A PROBABILISTIC FRAMEWORK TO LOCATE AND TRACK
MANEUVERING SATELLITES

A Dissertation in
Aerospace Engineering
by
Zachary Hall

© 2021 Zachary Hall

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Doctor of Philosophy

August 2021
The dissertation of Zachary Hall was reviewed and approved by the following:

Puneet Singla  
Professor of Aerospace Engineering  
Dissertation Advisor and Committee Chair

Robert G. Melton  
Professor of Aerospace Engineering

Christopher D. Rahn  
Professor of Mechanical Engineering

Jacob Langelaan  
Professor of Aerospace Engineering

Kirk W. Johnson  
Air Force Institute of Technology  
Special Member

Amy R. Pritchett  
Professor of Aerospace Engineering  
Head of the Department of Aerospace Engineering
Abstract

This dissertation develops a novel probabilistic framework for space domain awareness (SDA) with a specific focus on tracking noncooperative maneuvering satellites while accurately accounting for uncertainties in maneuver parameters and sensor observations. The proposed framework consists of a series of interconnected modules to accomplish the tasks of model prediction, sensor tasking, and model-data fusion for accurate tracking and maneuver estimation in a data-sparse environment. An important feature of the developed framework is that it provides accurate confidence intervals along with the point estimates for orbit states and maneuver parameters. Target satellite maneuvers are defined stochastically, and a Taylor series-based method termed higher-order sensitivity matrix (HOSM) method is developed to determine the mapping of stochastic initial conditions, maneuvers, and model parameters to future states, i.e., reachability set. The reachability set represents all possible target states given known control bounds and defines a search space for the target satellite. A systematic reachability set search (RSS) method is developed using a maximum detection likelihood criterion such that a maneuvering target for which tracking custody has been lost may be searched for and located. A higher-order moment matching (HOMM) method is developed to curtail the curse of dimensionality associated with multiple maneuvers. The independence of orbit state and future maneuver parameter is exploited to maintain constant target state dimensionality throughout the developed framework. The HOSM, RSS, and HOMM methods are validated via extensive numerical simulations throughout the dissertation including tracking a maneuvering satellite in cislunar space. Simulation results provides a basis of optimism for the ideas described in this dissertation. The HOSM method is shown to provide an increasingly accurate mapping of stochastic inputs to target reachability set by increasing the order of included sensitivity matrices. The nominal error in HOSM approximation is dependent on simulation parameters like propagation time, dynamic model, and uncertainty bounds; however, in all test case the root mean square (RMS) error between linear sensitivity matrix and fourth-order sensitivity matrix approximation decreases by approximately four
orders of magnitude. Implementing the RSS method illustrates a number of key characteristics, namely that sensor data where the target is not detected are used to improve the estimate of target state and maneuver parameters. The true target state and maneuver is always contained within the estimated probability density function (pdf) of the target, even without detecting the target. This means that sensor data observing vacant regions of space can be used to inform sensor tasking at future timesteps. Using the simulation parameters defined in each test case examined, and randomly selecting 250 realizations of true maneuver parameters, the target was detected within half an orbital period for 100% of cases for a target making a single maneuver in Geosynchronous Equatorial Orbit (GEO), 96% of cases for a target making two maneuvers in GEO, and 96.8% of cases for a target in an $L_1$ cislunar Halo orbit. These results clearly show the utility of the developed framework for accurately tracking a maneuvering satellite in a data-sparse environment and different orbital regimes.
Table of Contents

List of Figures viii
List of Tables xi
Acknowledgments xiii

Chapter 1
Introduction 1
1.1 Space Domain Awareness ....................... 1
1.2 Maneuvering Satellite Tracking ................. 4
1.3 Research Objectives ................................ 5

Chapter 2
Mathematical Background 10
2.1 Introduction ................. 10
2.2 Probability Theory ......................... 11
2.3 Estimation and Filtering ............................ 14
  2.3.1 Uncertainty Propagation ........................ 15
  2.3.2 Measurement Update ........................... 18
    2.3.2.1 Maximum A-Posteriori .................... 18
    2.3.2.2 Best Linear Unbiased Estimator .......... 20
  2.3.3 The Kalman Filter ......................... 24
  2.3.4 Nonlinear Kalman Filtering .................. 26
2.4 Expectation Value Computation ............... 29
  2.4.1 The Monte Carlo Method .................... 30
  2.4.2 Gaussian Quadrature ......................... 32
  2.4.3 The Unscented Transform .................... 36
  2.4.4 The Conjugate Unscented Transform ........ 38
2.5 higher-order Filtering ....................... 40
2.6 Conclusion .................................. 40
# Chapter 3

## Reachability Set Computation

3.1 Introduction ........................................................................... 42
3.2 Problem Description ............................................................... 43
3.3 higher-order Sensitivity Matrix Method ................................. 45
3.4 Spherically Uniform Quadrature ........................................... 48
3.5 Designed Quadrature for Mixed Distributions ....................... 55
   3.5.1 Penalization ................................................................. 58
   3.5.2 Gauss-Newton Iteration ................................................. 58
   3.5.3 Regularization .............................................................. 60
   3.5.4 Initialization and Enriching/Pruning ............................... 62
3.6 Numerical Simulations .......................................................... 63
   3.6.1 Cooperative Maneuvering Satellites ............................... 64
      3.6.1.1 Test Case 1: Two Burn Maneuver .............................. 65
   3.6.2 Noncooperative Targets ................................................ 72
      3.6.2.1 Test Case 3: Single Unknown Maneuver .................. 72
3.7 Conclusion ............................................................................ 78

# Chapter 4

## Reachability Set Search

4.1 Introduction ........................................................................... 79
4.2 Problem Description ............................................................... 81
4.3 Sensor Tasking ...................................................................... 83
4.4 Filtering and Estimation ......................................................... 86
4.5 Numerical Simulations ........................................................... 91
   4.5.1 Test Case 1: Single Maneuver LEO ............................... 93
   4.5.2 Test Case 2: Single Maneuver GEO ............................... 94
   4.5.3 Test Case 3: Two Maneuvers GEO ............................... 103
4.6 Conclusions ........................................................................ 106

# Chapter 5

## Higher-Order Moment Matching

5.1 Introduction ........................................................................... 108
5.2 Problem Description ............................................................... 110
5.3 Higher-Order Moment Computation for Additive Systems .... 111
   5.3.1 HOMM for Measurement Update ................................. 116
5.4 Numerical Simulations ........................................................... 118
   5.4.1 Test Case 1: Multiple Maneuver Reachability Set ........... 118
   5.4.2 Test Case 2: Higher Measurement Update ...................... 124
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.4.3</td>
<td>Test Case 3: Higher-Order Filter</td>
<td>129</td>
</tr>
<tr>
<td>5.5</td>
<td>Conclusion</td>
<td>134</td>
</tr>
<tr>
<td>Chapter 6</td>
<td>Cislunar Applications</td>
<td>135</td>
</tr>
<tr>
<td>6.1</td>
<td>Introduction</td>
<td>135</td>
</tr>
<tr>
<td>6.2</td>
<td>The Circular Restricted Three Body Problem</td>
<td>136</td>
</tr>
<tr>
<td>6.3</td>
<td>Periodic Orbits in the CRTBP</td>
<td>139</td>
</tr>
<tr>
<td>6.4</td>
<td>Numerical Simulations</td>
<td>141</td>
</tr>
<tr>
<td>6.4.1</td>
<td>Test Case 1: Reachability Set Propagation</td>
<td>142</td>
</tr>
<tr>
<td>6.4.2</td>
<td>Test Case 2: Reachability Set Search</td>
<td>144</td>
</tr>
<tr>
<td>6.5</td>
<td>Conclusion</td>
<td>151</td>
</tr>
<tr>
<td>Chapter 7</td>
<td>Conclusion</td>
<td>153</td>
</tr>
<tr>
<td>7.1</td>
<td>Future Work</td>
<td>155</td>
</tr>
<tr>
<td>References</td>
<td></td>
<td>157</td>
</tr>
</tbody>
</table>
List of Figures

1.1 Space Catalog Objects vs Time .......................... 2
1.2 IDA definition of SSA Framework ........................ 3
1.3 Diagram of SDA Framework .............................. 6

2.1 Probability Density and Cumulative Density Functions ............ 13
2.2 Generic Filter Cycle ..................................... 15
2.3 Error in Moment Constraints vs N .......................... 32
2.4 UT points for n=2 ....................................... 37
2.5 A Schematic of CUT Axes ................................. 38
2.6 Comparison of Quadrature Schemes for Same Order of Accuracy 39

3.1 Spherical Coordinate System ............................... 48
3.2 Illustration of Marginalized cdfs ........................... 51
3.3 Unscented Transformation Points for n=3 ...................... 52
3.4 CUT4 Scaling Parameters vs Central Weight ................... 55
3.5 Singular Values of $\tilde{J}$ ................................... 62
3.6 Diagram of Maneuver Geometry in NTW Frame ................. 64
3.7 Test Case 1: Evolution of 4th Order Reachability Set ............ 67
3.8 Test Case 1: $|r_2|$ Approximation % Error. The color of the particle corresponds to the logarithm of % error, i.e., log%ε. The cold blue color represents smaller values ($\approx O(10^{-5})$) for %ε while red color represents higher values ($\approx O(10)$) for %ε. ........................................... 68
3.9 Test Case 2: Diagram of Arrival Geometry ..................... 70
3.10 Test Case 2: Evolution of 4th Order Reachability Set .......... 71
3.11 Test Case 2: $|r_2|$ Approximation % Error. The color of the particle corresponds to the logarithm of % error, i.e., log%ε. The cold blue color represents smaller values ($\approx O(10^{-7})$) for %ε while red color represents higher values ($\approx O(10^{-1})$) for %ε. .................. 73
3.12 Test Case 3: Approximation Errors for Reachability Sets Computed Via Tensor Product and DQ Methods ...................... 75
3.13 Test Case 3: Scatterplot of Reachability Set Position Errors  
3.14 Test Case 3: Position Approximation Error for Varying Order $D$  
3.15 Test Case 3: RMS Position Error (log scale) vs. Time  

4.1 Reachability Set Search Diagram  
4.2 Observer Geometry  
4.3 Noncooperative Satellite Search Method  
4.4 RSW Coordinate System  
4.5 Test Case 1: Reachable Set at $t=1$ hr  
4.6 Test Case 1: State Estimate Error  
4.7 Test Case 1: Final State Distribution  
4.8 Test Case 1: Maneuver Time Estimate Error  
4.9 Test Case 2: $5^{th}$ Order Polynomial Reachability Set at $t=12$ hr  
4.10 Test Case 2: Evolution of Target PDF in Orbital Plane During Search ($5^{th}$ Order Polynomial Approximation)  
4.11 Test Case 2: Detection Time for 250 Simulations and Varying $D$  
4.12 Test Case 2: Maneuver Estimate Error $\epsilon_u$ for 250 Simulations and Varying $D$  
4.13 Test Case 2: Maneuver Time Estimate Error $\epsilon_t$ for 250 Simulations and Varying $D$  
4.14 Test Case 3: $4^{th}$ Order Polynomial Reachability Set at $t=12$ hr  
4.15 Test Case 3: Evolution of Target PDF in Orbital Plane During Search ($4^{th}$ Order Polynomial Approximation)  
4.16 Test Case 3: Maneuver Estimate Errors for 250 Simulations  

5.1 Growth of Sample Points for Sequential Reachability Sets  
5.2 Test Case 1: Nominal Three-Burn Transfer  
5.3 Test Case 1: Normalized Velocity Quadrature Sets for the First Maneuver  
5.4 Test Case 1: Post Maneuver Reachability Sets at Each Maneuver  
5.5 Test Case 1: Reachability Set Contours for $2^{nd}$ Order Polynomial Approximation at Intermediate Timesteps  
5.6 Test Case 2: Normalized X-Position Moments Propagated to 1$^{st}$ Measurement  
5.7 Test Case 2: Propagated Prior Approximated Using UT and CUT Points at 1$^{st}$ Measurement  
5.8 Test Case 2: Posterior Quadrature Sets at 1$^{st}$ Measurement  
5.9 Test Case 2: Posterior Quadrature Sets at 2$^{nd}$ Measurement  
5.10 Test Case 3: Prior and Posterior Distributions for the First Three Timesteps
5.11 Test Case 3: Prior and Posterior Estimation Errors for All Filters vs Time ......................................................... 133

6.1 Circular Restricted Three Body Problem Diagram .................. 137
6.2 Lagrange Points in the CRTBP ........................................... 138
6.3 $L_1$ Halo Orbit Family .................................................... 141
6.4 Initial Target Halo Orbit $A_z = 60,000 km$ .......................... 142
6.5 Test Case 1: Target Reachability Set at $t = P$ ...................... 143
6.6 Test Case 1: Reachability Set Position Error at $t = P/2$ ............ 144
6.7 Test Case 1: Reachability Set Position Error at $t = P$ ............ 145
6.8 Test Case 1: $||\epsilon_r||_2$ vs Time for Varying $D$ ................ 146
6.9 Observer and Target Orbits in CRTBP Coordinates ................. 147
6.10 Test Case 2: Evolution of Target PDF in y-z Plane During Search (5th order approximation) ................................. 148
6.11 Expected Detection Probability vs. Time ............................. 149
6.12 Detection Time Histogram ............................................. 150
6.13 Test Case 2: Histograms of Maneuver Estimation Error at $t_f$ for 250 Simulations .................................................... 151
6.14 Test Case 2: Histograms of State Estimation Error at $t_f$ for 250 Simulations .................................................... 151
List of Tables

2.1 Table of Common Orthogonal Polynomials .......................... 35

3.1 Summary of Input PDFs ............................................. 45
3.2 Moments of Spherically Uniform Distribution ...................... 50
3.3 Test Case 1: Nominal Transfer Orbit Parameters ............... 66
3.4 Test Case 1: Nominal Burn Parameters ............................ 66
3.5 Test Case 1: Nominal Maneuver Uncertainties .................. 66
3.6 Test Case 1: $|\mathbf{r}_2|$ Position Error ......................... 67
3.7 Test Case 1: $\mathbf{r}_2$ Statistical Moments ....................... 69
3.8 Test Case 2: Nominal Orbit Elements ............................. 70
3.9 Test Case 4: Input Uncertainty .................................. 71
3.10 Test Case 2: $\mathbf{r}_2$ Statistical Moments ....................... 71
3.11 Test Case 4: $|\mathbf{r}_2|$ Position Error ......................... 72
3.12 LEO Initial Orbital Elements .................................. 75

4.1 Test Case 1: Initial Target Nominal Orbital Elements ........... 93
4.2 Initial Target Nominal Orbital Elements .......................... 97
4.3 Initial Target Nominal State (ECI Coordinates) ................. 97
4.4 Test Case 2: Observer State at $t=12$hr (RSW Coordinates) ... 98
4.5 Test Case 2: Summary of Quadrature Sets for Single Maneuver Case (n=4) .......................................................... 99
4.6 Test Case 2: Target Maneuver and State Estimate Error at $t_d$ ... 100
4.7 Test Case 2: Target Maneuver and State Estimate Error at $t_f$ ... 100
4.8 Test Case 2: Normalized Target State and Maneuver Estimate Error Summary at $t_f$: Mean Error±1σ ............................. 103
4.9 Test Case 3: Summary of Quadrature Sets for Two Maneuver Case (n=8) .......................................................... 103
4.10 Test Case 3: Normalized Target State and Maneuver Estimate Error Summary at $t_f$ for 4th Order Polynomial Model: Mean Error±1σ . 106
5.1 Test Case 1: Nominal Maneuver Sequence ................. 119
5.2 Test Case 1: Initial Orbital Elements .................... 119
5.3 Test Case 2: Target Initial Orbital Elements ............. 125
5.4 Position RMSE ........................................... 131
5.5 Velocity RMSE ............................................ 131
6.1 RMSE of Maneuver and State Estimates at $t_f$ ............ 151
Acknowledgments

This dissertation is the culmination of several years of work; the completion of which would not have been possible without selfless and continuous support from others along the way. I would like to offer my sincerest thanks to my academic advisor Dr. Singla for all of his help throughout my academic career. Dr. Singla has taught me everything I know about engineering, and without his vast and frequently intimidating understanding of dynamical systems, linear algebra, optimization, probability theory, estimation, filtering, and astrodynamics, I would not be where I am today. To an even greater degree, I would like to thank Dr. Singla for always going above and beyond his duties as an academic advisor to be understanding and accommodating if I was struggling, and always making himself available to myself and other grad students to the greatest extent possible. I would also like to extend my gratitude to the members of my doctoral committee for their guidance and feedback throughout the process of writing my dissertation.

Next I would like to thank my family for all of their love and support throughout the way. To my brother Nate, for always being my closest friend and whether he believes it or not, for being my best role model. To my girlfriend Sophie, for being there to support me when I’m stressed, and for loving me in spite of my objectively terrible sense of humor. And most importantly to my parents, for raising me to be who I am today and instilling in me a lifelong love of learning.

I would also like to thank all of my friends who helped me along the way. To my fellow graduate students, for always being around to collaborate on research and offer help with coursework; and for always being willing to let me win their money in poker. To my friends from growing up of whom there are too many to name, thanks for always supporting my professional aspirations even though you don’t understand what I do. I would also like to give a special thanks to my close friend Troy. Even though you aren’t around to see me finish, I know how proud of me you were and I hope you knew how proud of you I was too.

