta)] = \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \tag{A.3}$$

$$[M_3(\theta)] = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{A.4}$$

Thus:

$$[EI] = [M_2(-\phi)][M_3(\theta)] = \begin{bmatrix} \cos\phi\cos\theta & \cos\phi\sin\theta & \sin\phi \\ -\sin\theta & \cos\theta & 0 \\ -\sin\phi\cos\theta & -\sin\phi\sin\theta & \cos\phi \end{bmatrix} \tag{A.5}$$

Furthermore, note that by a property of DCMs$[IE] = [EI]^{-1} = [EI]^T$:

$$[IE] = [EI]^T = \begin{bmatrix} \cos\phi\cos\theta & -\sin\theta & -\sin\phi\cos\theta \\ \cos\phi\sin\theta & \cos\theta & -\sin\phi\sin\theta \\ \sin\phi & 0 & \cos\phi \end{bmatrix} \tag{A.6}$$

The angular velocity of the $\mathcal{E}$ frame w.r.t. the $\mathcal{I}$ frame is:

$$\boldsymbol{\omega}_{\mathcal{E}/\mathcal{I}} = \dot{\theta}\hat{\boldsymbol{\imath}}_3 - \dot{\phi}\hat{\boldsymbol{e}}_2 \tag{A.7}$$

or, equivalently:

$$\boldsymbol{\omega}_{\mathcal{E}/\mathcal{I}} = \dot{\theta}\sin\phi\hat{\boldsymbol{e}}_1 - \dot{\phi}\hat{\boldsymbol{e}}_2 + \dot{\theta}\cos\phi\hat{\boldsymbol{e}}_3 \tag{A.8}$$

In summary:

- $\hat{r} = \hat{\boldsymbol{e}}_1$
- From $\{I\}$, rotate about $\hat{\boldsymbol{\imath}}_3$ by θ, and then rotate negatively about the rotated $\hat{\boldsymbol{\imath}}_2$ by ϕ.
- $[EI] = [M_2(-\phi)][M_3(\theta)]$
- $\boldsymbol{\omega}_{\mathcal{E}/\mathcal{I}} = \dot{\theta}\sin\phi\hat{\boldsymbol{e}}_1 - \dot{\phi}\hat{\boldsymbol{e}}_2 + \dot{\theta}\cos\phi\hat{\boldsymbol{e}}_3$

A.3.4 Velocity Frame $\mathcal{S} : \{\hat{\boldsymbol{s}}_1, \hat{\boldsymbol{s}}_2, \hat{\boldsymbol{s}}_3\}$

Define the velocity frame such that the third basis vector aligns with the air-relative velocity, $\hat{U} = \hat{\boldsymbol{s}}_3$. This is defined as a sequence of two principal rotations from the position frame: first, a negative rotation about the position vector $\hat{\boldsymbol{e}}_1$ by the heading angle ψ, and second, a rotation about the (rotated) $\hat{\boldsymbol{e}}_2$ by the (planet-relative) flight path angle γ. Thus, the DCM to rotate from the position frame to the velocity frame can be assembled as the product of two principal rotation matrices: $[SE] = [M_2(\gamma)][M_1(-\psi)]$. The orientation of the velocity frame w.r.t. the position frame is illustrated in Fig. A.2.

Drag is always directed exactly opposite the air-relative velocity vector. Additionally, note that the lift vector is directed orthogonal to the air-relative velocity, and for a full-lift-up condition is aligned with the $\hat{\boldsymbol{s}}_1$ vector. To capture out-of-plane components of lift, define the bank angle σ as the rotation of the lift vector about the velocity vector, such that for $\sigma = 0$ $\hat{L} = \hat{\boldsymbol{s}}_1$ (full-lift-up) and for $\sigma = 90\hat{L} = \hat{\boldsymbol{s}}_2$ (lift-starboard). The direction of lift is thus always in this $\hat{\boldsymbol{s}}_1 - \hat{\boldsymbol{s}}_2$ plane, rotated by the bank angle, as illustrated in Fig. A.3.

$$[SE] = [M_2(\gamma)][M_1(-\psi)] = \begin{bmatrix} \cos\gamma & -\sin\gamma\sin\psi & -\sin\gamma\cos\psi \\ 0 & \cos\psi & -\sin\psi \\ \sin\gamma & \cos\gamma\sin\psi & \cos\gamma\cos\psi \end{bmatrix} \tag{A.9}$$

Again, write out $[ES] = [SE]^T$ for later convenience:

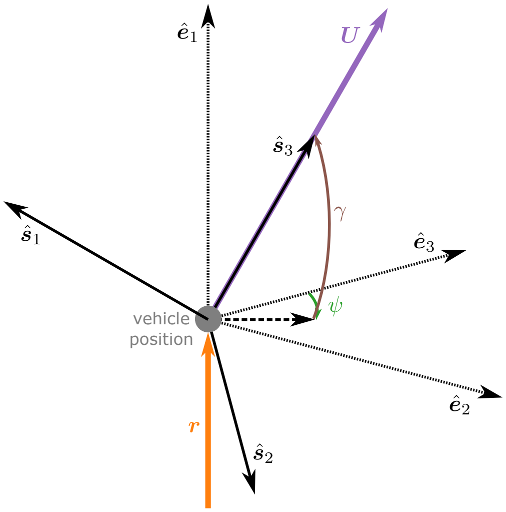


Figure A.2: Relating the position and velocity frames

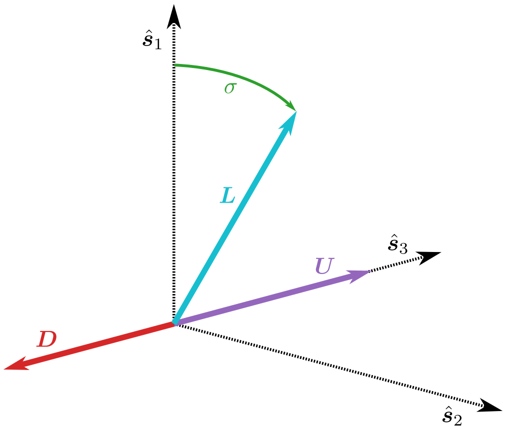


Figure A.3: Relating velocity, lift, and drag in the velocity frame

$$[ES] = [SE]^T = \begin{bmatrix} \cos\gamma & 0 & \sin\gamma \\ -\sin\gamma\sin\psi & \cos\psi & \cos\gamma\sin\psi \\ -\sin\gamma\cos\psi & -\sin\psi & \cos\gamma\cos\psi \end{bmatrix} \tag{A.10}$$

The angular velocity of the $\mathcal{S}$ frame w.r.t. the $\mathcal{E}$ frame is:

$$\boldsymbol{\omega}_{\mathcal{S}/\mathcal{E}} = -\dot{\psi}\hat{\boldsymbol{e}}_1 + \dot{\gamma}\hat{\boldsymbol{s}}_2 \tag{A.11}$$

Or, equivalently:

$$\boldsymbol{\omega}_{\mathcal{S}/\mathcal{E}} = -\dot{\psi}\cos\gamma\hat{\boldsymbol{s}}_1 + \dot{\gamma}\hat{\boldsymbol{s}}_2 - \dot{\psi}\sin\gamma\hat{\boldsymbol{s}}_3 \tag{A.12}$$

In summary:

- $\hat{U} = \hat{\boldsymbol{s}}_3$
- $\hat{D} = -\hat{\boldsymbol{s}}_3$
- $\hat{L} = \cos\sigma\hat{\boldsymbol{s}}_1 + \sin\sigma\hat{\boldsymbol{s}}_2$
- $[SE] = [M_2(\gamma)][M_1(-\psi)]$
- $\boldsymbol{\omega}_{\mathcal{S}/\mathcal{E}} = -\dot{\psi}\cos\gamma\hat{\boldsymbol{s}}_1 + \dot{\gamma}\hat{\boldsymbol{s}}_2 - \dot{\psi}\sin\gamma\hat{\boldsymbol{s}}_3$

A.3.5 Finding Unit Vectors Without DCMs

$$\hat{\boldsymbol{e}}_1 = \boldsymbol{r}/r \tag{A.13}$$

$$\hat{\boldsymbol{e}}_2 = (\hat{\boldsymbol{\imath}}_3 \times \hat{\boldsymbol{e}}_1)/|\hat{\boldsymbol{\imath}}_3 \times \hat{\boldsymbol{e}}_1| \tag{A.14}$$

$$\hat{\boldsymbol{e}}_3 = \hat{\boldsymbol{e}}_1 \times \hat{\boldsymbol{e}}_2 \tag{A.15}$$

$$\tag{A.16}$$

$$\hat{\boldsymbol{s}}_1 = \hat{\boldsymbol{s}}_2 \times \hat{\boldsymbol{s}}_3 \tag{A.17}$$

$$\hat{\boldsymbol{s}}_2 = (\hat{\boldsymbol{s}}_3 \times \hat{\boldsymbol{e}}_1)/|\hat{\boldsymbol{s}}_3 \times \hat{\boldsymbol{e}}_1| \tag{A.18}$$

$$\hat{\boldsymbol{s}}_3 = \boldsymbol{u}/u \tag{A.19}$$

A.4 Kinematic Equations

The position vector is trivially written as

$$\boldsymbol{r} = r\hat{\boldsymbol{e}}_1 \tag{A.20}$$

Air-relative velocity is defined as

$$\boldsymbol{U} \equiv \frac{{}^{\mathcal{I}}\mathrm{d}}{\mathrm{d}t}\boldsymbol{r} \tag{A.21}$$

where this notation indicates the time derivative of the position vector with respect to the planet-fixed frame. Via transport theorem:

$$\frac{{}^{\mathcal{I}}\mathrm{d}}{\mathrm{d}t}(\boldsymbol{r}) = \frac{{}^{\mathcal{E}}\mathrm{d}}{\mathrm{d}t}(\boldsymbol{r}) + \boldsymbol{\omega}_{\mathcal{E}/\mathcal{I}} \times \boldsymbol{r} \tag{A.22}$$

The first term of Eq. (A.22) evaluates to

$$\frac{{}^{\mathcal{E}}\mathrm{d}}{\mathrm{d}t}(\boldsymbol{r}) = \dot{r}\hat{\boldsymbol{e}}_1 \tag{A.23}$$

and the second term evaluates to

$$\boldsymbol{\omega}_{\mathcal{E}/\mathcal{I}} \times r\hat{\boldsymbol{e}}_1 = r\dot{\theta}\cos\phi\hat{\boldsymbol{e}}_2 + r\dot{\phi}\hat{\boldsymbol{e}}_3 \tag{A.24}$$

Thus:

$$\frac{{}^{\mathcal{I}}\mathrm{d}}{\mathrm{d}t}(\boldsymbol{r}) = \dot{r}\hat{\boldsymbol{e}}_1 + r\dot{\theta}\cos\phi\hat{\boldsymbol{e}}_2 + r\dot{\phi}\hat{\boldsymbol{e}}_3 \tag{A.25}$$