Lastly, I want to thank my financial supporters who helped fund this dissertation. Thanks to the Air Force Office of Scientific Research (AFOSR) for funding my
early graduate career under joint grants FA9550-15-1-0313 and FA9550-17-1-0088, and to both the SMART scholarship program and the Air Force Research Lab for funding my PhD. The results and conclusions of this work do not necessarily reflect the views of the funding agencies.
Chapter 1  
Introduction

1.1 Space Domain Awareness

Space domain awareness (SDA) is broadly defined as the knowledge of all elements in the space environment including current and predicted physical state, as well as characterization of space objects, the operational environment, human factors, and interpretation of events [1, 2]. A main objective of SDA is to provide decision makers with quantifiable information on which to base threat assessment, deterrence, mitigation, or recovery to ensure the longevity of space assets. According to [3] “SDA encompasses all activities of information tasking, collection, fusion, exploitation, quantification, and extraction to end in credible threat and hazard identification and prediction”.

Steadily increasing utilization of space by government, commercial, and private entities coupled with an anticipated surge of satellite launches over the next decade is driving an increasingly urgent need for efficient and robust SDA. As of April 29, 2021 there are 22,834 cataloged on-orbit objects in the space catalog, the highest number to date. Figure 1.1 depicts the number of on-orbit, decayed, and total cataloged resident space objects (RSOs) since the beginning of the space-era [4]. Tracking and characterizing these roughly 23,000 RSOs currently on-orbit to provide actionable SDA given the limited coverage and availability of ground sensor resources is already a daunting task in the present. Massive commercial high-speed internet mega-constellations like Starlink and Project Kuiper in commonly used low Earth orbit (LEO), are expected to greatly exacerbate the problem of maintaining adequate SDA. [5-8].
In addition to overcrowding in LEO, cislunar space is emerging as a new frontier and is driving the need for advanced satellite tracking methods. NASA’s lunar gateway and exploration goals tied to the Artemis program have driven a keen interest in stable Lagrange point orbits which have the promise of enabling complex mission architectures [9–12]. Other countries have also either already begun, or announced lunar missions for the coming decade including China [13,14], Russia [15,16], India [17], and Japan [18]. This increase in cislunar space traffic poses its own unique set of SDA challenges, especially in the likely scenario of noncooperative satellites from multiple entities operating simultaneously in this domain [19]. The much greater distances to cislunar space than traditional LEO or GEO regimes means that ground-based sensors subject to atmospheric blurring effects need to deal with very low signal-to-noise ratio (SNR), which poses serious difficulties to tracking objects in this region given current technologies. The design of space-based sensor networks to track objects in cislunar space has received much attention recently [20,21], and appears to be a critical technology for enabling operational SDA in the future.

A comprehensive assessment of space situational awareness (SSA) and space traffic management (STM) can be found in a recent report by the Institute for Defense Analyses (IDA) [22]. SSA and SDA are nearly synonymous and will be
used as such without delving into the subtle differences pointed out by [3]. The SDA framework defined in [22] is shown in Figure 1.2. The Space Surveillance Network (SSN) has been tasked to maintain a catalog of space objects for safe operation of satellites and other mission design needs of the USAF. Although, the number of objects in the orbital population (Figure 1.1) that can be tracked will grow abruptly with the deployment of new advanced sensors, (e.g., the Space Fence and DARPA’s Space Surveillance Telescope), there is a need to develop sensing technologies to track smaller objects (<20cm). In addition to DoD’s efforts, NASA’s orbital debris monitoring efforts primarily receive data from four radars [23,24] with three of them located in the Northern hemisphere. The most important limitation of these radars is their location; their higher latitude limits any observations of uncatalogued debris to orbital inclinations greater than about 28 degrees; in addition, debris from objects in a Molniya-type orbit (which is highly elliptical, with an apogee near geosynchronous altitude and a perigee in the Southern Hemisphere) cannot be detected, because most of the U.S. ground-based sensor systems are in the Northern Hemisphere.

Parallelly, this order of magnitude increase in the number of sensed objects will severely impact current manually intensive operational catalog maintenance.
activities. While progress on this problem has been made with the AFRL Rapid Innovation program GEO Odyssey and DARPA’s Orbit Outlook program, SDA capabilities remain far from what is needed for dealing with the future challenges. Effective detection, estimation and characterization of the RSO population and their dynamical interactions with the space environment underpin the realization of the next generation tools for enhanced SSA.

Using the modern day SDA paradigm, targets are assigned a state mean and covariance which are propagated forward in time using a linear extended Kalman filter (EKF) to predict where and when to task a sensor. Propagating mean and covariance using linear methods often introduces significant error from nonlinear dynamic and measurement models, as well as long propagation times due to data sparsity. To account for these errors, non-Gaussian effects must be addressed in filtering. The computational burden of addressing these nonlinear dynamical effects is already a significant problem; however, the problem is further exacerbated by the fact that active satellites are able to make unknown maneuvers. These maneuvers can change the trajectory of target satellites to the point where tracking custody may be lost altogether using traditional means.

1.2 Maneuvering Satellite Tracking

Maneuvering target tracking is a well-studied problem in the literature, with a history dating back to the 1970s [25–27]. In most tracking applications, this involves using real-time measurements to inform the dynamic model, as well as reconstruct or estimate the maneuver as it occurs. A comprehensive introduction to maneuvering target tracking is presented in a six part paper series [28–33] that covers literature up to the early 2000’s. Methods for maneuvering target tracking in this category include decision-based methods [34–36] and multiple-model methods [37–39]. There is a diverse body of existing methods in the literature for tracking maneuvering targets using real-time measurements [34,40–42]; however, it is emphasized here that all the literature cited above assume data-richness, such that measurements of the target are acquired during the target maneuver.

Methods using real-time measurements are useful in air and ground target tracking scenarios where it is reasonable to assume that measurements of the target are available throughout the entire trajectory. Unfortunately, due to the
limited coverage and availability of sensor resources, satellite tracking applications frequently have large time delays between observations on the order of hours or days. In these data-sparse situations, unobserved maneuvers can easily cause a target to be lost.

The problem of detecting and reconstructing maneuvers in data-sparse situations has received some, albeit very limited, coverage in tracking literature. Patera [43] addresses the problem of detecting maneuvers and other events (collisions, reentry, etc) in terms of statistically significant changes in orbital energy. An optimal control-based method has also been developed to reconstruct finite maneuvers [44–46]. The underlying technique for this method was first formulated in 1988 as the minimum model error method [47]. The novelty of the minimum model error method is to treat the control as an unmodeled deviation from the dynamics, and minimize this deviation such that the state estimate is statistically consistent with the observations. When applied to the satellite tracking problem, this method formulates the maneuver detection process as an optimal control problem connecting two sparse measurements (two point boundary value problem) using an assumed minimum fuel control policy. Whenever the optimal control profile rises above the level of system noise, it is assumed that a maneuver has occurred. Although the minimum fuel maneuver is not necessarily a bad assumption, this method does not account for the many sub-optimal trajectories that can explain the same observation data. Furthermore, both the orbital energy method as well as the minimum model error method assume that measurements of the target can be acquired after making a maneuver. One of the primary contributions of this dissertation is the development of methods utilizing sensor data in conjunction with dynamic models and bounds on maneuver parameters to seek and locate a target satellite that has been lost due to unknown maneuvers.

1.3 Research Objectives

A major premise of the research proposed in this dissertation is to formulate the maneuvering satellite tracking problem in a probabilistic framework by defining target state, model parameters, and unknown maneuvers as stochastic variables. In this context, the set of all reachable target states corresponds to the full non-Gaussian pdf resulting from the propagation of all stochastic inputs. Assuming bounds on
input uncertainties are not violated, the true target state will always lie within the reachable set which provides a search area for sensor tasking. Observations of this set can then be used to update the target pdf and inform future sensor tasking.

The SDA framework depicted in Figure 1.3 will be adopted in this dissertation. There are three primary modules in this framework: 1) Model-Data Fusion, 2) Reachability Set Propagation, and 3) Sensor Tasking. The model-data fusion module comprises filtering/estimation, data-association, orbit determination, and detection algorithms. For non-cooperative maneuvering satellites, it also includes the estimation of maneuvering parameters of a satellite. Data association will not be a primary concern in this work, so all measurement data are assumed to be perfectly associated. Estimates from the model-data fusion module are fed into the reachability set propagation module, which uses the current estimates, as well as knowledge of maneuver bounds to predict the future reachable set. RSO reachability sets represent the target search area and are used as input to the sensor tasking module. Sensor tasking is all methods and algorithms that enable the selection of a set of sensor commands for data acquisition. Once the sensors are tasked and data are available, the data are fed back to the model-data fusion module and the cycle is repeated.

The overarching goal of this dissertation is to develop a unified framework for search, detection, tracking and maneuver estimation of a maneuvering satellite.
using available sensor resources and known uncertainty bounds on target inputs. This dissertation will focus on the following specific objectives

1. The first objective is to develop computationally efficient methods to compute probabilistic reachability sets while accounting for uncertainty in model state, maneuver and model parameters.

2. The second objective is to leverage probabilistic reachability sets of a target RSO to task an active space-based sensor.

3. The third objective is to fuse sensor data with reachability sets in a statistically consistent manner to estimate orbit states along with maneuver parameters such as magnitude and maneuver time.

4. The fourth objective is to develop a systematic method to update the target reachability set and extend the computation of reachability sets to multiple maneuvers in a statistically consistent manner while keeping the dimension of the sampling space to be minimal.

5. The fifth objective is to extend the reach of space domain awareness past the traditional GEO belt into the cislunar domain.

6. The fifth and final objective is to demonstrate the utility and performance of developed algorithms through rigorous analysis and simulations. Several simulations depicting actual engineering applications will be considered to demonstrate the efficacy and illustrate the maturity of the ideas described in this dissertation.

The crux of this dissertation research is to lay the foundation for a computationally tractable framework for enhanced SDA. Broadly speaking, the developed tools are anticipated to advance the state-of-the-art in the general area of uncertainty quantification, nonlinear filtering, reachability set computations and sensor tasking in any engineering application. The chapters in this dissertation are structured in the following manner.

Chapter 2 provides a background of mathematical concepts and techniques that will be used to develop the framework identified in Figure 1.3. These concepts include an introduction to basic probability, filtering, estimation, and numerical
integration theory. The topics discussed in this chapter will be utilized and expanded upon in the main body of the dissertation to achieve the objectives stated above.

Chapter 3 presents a generic uncertainty propagation technique known as the higher-order sensitivity matrix (HOSM) method. The HOSM method assumes a Taylor series expansion of the system dynamics, and replaces the explicit dynamic model with a $d^{th}$ order polynomial approximation. Least squares error polynomial coefficients are computed numerically via quadrature techniques. The curse of dimensionality associated with multi-dimensional integration is discussed in detail, and competing high-dimensional quadrature schemes, including sparse-grids, the conjugate unscented transform (CUT), and designed quadrature (DQ) are compared. Quadrature sets for a novel uniformly spherical distribution, to be used later in the dissertation, are derived in detail.

Chapter 4 outlines a Bayesian framework for search and detection of a target within a reachability set. To provide a systematic method for searching the reachability set, this chapter introduces the notion of detection probability. Reachability samples propagated using the HOSM polynomial model are used to determine the sensor parameters which maximize the likelihood of detecting the target. A generic particle filter update procedure is used to update the target satellite reachability set with sensor data. The measurement update is intimately related to the detection probability, and has the critical property of reducing uncertainty in the target search area using measurements of vacant regions of the reachability set. Using the HOSM model, updating the target state estimate is synonymous with updating the estimated maneuver sequence, and the link between the two is discussed. Examples are provided for single-observer, single-target applications.

Chapter 5 presents a higher-order moment matching scheme, which enables non-Gaussian effects to be preserved between target maneuvers and measurement updates. Higher-order moments of the output are computed by separating the expected values of independent variables and employing numerical integration methods. A direct moment matching numerical integration scheme is then used to determine quadrature sets which enable estimation of the pdf. Limitations and drawbacks of real-time implementation are discussed.

Chapter 6 applies the methods previously developed to cislunar scenarios. The cislunar domain is of particular interest for current and future space missions, and the examples in this chapter illustrate potential applications of the framework
proposed by this dissertation to unsolved engineering problems.

Chapter 7 summarizes the contributions of this dissertation, comments on open areas for future research and provides concluding remarks.
Chapter 2  
Mathematical Background

2.1 Introduction

This chapter presents the preliminary mathematics underlying the methods developed in this dissertation. The concepts presented in this chapter include introductory probability theory, filtering and estimation, and numerical integration methods.

All introductory probability concepts are discussed with respect to continuous variables, as is typical in dynamic systems theory, and include the some of the basic properties of probability density functions (pdfs), cumulative density functions (cdfs), joint densities, marginal densities and conditional densities. These definitions are used to derive Bayes rule and form the basis of estimation theory. The concept of expectation values and statistical moments are also introduced.

These probability concepts are extended to various estimation and filtering applications. The filter cycle provides the generic steps involved to estimate a target state given measurement data. The two main steps in this cycle, uncertainty propagation, and measurement update are explored in detail. Uncertainty propagation via the Chapman-Kolmogorov equation (CKE) is discussed, and several measurement update methods are examined including the Bayesian maximum a-posteriori (MAP) approach, and minimum variance formulations. The relationship between the best linear unbiased estimator (BLUE) and linear Kalman filter (KF) is examined, and application of the KF to nonlinear systems is discussed.

Numerical integration methods that enable nonlinear filtering are discussed. A Taylor series expansion approach is used to derive the moment constraint equations (MCEs), and the pros and cons of various quadrature methods that satisfy the
MCEs are discussed. In particular, non-product quadrature methods that apply to high dimensional, high MCE order, are examined. Finally, extension of Kalman filtering theory to include higher-order moment updates is introduced.

The structure of this chapter is as follows: Section 2.1 outlines basic probability concepts and Section 2.2 discusses how these concepts relate to filtering and estimation theory. Section 2.3 discusses existing numerical integration methods with specific focus on computational efficiency, and Section 2.4 provides a conceptual introduction to a generic higher-order moment matching method which will be developed later.

2.2 Probability Theory

One of the main premises of the work in this dissertation is that uncertainty in satellite state, maneuver history, and model parameters can be quantitatively characterized by random variables. That is to say, solutions to many astrodynamics applications can be analyzed in a novel way by substituting continuous random vectors with associated probability distribution functions (pdfs) for the traditional deterministic solution. This section will provide some of the basic definitions and properties of random variables.

There is a duality in probability theory between analogous concepts for discrete and continuous variables. Since all of the dynamical systems applications considered in this dissertation pertain to continuous variables, only random variables with continuous probability density functions (pdfs) will be discussed. Consider an \((n \times 1)\) random vector \(x\) with the domain \(\Omega\). This random vector is prescribed a pdf \(\pi(x)\) with the properties

\[
\pi(x) \geq 0, \forall x \\
\int_{\Omega} \pi(x)dx = 1
\]  

such that the total probability within the domain is 1, and the probability of any state in the domain must be equal to or greater than zero. The pdf describes the relative likelihood that a sample will take on a certain probability value over the
continuous domain. Two very commonly used distributions are the uniform pdf

$$
\pi(x) = \prod_{i=1}^{n} \frac{1}{(b_i - a_i)}, \quad \Omega = [a, b]
$$

(2.2)

where \(a\) and \(b\) are vectors of lower and upper bounds of \(x\), and the Gaussian
distribution

$$
\pi(x) = \frac{1}{\sqrt{(2\pi)^n|\Sigma|}} \exp \left[ -\frac{1}{2}(x - \mu)^T\Sigma^{-1}(x - \mu) \right], \quad \Omega = [-\infty, \infty]
$$

(2.3)

where \(\mu\) and \(\Sigma\) are the mean and covariance of \(x\) respectively. The absolute probability
associated with a given pdf is always defined as an integral of the pdf over some interval of interest

$$
P(a < x < b) = \int_{a_1}^{b_1} \cdots \int_{a_n}^{b_n} \pi(x) dx_1 dx_2 \cdots dx_n
$$

(2.4)

where \(a\) and \(b\) are the interval bounds. As a result, the absolute probability of any infinitesimally small point is equal to zero. The idea of absolute probability is related to the cumulative density function (cdf) \(\Pi(x)\), which is defined as the absolute probability from the lower bound of \(\Omega\) up to \(x\)

$$
\Pi(x) = \int_{a_n}^{x_n} \cdots \int_{a_2}^{x_2} \int_{a_1}^{x_1} \pi(x) dx_1 dx_2 \cdots dx_n
$$

(2.5)

Therefore, the cdf evaluated at the upper bound of \(\Omega\) must fulfill the unity constraint given by (2.1), i.e. \(\Pi(b) = 1\). A plot of the pdf and cdf for normalized Gaussian and Uniform scalar variables is shown in Figure 2.1.

Another important concept in probability theory is the relationship between joint pdf, marginal pdf, and conditional pdf for two random vectors \(x\) and \(y\). The joint pdf \(\pi(x, y)\) describes the likelihood of both \(x\) and \(y\) occurring, the marginalized pdf \(\pi(x)\) describes the likelihood of \(x\) regardless of \(y\), and the conditional pdf \(\pi(x|y)\) describes the likelihood of \(x\) given knowledge of \(y\). A joint pdf can be marginalized by integrating out one of the variables

$$
\pi(x) = \int_{\Omega_y} \pi(x, y) dy
$$

(2.6)
to give the pdf in terms of one variable. The joint probability can be defined in terms of both the conditional and marginalized probability defined as

$$\pi(x, y) = \pi(x|y)\pi(y) = \pi(y|x)\pi(x)$$  \hspace{1cm} (2.7)

These definitions can be used to derive Bayes’ rule where $y$ is a measurement and $x$ is the state. Rearranging (2.7), the probability of $x$ conditioned on $y$ is given by

$$\pi(x|y) = \frac{\pi(y|x)\pi(x)}{\pi(y)}$$  \hspace{1cm} (2.8)

Substituting the definition of the marginalized distribution $\pi(y)$ gives Bayes rule in the following form

$$\pi(x|y) = \frac{\pi(y|x)\pi(x)}{\int_{\Omega} \pi(y|x)\pi(x)dx}$$  \hspace{1cm} (2.9)

Bayes rule has profound implications on filtering and estimation theory. This formula provides a convenient definition of the state pdf conditioned on information from measurement $y$.

An important mathematical operator related to the pdf of a random variable is known as the expected value operator $E[\cdot]$. The expected value is often described to students in passing as the “mean”, but this can be misleading. The definition of the expected value operator is the integral of the argument multiplied by the pdf of $x$ over the domain $\Omega$

$$E[\cdot] = \int_{\Omega} (\cdot)p(x)dx$$  \hspace{1cm} (2.10)
Thus, the expected value is equal to the mean of $x$ if and only if the argument of the expected value is $x$. The expectation values of other polynomials are very important in statistical analysis, and are known as moments. Another commonly used expected value is the centralized, second-order moment, known as the covariance, given by

$$E[(x_i - E[x_i])(x_j - E[x_j])] = \Sigma_{ij}$$  \hspace{1cm} (2.11)

In general, moments can be normalized such that the second-order moment is one and the mean is zero

$$\zeta = \Sigma^{-1}(x - E[x]), \ E[\zeta] = 0, \ E[\zeta^2] = 1$$  \hspace{1cm} (2.12)

Using this normalization, any $d^{th}$-order statistical moment can be computed

$$E[\zeta_{\alpha_1}\zeta_{\alpha_2} \ldots \zeta_{\alpha_m}], \text{ for } \alpha_1, \alpha_2, \ldots \alpha_m = 1, 2 \ldots n$$  \hspace{1cm} (2.13)

Properties of the expectation value stem from the properties of the integral. The expectation operator is linear

$$E[A + B] = E[A] + E[B], \ E[\alpha A] = \alpha E[A]$$  \hspace{1cm} (2.14)

where $A$ and $B$ are random variables, and $\alpha$ is a constant. The expectation value operator is generally non-multiplicative

$$E[AB] \neq E[A]E[B]$$  \hspace{1cm} (2.15)

where $A, B$ are dependent random variables. If $A, B$ are independent, however, the expectation value is multiplicative. This property will become important later in the dissertation.

### 2.3 Estimation and Filtering

This section will extend some of the introductory probability concepts discussed in the previous section to filtering and estimation applications. There are three major components of any filter: 1) prior propagation and 2) measurement update, and 3) time-shifting the posterior to the prior. The majority of the focus will be placed on
the first two steps but the third step will be briefly introduced. The generic filter cycle is depicted in Figure 2.2.

There are a wide range of techniques available to accomplish these steps, several of which will be discussed here. First, the continuous and discrete Kolmogorov equations will be explored for the purposes of uncertainty propagation. Then the measurement update will be discussed in the context of minimum variance estimation and the maximum likelihood estimation. Finally, the idea of higher-order filtering will be introduced to improve accuracy between the posterior to prior time-shift.

2.3.1 Uncertainty Propagation

This section will discuss methods for propagating uncertainty forward in time through a dynamic system. An overview of the Chapman-Kolmogorov equation (CKE) and its application to uncertainty propagation will be provided here; however, a more detailed discussion can be found in [48]. Consider a discrete-time stochastic system

$$\mathbf{x}_{k+1} = \mathbf{\chi}(\mathbf{x}_k, t_k) + \mathbf{\omega}_k$$  \hspace{1cm} (2.16)

where $\mathbf{x}$ is an $(n \times 1)$ state vector, $\mathbf{\omega}$ is an $(n \times 1)$ process noise vector with pdf $\pi_{\omega}(\mathbf{\omega})$, and $\mathbf{\chi}$ is the system flow. The CKE relates the propagated state pdf to the prior state pdf, and is given by

$$\pi(\mathbf{x}_{k+1}) = \int \pi(\mathbf{x}_{k+1} | \mathbf{x}_k) \pi(\mathbf{x}_k) d\mathbf{x}_k$$  \hspace{1cm} (2.17)
where the conditional likelihood \( \pi(x_{k+1}|x_k) \) is defined by

\[
\pi(x_{k+1}|x_k) = \pi_\omega(x_{k} - \chi(x_{k}, t_k))
\]  

(2.18)

The continuous-time differential form of the CKE is known as the Fokker-Planck-Kolmogorov equation (FPKE) or sometimes as the forward-time Kolmogorov equation. If we consider the stochastic differential equation

\[
dx(t) = f(x(t), t)dt + g(x(t), t)d\omega(t)
\]  

(2.19)

where \( x(t) \) is the state, \( f \) is the stochastic drift, \( g \) is the diffusion, and \( \omega(t) \) is a Wiener process, then the FPKE which governs the continuous-time evolution of the pdf \( \pi(x, t) \) is given by

\[
\frac{\partial \pi(x, t)}{\partial t} = -\sum_{i=1}^{n} \frac{\partial}{\partial x_i} [f_i(x, t)\pi(x, t)] + \sum_{i=1}^{n} \sum_{j=1}^{n} \frac{\partial^2}{\partial x_i \partial x_j} [D_{ij}(x, t)\pi(x, t)]
\]  

(2.20)

where \( D \) is a diffusion tensor given by \( D = \frac{1}{2}gg^T \). The CKE and FPKE can be used to propagate uncertainty for discrete and continuous systems respectively; however, solving them for generic systems can be a formidable problem. Under certain simplifications there can be analytical solutions, but in general these equations need to be solved numerically [49–51].

Consider an analytical example for linear systems. Assume we wish to propagate uncertainty in using CKE for a discrete time linear system

\[
x_{k+1} = F_k x_k + \omega_k
\]  

(2.21)

where \( F_k \) is a constant matrix, \( \omega_k \) is zero-mean Gaussian process noise \( \omega_k : \mathcal{N}(0, Q_k) \), and \( x_k \) is a Gaussian prior \( x_k : \mathcal{N}(\mu_k, \Sigma_k) \). The pdfs using this system definition are given by

\[
\pi(x_k) = \frac{1}{(2\pi)^{n/2} \sqrt{\Sigma_k}} exp \left[ -\frac{1}{2} (x_k - \mu_k)^T \Sigma_k^{-1} (x_k - \mu_k) \right]
\]

\[
\pi(x_{k+1}|x_k) = \frac{1}{(2\pi)^{n/2} \sqrt{Q_k}} exp \left[ -\frac{1}{2} (x_{k+1} - F_k x_k)^T Q_k^{-1} (x_{k+1} - F_k x_k) \right]
\]  

(2.22)
Substituting these pdfs into (2.17), leads to the following expression for the propagated prior distribution

\[
\pi(x_{k+1}) = \int \frac{1}{(2\pi)^n \sqrt{\det(Q_k)}} \exp \left[ -\frac{1}{2} \left( (x_k - \mu_k)^T \Sigma_k^{-1} (x_k - \mu_k) + \ldots + (x_{k+1} - F_k x_k)^T Q_k^{-1} (x_{k+1} - F_k x_k) \right) \right] dx_k
\]

(2.23)

By expanding out the exponential term, and grouping in terms of \(x_k\)

\[
-\frac{1}{2} \left( (x_k - \mu_k)^T \Sigma_k^{-1} (x_k - \mu_k) + (x_{k+1} - F_k x_k)^T Q_k^{-1} (x_{k+1} - F_k x_k) \right)
\]

\[
= -\frac{1}{2} \left( x_k^T A x_k - 2b^T x_k + d \right)
\]

(2.24)

where simplifying variables \(A, b, d\) are given by

\[
A = \Sigma_k^{-1} + F_k^T Q_k^{-1} F_k, \quad b = \mu_k \Sigma_k^{-1} + x_{k+1}^T Q_k^{-1} F_k, \quad d = x_{k+1}^T Q_k^{-1} x_{k+1} + \mu_k \Sigma_k^{-1} \mu_k
\]

(2.25)

Separating the constant terms with respect to the integral, the propagated prior can be written

\[
\pi(x_{k+1}) = \frac{1}{(2\pi)^n \sqrt{\det(A)}} \exp \left[ -\frac{1}{2} d \right] \int \exp \left[ -\frac{1}{2} x_k^T A x_k - 2b^T x_k \right] dx_k
\]

(2.26)

and further simplified using the identity

\[
\int \exp \left[ \frac{1}{2} x_k^T A x_k + b^T x_k \right] dx_k = \sqrt{(2\pi)^n |A^{-1}|} \exp \left[ -\frac{1}{2} b^T A^{-T} b \right]
\]

(2.27)

The propagated prior can now be written as

\[
\pi(x_{k+1}) = \frac{(2\pi)^{n/2} \sqrt{|A^{-1}|}}{(2\pi)^n \sqrt{\det(Q_k)}} \exp \left[ -\frac{1}{2} d - \frac{1}{2} b^T A^{-T} b \right]
\]

(2.28)

By re-substitution of \(A, b^T, d\), and application of Sylvester’s determinant theorem [52] to the constant term, the propagated prior can be simplified to the final form
given by
\[
\pi(x_{k+1}) = \frac{1}{(2\pi)^{n/2} \sqrt{|F_k \Sigma_k F_k^T + Q_k|}} \times \exp \left[ -\frac{1}{2} (x_{k+1} - F_k \mu_k)^T (F_k \Sigma_k F_k^T + Q_k)^{-1} (x_{k+1} - F_k \mu_k) \right]
\]

In this form, it is immediately recognizable that the posterior distribution is Gaussian with mean and covariance given by
\[
\begin{align*}
\mu_{k+1} &= F_k \mu_k \\
\Sigma_{k+1} &= F_k \Sigma_k F_k^T + Q_k
\end{align*}
\]

This is a powerful result which indicates that the pdf of a Gaussian distribution remains Gaussian when propagated through a linear system. The following section will discuss the measurement update step in a filter, provide a similar analysis of linear systems, and discuss the extension of both state propagation and measurement update to nonlinear systems.

### 2.3.2 Measurement Update

The measurement update step in a filter uses observed data to improve the estimated system state. Various measurement update methods will be shown for systems under different assumptions. First a generic Bayesian pdf update known as the maximum a-posteriori (MAP) estimator will be discussed, then the best linear unbiased estimator (BLUE) is presented for linear systems. The close relationship between the BLUE and the Kalman filter (KF) is discussed, and lastly the extension of the classical Kalman filter to nonlinear systems is discussed.

#### 2.3.2.1 Maximum A-Posteriori

This section presents a generic Bayesian update, known as maximum a-posteriori (MAP) estimation. This method seeks to determine the state $x$ which maximizes the posterior pdf ($\pi(x|y)$) using Bayes rule. Recalling the definition of Bayes rule
(2.9), the MAP estimator can be posed as

\[
\hat{x} = \max_x \left\{ \pi(x|y) = \frac{\pi(y|x) \pi(x)}{\int_{\Omega} \pi(y|x) \pi(x) dx} \right\}
\]  

(2.31)

Note that since the denominator is a scaling constant it can be removed from the maximization. Additionally, since the logarithm is a monotonically increasing function, maximizing the posterior pdf is identical to maximizing a log-posterior pdf

\[
\hat{x} = \max_x \{ \ln(\pi(x)) + \ln(\pi(y|x)) \}
\]  

(2.32)

In this formulation, the prior distribution acts as a preconditioning term, also known as a regularization. The optimization problem in (2.32) can take various standard forms depending of the pdfs \( \pi(x) \) and \( \pi(y|x) \). Unfortunately, if the prior and likelihood distributions are non-standard, solving for the MAP estimator can be quite difficult and must be done numerically.

Consider a linear measurement model

\[
y = Hx + \nu
\]  

(2.33)

where \( \nu \) is Gaussian measurement noise \( \nu \in \mathcal{N}(0, R) \) and the state \( x \) is also Gaussian \( x \in \mathcal{N}(\mu, \Sigma) \). The measurement likelihood function is given by the pdf of the Gaussian measurement noise \( \pi(y|x) = \pi_\nu(y - Hx) \). The MAP optimization problem for this system can be stated as

\[
\hat{x} = \min_x \left\{ (x - \mu)^T \Sigma^{-1} (x - \mu) + (y - Hx)^T R^{-1} (y - Hx) \right\}
\]  

(2.34)

This cost function can also be written using the 2-norm

\[
\min_x \left\{ \| R^{-1/2} (y - CX) \|_2 + \| \Sigma^{-1/2} (x - \mu) \|_2 \right\}
\]  

(2.35)

which is recognizable as a Tikhanov regularization problem, where the prior pdf serves as the regularizing term. This problem can be solved analytically by taking the partial derivative with respect to \( x \) and setting equal to zero.

\[
\frac{\partial \ln \pi(y|x)}{\partial x} = 2(x - \mu)^T \Sigma^{-1} + 2(Hx - y)^T R^{-1} H = 0
\]  

(2.36)
Solving this for \( x \) yields

\[
(H^T R^{-1} H + \Sigma^{-1}) x = - \Sigma^{-1} \mu + H^T R^{-1} y \\
x = (H^T R^{-1} H + \Sigma^{-1})^{-1} (H^T R^{-1} y + \Sigma^{-1} \mu)
\] (2.37)

This result gives the exact MAP estimate for linear systems with Gaussian pdfs. The following section will discuss the best linear unbiased estimator (BLUE), and the connection between MAP, BLUE, and the Kalman filter.

### 2.3.2.2 Best Linear Unbiased Estimator

An overview on the measurement update for a linear system using the best linear unbiased estimator (BLUE) is presented here; however, more details can be found in [53]. Given a linear measurement model

\[
y = Hx_t + \nu
\] (2.38)

where \( y \) is the \((m \times 1)\) measurement, \( H \) is a constant \((m \times n)\) matrix, \( x_t \) is the true state, and \( \nu \) is zero-mean Gaussian noise with covariance \( R \). Assume that \( \pi(x) \) is unknown, and it is desired to estimate \( x \) as a linear function of \( y \)

\[
x = M y + n
\] (2.39)

where \( M \) is a \((n \times m)\) update matrix, and \( n \) is an \((n \times 1)\) constant vector. The objective of BLUE is to determine the update variables \( n \) and \( M \) such that the estimate of \( x \) minimizes the variance of the updated variable

\[
\min J = \frac{1}{2} E \left[ (x - x_t)(x - x_t)^T \right]
\] (2.40)

such that the estimate \( E[x] \) is unbiased. To define estimation bias, consider expanding equation (2.40)

\[
E \left[ (x - x_t)(x - x_t)^T \right] = E[x]E[x^T] - E[x]E[x_t^T] - E[x_t x_t^T] + E[x_t x_t^T]
\] (2.41)
Adding and subtracting $2E[x]E[x^T]$ terms, the variance can be grouped into two terms

$$E \left[ (x - x_t)(x - x_t)^T \right] = E \left[ (x - E[x])(x - E[x])^T \right] + E \left[ (x_t - E[x])(x_t - E[x])^T \right]$$

Variance, Bias

(2.42)

It can be seen that for an estimate to be unbiased, the expected value of the state must equal the true state $E[x] = x_t$. Substituting (2.38) into (2.39)

$$x = M(Hx_t + \nu) + n$$

$$E[x] = MHx_t + n$$

(2.43)

the unbiased condition can be used to determine the constraints

$$MH = I, \; n = 0$$

(2.44)

Using these constraints, the estimated state is given by $x = x_t + M\nu$. The variance is given by

$$E[xx^T] - E[x]E[x^T] = E[x_t x_t^T] + ME[\nu\nu^T]M^T - x_t x_t^T$$

$$= MRM^T$$

(2.45)

The constraints can now be augmented to the cost function using Lagrange multiplier $\Lambda$, and the BLUE can be found by solving the minimization problem given by

$$\min_{M,\Lambda} J = Tr \left\{ \frac{1}{2} MRM^T + \Lambda(I - MH) \right\}$$

(2.46)

Using the identities

$$\frac{\partial Tr(ABA^T)}{\partial A} = A(B + B^T), \quad \frac{\partial Tr(BAC)}{\partial A} = B^T C^T$$

(2.47)
the partial derivative of the cost function with respect to \( M \) and \( \Lambda \) are set equal to zero

\[
\frac{\partial J}{\partial M} = MR - \Lambda^T H^T = 0
\]

\[
\frac{\partial J}{\partial \Lambda} = I - MH = 0
\]  

(2.48)

Solving these equations for \( \Lambda \)

\[
\Lambda^T = (H^T R^{-1} H)^{-1}
\]  

(2.49)

and substituting back to solve for \( M \) gives

\[
M = (H^T R^{-1} H)^{-1} H^T R^{-1} y
\]  

(2.50)

Therefore the exact optimal update for linear systems in a minimum variance context is given by

\[
x = (H^T R^{-1} H)^{-1} H^T R^{-1} y
\]  

(2.51)

This solution is the well known least squares error solution; however, consider a similar problem where a-priori information is known about the state. Assume that information about the a-priori state \( x_a = x_t + \omega \) is given by

\[
E[x_a] = \mu_a, \quad E[(x_a - \mu_a)(x_a - \mu_a)^T] = \Sigma_a
\]  

(2.52)

and the state is now to be updated linearly using both the measurement and a-priori state

\[
x = My + Nx_a + n
\]  

(2.53)

where the objective is to determine \( M, N \). A similar procedure as the BLUE without a-priori information can be used to determine unbiased conditions

\[
MH + N = I, \quad n = 0
\]  

(2.54)

The updated state can now be written

\[
x = x_t + M\nu + N\omega
\]  

(2.55)
which gives the variance of the updated state as

\[ E[(x - E[x])(x - E[x])^T] = MRM^T + N\Sigma_a N^T \quad (2.56) \]

Therefore the BLUE with a-priori information can be found by solving the minimization problem

\[
\min_{M,N,\Lambda} J = Tr \left\{ \frac{1}{2} MRM^T + \frac{1}{2} N\Sigma_a N^T + \Lambda(I - MH - N) \right\} \quad (2.57)
\]

The partial derivatives of the cost function with respect to the three unknown variables are given by

\[
\frac{\partial J}{\partial M} = MR - \Lambda^T H^T = 0 \\
\frac{\partial J}{\partial N} = N\Sigma_a - \Lambda^T = 0 \\
\frac{\partial J}{\partial \Lambda} = I - MH - N = 0
\]

(2.58)

Solving for the Lagrange multiplier, and substituting to solve for update matrices \( M \) and \( N \) gives

\[
\Lambda^T = (H^T R^{-1} H + \Sigma_a^{-1})^{-1} \\
M = (H^T R^{-1} H + \Sigma_a^{-1})^{-1} H^T R^{-1} \\
N = (H^T R^{-1} H + \Sigma_a^{-1})^{-1} \Sigma_a^{-1}
\]

(2.59)

such that the BLUE with a-priori information is given by

\[
x = (H^T R^{-1} H + \Sigma_a^{-1})^{-1} (H^T R^{-1} y + \Sigma_a^{-1} x_a)
\]

(2.60)

This solution provides an exact minimum variance solution for linear measurement equations given Gaussian a-priori knowledge of the state. Notably, the BLUE solution is identical to the MAP solution found in the previous section for linear systems. This is not true in general, and only holds for the special case of linear systems with Gaussian uncertainty.