Now consider the left-hand side of Eq. A.21. In the velocity frame this expression is trivial:

$$\boldsymbol{U} = U\hat{\boldsymbol{s}}_3 \tag{A.26}$$

Rotating into the position frame gives:

$$\boldsymbol{U} = \sin\gamma U\hat{\boldsymbol{e}}_1 + \cos\gamma\sin\psi U\hat{\boldsymbol{e}}_2 + \cos\gamma\cos\psi U\hat{\boldsymbol{e}}_3 \tag{A.27}$$

Equating the left- (Eq. (A.27)) and right- (Eq. (A.25)) hand sides of Eq. A.21 and solving the time derivative of each state component gives the desired kinematic equations:

$$\dot{r}\hat{\boldsymbol{e}}_1 + r\dot{\theta}\cos\phi\hat{\boldsymbol{e}}_2 + r\dot{\phi}\hat{\boldsymbol{e}}_3 = \sin\gamma U\hat{\boldsymbol{e}}_1 + \cos\gamma\sin\psi U\hat{\boldsymbol{e}}_2 + \cos\gamma\cos\psi U\hat{\boldsymbol{e}}_3 \tag{A.28}$$

$$\dot{r} = U\sin\gamma \tag{A.29}$$

$$\dot{\theta} = \frac{U\cos\gamma\sin\psi}{r\cos\phi} \tag{A.30}$$

$$\dot{\phi} = \frac{U\cos\gamma\cos\psi}{r} \tag{A.31}$$

A.5 Kinetic Equations

Now apply Newton's second law:

$$\frac{\Sigma \boldsymbol{F}}{m} = \ddot{\boldsymbol{r}} \tag{A.32}$$

where $\Sigma \boldsymbol{F}$ is the sum of external forces acting through the center of mass of the vehicle, m is the vehicle mass, and $\ddot{\boldsymbol{r}}$ is shorthand for the second time derivative of the position vector with respect to an inertial frame, $\ddot{\boldsymbol{r}} \equiv \frac{{}^{\mathcal{N}}\mathrm{d}^2}{\mathrm{d}t^2}\boldsymbol{r}$. Likewise, write the inertial velocity as $\dot{\boldsymbol{r}} \equiv \frac{{}^{\mathcal{N}}\mathrm{d}}{\mathrm{d}t}\boldsymbol{r}$.

A.5.1 Inertial Acceleration Vector

At this point it might be tempting to start with the position vector and take its time derivative w.r.t. the inertial frame twice. However, although that would yield a correct expression, we want to take a different path here. Recall that our objective is to write EOMs in terms of our state variables, and our state components associated with velocity are defined based on the air-relative (planet-relative) velocity vector, not the inertial velocity. Therefore, what we actually want here is to write the inertial acceleration vector *in terms of the planet-relative velocity.*

First, write the inertial velocity vector in terms of the planet-relative velocity vector by applying transport theorem to get from the $\mathcal{I}$ frame to the $\mathcal{N}$ frame then substituting in the definition of $\boldsymbol{U}$ from Eq. A.21:

$$\dot{\boldsymbol{r}} \equiv \frac{{}^{\mathcal{N}}\mathrm{d}}{\mathrm{d}t}(\boldsymbol{r}) = \frac{{}^{\mathcal{I}}\mathrm{d}}{\mathrm{d}t}(\boldsymbol{r}) + \boldsymbol{\omega}_{\mathcal{I}/\mathcal{N}} \times \boldsymbol{r} = \boldsymbol{U} + \boldsymbol{\omega}_{\mathcal{I}/\mathcal{N}} \times \boldsymbol{r} \tag{A.33}$$

Recall that $\boldsymbol{\omega}_{\boldsymbol{I/N}} = \omega_p \hat{\boldsymbol{\imath}}_3$. Write $\hat{\boldsymbol{\imath}}_3$ in terms of $\mathcal{E}$ frame basis vectors

$$\hat{\boldsymbol{\imath}}_3 = \sin\phi \hat{\boldsymbol{e}}_1 + \cos\phi \hat{\boldsymbol{e}}_3 \tag{A.34}$$

$$\boldsymbol{\omega}_{\mathcal{I}/\mathcal{N}} \times \boldsymbol{r} = \omega_p \left(\sin\phi \hat{\boldsymbol{e}}_1 + \cos\phi \hat{\boldsymbol{e}}_3\right) \times r\hat{\boldsymbol{e}}_1 = r\omega_p \cos\phi \hat{\boldsymbol{e}}_2 \tag{A.35}$$

then relate $\dot{\boldsymbol{r}}$ and $\boldsymbol{U}$ as below:

$$\dot{\boldsymbol{r}} = \boldsymbol{U} + r\omega_p \cos\phi \hat{\boldsymbol{e}}_2 \tag{A.36}$$

Next, take the inertial derivative of the right-hand side of Eq. (A.36) to find the inertial acceleration in terms of the air-relative velocity.

A.5.1.1 Term 1

Recall that $\boldsymbol{U} = U\hat{\boldsymbol{s}}_3$. Then,

$$\frac{{}^{\mathcal{N}}\mathrm{d}}{\mathrm{d}t}(\boldsymbol{U}) = \frac{{}^{\mathcal{S}}\mathrm{d}}{\mathrm{d}t}(\boldsymbol{U}) + \boldsymbol{\omega}_{\mathcal{S}/\mathcal{N}} \times \boldsymbol{U} \tag{A.37}$$

The first term evaluates to:

$$\frac{{}^{\mathcal{S}}\mathrm{d}}{\mathrm{d}t}(\boldsymbol{U}) = \dot{U}\hat{\boldsymbol{s}}_3 \tag{A.38}$$

To evaluate the second term, note that:

$$\boldsymbol{\omega}_{\mathcal{S}/\mathcal{N}} = \boldsymbol{\omega}_{\mathcal{S}/\mathcal{E}} + \boldsymbol{\omega}_{\mathcal{E}/\mathcal{I}} + \boldsymbol{\omega}_{\mathcal{I}/\mathcal{N}} \tag{A.39}$$

and, equivalently,

$${}^{\mathcal{S}}\boldsymbol{\omega}_{\mathcal{S}/\mathcal{N}} = {}^{\mathcal{S}}\boldsymbol{\omega}_{\mathcal{S}/\mathcal{E}} + [SE]^{\mathcal{E}}\boldsymbol{\omega}_{\mathcal{E}/\mathcal{I}} + [SE][EI]^{\mathcal{I}}\boldsymbol{\omega}_{\mathcal{I}/\mathcal{N}} \tag{A.40}$$

Thus, Eq. (A.37)is equal to:

$$\frac{{}^{\mathcal{N}}\mathrm{d}}{\mathrm{d}t}(\boldsymbol{U}) = \dot{U}\hat{\boldsymbol{s}}_3 + \boldsymbol{\omega}_{\mathcal{S}/\mathcal{N}} \times U\hat{\boldsymbol{s}}_3 \tag{A.41}$$

A.5.1.2 Term 2

Using transport theorem,

$$\frac{{}^{\mathcal{N}}\mathrm{d}}{\mathrm{d}t}(r\omega_p \cos\phi \hat{\boldsymbol{e}}_2) = \frac{{}^{\mathcal{E}}\mathrm{d}}{\mathrm{d}t}(r\omega_p \cos\phi \hat{\boldsymbol{e}}_2) + \boldsymbol{\omega}_{\mathcal{E}/\mathcal{N}} \times (r\omega_p \cos\phi)\hat{\boldsymbol{e}}_2 \tag{A.42}$$

Noting again that $\dot{\omega}_p = 0$,

$$\frac{{}^{\mathcal{E}}\mathrm{d}}{\mathrm{d}t}(r\omega_p \cos\phi \hat{\boldsymbol{e}}_2) = \left(\dot{r}\omega_p \cos\phi - r\dot{\phi}\omega_p \sin\phi\right)\hat{\boldsymbol{e}}_2 \tag{A.43}$$

The angular velocity $\boldsymbol{\omega}_{\mathcal{E}/\mathcal{N}}$ is found with nearly the same expression as Eq. A.39, just without the first term:

$$\boldsymbol{\omega}_{\mathcal{E}/\mathcal{N}} = \boldsymbol{\omega}_{\mathcal{E}/\mathcal{I}} + \boldsymbol{\omega}_{\mathcal{I}/\mathcal{N}} \tag{A.44}$$

The right-hand side of Eq. (A.42) is then written as:

$$\frac{{}^{\mathcal{N}}\mathrm{d}}{\mathrm{d}t}(r\omega_p \cos\phi \hat{\boldsymbol{e}}_2) = \left(\dot{r}\omega_p \cos\phi - r\dot{\phi}\omega_p \sin\phi\right)\hat{\boldsymbol{e}}_2 + \boldsymbol{\omega}_{\mathcal{E}/\mathcal{N}} \times (r\omega_p \cos\phi)\hat{\boldsymbol{e}}_2 \tag{A.45}$$

A.5.1.3 Combining Terms

Combining terms yields the expression:

$$\begin{aligned}\ddot{\boldsymbol{r}} &= \left[\frac{{}^{\mathcal{N}}\mathrm{d}}{\mathrm{d}t}(\boldsymbol{U})\right] + \left[\frac{{}^{\mathcal{N}}\mathrm{d}}{\mathrm{d}t}(r\omega_p \cos\phi \hat{\boldsymbol{e}}_2)\right] \\ &= \left[\dot{U}\hat{\boldsymbol{s}}_3 + \boldsymbol{\omega}_{\mathcal{S}/\mathcal{N}} \times U\hat{\boldsymbol{s}}_3\right] + \left[\left(\dot{r}\omega_p \cos\phi - r\dot{\phi}\omega_p \sin\phi\right)\hat{\boldsymbol{e}}_2 + \boldsymbol{\omega}_{\mathcal{E}/\mathcal{N}} \times (r\omega_p \cos\phi)\hat{\boldsymbol{e}}_2\right]\end{aligned} \tag{A.46}$$

A.5.2 Acceleration Due to Forces

Now evaluate the gravity, lift, and drag forces one at a time, then write their sum in the $\mathcal{S}$ frame.

A.5.2.1 Acceleration Due to Gravity

Derive an expression for the gravitational acceleration due to the two-body plus J2 zonal perturbation in terms of the $\hat{\boldsymbol{e}}_1$ and $\hat{\boldsymbol{e}}_3$ directions. Several dot products of basis vectors will be useful in this derivation. Note that from the previously defined DCMs:

$$\hat{\boldsymbol{\imath}}_3 = \sin\phi \hat{\boldsymbol{e}}_1 + \cos\phi \hat{\boldsymbol{e}}_3 \tag{A.47}$$

Thus, $\hat{\boldsymbol{\imath}}_3 \cdot \hat{\boldsymbol{e}}_1 = \sin\phi$ and $\hat{\boldsymbol{\imath}}_3 \cdot \hat{\boldsymbol{e}}_3 = \cos\phi$. Note also that $\hat{\boldsymbol{\imath}}_3 \cdot \hat{\boldsymbol{e}}_2 = 0$, $\hat{\boldsymbol{e}}_1 \cdot \hat{\boldsymbol{e}}_1 = 1$, and $\hat{\boldsymbol{e}}_1 \cdot \hat{\boldsymbol{e}}_2 = \hat{\boldsymbol{e}}_1 \cdot \hat{\boldsymbol{e}}_3 = 0$. Finally, note that $\hat{\boldsymbol{e}}_1 \equiv \hat{r}$.