In fact, the special case of linear systems with Gaussian uncertainty has further implications. Uncertainty propagation using the CKE for linear dynamics (2.30)
combined with the BLUE measurement update is equivalent to a classical Kalman filter. The following section will summarize the Kalman filter for linear systems and discuss how this filtering framework may be extended to nonlinear systems.

2.3.3 The Kalman Filter

The previous sections provided examples of CKE uncertainty propagation and BLUE/MAP measurement update for linear systems and Gaussian noise distributions. This section will summarize how these examples are equivalent to the Kalman filter, and discuss ways in which the Kalman filtering framework may be applied to nonlinear systems. Consider the linear dynamic and measurement model

\[
x_{k+1} = F_k x_k + \omega_k \\
y_{k+1} = H_{k+1} x_{k+1} + \nu_{k+1}
\]

with Gaussian initial state \( x_k \in \mathcal{N}(\mu_k, \Sigma_k) \), process noise \( \omega_k \in \mathcal{N}(0, Q_k) \), and measurement noise \( \nu_{k+1} \in \mathcal{N}(0, R_{k+1}) \). The propagated prior state mean and covariance can be determined by

\[
\begin{align*}
\mu_{k+1}^- &= E[x_{k+1}^-] = F_k \mu_k \\
\Sigma_{k+1}^- &= E[(x_{k+1}^- - \mu_{k+1}^-)(x_{k+1}^- - \mu_{k+1}^-)^T] \\
&= F_k (E[x_k x_k^T] - E[x_k]E[x_k]^T) F_k^T + E[\omega_k \omega_k^T] \\
&= F_k \Sigma_k F_k^T + Q_k
\end{align*}
\]

Notice that this result is equivalent to (2.30) found through analytical evaluation of the CKE. Assume that the propagated prior state \( x_{k+1}^- \) is updated to the posterior state \( x_{k+1}^+ \) using a linear gain matrix

\[
x_{k+1}^+ = x_{k+1}^- + K_{k+1} (\hat{y}_{k+1} - y_{k+1})
\]
where \( \tilde{y}_{k+1} \) is the measurement data, and \( K_{k+1} \) is the Kalman gain. The posterior mean and covariance are given by

\[
\begin{align*}
\mu_{k+1}^+ &= E[x_{k+1}^+] = \mu_{k+1}^- + K_{k+1}(\tilde{y}_{k+1} - H_{k+1}\mu_{k+1}^-) \\
\Sigma_{k+1}^+ &= E[(x_{k+1}^+ - \mu_{k+1}^+)(x_{k+1}^+ - \mu_{k+1}^+)^T] \\
&= \Sigma_{k+1}^- - K_{k+1}H_{k+1}\Sigma_{k+1}^- H_{k+1}^T K_{k+1}^T + K_{k+1}(H_{k+1}\Sigma_{k+1}^- H_{k+1}^T + R_{k+1})K_{k+1}^T 
\end{align*}
\]

This expression for posterior covariance can be used to solve for Kalman gain using a minimum variance criteria

\[
\min_{K_{k+1}} J = \frac{1}{2} Tr \{ \Sigma_{k+1}^+ \}
\]

\[
\frac{\partial J}{\partial K_{k+1}} = -2\Sigma_{k+1}^- H_{k+1}^T + 2K_{k+1}(H_{k+1}\Sigma_{k+1}^- H_{k+1}^T + R_{k+1}) = 0
\]

\[
K_{k+1} = \Sigma_{k+1}^- H_{k+1}^T (H_{k+1}\Sigma_{k+1}^- H_{k+1}^T + R_{k+1})^{-1} 
\]  

(2.65)

Substituting the Kalman gain back into (2.64) and simplifying, the posterior mean and covariance can be written

\[
\begin{align*}
\mu_{k+1}^+ &= E[x_{k+1}^+] = \mu_{k+1}^- + K_{k+1}(\tilde{y}_{k+1} - H_{k+1}\mu_{k+1}^-) \\
\Sigma_{k+1}^+ &= \Sigma_{k+1}^- - K_{k+1}H_{k+1}\Sigma_{k+1}^- 
\end{align*}
\]

(2.66)

In the previous section it was stated that the BLUE update is equivalent to the linear Kalman filter update. This can be easily shown using equation (2.53) where the a-priori information is considered to be the propagated prior \( x_a = \tilde{x}_{k+1}^- \)

\[
x_{k+1}^+ = M\tilde{y}_{k+1} + Nx_{k+1}^- + n 
\]

(2.67)

Substituting the linear measurement model for \( \tilde{y}_{k+1} = H_{k+1}\tilde{x}_{k+1}^- + \nu_{k+1} \) gives

\[
x_{k+1}^+ = M(H_{k+1}\tilde{x}_{k+1}^- + \nu_{k+1}) + Nx_{k+1}^- + n 
\]

(2.68)

Using the unbiased condition, it can be shown that

\[
E[x_{k+1}^+] = (MH_{k+1} + N)E[x_{k+1}^-] + n = E[x_{k+1}^-] \\
n = 0, \quad N = I - MH_{k+1} 
\]

(2.69)
Resubstituting these values for \( n, N \) into (2.67), it can be easily seen that the BLUE update with a-priori information is identical to the Kalman update

\[
x^+_{k+1} = x^-_{k+1} + M(\hat{y}_{k+1} - H_{k+1}x^-_{k+1})
\]

(2.70)

where \( M = K_{k+1} \) is the Kalman gain. The Kalman filter provides the exact minimum variance optimal solution for a linear system, and is a powerful theoretical tool for estimation and filtering; however, real-world systems are never exactly linear. The following section will discuss how this framework can be extended to nonlinear systems.

### 2.3.4 Nonlinear Kalman Filtering

The classical Kalman filter applies only to linear systems, and although it provides an elegant theoretical framework for minimum variance filters, linear systems do not exist in the real-world applications. Thus, the Kalman filter must be modified to apply to nonlinear systems as well. Consider a generic nonlinear system

\[
x_{k+1} = f(x_k) + \omega_k
\]

\[
y_{k+1} = h(x_{k+1}) + \nu_{k+1}
\]

(2.71)

where \( f(x_k) \) is the dynamic model, \( h(x_{k+1}) \) is the measurement model, \( \omega, \nu \) are zero mean Gaussian noise sequences with covariance \( Q, R \) respectively, and \( x_k \) is a random vector with known pdf \( \pi(x_k) \). The framework for propagating and updating the state mean and covariance are similar between linear and nonlinear Kalman filters; however, for nonlinear systems everything must be left as generic expectation values. Consider propagating the prior mean and covariance

\[
\mu^-_{k+1} = E[f(x_k)]
\]

\[
\Sigma^-_{k+1} = E[f(x_k)f(x_k)^T] - E[f(x_k)]E[f(x_k)^T] + Q_k
\]

(2.72)

Similar to the linear Kalman filter, assume the posterior update is given by the linear gain matrix \( K_{k+1} \)

\[
x^+_{k+1} = x^-_{k+1} + K_{k+1}(\hat{y}_{k+1} - y_{k+1})
\]

\[
\mu^+_{k+1} = \mu^-_{k+1} + K_{k+1}(\hat{y}_{k+1} - E[y_{k+1}])
\]

(2.73)
Using a couple of notation simplifications

\[
\begin{align*}
\Delta x_{k+1}^- &= x_{k+1}^- - \mu_{k+1} \\
\Delta y_{k+1} &= y_{k+1}^- - E[y_{k+1}] \\
\Delta x_{k+1}^+ &= \Delta x_{k+1}^- - K_{k+1} \Delta y_{k+1}
\end{align*}
\] (2.74)

the posterior covariance can be written as

\[
\Sigma_{k+1}^+ = E[\Delta x_{k+1}^+ \Delta x_{k+1}^+ T] \\
= E[\Delta x_{k+1}^- \Delta x_{k+1}^+ T] - K_{k+1} E[\Delta y_{k+1} \Delta x_{k+1}^+ T] \\
- E[\Delta x_{k+1}^- \Delta y_{k+1} T] K_{k+1}^T + K_{k+1} E[\Delta y_{k+1} \Delta y_{k+1} T] K_{k+1}^T
\] (2.75)

This expression can be made more compact by defining variance matrices for each of the expectation value terms in the above expression

\[
\Sigma_{k+1}^+ = \Sigma_{k+1}^- - K_{k+1} \Sigma_{k+1}^{(xy),T} - \Sigma_{k+1}^{(xy)} K_{k+1}^T + K_{k+1} \Sigma_{k+1}^{(yy)} K_{k+1}^T
\] (2.76)

where the matrices \( \Sigma_{k+1}^{(xy)} \) and \( \Sigma_{k+1}^{(yy)} \) are given by

\[
\begin{align*}
\Sigma_{k+1}^{(xy)} &= E[\Delta x_{k+1}^- \Delta y_{k+1} T] \\
&= \mathbb{E} [f(x_k) h^T(x_{k+1})] - \mathbb{E} [f(x_k)] \mathbb{E} [h^T(x_{k+1})] \\
\Sigma_{k+1}^{(yy)} &= E[\Delta y_{k+1} \Delta y_{k+1} T] \\
&= \mathbb{E} [h(x_{k+1}) h^T(x_{k+1})] - \mathbb{E} [h(x_{k+1})] \mathbb{E} [h^T(x_{k+1})] + R_{k+1}
\end{align*}
\] (2.77)

The posterior covariance (2.76) can be minimized with respect to Kalman gain using a procedure similar to that used in the linear Kalman filter. The minimum variance Kalman gain found in this manner is given by

\[
K_{k+1} = \Sigma_{k+1}^{(xy)} \left( \Sigma_{k+1}^{(yy)} \right)^{-1}
\] (2.78)

Substituting (2.78) into (2.76) gives the posterior covariance update

\[
\Sigma_{k+1}^+ = \Sigma_{k+1}^- - K_{k+1} \Sigma_{k+1}^{(xy),T}
\] (2.79)

The prior mean and covariance propagation as well as the mean and covariance measurement update presented here are valid for any nonlinear system.
problem now becomes how to evaluate the expectation values involved for a generic function.

To motivate this discussion, consider a first-order Taylor series expansion on the dynamic and measurement models

\[
\begin{align*}
\mathbf{f}(\mathbf{x}_k) &= \mathbf{f}(\mu_k) + \mathbf{F}_k \Delta \mathbf{x}_k \\
\mathbf{h}(\mathbf{x}_{k+1}) &= \mathbf{h}(\mu_{k+1}) + \mathbf{H}_{k+1} \Delta \mathbf{x}_{k+1}
\end{align*}
\]

where the matrices \( \mathbf{F}_k \) and \( \mathbf{H}_{k+1} \) are the Jacobian matrices of the dynamic and measurement models respectively. Substituting the linearized expression for \( \mathbf{f}(\mathbf{x}_k) \) into equation (2.72), gives the propagated prior mean and covariance as

\[
\begin{align*}
\mu^- &= \mathbf{f}(\mu_k) \\
\Sigma^- &= \mathbf{F}_k \Sigma_k \mathbf{F}_k^T + \mathbf{Q}_k
\end{align*}
\]

Notice that this form is identical to that of the linear Kalman filter (2.62), except that the \( \mathbf{F}_k \) is defined as Jacobian of the dynamics rather than as the linear dynamics themselves. This similarity extends to the measurement update such that the posterior mean and covariance can be computed using measurement model Jacobian \( \mathbf{H}_{k+1} \) in the classical Kalman filter equations (2.66).

Linearizing a generic system in this manner and applying the classical Kalman filter equations is the well-known extended Kalman filter (EKF), and is widely used for filtering nonlinear systems. The EKF extends the applicability of the KF to nonlinear systems; however, if the true dynamic or measurement model equations are highly nonlinear or the timestep \( \Delta t = t_{k+1} - t_k \) is too large, then the first order approximation can quickly cause the accuracy of the expectation values in (2.72), (2.77) to deteriorate, and in some cause the mean and covariance estimates to diverge entirely.

If a first-order approximation is not sufficient to accurately capture the state propagation or update for a given system, the next logical step is to use a higher-order Taylor series expansion to evaluate the expectation values more accurately \([54]\). The premise is simple enough; however manual evaluation of higher-order partial derivative tensors can become burdensome even for moderately high dimension systems. This inconvenience is further exacerbated, or rendered entirely preventative, if there is not an explicit analytical form of the dynamic or measurement
model. An alternative to the analytical Taylor series expansion is to evaluate the expectation values numerically via sampling-based methods. The following section will discuss several sampling-based methods, and describe how they can be used to accurately and efficiently integrate expectation values.

### 2.4 Expectation Value Computation

This section presents various numerical expectation value computation methods, i.e. sampling methods, and provides commentary on the pros and cons of each. Sampling-based methods are powerful tools which enable integration of nonlinear functions in a derivative-free manner. All numerical integration methods are variations of the same basic technique: to approximate the integral of a function as the summation of a finite number of weighted function evaluations

\[
E[f(x)] = \sum_{i=1}^{N} w^{(i)} f(x^{(i)})
\]  

(2.82)

where \( N \) is the number function evaluations, \( x^{(i)} \) are sample points and \( w^{(i)} \) are corresponding weights. The difference between the multitude of existing numerical integration methods lies only in how the points and weights are selected. To motivate a discussion of numerical integration, consider substituting a \( d \)-th order Taylor series expansion into (2.82)

\[
E \left[ f_{j}(x^{*}) + \frac{\partial f_{j}(x^{*})}{\partial x_{\alpha_1}} \delta x_{\alpha_1} + \ldots + \frac{1}{d!} \frac{\partial^d f_{j}(x^{*})}{\partial x_{\alpha_1} \ldots \partial x_{\alpha_d}} \delta x_{\alpha_1} \ldots \delta x_{\alpha_d} \right]
\]

\[
= \sum_{i=1}^{N} w^{(i)} \left( f_{j}(x^{*}) + \frac{\partial f_{j}(x^{*})}{\partial x_{\alpha_1}} \delta x_{\alpha_1}^{(i)} + \ldots + \frac{1}{d!} \frac{\partial^d f_{j}(x^{*})}{\partial x_{\alpha_1} \ldots \partial x_{\alpha_d}} \delta x_{\alpha_1}^{(i)} \ldots \delta x_{\alpha_d}^{(i)} \right)
\]

(2.83)

Noting that the partial derivative terms are constant and therefore independent of the expectation value and summation operators, the above equality can be
re-written as
\[ f_j(x^*)E[1] + \frac{\partial f_j(x^*)}{\partial x_{\alpha_1}}E[\delta x_{\alpha_1}] + \ldots \frac{1}{d!} \frac{\partial^d f_j(x^*)}{\partial x_{\alpha_1} \ldots \partial x_{\alpha_d}}E[\delta x_{\alpha_1} \ldots \delta x_{\alpha_d}] \]
\[
= f_j(x^*) \left[ \sum_{i=1}^{N} w^{(i)} \right] + \frac{\partial f_j(x^*)}{\partial x_{\alpha_1}} \left[ \sum_{i=1}^{N} w^{(i)} \delta x^{(i)}_{\alpha_1} \right] + \ldots \frac{1}{d!} \frac{\partial^d f_j(x^*)}{\partial x_{\alpha_1} \ldots \partial x_{\alpha_d}} \left[ \sum_{i=1}^{N} w^{(i)} \delta x^{(i)}_{\alpha_1} \ldots \delta x^{(i)}_{\alpha_d} \right]
\]
(2.84)

Since the partial derivative terms on each side are equal, the only requirement for the above expression to hold is that the moments in \( \delta x_{\alpha} \) must equal the summation terms in \( w^{(i)} \) and \( \delta x^{(i)}_{\alpha} \). This leads to a set of equations known as the moment constraint equations (MCEs)

\[
\sum_{i=1}^{N} w^{(i)} = E[1]
\]
\[
\sum_{i=1}^{N} w^{(i)} \delta x^{(i)}_{\alpha_1} = E[\delta x_{\alpha_1}]
\]
\[
\sum_{i=1}^{N} w^{(i)} \delta x^{(i)}_{\alpha_1} \delta x^{(i)}_{\alpha_2} = E[\delta x_{\alpha_1} \delta x_{\alpha_2}]
\]
\[
\vdots
\]
\[
\sum_{i=1}^{N} w^{(i)} \delta x^{(i)}_{\alpha_1} \ldots \delta x^{(i)}_{\alpha_d} = E[\delta x_{\alpha_1} \ldots \delta x_{\alpha_d}]
\]
(2.85)

Assuming the expectation values of \( x \) are known, the expectation value \( E[f(x)] \) can be evaluated up to the accuracy of a \( d^{th} \)-order Taylor series by solving for a set of points and weights which satisfy the MCEs. Both random sampling and deterministic sampling techniques will be discussed.