These preliminaries established, start with an expression for the gravitational potential function based on Eq. 11.63 from [239]:

$$U(\boldsymbol{r}) = \frac{\mu}{r}\left(1 - \frac{J_2 R^2}{2r^2}\left(3\sin^2\phi - 1\right)\right) \tag{A.48}$$

where R is the equatorial radius of the central body and J_2 is the corresponding spherical harmonic coefficient. Eq. A.48 includes the effects of the point-mass gravity plus the J2 zonal term from spherical harmonics.

To find the acceleration from the potential function, take the gradient:

$$\ddot{\boldsymbol{r}}_{grav} = \nabla U(\boldsymbol{r}) \tag{A.49}$$

First, substitute $(\hat{\boldsymbol{e}}_1 \cdot \hat{\boldsymbol{\imath}}_3) = \sin\phi$ into Eq. A.48, then multiply by a factor of r inside the parenthetical to convert $\hat{\boldsymbol{e}}_1$ to $\boldsymbol{r}$, multiplying by a factor of $1/r^2$ outside the parenthetical to compensate:

$$\ddot{\boldsymbol{r}}_{grav} = \nabla \left[\frac{\mu}{r} \left(1 - \frac{J_2 R^2}{2r^2} \left(\frac{3}{r^2} \left(\boldsymbol{r} \cdot \hat{\boldsymbol{\imath}}_3 \right)^2 - 1 \right) \right) \right] \tag{A.50}$$

Distribute the gradient operator (written now as a partial vector derivative):

$$\ddot{\boldsymbol{r}}_{grav} = \frac{\partial}{\partial \boldsymbol{r}} \left[\frac{\mu}{r} \right] - \frac{\partial}{\partial \boldsymbol{r}} \left[\frac{\mu J_2 R^2}{2r^5} 3 \left(\boldsymbol{r} \cdot \hat{\boldsymbol{\imath}}_3 \right)^2 \right] + \frac{\partial}{\partial \boldsymbol{r}} \left[\frac{\mu J_2 R^2}{2r^3} \right] \tag{A.51}$$

Moving constants outside of the gradient expressions:

$$\ddot{\boldsymbol{r}}_{grav} = \mu \frac{\partial}{\partial \boldsymbol{r}} \left[\frac{1}{r} \right] - \frac{3\mu J_2 R^2}{2} \frac{\partial}{\partial \boldsymbol{r}} \left[\frac{1}{r^5} \left(\boldsymbol{r} \cdot \hat{\boldsymbol{\imath}}_3 \right)^2 \right] + \frac{\mu J_2 R^2}{2} \frac{\partial}{\partial \boldsymbol{r}} \left[\frac{1}{r^3} \right] \tag{A.52}$$

Now take the gradient of each term of Eq. A.52 one at a time. Make use of an expansion of the partial vector derivative as below:

$$\frac{\partial}{\partial \boldsymbol{r}} = \frac{\partial}{\partial r} \frac{\partial r}{\partial \boldsymbol{r}} \tag{A.53}$$

In addition, note the below identity, which can be readily verified by expanding the unit vector and magnitude in terms of their components $\boldsymbol{r} = [x, y, z]^T$:

$$\frac{\partial r}{\partial \boldsymbol{r}} = \hat{r} \tag{A.54}$$

Turning now to the first term of Eq. A.52, take the scalar portion of the derivative:

$$\frac{\partial}{\partial r} \left[\frac{1}{r} \right] = -\frac{1}{r^2} \tag{A.55}$$

The full gradient of this term is then found simply by multiplying Eq. A.55 by the unit vector:

$$\frac{\partial}{\partial \boldsymbol{r}} \left[\frac{1}{r} \right] = -\frac{1}{r^2} \hat{\boldsymbol{e}}_1 \tag{A.56}$$

Next, take the second term of Eq. A.52 and expand via chain rule:

$$\frac{\partial}{\partial \boldsymbol{r}} \left[\frac{1}{r^5} \left(\boldsymbol{r} \cdot \hat{\boldsymbol{\imath}}_3 \right)^2 \right] = \frac{\partial}{\partial \boldsymbol{r}} \left[\frac{1}{r^5} \right] \left(\boldsymbol{r} \cdot \hat{\boldsymbol{\imath}}_3 \right)^2 + \frac{1}{r^5} \frac{\partial}{\partial \boldsymbol{r}} \left[\left(\boldsymbol{r} \cdot \hat{\boldsymbol{\imath}}_3 \right)^2 \right] \tag{A.57}$$

The first term of Eq. A.57 is another application of the scalar derivative times the unit vector:

$$\frac{\partial}{\partial \boldsymbol{r}} \left[\frac{1}{r^5} \right] = -\frac{5}{r^6} \hat{\boldsymbol{e}}_1 \tag{A.58}$$

The second term of Eq. A.57 is resolved by noting the derivative of the parenthetical is simply $\hat{\boldsymbol{\imath}}_3$:

$$\frac{\partial}{\partial \boldsymbol{r}}\left[\left(\boldsymbol{r}\cdot\hat{\boldsymbol{\imath}}_3\right)^2\right] = 2\left(\boldsymbol{r}\cdot\hat{\boldsymbol{\imath}}_3\right)\hat{\boldsymbol{\imath}}_3 \tag{A.59}$$

Going back now to Eq. A.52, consider the third term. Again, we apply the identity from Eq. A.54:

$$\frac{\partial}{\partial \boldsymbol{r}}\left[\frac{1}{r^3}\right] = -\frac{3}{r^4}\hat{\boldsymbol{e}}_1 \tag{A.60}$$

With this, plug in the derivatives and simplify. Substituting Eqs. A.56, A.58, A.59, and A.60 into Eq. A.52:

$$\ddot{\boldsymbol{r}}_{grav} = \mu\left[-\frac{1}{r^2}\hat{\boldsymbol{e}}_1\right] - \frac{3\mu J_2 R^2}{2}\left[-\frac{5}{r^6}\left(\boldsymbol{r}\cdot\hat{\boldsymbol{\imath}}_3\right)^2\hat{\boldsymbol{e}}_1 + \frac{2}{r^5}\left(\boldsymbol{r}\cdot\hat{\boldsymbol{\imath}}_3\right)\hat{\boldsymbol{\imath}}_3\right] + \frac{\mu J_2 R^2}{2}\left[-\frac{3}{r^4}\hat{\boldsymbol{e}}_1\right] \tag{A.61}$$

Pull the r factor back out of the $\boldsymbol{r}$ vectors to form $\hat{\boldsymbol{e}}_1$ and collect terms by basis vector:

$$\ddot{\boldsymbol{r}}_{grav} = -\frac{\mu}{r^2}\left[1 + \frac{3J_2R^2}{2r^2}\left(1 - 5\left(\hat{\boldsymbol{e}}_1\cdot\hat{\boldsymbol{\imath}}_3\right)^2\right)\right]\hat{\boldsymbol{e}}_1 - \frac{\mu}{r^2}\left[\frac{3J_2R^2}{2r^2}2\left(\hat{\boldsymbol{e}}_1\cdot\hat{\boldsymbol{\imath}}_3\right)\right]\hat{\boldsymbol{\imath}}_3 \tag{A.62}$$

Finally, substitute $\hat{\boldsymbol{e}}_1\cdot\hat{\boldsymbol{\imath}}_3 = \sin\phi$ and Eq. A.47 back into Eq. A.62 to get an expression entirely in terms of $\mathcal{E}$ frame basis vectors:

$$\frac{\boldsymbol{F}_{\text{grav}}}{m} = -\frac{\mu}{r^2}\left[1 + \frac{3J_2R^2}{2r^2}\left(1 - 3\sin^2\phi\right)\right]\hat{\boldsymbol{e}}_1 - \frac{\mu}{r^2}\left[\frac{3J_2R^2}{2r^2}2\sin\phi\cos\phi\right]\hat{\boldsymbol{e}}_3 \tag{A.63}$$

As desired, Eq. A.63 gives the acceleration due to gravity in terms of basis vectors of a single reference frame, where gravity is modeled as a point mass plus the oblateness perturbation.

A.5.2.2 Acceleration Due to Drag

Define the magnitude of the drag force and assign it to the variable D for simplification:

$$\frac{F_D}{m} \equiv D = \frac{1}{2m}\rho U^2 C_D A = \frac{\rho U^2}{2\beta} \tag{A.64}$$

where $\beta = m/(C_D A)$ is the ballistic coefficient of the vehicle, which can be understood as a ratio of the inertial forces to aerodynamics forces acting on the vehicle. Write the vector form of acceleration due to drag, which is always directed opposite the air-relative velocity:

$$\frac{\boldsymbol{F}_D}{m} = -D\hat{\boldsymbol{s}}_3 \tag{A.65}$$

A.5.2.3 Acceleration Due to Lift

Similarly, define the magnitude of the lift force and assign it to the variable L:

$$\frac{F_L}{m} \equiv L = \frac{1}{2m}\rho U^2 C_L A = \frac{\rho U^2}{2\beta} L/D = D \cdot L/D \tag{A.66}$$

where L/D is the lift-to-drag ratio of the vehicle. To write the vector form of acceleration due to lift, consider the bank angle. In the $\mathcal{S}$ frame, full-lift-up is directed along $\hat{\boldsymbol{s}}_1$ whereas full-lift-starboard is along $\hat{\boldsymbol{s}}_2$. It is straightforward to go from this and Fig. A.3 to the vector expression below:

$$\frac{\boldsymbol{F}_L}{m} = (L\cos\sigma)\,\hat{\boldsymbol{s}}_1 + (L\sin\sigma)\,\hat{\boldsymbol{s}}_2 \tag{A.67}$$

A.5.2.4 Combining Terms

Combining terms:

$$\frac{\Sigma\boldsymbol{F}}{m} = \frac{\boldsymbol{F}_{\text{grav}}}{m} + \frac{\boldsymbol{F}_D}{m} + \frac{\boldsymbol{F}_L}{m} \tag{A.68}$$

A.5.3 Equate, Substitute, and Solve

Equate $\Sigma\boldsymbol{F}/m$ and $\ddot{\boldsymbol{r}}$ as below:

$$\begin{aligned} \frac{1}{m}(\boldsymbol{F}_{\text{grav}} + \boldsymbol{F}_D + \boldsymbol{F}_L) = &\left[\dot{U}\hat{\boldsymbol{s}}_3 + \boldsymbol{\omega}_{\mathcal{S}/\mathcal{N}} \times U\hat{\boldsymbol{s}}_3\right] \\ &+ \left[\left(\dot{r}\omega_p\cos\phi - r\dot{\phi}\omega_p\sin\phi\right)\hat{\boldsymbol{e}}_2 + \boldsymbol{\omega}_{\mathcal{E}/\mathcal{N}} \times (r\omega_p\cos\phi)\hat{\boldsymbol{e}}_2\right] \end{aligned} \tag{A.69}$$

A.5.4 Results

Finally, substitute the kinematic equations (Eqs. A.29 - A.31) into Eq. A.69 and its component terms and solve:

$$\dot{r} = U\sin\gamma \tag{A.70a}$$

$$\dot{\theta} = \frac{U\cos\gamma\sin\psi}{r\cos\phi} \tag{A.70b}$$

$$\dot{\phi} = \frac{U\cos\gamma\cos\psi}{r} \tag{A.70c}$$

$$\dot{U} = -D - g_r \sin\gamma - g_\phi \cos\gamma\cos\psi$$
$$+ \omega_p^2 r \cos\phi \left(\cos\phi\sin\gamma - \sin\phi\cos\gamma\cos\psi\right) \tag{A.71a}$$
$$\dot{\gamma} = \frac{1}{U}\left[L\cos\sigma + \cos\gamma\left(\frac{U^2}{r} - g_r\right) + g_\phi\sin\gamma\cos\psi + 2\omega_p U\cos\phi\sin\psi\right.$$
$$\left. + \omega_p^2 r\cos\phi\left(\cos\phi\cos\gamma + \sin\phi\sin\gamma\cos\psi\right)\right] \tag{A.71b}$$
$$\dot{\psi} = \frac{1}{U}\left[\frac{L\sin\sigma}{\cos\gamma} + \frac{U^2}{r}\tan\phi\cos\gamma\sin\psi + g_\phi\frac{\sin\psi}{\cos\gamma}\right.$$
$$\left. - 2\omega_p U\left(\cos\phi\tan\gamma\cos\psi - \sin\phi\right) + \frac{\omega_p^2 r}{\cos\gamma}\cos\phi\sin\phi\sin\psi\right] \tag{A.71c}$$

$$L = \frac{\rho U^2}{2\beta} L/D \tag{A.72a}$$
$$D = \frac{\rho U^2}{2\beta} \tag{A.72b}$$
$$g_r \equiv |\ddot{\boldsymbol{r}}_{grav} \cdot \hat{\boldsymbol{e}}_1| = \frac{\mu}{r^2}\left[1 + \frac{3J_2R^2}{2r^2}\left(1 - 3\sin^2\phi\right)\right] \tag{A.72c}$$
$$g_\phi \equiv |\ddot{\boldsymbol{r}}_{grav} \cdot \hat{\boldsymbol{e}}_3| = \frac{\mu}{r^2}\left[\frac{3J_2R^2}{2r^2} 2\sin\phi\cos\phi\right] \tag{A.72d}$$

A.6 Coordinate Conversion

A.6.1 Inertial and Planet-Relative Velocity Conversion

It is important to note that inertial vs. planet-relative velocity and spherical vs. Cartesian coordinates are two separate choices, and their permutations give 4 ways to express the state. This is illustrated by Table A.1. All four locations in this grid are equally valid and all are used in the literature. Often, planet-relative velocity is used with spherical coordinates and inertial velocity with the Cartesian vectors. If a non-rotating planet and atmosphere is assumed with no wind, the two columns of Table A.1 collapse as identical.

Converting from the inertial Cartesian vectors in the upper-left of Table A.1 to the planet-relative spherical coordinates in the lower-right, and back again is a simple application of transport theorem:

$$\dot{\boldsymbol{r}} = \boldsymbol{U} + \boldsymbol{\omega}_{\mathcal{I}/\mathcal{N}} \times \boldsymbol{r} \tag{A.73}$$

$$\boldsymbol{U} = \dot{\boldsymbol{r}} - \boldsymbol{\omega}_{\mathcal{I}/\mathcal{N}} \times \boldsymbol{r} \tag{A.74}$$

where $\boldsymbol{\omega}_{\mathcal{I}/\mathcal{N}} = \omega_p \hat{\boldsymbol{\imath}}_3 = \omega_p \hat{\boldsymbol{n}}_3 = \text{const.}$

A.6.2 Spherical to Cartesian Conversion

Define the $[NI]$ DCM using a single-axis rotation about the 3-axis by an angle of $\Omega = \omega_p(t - t_0)$, where $(t - t_0)$ is the time elapsed since simulation start in seconds:

$$[NI] = \begin{bmatrix} \cos\Omega & -\sin\Omega & 0 \\ \sin\Omega & \cos\Omega & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{A.75}$$

Then,

$$\begin{aligned} {}^{\mathcal{N}}\boldsymbol{r} &= r[NI][IE]^{\mathcal{E}}\hat{\boldsymbol{e}}_1 \\ &= r \begin{bmatrix} \cos\phi\cos\theta\cos\Omega - \cos\phi\sin\theta\sin\Omega \\ \cos\phi\cos\theta\sin\Omega + \cos\phi\sin\theta\cos\Omega \\ \sin\phi \end{bmatrix} \end{aligned} \tag{A.76}$$

Follow a similar process for the inertial velocity vector:

$$\begin{aligned} {}^{\mathcal{N}}\boldsymbol{U} &= U[NI][IE][ES]^{\mathcal{S}}\hat{\boldsymbol{s}}_3 \\ &= U \begin{bmatrix} \cos(\Omega+\theta)\cos\phi\sin\gamma - \sin(\Omega+\theta)\cos\gamma\sin\psi - \cos(\Omega+\theta)\cos\gamma\cos\psi\sin\phi \\ \cos(\Omega+\theta)\cos\gamma\sin\psi + \sin(\Omega+\theta)\cos\phi\sin\gamma - \sin(\Omega+\theta)\cos\gamma\cos\psi\sin\phi \\ \sin\gamma\sin\phi + \cos\gamma\cos\phi\cos\psi \end{bmatrix} \end{aligned} \tag{A.77}$$

	Inertial Velocity	**Planet-Relative Velocity**
Cartesian vectors	$\boldsymbol{r}, \dot{\boldsymbol{r}}$	$\boldsymbol{r}, \boldsymbol{U}$
Spherical	—	$r, \theta, \phi, U, \gamma, \psi$

Table A.1: Some possible state representations

A.6.3 Cartesian to Spherical Conversion

Converting from Cartesian vectors to spherical coordinates involves a bit more visualization/geometry. Note that this process is again agnostic as to which velocity vector is used, but here denote velocity as $\boldsymbol{U}$ under the assumption planet-relative velocity is being used. Plugging in $\dot{\boldsymbol{r}}$ would result in the correct expressions for spherical coordinates using inertial velocity.

Find the first two state elements by simply taking the magnitude of the position and velocity vectors:

$$r = |\boldsymbol{r}| \tag{A.78}$$

$$U = |\boldsymbol{U}| \tag{A.79}$$

Next, find longitude θ. Figure A.4 visualizes the projection of the position vector in the equatorial plane (note that the solid orange vector is not labeled because it is this projection, not the full position vector $\boldsymbol{r}$). Longitude θ is then the angle between the $\hat{\boldsymbol{\imath}}_1$ basis vector and this projection:

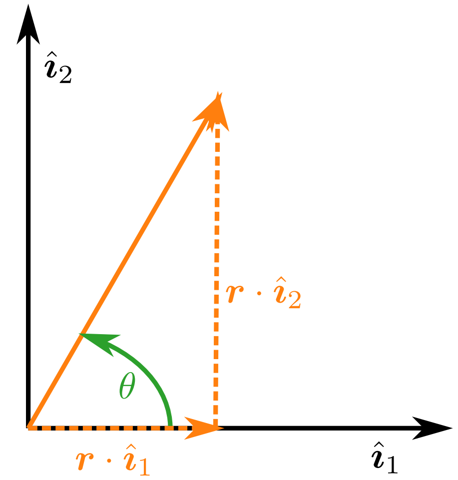


Figure A.4: Finding longitude from the position vector

$$\theta = \tan^{-1}\left(\frac{\boldsymbol{r} \cdot \hat{\boldsymbol{\imath}}_2}{\boldsymbol{r} \cdot \hat{\boldsymbol{\imath}}_1}\right) \tag{A.80}$$

Find the latitude ϕ through a similar process, starting with Fig. A.5. The dotted black vector is defined as orthogonal to the $\hat{\boldsymbol{\imath}}_3$ basis vector and is thus in the equatorial plane. The latitude ϕ is defined as the angle between this plane and the position vector $\boldsymbol{r}$:

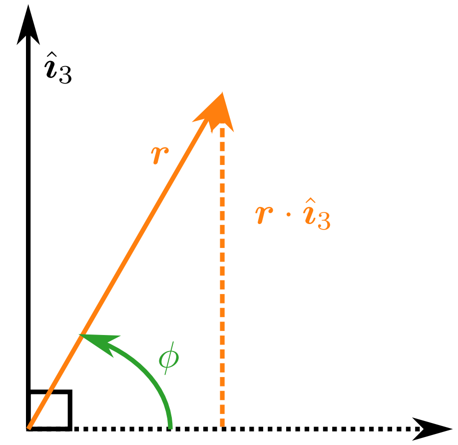


Figure A.5: Finding latitude from the position vector

$$\phi = \sin^{-1}\left(\frac{\boldsymbol{r} \cdot \hat{\boldsymbol{\imath}}_3}{r}\right) \tag{A.81}$$

Flight-path angle and heading angle are now found in much the same way as longitude and latitude. Flight-path angle is visualized in Fig. A.6 and given by the expression below:

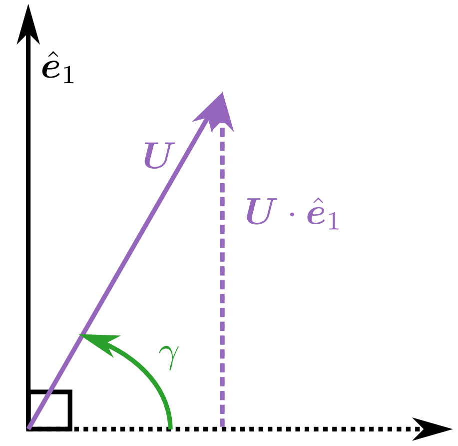


Figure A.6: Finding flight-path angle from the velocity vector

$$\gamma = \sin^{-1}\left(\frac{\boldsymbol{U} \cdot \hat{\boldsymbol{e}}_1}{U}\right) \tag{A.82}$$

Finally, heading angle is visualized in Fig. A.7 and given by the expression below. Note that, as visualized here, heading angle is measured down from the y-axis as opposed to up from the x-axis like longitude.

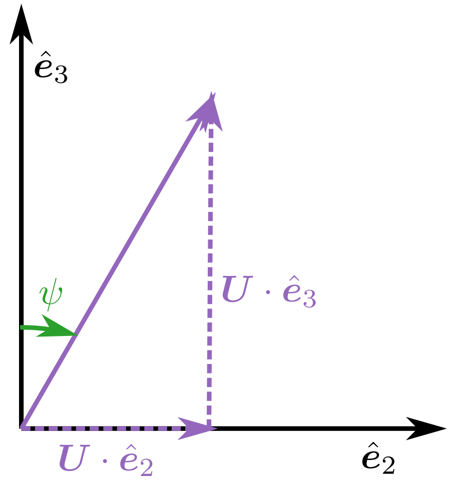


Figure A.7: Finding heading angle from the velocity vector

$$\psi = \tan^{-1}\left(\frac{\boldsymbol{U} \cdot \hat{\boldsymbol{e}}_2}{\boldsymbol{U} \cdot \hat{\boldsymbol{e}}_3}\right) \tag{A.83}$$

Appendix B

Partial Derivatives

The following partial derivatives are assumed to be evaluated along a nominal trajectory as a function of time, but explicit dependence on time is suppressed for notational simplicity.