2.4.1 The Monte Carlo Method

The Monte Carlo (MC) method is the traditional method for numerical integration. MC is a random sampling technique where \( N \) points \( x^{(i)} \) are randomly drawn from the pdf \( \pi(x) \) and assigned equal weight.

\[
x^{(i)} \in \pi(x), \quad w^{(i)} = \frac{1}{N}
\]
(2.86)
By randomly sampling in this manner, as $N \to \infty$ the MCEs will be satisfied asymptotically. Using MC, the MCEs are approximated as $N$ increases, but never exactly satisfied. The number of samples required to accurately satisfy the moment constraints depends on several factors including the desired accuracy, state dimension $n$ and the input domain. To illustrate this method a scalar example is given, where the first four moments of a standard zero mean, unit variance Gaussian pdf $x \in \mathcal{N}(0, 1)$ will be computed. The expectation values for a standard Gaussian pdf can be computed analytically

$$E[x] = 0, \ E[x^2] = 1, \ E[x^3] = 0, \ E[x^4] = 3 \quad (2.87)$$

The error $\epsilon$ in each moment constraint is computed as

$$\epsilon_j = \left| E[x^j] - \sum_{i=1}^{N} w(i)x^{(i)j} \right| \quad (2.88)$$

for varying $N$. Figure 2.3 plots $\epsilon_j$ vs. the number of random samples shown on a log scale, and illustrates the convergence of the moment constraints with increasing $N$. Using random sampling the approximation error is proportional to the inverse square root of $N$

$$\epsilon_j \propto \frac{1}{\sqrt{N}} \quad (2.89)$$

This implies that to gain one decimal place of accuracy, the number of MC samples must increase by a factor of 100. Although this method is exceedingly simple to implement, it can quickly become computationally intractable for problems requiring high accuracy due to slow convergence. This problem is exacerbated when the evaluation of a single point $f(x^{(i)})$ is computationally expensive, such as dynamic system propagation or an optimization procedure. The computational cost of numerical integration is directly proportional to the number of function evaluations required, and for this reason the objective is to evaluate the integral to the desired accuracy using the fewest number of points possible. Other random sampling and pseudo-random sampling methods have been developed which provide more favorable convergence properties; however, deterministic sampling methods, known as quadrature methods, are specifically designed to satisfy the MCEs and provide far superior performance. Therefore, the following sections will focus
2.4.2 Gaussian Quadrature

In general, deterministic alternatives to the random sampling scheme are known as quadrature rules. A fundamental assumption of quadrature methods is that the function being integrated can be approximated as a polynomial. If this is not the case, then these methods may prove inaccurate.

There are many quadrature rules formulated to carefully leverage symmetries in standard pdfs such that points and weights can exactly satisfy the MCEs. Perhaps the most famous quadrature method is the Gaussian quadrature rule (GQR), which provides the minimum number of points for scalar \((n = 1)\) variables. As a simple example, consider integration of a third-order scalar polynomial \(f(x)\) with respect to the standard uniform pdf \(x \in \mathcal{U}(-1, 1)\), \(p(x) = \frac{1}{2}\)

\[
\frac{1}{2} \int_{-1}^{1} f(x) dx = \frac{1}{2} \int_{-1}^{1} c_0 + c_1 x + c_2 x^2 + c_3 x^3 \\
= c_0 + \frac{c_2}{3}
\]

\[
(2.90)
\]
Consider attempting to replicate this result using two points \( x_1, x_2 \) with weights \( w_1, w_2 \). The polynomial \( f(x) \) and the derivative \( f'(x) \) evaluated at \( x_1 \) and \( x_2 \) are given by

\[
\begin{align*}
    f(x_1) &= c_0 + c_1 x_1 + c_2 x_1^2 + c_3 x_1^3 \\
    f(x_2) &= c_0 + c_1 x_2 + c_2 x_2^2 + c_3 x_2^3 \\
    f'(x_1) &= c_1 + 2c_2 x_1 + 3c_3 x_1^2 \\
    f'(x_2) &= c_1 + 2c_2 x_2 + 3c_3 x_2^2
\end{align*}
\] (2.91)

solving this system of equations for the coefficients \( c_i \)

\[
\begin{bmatrix}
    c_0 \\
    c_1 \\
    c_2 \\
    c_3
\end{bmatrix} = 
\begin{bmatrix}
    1 & x_1 & x_1^2 & x_1^3 \\
    1 & x_2 & x_2^2 & x_2^3 \\
    0 & 1 & 2x_1 & 3x_1^2 \\
    0 & 1 & 2x_2 & 3x_2^2
\end{bmatrix}^{-1}
\begin{bmatrix}
    f(x_1) \\
    f(x_2) \\
    f'(x_1) \\
    f'(x_2)
\end{bmatrix}
\] (2.92)

and substituting into (2.90) provides an expression for the integral in terms of the \( f(x) \) and \( f'(x) \)

\[
c_0 + \frac{c_2}{3} = \frac{x_2 - x_2^3 + x_1(1 + 3x_2^2)}{(x_1 - x_2)^3} f(x_1) + \frac{x_1^3 - x_1 - x_2(1 + 3x_1^2)}{(x_1 - x_2)^3} f(x_2) \\
- \frac{2x_2 + x_1(1 + 3x_2^2)}{3(x_1 - x_2)^2} f'(x_1) - \frac{2x_1 + x_2(1 + 3x_1^2)}{3(x_1 - x_2)^2} f'(x_2)
\] (2.93)

The fundamental form of numerical integration requires that the integral be replaced by a summation of weighted function evaluations

\[
\frac{1}{2} \int_{-1}^{1} f(x) dx = w_1 f(x_1) + w_2 f(x_2)
\] (2.94)

Therefore, in (2.93) the coefficients of the \( f(x) \) terms must equal the weights, and the coefficients of \( f'(x) \) must equal zero. Solving this system of equations for the points and weights give

\[
x_1 = \frac{1}{\sqrt{3}}, \quad x_2 = -\frac{1}{\sqrt{3}}, \quad w_1 = \frac{1}{2}, \quad w_2 = \frac{1}{2}
\] (2.95)

Thus, the two points and weights given above exactly integrate any polynomial
up to third-order, or approximate the integral of a non-polynomial function up to the accuracy of a third-order Taylor series with respect to a uniform weight function. Using GQR, \( N \) points are required to integrate up to \((2N - 1)^{th}\)-order polynomials. The fundamental theorem for Gaussian quadrature, and a discussion of the methods and algorithms needed to compute \( x_i, w_i \) can be found in [55].

Points \( x_i \) and weights \( w_i \) found using GQR are closely related to orthogonal polynomials. If two polynomials \( \phi_1(x), \phi_2(x) \) are orthogonal with respect to weighting function \( \pi(x) \), then they satisfy the property

\[
\int_{\Omega} \phi_1(x)\phi_2(x)\pi(x)dx = 0
\]

where \( \Omega \) is the support of \( x \). There is a rich theoretical foundation for orthogonal polynomials; however, it will not be discussed in its entirety here. See [56] for additional details on orthogonal polynomials. Orthogonal polynomials can be computed via application of the Gram-Schmidt method [57,58], and in general obey the three-term recursion relation

\[
\phi_{k+1}(x) = (a_k x + b_k)\phi_k(x) - c_k\phi_{k-1}(x)
\]

A useful property of orthogonal polynomials pertaining to Gaussian quadrature is that points \( x_i \) can be found by simply finding the roots of \( \phi_k(x) \), and the weights can be calculated using

\[
w_i = \int_{\Omega} \frac{\phi_k(x)}{(x - x_i)\phi'_k(x_i)}\pi(x)dx
\]

There are many different types of orthogonal polynomials for which this relationship is valid, but two frequently used are Legendre polynomials for the uniform distribution and Hermite polynomials for the Gaussian distribution. Due to the close relationship with orthogonal polynomials, GQR points with respect to a uniform distribution are known as Gauss-Legendre points and GQR points with respect to a Gaussian distribution are known as Gauss-Hermite points. Table 2.1 is adapted from [56] and summarizes a select few orthogonal polynomials, their pdfs, and their recursion relation coefficients.

Gaussian quadrature is a powerful tool that provides the minimal number of
Orthogonal Polynomial  |  Interval  |  $p(x)$  |  $a_k$  |  $b_k$  |  $c_k$
---|---|---|---|---|---
Legendre  |  $[-1, 1]$  |  $\frac{1}{2}$  |  $\frac{2k+1}{k+1}$  |  0  |  $\frac{k}{n+1}$
Hermite  |  $(-\infty, \infty)$  |  $e^{-\frac{x^2}{2}}$  |  1  |  0  |  $k$
Ultraspherical  |  $[-1, 1]$  |  $(1-x^2)^{\frac{\lambda-1}{2}}$  |  $\frac{2k+\lambda}{k+1}$  |  0  |  $\frac{k+2\lambda-1}{k+1}$
Laguerre  |  $[0, \infty)$  |  $e^{-\frac{x}{2}}x^\alpha$  |  $-\frac{1}{k+1}$  |  $\frac{\alpha+n+1}{k+1}$  |  $\frac{k+\alpha}{k+1}$
Chebyshev (First Kind)  |  $(-1, 1)$  |  $(1-x^2)^{-\frac{1}{2}}$  |  2  |  0  |  1
Chebyshev (Second Kind)  |  $[-1, 1]$  |  $(1-x^2)^{\frac{1}{2}}$  |  2  |  0  |  1

Table 2.1. Table of Common Orthogonal Polynomials

points needed to compute polynomial integrals for $n = 1$. Unfortunately, integrating polynomials in higher dimensions becomes significantly more difficult. To integrate a polynomial of the same degree in higher dimensions using Gaussian quadrature, one must construct a grid of quadrature points by taking the tensor product of 1D points. This leads to exponential growth in the necessary number of points (i.e. computational expense) with increasing dimension. GQR using this tensor product approach for an $n$ dimensional, $d = 2N - 1$ degree polynomial, requires $N^n$ Gaussian quadrature points. This means that even a moderately high polynomial degree and system dimensionality, say, $n = 6$ and $N = 5$, requires 15,625 points to compute the integral. The difficulty associated with integrating high dimensional variables due to this exponential growth is often referred to as the curse of dimensionality. Due to the computational cost of numerically integrating high dimensional systems, increasing $n$ can very quickly restrict the accuracy of the solution.

Fortunately, other quadrature methods exist which reduce the number of points needed in higher dimensions. Sparse grid methods [59–61] are a popular alternative, which can produce quadrature rules with far fewer points than full tensor-product methods. Unfortunately, the sparse tensor product procedure can cause weights $w_i$ to be negative, which introduces error into the integral. Ideally, a quadrature method must have the minimal number of points $N$, while maintaining positive weights. The following section will discuss non-product quadrature schemes, namely the unscented transform (UT) and conjugate unscented transform (CUT), which attempt to satisfy these requirements.
2.4.3 The Unscented Transform

The exponential growth incurred by tensor product methods has led to a great amount of effort put towards developing methods which can integrate high dimensional variables. One of the most popular non-product methods is known as the unscented transform and can integrate \( n \) dimensional variables up to third-order accuracy with only linear growth in the number of points [62]. Consider the problem of satisfying up to third-order MCEs of a standard \( n\)-D multivariate Gaussian distribution \( \mathbf{x} \in \mathcal{N}(\mathbf{0}, \mathbf{I}) \)

\[
\begin{align*}
E[1] &= 1 \\
E[x_i] &= 0 \\
E[x_i^2] &= 1, \ E[x_i x_j] = 0 \\
E[x_i^3] &= 0, \ E[x_i^2 x_j] = 0, \ E[x_i x_j x_k] = 0
\end{align*}
\] (2.99)

Notice that due to the symmetry of the distribution, any moments with at least one odd dimension component equal zero. Assume that an \((N \times n)\) set of quadrature points \( \mathbf{X} \) is placed on the principal axes symmetrically such that all points are the same distance from the origin \( r_1 \), and are assigned an \((N \times 1)\) weighting vector \( \mathbf{w} \) where all weights are equal \( w_1 \)

\[
\mathbf{X} = \begin{bmatrix}
  r_1 & 0 & \cdots & 0 \\
  0 & r_1 & \cdots & 0 \\
  \vdots & \vdots & \ddots & \vdots \\
  0 & 0 & \cdots & r_1 \\
- r_1 & 0 & \cdots & 0 \\
  0 & - r_1 & \cdots & 0 \\
  \vdots & \vdots & \ddots & \vdots \\
  0 & 0 & \cdots & - r_1
\end{bmatrix}, \quad \mathbf{w} = \begin{bmatrix}
  w_1 \\
  w_1 \\
  \vdots \\
  w_1
\end{bmatrix}
\] (2.100)

Each row of \( \mathbf{X} \) gives the coordinates of one quadrature point. Figure 2.4 illustrates the placement of UT points on the principal axes for \( n = 2 \). Since the points are placed symmetrically, all odd-order moments are automatically satisfied, so only...
the even-order MCEs must be solved for the distance $r_1$ and weight $w_1$.

$$E[1] = \sum_{\alpha=1}^{N} w_\alpha = 2nw_1 = 1$$

$$E[x_1^2] = \sum_{\alpha=1}^{N} w_\alpha X_{\alpha i}^2 = 2w_1r_1^2 = 1$$

(2.101)

Solving these two equations provides the simple result

$$w_1 = \frac{1}{2n}, \quad r_1 = \sqrt{n}$$

(2.102)

Therefore, using the quadrature set $X$ and weights $w$ to compute integrals will automatically approximate the integral up to the accuracy of a third-order Taylor series. This formulation requires only $N = 2n$ points, and is thus highly efficient with increasing dimension. Sometimes a central weight or tuning parameter are included in the UT formulation, however, these will not be explored here.

Assume we would now like to satisfy up to fifth-order moment constraints using a similar approach. The even moment constraint equations are given by

$$E[1] = 1, \quad E[x_1^2] = 1, \quad E[x_1^4] = 3, \quad E[x_1^2x_2^2] = 1$$

(2.103)

Without going any further, it can be readily seen that quadrature points placed on the principal axes cannot satisfy the cross-moment constraint. In other words, since all components except one equal zero, the constraint $E[x_1^2x_2^2] = 1$ cannot be
satisfied

\[ E[x_i^2x_j^2] = \sum_{\alpha=1}^{N} w_{\alpha} X_{\alpha,i}^2 X_{\alpha,j}^2 = 0 \neq 1 \quad (2.104) \]

This implies that regardless of how \( w_1, r_1 \) are selected, any moment with more than a single dimension cannot be satisfied with points placed on the principal axis. Unfortunately, this fact limits the UT to third-order accuracy, so if better accuracy is required for highly nonlinear problems, the method fails. This shortcoming is the motivation behind the generalized extension of the UT method, the Conjugate Unscented Transformation (CUT) method.

### 2.4.4 The Conjugate Unscented Transform

The CUT method, is an extension of the UT which seeks to satisfy higher-order moment constraints for multi-dimensional inputs [63–67]. Much like the UT, the CUT method seeks to replicate the moments of a distribution by symmetrically placing quadrature points such that odd-order moments are automatically satisfied and even-order moments are used to solve for scaling parameters and weights.