B.1 Common Terms

The equations of motion are rewritten below in the form $\dot{x} = f(t, x, u(t, x), p_0) = (f_r, f_V, f_\gamma, f_R)$ as

$$\dot{r} = f_r = V \sin\gamma, \tag{B.1a}$$

$$\dot{V} = f_V = -\frac{\rho V^2}{2\beta} - \frac{\mu_{\text{grav}} \sin\gamma}{r^2}, \tag{B.1b}$$

$$\dot{\gamma} = f_\gamma = \frac{\rho V}{2\beta} L/D \cos\sigma - \left(\frac{\mu_{\text{grav}}}{r^2} - \frac{V^2}{r}\right)\frac{\cos\gamma}{V}, \tag{B.1c}$$

$$\dot{R} = f_R = V \cos\gamma. \tag{B.1d}$$

Each component of the A matrix, defined by (3.24), is given below for atmospheric flight. Note that these sensitivities make use of the atmospheric density scale height H defined such that $\partial\rho/\partial h = -\rho/H$, but this is not equivalent to assuming an exponential atmosphere, and H can be numerically estimated at each altitude for any density profile. Thus:

$$\frac{\partial f_r}{\partial r} = 0, \quad \frac{\partial f_r}{\partial V} = \sin\gamma, \quad \frac{\partial f_r}{\partial \gamma} = V\cos\gamma, \quad \frac{\partial f_r}{\partial R} = 0, \tag{B.2}$$

$$\frac{\partial f_V}{\partial r} = \frac{\rho V^2}{2H\beta} + \frac{2\mu_{\text{grav}} \sin\gamma}{r^3}, \quad \frac{\partial f_V}{\partial V} = -\frac{\rho V}{\beta}, \quad \frac{\partial f_V}{\partial \gamma} = -\frac{\mu_{\text{grav}} \cos\gamma}{r^2}, \quad \frac{\partial f_V}{\partial R} = 0, \tag{B.3}$$

$$\frac{\partial f_\gamma}{\partial r} = -\frac{\rho V}{2H\beta} L/D \cos\sigma + \left(\frac{2\mu_{\text{grav}}}{r^3} - \frac{V^2}{r^2}\right)\frac{\cos\gamma}{V}, \quad \frac{\partial f_\gamma}{\partial V} = \frac{\rho}{2\beta} L/D \cos\sigma + \frac{\cos\gamma}{r}\left(\frac{\mu_{\text{grav}}}{V^2 r} + 1\right), \tag{B.4a}$$

$$\frac{\partial f_\gamma}{\partial \gamma} = \left(\frac{\mu_{\text{grav}}}{r^2} - \frac{V^2}{r}\right)\frac{\sin\gamma}{V}, \quad \frac{\partial f_\gamma}{\partial R} = 0, \tag{B.4b}$$

$$\frac{\partial f_R}{\partial r} = 0, \quad \frac{\partial f_R}{\partial V} = \cos\gamma, \quad \frac{\partial f_R}{\partial \gamma} = -V\sin\gamma, \quad \frac{\partial f_R}{\partial R} = 0. \tag{B.5}$$

Each component of the B matrix, defined by Eq. (3.36), is given below for atmospheric flight where the control is $u = \cos\sigma$:

$$\frac{\partial f_r}{\partial u} = 0, \quad \frac{\partial f_V}{\partial u} = 0, \quad \frac{\partial f_\gamma}{\partial u} = \frac{\rho V}{2\beta} L/D, \quad \frac{\partial f_R}{\partial u} = 0. \tag{B.6}$$

The C matrix is computed based on the terms in the KL expansion of density variability, following (3.30) and (3.31) where the GRF in this case is density $\rho(z)$. The partial derivatives of the dynamics with respect to density are given by (B.7), and the partial derivatives of density with respect to the uncertain parameters p_0 are given by (B.8) below

$$\frac{\partial f_r(t)}{\partial \rho(h)} = 0, \qquad \frac{\partial f_V(t)}{\partial \rho(h)} = -\frac{V^2}{2\beta}, \qquad \frac{\partial f_\gamma(t)}{\partial \rho(h)} = \frac{Vu}{2\beta} L/D, \qquad \frac{\partial f_R(t)}{\partial \rho(h)} = 0, \tag{B.7}$$

$$\frac{\partial \rho(h)}{\partial p_0} = \begin{bmatrix} \frac{\partial \rho(h)}{\partial w_1} & \cdots & \frac{\partial \rho(h)}{\partial w_q} \end{bmatrix}, \quad \text{where} \quad \frac{\partial \rho(h)}{\partial w_i} = \sqrt{\lambda_i}\phi_i(h). \tag{B.8}$$

As per (3.24), the ith row of the C matrix is then built by multiplying $\partial f_i(t)/\partial \rho(h)$ (a scalar) by $\partial \rho(h)/\partial p_0$ (a $1 \times q$ matrix), then prepending n zeros to the row, such that the dimensions of C are $n \times (n+q)$.

B.2 Aerocapture

The aerocapture guidance law uses the partial derivative of apoapsis radius at the final state (at or after atmospheric exit) to compute the adjoint state terminal condition. Partial derivatives of the apoapsis velocity and ΔV_1 with respect to the final state are also used during analysis. The states in the following equations must be *inertial*, meaning either a nonrotating planet is assumed or the full state is converted from planet-relative to inertial before these expressions are evaluated.

Begin by noting that downrange distance has no impact on any of these derivatives.

$$\frac{\partial r_a}{\partial R_f} = 0, \quad \frac{\partial V_a}{\partial R_f} = 0, \quad \frac{\partial \Delta V_1}{\partial R_f} = 0. \tag{B.9}$$

Next, note that the circular (final) orbit velocity is not impacted by the final states and so this term drops out of the ΔV partial derivative:

$$\frac{\partial V_c}{\partial r_f} = 0, \quad \frac{\partial V_c}{\partial V_f} = 0, \quad \frac{\partial V_c}{\partial \gamma_f} = 0. \tag{B.10}$$

Now write the partial derivatives of energy and magnitude of angular momentum with respect to the states,

$$\frac{\partial \varepsilon_f}{\partial r_f} = \frac{\mu_{\text{grav}}}{r_f^2}, \quad \frac{\partial \varepsilon_f}{\partial V_f} = V_f, \quad \frac{\partial \varepsilon_f}{\partial \gamma_f} = 0, \tag{B.11}$$

$$\frac{\partial |\boldsymbol{h}_f|}{\partial r_f} = V_f \cos\gamma_f, \quad \frac{\partial |\boldsymbol{h}_f|}{\partial V_f} = r_f \cos\gamma_f, \quad \frac{\partial |\boldsymbol{h}_f|}{\partial \gamma_f} = -r_f V_f \sin\gamma_f. \tag{B.12}$$

The states affect the velocity at apoapsis through the energy and the magnitude of angular momentum, so take its partial derivatives with respect to these quantities.

$$\frac{\partial V_a}{\partial |\boldsymbol{h}_f|} = -\left(\frac{V_a}{|\boldsymbol{h}_f|} + \frac{2\varepsilon_f}{\sqrt{\mu_{\text{grav}}^2 + 2\varepsilon_f |\boldsymbol{h}_f|^2}}\right), \quad \frac{\partial V_a}{\partial \varepsilon_f} = -\frac{|\boldsymbol{h}_f|}{\sqrt{\mu_{\text{grav}}^2 + 2\varepsilon_f |\boldsymbol{h}_f|^2}}. \tag{B.13}$$

The partial derivatives of velocity at apoapsis with respect to the states can then be found via chain rule using the partial derivatives we've already computed.

$$\frac{\partial V_a}{\partial r_f} = \frac{\partial V_a}{\partial |\boldsymbol{h}_f|}\frac{\partial |\boldsymbol{h}_f|}{\partial r_f} + \frac{\partial V_a}{\partial \varepsilon_f}\frac{\partial \varepsilon_f}{\partial r_f}, \quad \frac{\partial V_a}{\partial V_f} = \frac{\partial V_a}{\partial |\boldsymbol{h}_f|}\frac{\partial |\boldsymbol{h}_f|}{\partial V_f} + \frac{\partial V_a}{\partial \varepsilon_f}\frac{\partial \varepsilon_f}{\partial V_f}, \quad \frac{\partial V_a}{\partial \gamma_f} = \frac{\partial V_a}{\partial |\boldsymbol{h}_f|}\frac{\partial |\boldsymbol{h}_f|}{\partial \gamma_f} + \frac{\partial V_a}{\partial \varepsilon_f}\frac{\partial \varepsilon_f}{\partial \gamma_f}. \tag{B.14}$$

Now make use of those derivatives in a similar chain rule expression for the derivatives of apoapsis radius, where the derivatives of r_a with respect to $|\boldsymbol{h}_f|$ and V_a were included directly in the expressions below.

$$\frac{\partial r_a}{\partial r_f} = \frac{1}{V_a}\frac{\partial |\boldsymbol{h}_f|}{\partial r_f} - \frac{|\boldsymbol{h}_f|}{V_a^2}\frac{\partial V_a}{\partial r_f}, \quad \frac{\partial r_a}{\partial V_f} = \frac{1}{V_a}\frac{\partial |\boldsymbol{h}_f|}{\partial V_f} - \frac{|\boldsymbol{h}_f|}{V_a^2}\frac{\partial V_a}{\partial V_f}, \quad \frac{\partial r_a}{\partial \gamma_f} = \frac{1}{V_a}\frac{\partial |\boldsymbol{h}_f|}{\partial \gamma_f} - \frac{|\boldsymbol{h}_f|}{V_a^2}\frac{\partial V_a}{\partial \gamma_f}. \tag{B.15}$$

The states affect the velocity at apoapsis after maneuver 1, V_1, only through the apoapsis radius r_a, so we take that derivative below.

$$\frac{\partial V_1}{\partial r_a} = -\frac{\mu_{\text{grav}} r_c (2r_a + r_c)}{V_1 r_a^2 (r_a + r_c)^2} \tag{B.16}$$

Now use chain rule to find the derivatives of V_1 with respect to the states.

$$\frac{\partial V_1}{\partial r_f} = \frac{\partial V_1}{\partial r_a}\frac{\partial r_a}{\partial r_f}, \quad \frac{\partial V_1}{\partial V_f} = \frac{\partial V_1}{\partial r_a}\frac{\partial r_a}{\partial V_f}, \quad \frac{\partial V_1}{\partial \gamma_f} = \frac{\partial V_1}{\partial r_a}\frac{\partial r_a}{\partial \gamma_f}, \tag{B.17}$$

Finally, combine the velocity partial derivatives to get the partial derivatives of ΔV_1 with respect to the states.