The main difference is that special sets of symmetric axes known as conjugate axes are used to construct quadrature points directly in n-D space as illustrated in Figure 2.5. Points placed along these conjugate axes are able to satisfy cross-moment constraints for multi-dimensional inputs, unlike points placed only along the principal axes like in UT. The specific set of symmetric axes needed is dependent on the order of moment constraint equations to be satisfied.
These CUT points are guaranteed to exactly evaluate multidimensional expectation integrals involving polynomial functions with significantly fewer points than other methods of comparable order. Figure 2.6 shows a comparison of the number of points required for several quadrature methods for similar accuracy, clearly illustrating the reduced growth exhibited by the CUT method. The CUT method has already been implemented successfully in cross-disciplinary fields of study, including: the study of volcanic ash plumes, conjunction analysis, control of glucose in diabetic patients, soil characterization for civil engineering purposes, and analysis of energy harvesters [68–72]. More details about the CUT methodology and its comparison with conventional quadrature rules can be found in [63–67, 73]. Tabulated values for the CUT scaling parameters \( r_i \) and weights \( w_i \) for both Gaussian and uniform pdfs can be found in [67].

CUT along with the other quadrature methods discussed here enable highly efficient and accurate numerical integration without the need to explicitly take partial derivatives. This proves quite beneficial for accurately evaluating the expectation values required to update the mean and covariance in Kalman-type filters. The following section will discuss how this can be extended even further, to update additional moments in what is known as higher-order filtering.

![Figure 2.6. Comparison of Quadrature Schemes for Same Order of Accuracy](image-url)
2.5 higher-order Filtering

The basic procedure of all Kalman filtering techniques discussed up to this point is to update the state mean and covariance using measurement data and redefine the posterior as Gaussian. This is convenient for analytical and sampling methods alike. Since the posterior is always Gaussian, quadrature points can be solved for a normalized Gaussian distribution and linearly scaled to fit any pdf without the need to explicitly resolve the quadrature points. Performing the update in this manner has the major drawback of losing fidelity of the generic posterior after every measurement update, which can cause extreme realizations of the state pdf to be improperly captured by the posterior pdf.

A natural extension to the Kalman filtering approach is to update the higher-order moments of the posterior as well. This is the idea behind the $J^{th}$ moment extended Kalman filter (JMEKF) derived in [54]. The JMEKF analytically expands the dynamic and measurement models using higher-order tensors and computes the measurement update to higher-order moments. Although mathematically valid, evaluation of higher-order moment updates analytically can prove daunting even for even the most algebraically gifted. Furthermore, these analytical expansions are system dependent and must be re-evaluated for every problem considered which can prove tedious.

Using numerical methods, the higher-order moments of the posterior may be computed; however, a major challenge is how to resample quadrature points for the updated non-Gaussian distribution. Later in this dissertation, a higher-order moment update method will be presented in conjunction with a generic quadrature point computation method. This approach will enable higher-order moments of the posterior distribution to be retained in a system independent, fully numerical manner.

2.6 Conclusion

The concepts laid out in this Chapter constitute the foundations of the research contributions developed in this dissertation. The following Chapters will apply and extend the methods discussed here to provide novel methods for analyzing stochastic systems. These methods include reachability set propagation, sensor
tasking to search the reachable space, and extension of Kalman filtering to higher-order moment updates. These generic methods are applicable to any stochastic systems; however, they will be discussed and validated in the context of satellite tracking applications. The next Chapter will develop and discuss reachability set propagation.
Chapter 3 | Reachability Set Computation

3.1 Introduction

This chapter utilizes several of the concepts from Chapter 2 to develop an efficient method for determining the reachable space of a target satellite. The reachability set problem has been investigated in many different contexts, often interpreting the reachability set as an envelope of possible future states generated analytically by some bounded control [74–78]. In this dissertation, a reachability set is defined as the target state pdf resulting from all sources of uncertainty including initial state, control, and model parameters. The uncertainties prescribed to these inputs are not necessarily symmetric, or Gaussian, as is the case in traditional uncertainty propagation. To this end, one of the major contributions of this chapter is the definition of the spherically uniform distribution for modeling unknown maneuvers, and development of quadrature sets for propagating mixed-distribution uncertainty.

Rather than linearizing the dynamic model and propagating covariance, as is frequently done via state transition matrices, stochastic input variables with known pdf are prescribed and propagated through the dynamics. The objective of this chapter is to develop a nonlinear method to map the input pdf to the output pdf by computing arbitrary order sensitivity matrices numerically. This method is termed the higher-order sensitivity matrix (HOSM) method, and involves computing the least squares polynomial coefficients analogous to the higher-order terms of a Taylor series expansion. These polynomial coefficients represent elements of the sensitivity matrices and are synonymous with polynomial chaos expansion coefficients in estimation literature [79–82].
Traditionally in polynomial chaos theory, the computation of higher-order sensitivity coefficients for high dimensional systems via tensor product methods requires an exponentially increasing number of quadrature points with increasing state dimension. To alleviate the computational burden associated with multidimensional expectation integral evaluation, this work utilizes non-product quadrature methods to compute the necessary multidimensional expectation integrals in a computationally attractive manner. The non-product CUT method provides the minimal quadrature points in $n$-D space for symmetric inputs; however, mixed input distributions are often required which re-introduce the need for tensor products. To circumvent this issue, a generic moment matching method known as designed quadrature (DQ) is introduced which enables constrained non-product quadrature points to be computed for any input distribution.

The structure of this chapter is as follows. Section 3.2 provides a mathematical description of the problem including known quantities and the problem objective. Section 3.3 discusses substituting a Taylor series expansion for the true system dynamics, and how to evaluate the coefficients of this expansion via least squares method. Section 3.4 provides a derivation of the spherically uniform distribution including a transformation from linear to spherically uniform coordinates, evaluation of the statistical moments, and development of quadrature points which match these moments. Section 3.5 describes the generic moment matching method designed quadrature (DQ) for computing mixed distributions, and Section 3.6 provides numerical validation of the HOSM method for various reachability set examples.

### 3.2 Problem Description

Consider the continuous-time dynamic system

$$\dot{x} = f(x, t) + g(u, t)$$  \hspace{1cm} (3.1)

where, $x$ is the $(n \times 1)$ target state vector, $u$ is the $(m \times 1)$ target control vector, and $f, g$ are the dynamic and control models respectively. Assume that the continuous control can be approximated by a series of $M$ impulsive maneuvers, $u_i$, and maneuver
times $t_i$, with known bounds.

$$
\mathbf{u} = \{ \mathbf{u}_1, t_1, \mathbf{u}_2, t_2, \ldots, \mathbf{u}_M, t_M \}
$$

$$
\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, t) + \sum_{i=1}^{M} \mathbf{g}(\mathbf{u}_i, t) \delta(t - t_i)
$$

(3.2)

where $\delta(\cdot)$ is the Dirac delta function. The system flow $\mathbf{\chi}$ can be written compactly as

$$
\mathbf{x}_{k+1} = \mathbf{\chi}(\mathbf{x}_0, \mathbf{u}, \mathbf{p}, k+1)
$$

(3.3)

where $\mathbf{p}$ is an $(l \times 1)$ vector of any dynamic or maneuver model parameters with uncertainty. Assume the system input $\mathbf{z}$ is given by all stochastic variables including initial state $\mathbf{x}_0$, model parameters $\mathbf{p}$ and maneuver sequence $\mathbf{u}_i, t_i$

$$
\mathbf{z}^T = [\mathbf{x}_0^T \mathbf{p}^T \mathbf{u}_1^T t_1 \mathbf{u}_2^T t_2 \cdots \mathbf{u}_M^T t_M]
$$

(3.4)

such that $\mathbf{z}$ is a $(n+l+M(m+1)) \times 1$ vector. Assume the initial target state $\mathbf{x}_0$ and model parameters $\mathbf{p}$ are defined as random variables with known pdfs $\pi(\mathbf{x}_0)$, and $\pi(\mathbf{p})$ respectively. Traditionally, target maneuvers are represented by equivalent Gaussian noise applied at each timestep; however, the uncertainty assigned to maneuvers is application dependent. Consider the example of a cooperative satellite with known nominal maneuver. In this case, a Gaussian pdf may be prescribed around the planned nominal maneuver to account for uncertainty in actuator performance or maneuver timing.

If the maneuver is for a non-cooperative satellite however, Gaussian uncertainty does not intuitively make sense. Instead, the maneuvers can be defined as impulsive zero-mean spherically uniform distributions up to maximum radius $\Delta V_{\text{max}}$, denoted by $\mathbf{u}_i \in \mathcal{U}_s(0, \Delta V_{\text{max}})$, and maneuver times can be defined as linearly uniform between two time limits $t_i \in \mathcal{U}(t_a, t_b)$. Defining the maneuvers in this manner imposes assumptions only on the maximum maneuver magnitude, and considers any target attitude, maneuver magnitude, and maneuver time for an unknown maneuver to be equally probable. Computing reachability sets using the statistics of this distribution will be discussed in Section 3.4.

For numerical accuracy, the input vector is typically normalized to a zero mean
vector \( \zeta \)

\[
\zeta = S^{-1}(z - \mu)
\]  

(3.5)

where \( S \) is a block-diagonal scaling matrix dependent on the pdf normalization of each component of \( z \), and \( \mu \) is the augmented mean input vector. Table 3.1 summarizes the various input types and their associated means and scaling matrices. Using the normalized input vector, the system flow \( \chi \) can be rewritten as

\[
x_k = \chi(\zeta, k)
\]  

(3.6)

The system flow represents the mapping of stochastic inputs with joint distribution \( \pi(\zeta) \) onto the reachability set \( \pi(x_k) \). The objective of this chapter is to develop a method to accurately and efficiently compute this mapping.

### 3.3 higher-order Sensitivity Matrix Method

This section develops the HOSM method for reachability set propagation, where the objective is efficient computation of the mapping given by (3.6). Consider a \( d^{th} \) order Taylor series expansion of the system flow

\[
\chi^{(d)}(\zeta) \approx \chi^{(0)}_{\text{Nominal Solution}} + \frac{\partial \chi^{(0)}}{\partial \zeta_{\alpha_1}} \zeta_{\alpha_1} + \frac{1}{2!} \frac{\partial^2 \chi^{(0)}}{\partial \zeta_{\alpha_1} \partial \zeta_{\alpha_2}} \zeta_{\alpha_1} \zeta_{\alpha_2} \ldots + \frac{1}{d!} \frac{\partial^d \chi^{(0)}}{\partial \zeta_{\alpha_1} \partial \zeta_{\alpha_2} \ldots \partial \zeta_{\alpha_d}} \zeta_{\alpha_1} \zeta_{\alpha_2} \ldots \zeta_{\alpha_d}, \quad \text{for } \alpha_1, \alpha_2, \ldots, \alpha_d = 1, 2, \ldots, n
\]  

(3.7)

where \( \zeta \) is the normalized random input variable with known pdf, and the partial derivative terms are constants evaluated at the mean input. Computing the partial derivatives of the system flow with respect to the inputs analytically can be difficult,
or even impossible depending on the dynamic model. Instead of explicitly computing these higher-order partial derivative tensors, we would like to numerically compute an equivalent polynomial model.

It is necessary that the Taylor series expansion contains all permutations of the input vector for each order, so the total number of basis functions $L$ follows the factorial relationship given by (3.8) [83].

$$L = (n+d)C_d = \frac{(n + d)!}{d!n!}$$

It follows naturally then, that the number of basis functions contained in just the $i^{th}$ order follows the relationship given by:

$$b_i = (n+i)C_i - (n+i-1)C_{(i-1)} = \frac{(n + i)! - i(n + i - 1)!}{i!n!}$$

Grouping the constant partial derivatives into $(n \times b_i)$ sensitivity matrices $C_i$ and the $\zeta$ terms into arbitrary $i^{th}$ order $(b_i \times 1)$ basis vectors $\phi_i$, the expansion can be rewritten as,

$$\mathbf{x}(\zeta) \approx C_0 + C_1 \phi_1 + C_2 \phi_2 + \ldots C_d \phi_d$$

These sensitivity matrices and basis vectors can be compiled into the compact polynomial model

$$\mathbf{x} \approx C \phi(\zeta)$$

where $C$ is an $(n \times L)$ matrix of coefficients, and $\phi$ is an $(L \times 1)$ vector of basis functions. Rather than explicitly evaluating the coefficients in terms of the partial derivatives, it is desired to compute them numerically via a least squares minimization procedure.

Least squares is a method for solving an over-constrained system which provides the globally minimum error between a polynomial approximation and a generic nonlinear function. Using index notation, the polynomial model approximation can be rewritten as

$$x_j \approx c_{ji} \phi_i(\zeta)$$

where the approximation error is defined as

$$\epsilon_j = x_j - c_{ji} \phi_i(\zeta)$$
A quadratic error squared cost function can be defined over the domain of the input variable \( \zeta \)

\[
J = \frac{1}{2} \int_{\Omega} \varepsilon_j \varepsilon_j \pi(\zeta) d\zeta = \frac{1}{2} E[\varepsilon_j \varepsilon_j]
\]  

(3.14)

The cost function can be minimized by setting the partial derivative of \( J \) with respect to the unknown coefficients equal to zero.

\[
\frac{\partial J}{\partial c_{jk}} = 0 = E \left[ \varepsilon_j \frac{\partial \varepsilon_j}{\partial c_{jk}} \right]
\]

\[
0 = E[(x_j - c_{ji}) \phi_i(\zeta)(-\phi_k(\zeta))]
\]

\[
E[x_j \phi_k(\zeta)] = c_{ji} E[\phi_i(\zeta) \phi_k(\zeta)]
\]

(3.15)

This can also be written in matrix form, known as the normal equations

\[
A = CB
\]

\[
A_{jk} = E[x_j \phi_k(\zeta)], \quad B_{ik} = E[\phi_i(\zeta) \phi_k(\zeta)]
\]

(3.16)

where \( A \) is an \((n \times L)\) matrix, and \( B \) is an \((L \times L)\) matrix. The coefficients can be computed as

\[
C = AB^{-1}
\]

(3.17)

using any polynomial basis. Least squares coefficients may be computed using (3.17) and arbitrary polynomial basis functions; however, if \( \phi_i \) are specifically selected to be orthogonal with respect to \( \pi(\zeta) \), then \( B \) becomes a diagonal matrix.

\[
B_{ik} = \begin{cases} 
E[\phi_i(\zeta) \phi_i(\zeta)] & \text{for } i = k \\
0 & \text{for } i \neq k 
\end{cases}
\]

(3.18)

A diagonal \( B \) matrix causes the coefficients to become linearly independent, and the solution for the best fit coefficients can be even more efficiently computed as

\[
c_{ji} = \frac{E[x_j \phi_i(\zeta)]}{E[\phi_i(\zeta) \phi_i(\zeta)]}
\]

(3.19)

Computation of elements in \( B \) can be done analytically and offline for any basis and known pdf \( \pi(\zeta) \); however, the expectation values in \( A \) are generic and require
numerical integration methods. The quadrature techniques discussed in Section 2.4 are well studied for both Gaussian or uniform inputs, and can be used if $\pi(\zeta)$ is fully symmetric with respect to one of these distributions.

Unfortunately, the inputs considered for reachability set computation here are not fully symmetric, and may include spherically uniform distributions, which are not covered in the literature. The following sections will discuss the computation of efficient quadrature techniques for spherically uniform and non-symmetric mixed distribution inputs.

### 3.4 Spherically Uniform Quadrature

In the case of a noncooperative maneuvering satellite, the user may have no information about the magnitude and direction of a maneuver. In this case, assuming a maximum maneuver magnitude and modeling the maneuver as spherically uniform is a much more realistic assumption than simply assigning Gaussian uncertainty. This section will derive the spherically uniform distribution, compute its statistical moments, and provide examples of quadrature methods which enable evaluation of reachability coefficients with respect to these maneuvers.

Consider the spherical coordinate system $(\theta, \phi, r)$ given in Figure 3.1 where $\theta$ is the azimuth, $\phi$ is the co-latitude, and $r$ is the radius. The transformation between
Cartesian coordinates \((\zeta_1, \zeta_2, \zeta_3)\) and spherical coordinates is

\[
\begin{align*}
\begin{cases}
\zeta_1 \\
\zeta_2 \\
\zeta_3
\end{cases}
&=egin{cases}
 r \sin(\phi) \cos(\theta) \\
r \sin(\phi) \sin(\theta) \\
r \cos(\phi)
\end{cases}, \\
\begin{cases}
 \theta \\
 \phi \\
r
\end{cases}
&=egin{cases}
 \tan^{-1}(\frac{\zeta_2}{\zeta_1}) \\
 \cos^{-1}(\frac{\zeta_3}{r}) \\
 \sqrt{\zeta_1^2 + \zeta_2^2 + \zeta_3^2}
\end{cases}
\end{align*}
\]

Assume there exist random variables \(\theta, \phi, r\) such that the pdf of a unit sphere is uniformly distributed in Cartesian space \(\pi(\zeta) = c\). The constant \(c\) must be determined such that the pdf integrates to one.

\[
\int_\Omega \pi(\zeta) d\zeta = 1
\]

where \(\Omega\) is the support of a sphere, i.e. \(r < 1\). Firstly, notice that since the domain of the pdf is spherical, the bounds cannot be directly expressed in Cartesian space, therefore a transformation must be applied to map the differential volume element from Cartesian coordinates \(d\zeta\) to spherical coordinates \(drd\theta d\phi\). This transformation is given by the determinant of the Jacobian of Cartesian variables with respect to spherical variables.

\[
|J| = \begin{vmatrix}
\frac{\partial \zeta_1}{\partial r} & \frac{\partial \zeta_1}{\partial \theta} & \frac{\partial \zeta_1}{\partial \phi} \\
\frac{\partial \zeta_2}{\partial r} & \frac{\partial \zeta_2}{\partial \theta} & \frac{\partial \zeta_2}{\partial \phi} \\
\frac{\partial \zeta_3}{\partial r} & \frac{\partial \zeta_3}{\partial \theta} & \frac{\partial \zeta_3}{\partial \phi}
\end{vmatrix} = r^2 \sin(\phi)
\]

Using the differential volume element transformation, the constant \(c\) can be found by directly integrating the expression

\[
1 = c \int_0^\pi \int_0^{2\pi} \int_0^1 r^2 \sin(\phi) dr d\theta d\phi
\]

\[
c = \left[ \int_0^\pi \int_0^{2\pi} \int_0^1 r^2 \sin(\phi) dr d\theta d\phi \right]^{-1} = \frac{3}{4\pi}
\]

where the constant \(c\) can be thought of as normalizing the sphere to unit volume. Therefore, pdf for a uniform sphere is given by

\[
\pi(\theta, \phi, r) = \frac{3}{4\pi} r^2 \sin(\phi)
\]
and the expectation value operator with respect to the uniform spherical distribution can now be defined as

\[ E[f(\zeta)] = \frac{3}{4\pi} \int_0^\pi \int_0^{2\pi} \int_0^1 f(\zeta) r^2 \sin(\phi) drd\theta d\phi \]  

Expression (3.25) can be used to analytically evaluate the moments of the uniform spherical distribution with respect to Cartesian space. These moments are evaluated and listed in Table 3.2. Note that since the uniformly spherical distribution is symmetric, the odd-order moments are all equal to zero.