$$\frac{\partial \Delta V_1}{\partial r_f} = \frac{\partial V_1}{\partial r_f} - \frac{\partial V_a}{\partial r_f}, \quad \frac{\partial \Delta V_1}{\partial V_f} = \frac{\partial V_1}{\partial V_f} - \frac{\partial V_a}{\partial V_f}, \quad \frac{\partial \Delta V_1}{\partial \gamma_f} = \frac{\partial V_1}{\partial \gamma_f} - \frac{\partial V_a}{\partial \gamma_f}. \tag{B.18}$$

Appendix C

Relative Orbit Elements Derivation Detail

In addition to the Hill frame of the chief spacecraft $\mathcal{O}$, define $\mathcal{D}$ as the Hill frame of the deputy spacecraft. Thus, ${}^{\mathcal{D}}\boldsymbol{r}_d = {}^{\mathcal{D}}(r_d, 0, 0)^T$, and recall that ${}^{\mathcal{O}}\boldsymbol{r}_d = {}^{\mathcal{O}}(x + x_c, y + y_c, z)^T$. The deputy position vector is mapped from the deputy Hill frame to the chief velocity frame via the inertial frame as

$$ {}^{\mathcal{V}}\boldsymbol{r}_d = [VO][ON][ND]{}^{\mathcal{D}}\boldsymbol{r}_d \tag{C.1} $$

As before, assume that the distance between deputy and chief is much less than the chief radius, $(x, y, z) \ll r_c$. Taking the first variations of $[ND]$ and r_d about the chief spacecraft gives the following first-order approximations [239]

$$ [ND] \approx [NO] + [\delta NO] \tag{C.2} $$

$$ r_d \approx r_c + \delta r \tag{C.3} $$

Substituting these approximations into Eq. (C.1) yields

$$ {}^{\mathcal{V}}\boldsymbol{r}_d = [VO]\left(\mathbb{I}_3 + [ON][\delta NO]\right)\begin{bmatrix} r_c + \delta r \\ 0 \\ 0 \end{bmatrix} \tag{C.4} $$

where $\mathbb{I}_3$ is the 3x3 identity matrix.

Note that the deputy position vector can be written as

$$ {}^{\mathcal{V}}\boldsymbol{r}_d = {}^{\mathcal{V}}\boldsymbol{\rho} + [VO]\,{}^{\mathcal{O}}\begin{bmatrix} r_c \\ 0 \\ 0 \end{bmatrix} \tag{C.5} $$

Substituting Eq. (C.5) into Eq. (C.4), dropping the second-order terms associated with $[\delta NO](\delta r, 0, 0)^T$, and simplifying, the following expression is obtained

$$ {}^{\mathcal{V}}\boldsymbol{\rho} = [VO]\left({}^{\mathcal{O}}\begin{bmatrix} \delta r \\ 0 \\ 0 \end{bmatrix} + [ON][\delta NO]\,{}^{\mathcal{O}}\begin{bmatrix} r_c \\ 0 \\ 0 \end{bmatrix} \right) \tag{C.6} $$

Ref. [239] shows that the parenthetical in the right-hand side of Eq. (C.6) is equivalent to ${}^{\mathcal{O}}\boldsymbol{\rho}$ and can be expressed as Eq. (6.30), repeated here for convenience:

$$ {}^{\mathcal{O}}\boldsymbol{\rho} = \begin{bmatrix} \delta r \\ r(\delta\theta + \cos i\delta\Omega) \\ r(\sin\theta\delta i - \cos\theta\sin i\delta\Omega \end{bmatrix} $$

The variation of orbit radius is expressed as [239]

$$ \delta r = \frac{r}{a}\delta a + \frac{V_r}{V_t}r\delta\theta - \frac{r}{p}(2aq_1 + r\cos\theta)\delta q_1 - \frac{r}{p}(2aq_2 + r\sin\theta)\delta q_2, \tag{C.7} $$

where

$$ V_r = \dot{r} = \frac{h}{p}(q_1\sin\theta - q_2\cos\theta) \tag{C.8a} $$

$$ V_t = r\dot{\theta} = \frac{h}{p}(1 + q_1\cos\theta + q_2\sin\theta), \tag{C.8b} $$

$$ q_1 = e\cos\omega \tag{C.9a} $$

$$ q_2 = e\sin\omega, \tag{C.9b} $$

and

$$\theta = f + \omega. \tag{C.10}$$

Note also the orbit identities:

$$r = \frac{p}{\alpha} \tag{C.11}$$

$$p = a(1 - e^2) \tag{C.12}$$

Taking the first-order variations of Eqs. (C.9) and (C.10) gives expressions for their corresponding orbit element differences:

$$\delta q_1 = \cos\omega\delta e - e\sin\omega\delta\omega \tag{C.13}$$

$$\delta q_2 = \sin\omega\delta e + e\cos\omega\delta\omega \tag{C.14}$$

$$\delta\theta = \delta f + \delta\omega \tag{C.15}$$

Appendix D

Useful Coordinate Relationships

Let θ and ϕ be longitude and latitude, respectively, and model the central body as a perfect sphere for the purpose of these equations. The range (great circle distance) between points (θ_1, ϕ_1) and (θ_2, ϕ_2) is

$$d = R\cos^{-1}\left(\sin\phi_1 \sin\phi_2 + \cos\phi_1 \cos\phi_2 \cos(|\theta_2 - \theta_1|)\right) \tag{D.1}$$

and the bearing between them (e.g. the heading angle of the great circle arc connecting the points) is

$$\psi_B = \tan^{-1}\left(\frac{\cos\phi_2 \sin(\theta_2 - \theta_1)}{\cos\phi_1 \sin\phi_2 - \sin\phi_1 \cos\phi_2 \cos(\theta_2 - \theta_1)}\right) \tag{D.2}$$

In the case where the coordinates of point 1 are known, along with the great circle distance and bearing between it and point 2, the coordinates of the second point can be computed as

$$\theta_2 = \theta_1 + \tan^{-1}\left(\frac{\sin\psi_B \sin(d/R)\cos\phi_1}{\cos(d/R) - \sin\phi_1 \sin\phi_2}\right) \tag{D.3a}$$

$$\phi_2 = \sin^{-1}\left(\sin\phi_1 \cos(d/R) + \cos\phi_1 \sin(d/R)\cos\psi_B\right) \tag{D.3b}$$

Appendix E

Notes on Iterative Covariance Steering

This appendix summarizes the key steps to take a chance-constrained nonlinear dynamical system influenced by stochastic perturbations and, through linearization and convex relaxation, reformulate it into a sequence of convex optimization problems. These notes follow closely from from Refs. [96, 287, 288].

Begin with the nonlinear stochastic system

$$\mathrm{d}\boldsymbol{x} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t)\mathrm{d}t + \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{u})\mathrm{d}\boldsymbol{w} \tag{E.1}$$

on the the interval $t \in [t_0, t_f]$, where $\boldsymbol{x} \in \mathbb{R}^n$ is the state vector, $\boldsymbol{u} \in \mathbb{R}^m$ is the control vector, and $\boldsymbol{w}(t)$ is an n-dimensional Brownian motion. Assume that the state at the initial time $\boldsymbol{x}(t_0) = \boldsymbol{x}_0$ is Gaussian distributed with known mean and covariance. For a reference control input $\hat{\boldsymbol{u}}(t)$, the reference state $\hat{\boldsymbol{x}}(t)$ is the solution to the deterministic system

$$\dot{\hat{\boldsymbol{x}}}(t) = \boldsymbol{f}(\hat{\boldsymbol{x}}(t), \hat{\boldsymbol{u}}(t), t), \tag{E.2}$$

noting that the disturbance term does not appear in the reference because it has zero mean.

Linearize the unperturbed dynamics function $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t)$ by taking a first-order Taylor series expansion about the reference to obtain

$$\boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t), t) \approx \boldsymbol{f}(\hat{\boldsymbol{x}}(t), \hat{\boldsymbol{u}}(t), t) + \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}(\boldsymbol{x}(t) - \hat{\boldsymbol{x}}(t)) + \frac{\partial \boldsymbol{f}}{\partial u}(\boldsymbol{u}(t) - \hat{\boldsymbol{u}}(t)). \tag{E.3}$$

Define the Jacobian matrices

$$A(t) = \left.\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}\right|_{\hat{\boldsymbol{x}}, \hat{\boldsymbol{u}}} \quad B(t) = \left.\frac{\partial \boldsymbol{f}}{\partial u}\right|_{\hat{\boldsymbol{x}}, \hat{\boldsymbol{u}}} \tag{E.4}$$

evaluated at the reference $(\hat{\boldsymbol{x}}(t), \hat{\boldsymbol{u}}(t))$, define the perturbed dynamics evaluated along the reference as

$$D(t) = \boldsymbol{g}(\hat{\boldsymbol{x}}(t), \hat{\boldsymbol{u}}(t)), \tag{E.5}$$

and let the vector $\boldsymbol{c}(t)$ be defined as

$$\boldsymbol{c}(t) = \boldsymbol{f}(\hat{\boldsymbol{x}}(t), \hat{\boldsymbol{u}}(t), t) - A(t)\hat{\boldsymbol{x}}(t) - B(t)\hat{\boldsymbol{u}}(t). \tag{E.6}$$

Using this condensed notation, the linearized system dynamics are written as

$$\mathrm{d}\boldsymbol{x} = [A(t)\boldsymbol{x}(t) + B(t)\boldsymbol{u}(t) + \boldsymbol{c}(t)]\,\mathrm{d}t + D(t)\mathrm{d}\boldsymbol{w}. \tag{E.7}$$

Equation (E.7) approximates the true dynamics Eq. (E.1) as a continuous-time linear time-varying system. To convert to a discrete-time system, let $t_0 < t_1 < \cdots < t_N = t_f$ be a discrete time partition of the interval $[t_0, t_f]$ with N nodes. Make the zero-order hold (ZOH) assumption that the control is piecewise constant between time nodes,

$$\boldsymbol{u}(t) = \boldsymbol{u}_k, \quad \forall t \in [t_k, t_{k+1}), \quad k = 0, \ldots, N-1, \tag{E.8}$$

where the subscript k denotes $\boldsymbol{x}_k = \boldsymbol{x}(t_k)$, $\boldsymbol{u}_k = \boldsymbol{u}(t_k)$, etc. Make use of the ZOH assumption to integrate Eq. (E.7) from time t_k to time t_{k+1} to find a solution to the linearized system,

$$\boldsymbol{x}_{k+1} = \Phi(t_{k+1}, t_k)\boldsymbol{x}_k + \int_{t_k}^{t_{k+1}} \Phi(t_{k+1}, t)\,[B(t)\boldsymbol{u}_k + c(t)]\,\mathrm{d}t + \int_{t_k}^{t_{k+1}} \Phi(t_{k+1}, t)D(t)\mathrm{d}\boldsymbol{w}(t) \tag{E.9}$$

where $\Phi(t_2, t_1)$ is the state transition matrix from t_1 to t_2,

$$\dot{\Phi}(t, t_0) = A(t)\Phi(t, t_0), \quad \Phi(t_0, t_0) = \mathbb{I}_n, \tag{E.10}$$

where $\mathbb{I}_n$ is the $n \times n$ identity matrix. Define the discrete-time system matrices A_k and B_k, along with the discrete-time vector c_k, as

$$A_k = \Phi(t_{k+1}, t_k), \tag{E.11a}$$

$$B_k = \int_{t_k}^{t_{k+1}} \Phi(t_{k+1}, t)B(t)\mathrm{d}t, \tag{E.11b}$$

$$c_k = \int_{t_k}^{t_{k+1}} \Phi(t_{k+1}, t)c(t)\mathrm{d}t. \tag{E.11c}$$

Following the previous assumption of Brownian motion $\boldsymbol{w}(t)$, let $\boldsymbol{w}_k$ a sequence of n-dimensional independent and identically distributed standard normal variables. Then, define D_k such that the vector $D_k\boldsymbol{w}_k$ has covariance

$$\mathrm{Cov}(D_k\boldsymbol{w}_k, D_k\boldsymbol{w}_k) = \int_{t_k}^{t_{k+1}} \Phi(t_{k+1}, t)D(t)D^{\intercal}(t)\Phi^{\intercal}(t_{k+1}, t)\mathrm{d}t = D_kD_k^{\intercal}. \tag{E.12}$$

Note that to obtain a value D_k from Eq. (E.12) one can first evaluate the integral in the middle term, then take the Cholesky decomposition to isolate a valid D_k matrix. Substituting Eqs. (E.11) and (E.12) into Eq. (E.9), the stochastic difference equation

$$\boldsymbol{x}_{k+1} = A_k\boldsymbol{x}_k + B_k\boldsymbol{u}_k + \boldsymbol{c}_k + D_k\boldsymbol{w}_k \tag{E.13}$$

provides an exact discretization of the continuous-time linearized system Eq. (E.7) [287].