Assume random samples are to be drawn from the uniform sphere. Since drawing linearly uniform samples is computationally efficient and readily available in most programming languages, two ways to sample the uniform sphere using linear samples will be shown: 1) rejection, and 2) transformation. Rejection is perhaps the more simplistic of the two. To draw samples using rejection, randomly sample the linearly uniform distribution \( \zeta^{(i)} \in \mathcal{U}(-1, 1) \) and reject all samples for which \( r > 1 \). Repeat this process until the desired number \( N \) samples are found and assign equal weights.

Sampling the uniform sphere using transformation is a slightly more elegant method. The cumulative density function of the uniform sphere can be found via (2.5) to be

\[ \Pi(r, \theta, \phi) = \frac{\theta r^3(1 - \cos(\phi))}{4\pi} \]  

and the cdf marginalized with respect to each variable is given by

\[ \Pi_r(r) = r^3, \quad \Pi_\theta(\theta) = \frac{\theta}{2\pi}, \quad \Pi_\phi(\phi) = \frac{(1 - \cos(\phi))}{2} \]  

where these marginalized cdfs are depicted in Figure 3.2. To find a transformation of

<table>
<thead>
<tr>
<th>Moment Value</th>
<th>Moment Value</th>
<th>Moment Value</th>
<th>Moment Value</th>
<th>Moment Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>( E[\zeta^2] )</td>
<td>( \frac{1}{3} )</td>
<td>( E[\zeta^4] )</td>
<td>( \frac{3}{7} )</td>
<td>( E[\zeta^2 \zeta_j^2] )</td>
</tr>
<tr>
<td>( E[\zeta^6] )</td>
<td>( \frac{1}{21} )</td>
<td>( E[\zeta^4 \zeta_j^2] )</td>
<td>( \frac{1}{105} )</td>
<td>( E[\zeta^2 \zeta_j^2 \zeta_k^2] )</td>
</tr>
<tr>
<td>( E[\zeta^8] )</td>
<td>( \frac{1}{315} )</td>
<td>( E[\zeta^4 \zeta_j^2 \zeta_k^2] )</td>
<td>( \frac{1}{385} )</td>
<td>( E[\zeta^2 \zeta_j^2 \zeta_k^2 \zeta_l^2] )</td>
</tr>
<tr>
<td>( E[\zeta^2 \zeta_j^2 \zeta_k^2 \zeta_l^2] )</td>
<td>( \frac{1}{1001} )</td>
<td>( E[\zeta^6 \zeta_j^2 \zeta_k^2] )</td>
<td>( \frac{1}{3003} )</td>
<td>( E[\zeta^4 \zeta_j^2 \zeta_k^2 \zeta_l^2] )</td>
</tr>
</tbody>
</table>

Table 3.2. Moments of Spherically Uniform Distribution
variables, the marginalized cdfs must vary linearly over the domain of the spherical coordinates. This can be done using the uniformly distributed vector \( \zeta' \)

\[
\zeta' \in \mathcal{U}(-1, 1), \quad \zeta' = \begin{cases} 
\zeta'_1 = \frac{\theta}{\pi} - 1 \\
\zeta'_2 = -\cos(\phi) \\
\zeta'_3 = 2r^3 - 1
\end{cases} \tag{3.28}
\]

such that evaluating the cdf at \( \Pi(\zeta' = -1) = 0 \) and \( \Pi(\zeta' = 1) = 1 \). Samples can now be taken directly from the uniform cube \( \zeta'^{(i)} \in \mathcal{U}(-1, 1) \), and transformed to a uniform sphere using

\[
\zeta_1 = -\cos(\pi \zeta'_1) \sqrt{1 - (\zeta'_2)^2} \left( \frac{\zeta_3' + 1}{2} \right)^{1/3} \\
\zeta_2 = -\sin(\pi \zeta'_1) \sqrt{1 - (\zeta'_2)^2} \left( \frac{\zeta_3' + 1}{2} \right)^{1/3} \tag{3.29} \\
\zeta_3 = -\zeta'_2 \left( \frac{\zeta_3' + 1}{2} \right)^{1/3}
\]

There are many quadrature methods that one might consider to match the moments of the spherically uniform distribution \( (n = 3) \). Perhaps the most well-known class of quadrature method, Gaussian quadrature rules, requires only \( N \) points to compute up to \( d = 2N - 1 \) order moments for systems where \( n = 1 \), which is minimal for one dimensional systems [84]. Unfortunately, higher dimensional systems require a tensor product to be taken, which leads to exponential growth in the number of points. Furthermore, the Cartesian coordinates of the spherically uniform distribution are not statistically independent, i.e \( E[\zeta_i^2 \zeta_j^2] \neq E[\zeta_i^2]E[\zeta_j^2] \).
This causes the tensor product of 1D points to incorrectly match moments, and thus be invalid.

The celebrated Unscented Transform (UT) [85–87] is a popular alternative to tensor product methods which avoids exponential growth in $N$ with increasing dimension, and maintains positive weights. The UT symmetrically places points on the principal axes of the input $\zeta$ to match up to $3^{rd}$ order moments while incurring only linear growth in $N$ with dimension. Consider matching up to $3^{rd}$ order moments of the uniform sphere using equally weighted $w_1$ points placed at a distance of $r_1$ on each principal axis. See Figure 3.3 for a schematic of the UT for $n = 3$. Since the points are symmetric, odd-order MCEs are automatically satisfied, and the even order equations are given by

$$E[1] = 1 = 2nw_1, \quad E[\zeta_i^2] = \frac{1}{5} = 2w_1r_1^2$$

which leads to the simple solution

$$w_1 = \frac{1}{2n} = \frac{1}{6}, \quad r_1 = \sqrt{nE[\zeta_i^2]} = \sqrt{\frac{3}{5}}$$

This solution matches up to $3^{rd}$ order moments of $\zeta$ using only 6 points. Furthermore, notice that $r_1 < 1$ lies within the unit radius constraint for a spherical distribution and is therefore a valid solution. Support constraints $\zeta^{(i)} \in \Omega$ become very important when defining higher-order quadrature sets. Notice that if a similar procedure is attempted for $5^{th}$ order moments, the cross-dimension expectation value $E[\zeta_i^2\zeta_j^2]$...
will never be replicated because at least one dimension will always have a zero component. Thus, an alternative method must be used.

The conjugate unscented transform (CUT) method is a higher-order generalization of the UT method specifically designed with the cross-moment problem in mind. The CUT method leverages special symmetric axes to directly construct points in nD space, circumventing the need for a tensor product. The details of the CUT method are discussed thoroughly in Adurthi et al [67]; however, this section will outline application of CUT to the spherically uniform distribution.

Let us consider a set of points with distance $r_1$ and weight $w_1$ on the principal axes, and a set of points with distance $r_2$ and weight $w_2$ on the $c^{(3)}$ axis to satisfy 5th order MCEs. The $c^{(3)}$ axis is a symmetric axis which yields the set of points $\{Z^{(3)}\} = \{[r_2 \ r_2 \ r_2], \ [-r_2 \ r_2 \ r_2], \ [r_2 \ -r_2 \ r_2], \ ... \ [-r_2 \ -r_2 \ -r_2]\}$ with all permutations of negative and positive scaling parameters. Note that there are a total of 6 points on the principal axes constrained such that $r_1 \leq 1$ and 8 points on the $c^{(3)}$ axis constrained such that $r_2 \leq \frac{1}{\sqrt{3}}$. Recalling that the symmetry automatically satisfies the odd-order moments, the even-order MCEs for the uniform sphere are given by

\[
E[1] = 1 = 6w_1 + 8w_2
\]
\[
E[\zeta_i^2] = \frac{1}{5} = 2w_1 r_1^2 + 8w_2 r_2^2
\]
\[
E[c_i^4] = \frac{3}{35} = 2w_1 r_1^4 + 8w_2 r_2^4
\]
\[
E[\zeta_i^2 \zeta_j^2] = \frac{1}{35} = 8w_2 r_2^4
\]

Analytically, these equations can be reduced to expressions for $w_1$, $w_2$, $r_1$

\[
w_1 = \frac{1}{35r_1^4}, \ w_2 = \frac{1}{280r_2^4}, \ r_1^2 = \frac{2r_2^2}{7r_2^2 - 1}
\]

and a characteristic polynomial which is quadratic in $r_2^2$

\[
r_2^4 - \frac{42}{7} r_2^2 + \frac{5}{77} = 0
\]
This equation leads to two positive solutions for the second scaling parameter
\[ r_2 = \left[ \sqrt{\frac{3}{11} + \frac{2}{77}\sqrt{14}}, \sqrt{\frac{3}{11} - \frac{2}{77}\sqrt{14}} \right] \approx [0.6082, 0.4190] \tag{3.35} \]

It is crucial now to notice the role of constraints. The only feasible value \( r_2 \leq \frac{1}{\sqrt{3}} \) is given by \( r_2 = 0.4190 \). Unfortunately, when substituting this solution for \( r_2 \) into the expression for \( r_1 \), a value of \( r_1 = 1.2388 \) is found, which lies outside of the unit radius constraint. Thus, there are no feasible solutions using this set of points.

This example highlights one of the most glaring difficulties with the CUT method. There is no guarantee that a selected set of CUT axes will satisfy the MCEs and support constraints, so selecting the axes is often a guess and check procedure. For example, consider adding a single central point \( \zeta^{(0)} = [0, 0, 0] \) with weight \( w_0 \) to previously examined set. The only MCE that is influenced by this change is
\[ E[1] = 1 = w_0 + 6w_1 + 8w_2 \tag{3.36} \]
which yields the modified characteristic equation
\[ r_4^2 + \frac{42}{70w_0 + 77}r_2^2 + \frac{5}{70w_0 + 77} = 0 \tag{3.37} \]
with solution
\[ r_2 = \left[ \sqrt{\frac{21 + \sqrt{14(4 - 25w_0)}}{7(10w_0 + 11)}}, \sqrt{\frac{21 - \sqrt{14(4 - 25w_0)}}{7(10w_0 + 11)}} \right] \tag{3.38} \]

Further analysis of this solution, shows that real solutions only exist for \( w_0 \leq \frac{4}{25} \); however, selection of the central weight must still allow the unit radius constraint to be satisfied. Figure 3.4 shows plots of \( r_1 \) and \( r_2 \) parameters vs. central weight \( w_0 \). The solid lines represent \( r_1 \), the dotted lines represent \( r_2 \), and red/blue lines represent coupled solutions (from either the plus or minus solution in (3.38)). It can be determined that the minimum central weight with a solution that satisfies the constraints is \( w_0 = 0.0571 \).

If \( r_1 \) is chosen to be fixed at the boundary value \( r_1 = 1 \), then the solution for the remaining parameters can be computed as \( r_2 = 0.4472, w_0 = 0.1143, w_1 = 0.0286, w_2 = 0.0893 \). This solution satisfies up to 5th order moments using
only $N = 15$ points. Finding an analytical solution can be a tedious process even for 5th order MCEs, and the complexity of extending analytical solutions to higher $n$ and $d$ can render analytical solutions impossible. Numerical solutions using CUT methodology can be found; however, selecting axes which satisfy the MCEs and support constraints while providing minimum $N$ is very difficult.

The CUT method can be used if only a single unknown maneuver with known time is used as the input, however, the generic input considered in this work (3.4) is mixed rather than solely spherically uniform. This introduces another problem. All of the previously discussed methods implicitly assume fully-symmetric input, which is problematic when constructing non-product quadrature sets for a mixed distribution. The following section will discuss a generic moment matching approach known as designed quadrature to compute quadrature sets for mixed distributions.

### 3.5 Designed Quadrature for Mixed Distributions

A method for generic moment matching known as designed quadrature (DQ) will be outlined here; however, more details can be found in the original paper [88]. Most quadrature methods that have been developed are specific to standard distribution; however, inputs may be prescribed a variety of different uncertainty bounds in the generic reachability sets considered in this dissertation. Fortunately, statistical moments of mixed distributions may be easily computed analytically. If there is a mixed variable, $\zeta$ for which $\zeta_1$ and $\zeta_2$ have different pdfs, moments between the two can simply be computed by separating expected values

$$
E[\zeta_1^2 \zeta_2^4] = E[\zeta_1^2]E[\zeta_2^4]
$$

(3.39)
Assume a set of $L$ normalized moments up to $d^{th}$ order have been computed for a mixed random vector. The set of moment constraint equations using $N$ quadrature points is given by

$$
E[1] = \sum_{i=1}^{N} w_i \\
E[\zeta_x] = \sum_{i=1}^{N} w_i \zeta_x^{(i)} \\
E[\zeta_x \zeta_y] = \sum_{i=1}^{N} w_i \zeta_x^{(i)} \zeta_y^{(i)} \\
\vdots \\
E[\zeta_x \zeta_y \ldots \zeta_\gamma] = \sum_{i=1}^{N} w_i \zeta_x^{(i)} \zeta_y^{(i)} \ldots \zeta_\gamma^{(i)}
$$

(3.40)

The objective of this section is to determine a method for computing an $(N \times n)$ set of points $X$ and $(N \times 1)$ vector of weights $w$ which match a generic set of moments.

$$
X = \begin{bmatrix}
\zeta_x^{(1)T} \\
\zeta_x^{(2)T} \\
\vdots \\
\zeta_x^{(N)T}
\end{bmatrix}, \quad w = \begin{bmatrix}
w_1 \\
w_2 \\
\vdots \\
w_N
\end{bmatrix}
$$

(3.41)

Each row of $X$ corresponds to a quadrature point. The moment constraint equations are linear in weights, so they can be written as a linear system

$$
\begin{bmatrix}
E[1] \\
E[\zeta_x] \\
E[\zeta_x \zeta_y] \\
\vdots \\
E[\zeta_x \zeta_y \ldots \zeta_\gamma]
\end{bmatrix} = \begin{bmatrix}
1 & 1 & \ldots & 1 \\
\zeta_x^{(1)} & \zeta_x^{(2)} & \ldots & \zeta_x^{(N)} \\
\zeta_x^{(1)} \zeta_y^{(1)} & \zeta_x^{(2)} \zeta_y^{(2)} & \ldots & \zeta_x^{(N)} \zeta_y^{(N)} \\
\vdots & \vdots & \ddots & \vdots \\
\zeta_x^{(1)} \zeta_y^{(1)} \ldots \zeta_\gamma^{(1)} & \zeta_x^{(2)} \zeta_y^{(2)} \ldots \zeta_\gamma^{(2)} & \ldots & \zeta_x^{(N)} \zeta_y^{(N)} \ldots \zeta_\gamma^{(N)}
\end{bmatrix} \begin{bmatrix}
w_1 \\
w_2 \\
\vdots \\
w_N
\end{bmatrix}
$$

(3.42)

which can be expressed in compact form as

$$
R = Z(X)w - E
$$

(3.43)
where $E$ is the $(L \times 1)$ vector of moments, $Z$ is an $(L \times N)$ matrix of polynomials evaluated at all quadrature points, $w$ is an $(N \times 1)$ vector of weights, and $R$ is the $(L \times 1)$ error vector. For notation simplicity, polynomial moments are denoted as

$$
\tau_i(\zeta) = \prod_{p=1}^{n} \zeta_{ip}^{\Lambda_{ip}}
$$

where each row of the $(L \times n)$ indexing matrix $\Lambda$ contains the exponents applied to each dimension of a given polynomial. Therefore, the $Z(X)$ matrix can be written as

$$
Z(X) = \begin{bmatrix}
\tau_1(\zeta^{(1)}) & \tau_1(\zeta^{(2)}) & \cdots & \tau_1(\zeta^{(N)}) \\
\tau_2(\zeta^{(1)}) & \tau_2(\zeta^{(2)}) & \cdots & \tau_2(\zeta^{(N)}) \\
\tau_3(\zeta^{(1)}) & \tau_3(\zeta^{(2)}) & \cdots & \tau_3(\zeta^{(N)}) \\
\vdots & \vdots & \ddots & \vdots \\
\tau_L(\zeta^{(1)}) & \tau_L(\zeta^{(2)}) & \cdots & \tau_L(\zeta^{(N)}) 
\end{bmatrix}
$$

A cost function $g$ can be defined as the MCE error squared

$$
g = ||R||_2^2 = R^T R
$$

If all decision variables, i.e. points $\zeta^{(i)}$ and weights $w^{(i)}$, are collected into an $(N_d \times 1)$ vector $d$, an optimization problem can be posed

$$
\min_d g, \text{ subject to: } C(d)
$$

where $g$ is the objective function, and $C$ is a vector of constraints. Constraints can be state as positive weights $w^{(i)} > 0$, and support constraints $\zeta^{(i)} \in \Omega$. Unfortunately, the above problem is a difficult non-linear constrained optimization problem. Furthermore, since the cost vector $R$ is comprised of polynomial moments varying many orders of magnitude, the problem is often very poorly conditioned, leading to numerical issues. To circumvent these issues, the above problem will be converted into a series of regularized unconstrained optimization problems.