Assume a linear state feedback law,

$$\boldsymbol{u}_k = K_k(\boldsymbol{x}_k - \boldsymbol{\mu}_k) + \boldsymbol{\nu}_k. \tag{E.14}$$

For a quadratic cost function in the absence of chance constraints, this is the optimal form of the control [289]. The resulting mean and covariance propagation equations are then

$$\boldsymbol{\mu}_{k+1} = A_k\boldsymbol{\mu}_k + B_k\boldsymbol{\nu}_k + \boldsymbol{c}_k, \tag{E.15a}$$

$$P_{k+1} = (A_k + B_kK_k)P_k(A_k + B_kK_k)^{\intercal} + D_kD_k^{\intercal}. \tag{E.15b}$$

From Eqs. (E.15a) and (E.15b), it is apparent that the nominal control steers the mean while the feedback gain steers the covariance; this is a key insight for covariance steering [290, 291]. However, as shown below, the chance constraints depend on both the mean and covariance of the state. This motivates a *joint optimization* over the nominal control ν and feedback gains K at each discretization point.

The products of decision variables K and P in Eq. (E.15b) make the problem nonconvex. One approach to addressing this issue is to implement state history feedback of the form

$$\boldsymbol{u}_k = \sum_{\ell=0}^{k} K_{k,\ell}(\boldsymbol{x}_\ell - \boldsymbol{\mu}_\ell) + \boldsymbol{\nu}_k, \tag{E.16}$$

which is equivalent to disturbance history feedback and ultimately results in a convex formulation [292, 293]. The trade-off of this approach, however, is that the number of decision variables in the convex optimization problem then grows as $\mathcal{O}(N^2)$ as opposed to $\mathcal{O}(N)$, where N is the number of time nodes in the discretized problem.

Instead, a change of variables can be used to implement a convex relaxation and transform the problem into a linear semidefinite program. First, define the new variable

$$Y_k \equiv K_k P_k. \tag{E.17}$$

The covariance dynamics (Eq. (E.15b)) can then be written in a relaxed form as

$$Y_k P_k^{-1} Y_k^\intercal - \Phi_k \preceq 0, \tag{E.18a}$$

$$A_k P_k A_k^\intercal + B_k Y_k A_k^\intercal + A_k Y_k B_k^\intercal + B_k \Phi_k B_k^\intercal + D_k D_k^\intercal - P_{k+1} = 0, \tag{E.18b}$$

where Φ_k acts similarly to a slack variable and, crucially, its trace at all time nodes except the final time, $\sum_{k=0}^{N-1} \operatorname{tr}(\Phi_k)$, is penalized in the cost function. Equation (E.18a) can be expressed as a linear matrix inequality using the Schur complement:

$$\begin{bmatrix} P_k & Y_k^\intercal \\ Y_k & \Phi_k \end{bmatrix} \succeq 0. \tag{E.19}$$

This has been shown to be a lossless relaxation [289, 288]. Note that the value of the slack variable Φ_k when the true covariance dynamics are respected (that is, when Eq. (E.18a) is satisfied at equality), the slack variable takes on a value of $\Phi_k = Y_k P_k^{-1} Y_k^\intercal = K_k P_k K_k^\intercal$. This is equivalent to the covariance of the control input.

Now consider chance constraints on the state and control of the form

$$\mathcal{P}\left(\boldsymbol{a}_k^\intercal \boldsymbol{x}_k \geq \alpha_{\max}\right) \leq p_k^x, \tag{E.20a}$$

$$\mathcal{P}\left(\boldsymbol{b}_k^\intercal \boldsymbol{u}_k \geq \beta_{\max}\right) \leq p_k^u, \tag{E.20b}$$

where the probability that the scalar $\boldsymbol{a}_k^\intercal \boldsymbol{x}_k$ equals or exceeds $\alpha_{\max}$ is constrained to stay within p_k^x and likewise for the control constraints. The probability of constraint violation p_k^x or p_k^u is typically set to 0.05 or less, such that constraints can be nearly guaranteed to be respected, but, for unbounded perturbations such as Brownian noise, deterministic constraints cannot readily be imposed. Under the assumption that the initial distribution of the state is Gaussian and a linear feedback law is employed, the scalar random variables $\boldsymbol{a}_k^\intercal \boldsymbol{x}_k$ and $\boldsymbol{b}_k^\intercal \boldsymbol{u}_k$ are Gaussian with mean and covariance as follows [294]:

$$\langle \boldsymbol{a}^\intercal \boldsymbol{x}_k \rangle = \boldsymbol{a}^\intercal \boldsymbol{\mu}_k, \tag{E.21a}$$

$$\langle \boldsymbol{b}^\intercal \boldsymbol{u}_k \rangle = \boldsymbol{b}^\intercal \boldsymbol{\nu}_k, \tag{E.21b}$$

$$\mathrm{Cov}(\boldsymbol{a}^\intercal \boldsymbol{x}_k) = \boldsymbol{a}^\intercal P_k \boldsymbol{a}, \tag{E.21c}$$

$$\mathrm{Cov}(\boldsymbol{b}^\intercal \boldsymbol{u}_k) = \boldsymbol{b}^\intercal K_k P_k K_k^\intercal \boldsymbol{b} = \boldsymbol{b}^\intercal Y_k P_k^{-1} Y_k^\intercal \boldsymbol{b}. \tag{E.21d}$$

Using these properties, Eqs. (E.20) are converted to

$$\mathrm{normcdf}^{-1}(1 - p_k^x)\sqrt{\boldsymbol{a}^\intercal P_k \boldsymbol{a}} + \boldsymbol{a}^\intercal \boldsymbol{\mu}_k - \alpha_{\mathrm{max}} \le 0 \tag{E.22a}$$

$$\mathrm{normcdf}^{-1}(1 - p_k^u)\sqrt{\boldsymbol{b}^\intercal Y_k P_k^{-1} Y_k^\intercal \boldsymbol{b}} + \boldsymbol{b}^\intercal \boldsymbol{\nu}_k - \beta_{\mathrm{max}} \le 0 \tag{E.22b}$$

where $\mathrm{normcdf}^{-1}(\cdot)$ is the inverse cumulative distribution function of the normal distribution. Additionally, note that if the disturbances are not assumed to be Gaussian then $\mathrm{normcdf}^{-1}(\cdot)$ can be conservatively replaced using Cantelli's concentration inequality with $Q(1-p) = \sqrt{p/(1-p)}$ [288, 295]. Substitute Eq. (E.17) into Eq. (E.22b) to obtain the relaxed form

$$\mathrm{normcdf}^{-1}(1 - p_k^u)\sqrt{\boldsymbol{b}^\intercal \Phi_k \boldsymbol{b}} + \boldsymbol{b}^\intercal \boldsymbol{\nu}_k - \beta_{\mathrm{max}} \le 0. \tag{E.23}$$

Linearize these constraints around some reference values of P_k and Φ_k, using the tangent line as a linear global overestimator of the square root function:

$$\sqrt{x} \le \frac{1}{2\sqrt{x_r}} x + \frac{\sqrt{x_0}}{2}, \quad \forall x, x_0 > 0 \tag{E.24}$$

As a result of the square roots of P_k and Φ_k, the constraints in Eqs. (E.22a) and (E.23) are nonconvex. A conservative approximation can be obtained by linearizing the constraints around some reference values P_r and Φ_r for a given problem as follows:

$$\text{normcdf}^{-1}(1-p_k^x)\left(\frac{1}{2\sqrt{\boldsymbol{a}^\intercal P_r \boldsymbol{a}}}\boldsymbol{a}^\intercal P_k \boldsymbol{a} + \frac{1}{2}\sqrt{\boldsymbol{a}^\intercal P_r \boldsymbol{a}}\right) + \boldsymbol{a}^\intercal \boldsymbol{\mu}_k - \alpha_{\max} \leq 0 \tag{E.25a}$$

$$\text{normcdf}^{-1}(1-p_k^u)\left(\frac{1}{2\sqrt{\boldsymbol{b}^\intercal \Phi_r \boldsymbol{b}}}\boldsymbol{b}^\intercal \Phi_k \boldsymbol{b} + \frac{1}{2}\sqrt{\boldsymbol{b}^\intercal \Phi_r \boldsymbol{b}}\right) + \boldsymbol{b}^\intercal \boldsymbol{\nu}_k - \beta_{\max} \leq 0 \tag{E.25b}$$

This approximation is convex and, because it is conservative, will always satisfy the true, nonconvex constraints as well [288].

Finally, note that for lower bound chance constraints,

$$\mathcal{P}\left(\boldsymbol{a}_k^\intercal \boldsymbol{x}_k \leq \alpha_{\min}\right) \leq p_k^x \tag{E.26a}$$

$$\mathcal{P}\left(\boldsymbol{b}_k^\intercal \boldsymbol{u}_k \leq \beta_{\min}\right) \leq p_k^u, \tag{E.26b}$$

a similar set of equations applies:

$$-\text{normcdf}^{-1}(1-p_k^x)\left(\frac{1}{2\sqrt{\boldsymbol{a}^\intercal P_r \boldsymbol{a}}}\boldsymbol{a}^\intercal P_k \boldsymbol{a} + \frac{1}{2}\sqrt{\boldsymbol{a}^\intercal P_r \boldsymbol{a}}\right) + \boldsymbol{a}^\intercal \boldsymbol{\mu}_k - \alpha_{\min} \leq 0 \tag{E.27a}$$

$$-\text{normcdf}^{-1}(1-p_k^u)\left(\frac{1}{2\sqrt{\boldsymbol{b}^\intercal \Phi_r \boldsymbol{b}}}\boldsymbol{b}^\intercal \Phi_k \boldsymbol{b} + \frac{1}{2}\sqrt{\boldsymbol{b}^\intercal \Phi_r \boldsymbol{b}}\right) + \boldsymbol{b}^\intercal \boldsymbol{\nu}_k - \beta_{\min} \leq 0 \tag{E.27b}$$

The finite-horizon constrained covariance steering problem can then be written in convex form as below:

$$\min_{P_k, Y_k, \Phi_k, \boldsymbol{\mu}_k, \boldsymbol{\nu}_k} J = \sum_{k=0}^{N-1} \text{tr}(QP_k) + \text{tr}(R\Phi_k) + \boldsymbol{\mu}_k^\intercal Q \boldsymbol{\mu}_k + \boldsymbol{\nu}_k^\intercal R \boldsymbol{\nu}_k \tag{E.28a}$$

where the matrices $Q \succeq 0$ and $R \succ 0$ tune the penalization of the state and control, respectively, and such that, for all $k = 0, \ldots, N-1$,

$$A_k \boldsymbol{\mu}_k + B_k \boldsymbol{\nu}_k + \boldsymbol{c}k - \boldsymbol{\mu}_{k+1} = 0 \tag{E.28b}$$

$$\boldsymbol{\mu}_0 - \boldsymbol{\mu}_i = 0, \quad \boldsymbol{\mu}_N - \boldsymbol{\mu}_f = 0 \tag{E.28c}$$

$$Y_k P_k^{-1} Y_k^\intercal - \Phi_k \preceq 0 \tag{E.28d}$$

$$A_k P_k A_k^\intercal + B_k Y_k A_k^\intercal + A_k Y_k B_k^\intercal + B_k \Phi_k B_k^\intercal + D_k D_k^\intercal - P_{k+1} = 0 \tag{E.28e}$$

$$P_0 - P_i = 0, \quad P_N - P_f \preceq 0 \tag{E.28f}$$

$$\pm\text{normcdf}^{-1}(1-p_k^x)\left(\frac{1}{2\sqrt{\boldsymbol{a}^\intercal P_r \boldsymbol{a}}}\boldsymbol{a}^\intercal P_k \boldsymbol{a} + \frac{1}{2}\sqrt{\boldsymbol{a}^\intercal P_r \boldsymbol{a}}\right) + \boldsymbol{a}^\intercal \boldsymbol{\mu}_k - (\alpha_{\max}|\alpha_{\min}) \leq 0 \tag{E.28g}$$

$$\pm\text{normcdf}^{-1}(1-p_k^u)\left(\frac{1}{2\sqrt{\boldsymbol{b}^\intercal \Phi_r \boldsymbol{b}}}\boldsymbol{b}^\intercal \Phi_k \boldsymbol{b} + \frac{1}{2}\sqrt{\boldsymbol{b}^\intercal \Phi_r \boldsymbol{b}}\right) + \boldsymbol{b}^\intercal \boldsymbol{\nu}_k - (\beta_{\max}|\beta_{\min}) \leq 0 \tag{E.28h}$$

where, here, the notation $(\alpha_{\max}|\alpha_{\min})$ indicates $\alpha_{\max}$ when the first term of the left-hand side is positive and $\alpha_{\min}$ when that term is negative, and likewise for $(\beta_{\max}|\beta_{\min})$.

Appendix F

Notes on Polynomial Chaos Expansion

Polynomial chaos expansion works by representing the QoI $u = u(\boldsymbol{y})$ of a system as a generalized Fourier series expansion in a multi-dimensional polynomial basis orthonormal with respect to the joint probability distribution of the inputs $\boldsymbol{y}$, $\mathcal{P}_{\boldsymbol{y}}$,

$$\hat{u}(\boldsymbol{y}) = \sum_{j=0}^{P} c_j \psi_j(\boldsymbol{y}) \xrightarrow{\text{m.s.}} u(\boldsymbol{y}), \text{ as } P \to \infty, \tag{F.1}$$

where in Eq. (F.1) u has finite variance and the approximation converges in the mean-square sense to the exact QoI value as the number of terms $P+1$ approaches infinity. The Fourier coefficients c_j are given by Eq. (F.2),

$$c_j = \langle u(\boldsymbol{y}) \psi_j(\boldsymbol{y}) \rangle. \tag{F.2}$$

The orthonormal basis functions that form this approximation space are chosen based on the probability measure of the random inputs, $\mathcal{P}_{\boldsymbol{y}}$. Assume here that the random inputs to this problem are d i.i.d. Gaussian random variables; in this case, the basis functions are orthonormal Hermite polynomials [49]. Note that other probability distributions can be represented using other choices of polynomials. Letting $\psi_{j_k}(\boldsymbol{y}_k)$ index these 1D polynomials in $\boldsymbol{y}_k$ by their degree, $j_k = 0, 1, \ldots$ for $k = 1, \ldots, d$, the d-dimensional polynomials are defined by Eq. (F.3),

$$\psi_j(\boldsymbol{y}) = \prod_{k=1}^{d} \psi_{j_k}(y_k). \tag{F.3}$$

Then, the expansion is truncated to a basis of *total order* p, by considering only those indices j_k for which $\sum_{k=1}^{d} j_k \leq p$. With this truncation method, the expansion will have $P+1$ total terms as given by Eq. (F.4), noting that the number of expansion terms then grows exponentially with the number of dimensions d.

$$P + 1 = \frac{(p+d)!}{p!d!} \tag{F.4}$$

It is important to note that the quality of the approximation in Eq. (F.1) depends on the regularity of the QoI $u(\boldsymbol{y})$ with respect to the dispersed inputs $\boldsymbol{y}$. For an infinitely smooth $u(\boldsymbol{y})$ the convergence is exponential, but for non-smooth behavior the convergence can become slow or fail to converge, and is susceptible to Gibbs phenomenon.

The preceding steps provide a method for approximating the QoI, but have not covered how to compute the coefficients c_j beyond providing Eq. (F.2), which may be inefficient or infeasible to compute directly. Two fundamentally different approaches exist for solving for these coefficients: *intrusive* and *non-intrusive* methods. Intrusive methods such as Galerkin projection can be highly efficient [48], but by definition they require modifying the deterministic simulation code for the system of interest. This would also require modification whenever fidelity is added to the model or additional dispersed inputs are included. Thus, non-intrusive methods, which treat the deterministic solver as a black-box, are desirable.

There are a variety of non-intrusive methods to solve for the unknown PCE coefficients [296]. The basic approach is to generate some number of realizations of the QoI, then use these pilot trials to estimate the coefficients. For N samples of the QoI, Eq. (F.1) can be arranged in matrix form as shown in Eq. (F.5). Note that the dimensions of the measurement matrix $\boldsymbol{\Psi}$ are $N \times (P+1)$.

$$\begin{bmatrix} \psi_0(\boldsymbol{y}_1) & \cdots & \psi_P(\boldsymbol{y}_1) \\ \vdots & \ddots & \vdots \\ \psi_0(\boldsymbol{y}_N) & \cdots & \psi_P(\boldsymbol{y}_N) \end{bmatrix} \begin{bmatrix} c_0 \\ \vdots \\ c_P \end{bmatrix} \approx \begin{bmatrix} u(\boldsymbol{y}_1) \\ \vdots \\ u(\boldsymbol{y}_N) \end{bmatrix} \Rightarrow \boldsymbol{\Psi C} \approx \boldsymbol{U} \tag{F.5}$$

Solving the matrix equation in Eq. (F.5) via the standard least squares provides an estimate of the coefficients vector $\boldsymbol{C}$ for an overdetermined problem, where the number of samples exceeds the number of coefficients ($N > (P+1)$). However, because the number of coefficients grows rapidly with the dimensionality as shown in Eq. (F.4), for high-dimensional problems this may require sufficiently many samples that the PCE convergence is no better than standard Monte Carlo, especially if a high truncation order p is required. This makes it desirable to apply methods to solve Eq. (F.5) for the underdetermined problem. In this case, however, there are infinitely many solutions, and the minimum ℓ_2-norm solution becomes unstable under truncation error. This reveals a need for regularization of the problem.

Compressed sensing (also known as compressive sampling) provides one solution to this problem [297, 298, 299]. This approach enforces sparsity of the coefficient vector by applying appropriate norms. Initially, the solution with minimum ℓ_0-norm is considered, where $||\boldsymbol{C}||_0 = \#\{j : c_j \neq 0\}$:

$$\hat{\boldsymbol{C}} = \arg\min_{\boldsymbol{C}} ||\boldsymbol{C}||_0 \text{ s.t. } \boldsymbol{\Psi C} = \boldsymbol{U}. \tag{F.6}$$

However, a number of issues arise here. The optimization is nonconvex, the solution is not always unique, and this minimum is NP-hard to compute. A number of heuristic workarounds have been developed to address the complexity of finding this sparsest approximation. The approach summarized here is convex relaxation via ℓ_1-minimization. This solution seeks instead the minimum ℓ_1-norm solution as defined in Eq. (F.7), where $||\boldsymbol{C}||_1 = \sum_{j=0}^{P} |c_j|$:

$$\hat{\boldsymbol{C}} = \arg\min_{\boldsymbol{C}} ||\boldsymbol{C}||_1 \text{ s.t. } \boldsymbol{\Psi C} = \boldsymbol{U}. \tag{F.7}$$

Eq. (F.7) is now a convex optimization problem, and may be solved using standard linear programs. Under some conditions, the minimum ℓ_1-norm solution is unique and identical to the sparsest solution [300].

In practical implementation, there is truncation error, so quadratic programming can be employed to solve Eq. (F.8) for some tolerance $\delta > 0$. For this study, δ is selected by defining $\delta = \sigma||\boldsymbol{U}||_2$, where $\boldsymbol{U}$ is a vector of N samples of the QoI $\boldsymbol{u}$ and σ is tuned manually.

$$\hat{\boldsymbol{C}} = \arg\min_{\boldsymbol{C}} ||\boldsymbol{C}||_1 \text{ s.t. } ||\boldsymbol{\Psi C} - \boldsymbol{U}||_2 \leq \delta \tag{F.8}$$

A number of tools exist for ℓ_1-minimization. The tool used for generating initial results is the basis pursuit denoise (BPDN) problem solver provided by SPGL1: Spectral Projected Gradient for L1 minimization [301], a Python port of the original MATLAB solver [302] and based on the theory outlined by van den Berg and Friedlander [303].

Once the PCE coefficients are known or estimated, the mean and variance of the QoI can be trivially computed as shown in Eqs. F.9 and F.10.

$$\langle \boldsymbol{u} \rangle \approx \langle \hat{\boldsymbol{u}} \rangle = c_0 \tag{F.9}$$

$$\sigma^2 \approx \sum_{i=1}^{P} c_i^2 \tag{F.10}$$

Further stochastic properties of the QoI such as its cumulative distribution function or probability density function can be estimated by inexpensively sampling Eq. (F.1) now that the coefficients are known. Furthermore, sensitivity analysis is a natural byproduct of a PCE solution, and Sobol indices can be easily determined from the coefficients [304, 305].