There are four main components to the DQ method which will be addressed sequentially 1) Penalization, 2) Gauss-Newton Iteration, 3) Regularization, and 4) Initialization, Enriching, and Pruning.
3.5.1 Penalization

To convert the constrained problem to unconstrained optimization, the constraint violation is added as a penalty to the cost function. Assume $P(d)$ is the $\left(N_l \times 1\right)$ constraint penalty where $P_i(d)$ is given by either equality or inequality constraints

$$C_i(d) = 0 \rightarrow P_i(d) = C_i(d)^2$$
$$a \leq C_i(d) \leq b \rightarrow P_i(d) = (max[0, C_i(d) - b, a - C_i(d)])^2$$

(3.48)

The constraint-penalized cost function can be written as

$$G = \|\tilde{R}_k\|^2 = \|R_k\|^2 + c_k^2 \sum_{i=1}^{N_l} P_i(d)$$

(3.49)

where $c_k$ is a constant factor, and the $((L + N_l) \times 1)$ augmented cost vector $\tilde{R}_k$ is given by

$$\tilde{R}_k = \begin{bmatrix} R_k \\ c_k P \end{bmatrix}$$

(3.50)

c_k is defined using the adaptive rule.

$$c_k = max \left[A, \frac{1}{\|R\|^2} \right]$$

(3.51)

such that as the MCE violation decreases, the penalty placed on constraints increases. $A$ is a constant meant to be large ($A = 1000$ is recommended) which enforces a minimum constraint penalty even if initially the MCE violation is very large. Therefore, as a solution converges and $\|R\|^2$ is driven to zero, the penalty due to violated constraints will increase and drive the constraints to be satisfied.

Standard unconstrained optimization tools can now be used to solve for the decision variables which satisfy the cost function $G$. The Gauss-Newton method is used to update the decision variables such that the augmented cost function $\tilde{R}$ is minimized.

3.5.2 Gauss-Newton Iteration

Gauss-Newton method is a well-known unconstrained minimization method which updates the decision vector $d$ using the Jacobian of the cost function. Assume a
linear update $\Delta \mathbf{d}$ must be found which minimizes $\tilde{R}$ the post update cost function.

$$ \mathbf{d}_{k+1} = \mathbf{d}_k - \Delta \mathbf{d} \quad (3.52) $$

Consider a first order Taylor series expansion of the augmented cost vector

$$ \tilde{R}_{k+1} = \tilde{R}_k - \tilde{J}_k \Delta \mathbf{d} \quad (3.53) $$

where $\tilde{J}$ is the $((L + N_l) \times N_d)$ Jacobian matrix of the augmented cost function with respect to the decision variable vector

$$ \tilde{J} = \frac{\partial \tilde{R}_k}{\partial \mathbf{d}} = \begin{bmatrix} \frac{\partial R}{\partial \mathbf{d}} \\ c_k \frac{\partial P}{\partial \mathbf{d}} \end{bmatrix} \quad (3.54) $$

Assume the decision vector $\mathbf{d}$ is compiled in the following manner

$$ \mathbf{d} = [\zeta_1^{(T)} \zeta_2^{(T)} \cdots \zeta_n^{(T)} \mathbf{w}^T]^T \quad (3.55) $$

where $\zeta_i^{(l)}$ is the $i^{th}$ column of $\mathbf{X}$. The Jacobian of the $\tilde{R}$ with respect to $\mathbf{d}$ can be further partitioned as

$$ \tilde{J} = \frac{\partial \tilde{R}}{\partial \mathbf{d}} = \begin{bmatrix} \frac{\partial R}{\partial \zeta_1} & \frac{\partial R}{\partial \zeta_2} & \cdots & \frac{\partial R}{\partial \zeta_n} \\ c_k \frac{\partial P}{\partial \zeta_1} & c_k \frac{\partial P}{\partial \zeta_2} & \cdots & c_k \frac{\partial P}{\partial \zeta_n} \end{bmatrix} \quad (3.56) $$

The partial derivative of MCE error with respect to points are simply the derivative of the polynomial functions $\tau_i(\zeta)$.

$$ \frac{\partial R_i}{\partial \zeta_k^{(l)}} = \frac{\partial \tau_i(\zeta^{(l)})}{\partial \zeta_k^{(l)}} w^{(j)} = \frac{\partial}{\partial \zeta_k^{(l)}} \left[ w^{(j)} \prod_{p=1}^{n} \zeta_p^{(l)} \right] $$

$$ = \begin{cases} 0 & \text{for } j \neq l \\ w^{(l)} \lambda_{i,k}^{\Lambda_{i,k-1}(l)} \prod_{p} \zeta_p^{\Lambda_{i,p}(l)} & \text{for } j = l, \ p = [1, \ldots k-1, k+1 \ldots n] \end{cases} \quad (3.57) $$
and the partial derivative of MCE error with respect to weights is even simpler and can be shown in matrix form to be

$$\frac{\partial \mathbf{R}}{\partial \mathbf{w}} = \frac{\partial}{\partial \mathbf{w}} [\mathbf{Z}(\mathbf{X})\mathbf{w} - \mathbf{E}] = \mathbf{Z}(\mathbf{X})$$ \hspace{1cm} (3.58)

The Jacobian terms with respect to the constraints are obviously dependent on the constraints themselves; however, since weight constraints $P_i^{(w)}$ are always included they can be shown to be

$$P_i^{(w)} = \max(0, -w_i)^2$$

$$\frac{\partial P_i^{(w)}}{\partial w_j} = \begin{cases} 0 & \text{for } j \neq i \\ -2\max(0, -w_j) & \text{for } j = i, \end{cases} \quad \frac{\partial P_i^{(w)}}{\partial \zeta_j^{(k)}} = 0$$ \hspace{1cm} (3.59)

One the Jacobian has been computed, the objective is to minimize expression (3.53). Using this first order expansion, the minimization can be posed as

$$\min_{\Delta \mathbf{d}} ||\mathbf{\tilde{R}}_k - \mathbf{\tilde{J}}_k \Delta \mathbf{d}||_2^2$$ \hspace{1cm} (3.60)

which is easily recognizable as a least squares problem. The classical solution to the problem is given by

$$\Delta \mathbf{d} = (\mathbf{\tilde{J}}_k^T \mathbf{\tilde{J}}_k)^{-1} \mathbf{\tilde{J}}_k^T \mathbf{\tilde{R}}_k$$ \hspace{1cm} (3.61)

Unfortunately, because the augmented cost vector $\mathbf{\tilde{R}}$ contains values that vary many orders of magnitude, the Jacobian $\mathbf{\tilde{J}}$ is an ill-conditioned matrix. This causes significant error to accumulate when computing the Gauss-Newton step via (3.61). To prevent numerical error, a critical regularization step is required to find a more reasonable solution.

### 3.5.3 Regularization

Regularization is a well-studied approach to solve poorly conditioned problems with a very diverse body of literature on regularization methods [89–91]; however, for the purposes of this dissertation, only a surface-level description will be discussed. Tikhanov regularization, which penalizes the 2-norm of the vector being optimized, will be used. The regularized decision vector update $\Delta \mathbf{d}_\lambda$ can be found by solving
the penalized least squares problem

$$\Delta d_{\lambda} = \arg \min_{\Delta d} \{ ||\tilde{R} - \tilde{J}\Delta d||^2_2 + \lambda ||\Delta d||^2_2 \}$$  \hspace{1cm} (3.62)$$

where $\lambda$ is a regularization parameter which has a large impact on the quality of the solution. The objective of regularization is to filter out extreme singular values via appropriate selection of regularization parameter $\lambda$. Consider the singular value decomposition (SVD) of the augmented Jacobian

$$\tilde{J} = U\Sigma V^T = \sum_{i=1}^{K} u_i\sigma_i v_i^T$$  \hspace{1cm} (3.63)$$

where $K = \min[L + N_l, N_d]$, $U$ and $V$ are matrices of left and right singular vectors respectively, and $\Sigma$ is a matrix with $K$ singular values on the diagonal. Substituting the SVD of $\tilde{J}$ into (3.61) and rearranging gives the least squares error decision variable update as

$$\Delta d = \sum_{i=1}^{K} \frac{u_i^T \tilde{R} v_i}{\sigma_i^2}$$  \hspace{1cm} (3.64)$$

The regularized least squares solution using the Tikhonov cost function $\Delta d_{\lambda}$ takes a similar form as the unregularized solution

$$\Delta d_{\lambda} = \sum_{i=1}^{K} \frac{\rho_i u_i^T \tilde{R} v_i}{\sigma_i}$$  \hspace{1cm} (3.65)$$

with the additional parameter $\rho_i$, known as Tikhonov filter factors. Assuming $\lambda$ is known quantity, the filter factors are given by

$$\rho_i = \frac{\sigma_i^2}{\sigma_i^2 + \lambda^2}$$  \hspace{1cm} (3.66)$$

such that for any $\sigma_i \ll \lambda$, $\rho_i \approx \sigma_i^2 / \lambda^2$. This ensures that singular values far below the regularization threshold contribute very little to the solution. The question now becomes how to select $\lambda$. Typically for poorly conditioned problems, the singular values $\sigma_i$ slowly decay towards zero with an abrupt drop off at some point in the spectrum. Figure 3.5 shows a plot of the singular values for a typical Jacobian matrix $\tilde{J}$ on a log scale. In practice, the regularization parameter $\lambda$ is calculated
by finding the maximum amplitude difference between the log-singular values and selecting $\lambda = \arg \max_{\sigma_i} (|\ln(\Delta\sigma_i)|)$.

Once the regularization parameter $\lambda$ is selected, the regularized update vector $\Delta d_\lambda$ can be computed. The Gauss-Newton method can then be used to iteratively update a set of quadrature points to converge to a constrained set of quadrature points which minimize the MCE error. Since this method uses a Jacobian method for a fixed number of points and weights, the solution is highly sensitive to the number of quadrature points $N$ as well as the initial guess for $X$ and $w$. The following section will discuss initialization methods for pruning and enriching the quadrature set.

### 3.5.4 Initialization and Enriching/Pruning

The Gauss-Newton update previously described assumes a constant number of quadrature points $N$ while iterating to converge to a solution. One of the biggest challenges with DQ is to find a set with minimum $N$ that will still converge to a solution. The original DQ algorithm given in [88] determines an upper bound on $N$ using sparse grid theory, and randomly samples equally weighted points within the support. The bound on $N$ for an $n$ dimensional input and $d^{th}$ order MCEs is given by

$$N \approx \kappa \frac{(2n)^{d-1}}{(d-1)!}$$

(3.67)
where $\kappa$ is a tuning parameter typically selected between $[0.5, 0.9]$. Unfortunately, there is no guarantee that the current set of quadrature points will converge to a solution. In fact, the cost function will frequently stagnate or diverge entirely during the Gauss-Newton iteration procedure. In these cases, additional decision variables can be added to the current set in what is termed 'enrichment'. Conversely, if the solution does converge to a cost function value below the specified tolerance, the number of points $N$ can be pruned and the Gauss-Newton iteration repeated to attempt to find a set with lower $N$.

There are many possible ways one could devise to prune and enrich the current quadrature set. The method recommended in [88] is to remove the points with the lowest weight once the current set converges to below a desired threshold, and increase the number of points by randomly sampling new points and assigning equal weight if the cost function stagnates or diverges. Rather than sampling random points to enrich the set, the DQ algorithm used in this dissertation takes the $\alpha$ lowest weighted samples and splits them into two new samples with half their original weight and a small amount of Gaussian noise added each.

\[
\mathbf{x}_{\text{new}} = \begin{bmatrix} \mathbf{x}_{\text{old}} \\ \mathbf{x}_{\text{old}} \end{bmatrix} + \epsilon, \quad \mathbf{w}_{\text{new}} = \frac{1}{2} \begin{bmatrix} \mathbf{w}_{\text{old}} \\ \mathbf{w}_{\text{old}} \end{bmatrix}
\]

(3.68)

where $\epsilon$ is a $(2\alpha \times 1)$ zero-mean Gaussian noise vector with variance $\approx 1e - 4$.

An alternative to randomly sampling an initial set of points and weights is to initialize using an existing set of normalized quadrature points and weights. This guarantees that at a minimum the initial guess satisfies the mean and covariance moment constraints. Combining all of the previously discussed components of the DQ methods enables generic quadrature sets to be computed. The following section will provide some examples of quadrature points computed using DQ as well as how to apply them to the reachability set problem.

### 3.6 Numerical Simulations

This section will validate the HOSM method presented in this chapter via numerical simulation. Examples are provided for both cooperative and noncooperative reachability set scenarios to illustrate how characterization of uncertainty may be
used to tailor the method for specific applications.

### 3.6.1 Cooperative Maneuvering Satellites

This section will present two examples for the case of a cooperative maneuvering satellite as described in Section 3.2. In these test cases the nominal target maneuver sequence is assumed to be known, and Gaussian uncertainty is prescribed around the nominal $\mathbf{u}_{t,i} \in \mathcal{N}(\mu_{u,i}, \Sigma_{u,i})$. Note that since the input distributions are purely Gaussian, CUT points may be used for the reachability sets.

The dynamic model $f(x, t)$ in (3.1) is given by two body equations of motion

$$f(x, t) = \begin{bmatrix} \mathbf{v} \\ -\frac{\mu r}{|r|^3} \end{bmatrix}$$

Each test case uses the NTW satellite coordinate system as defined in Vallado 2007 [92] to describe impulsive maneuvers. For this system, $\hat{N}$ is in the orbital plane and perpendicular to velocity, $\hat{T}$ is parallel to the velocity vector, and $\hat{W}$ is parallel to the angular momentum vector to complete the right handed system. The maneuvers are described by a magnitude $|\Delta v|$, as well as pitch and yaw angles $\psi$ and $\theta$ such that $\mathbf{u} = [|\Delta v| \ \theta \ \psi]$ ($m = 3$) as shown in Figure 3.6.

![Figure 3.6. Diagram of Maneuver Geometry in NTW Frame](image)

Figure 3.6. Diagram of Maneuver Geometry in NTW Frame
Maneuvers in the NTW frame can be computed using (3.70)

$$\Delta v^{NTW} = |\Delta v| \begin{bmatrix} \sin \theta \cos \psi & \hat{N} \\ \cos \theta \cos \psi & \hat{T} \\ -\sin \psi & \hat{W} \end{bmatrix}$$

(3.70)

then be rotated to the Earth Centered Inertial (ECI) frame with rotation matrix (T). Therefore the maneuver model $g(u, t)$ in (3.1) is given by.

$$g(u, t) = \begin{bmatrix} 0 \\ T\Delta v^{NTW} \end{bmatrix}$$

(3.71)

The maneuver time for all cooperative satellite maneuvers are assumed to be known deterministically, and the maneuvers approximated as impulsive.

### 3.6.1.1 Test Case 1: Two Burn Maneuver

Test case 1 is based on Betts [93] which determines the optimal three burn transfer orbit from a LEO parking orbit to a final operations orbit of a specific inclination. This trajectory consists of three impulsive maneuvers to several target orbits with varying inclinations; however, this test case will only look at the first example with target $i = 63.4^\circ$. Additionally, this case only considers the first two burns, i.e. $M=2$, for the nominal optimal transfer orbit given in [93], and the end of the simulation is taken to be the position where the third burn would occur. Only the first two burns are considered due to the fact that the input dimension is $Mm$, and CUT points are not tabulated up to 9 dimensions for all orders.

The parking orbit is a circular LEO with $i = 37.4^\circ$, and the first burn is applied at an argument of latitude $u = 255^\circ$. The right ascension of the ascending node (RAAN) is not important for this analysis and is set to zero for simplicity. The nominal orbital elements for the coasting transfer orbits are given in Table 3.3. The first burn occurs at $t_1 = 0$, and the second burn occurs at $t_2 = 6078.2(s)$. The initial state is assumed to be deterministic, and is given by the state corresponding to the parking orbit with argument of latitude $\eta = 255^\circ$. The nominal maneuvers $\mu_{u,i}$ as well as the nominal argument of latitude ($\eta^*$) and time that each maneuver occurs are given in Table 3.4. The standard deviations for all maneuver variables are given by Table 3.5. Note that the time of the second burn is fixed regardless of
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Parking Orbit</th>
<th>1st Transfer</th>
<th>2nd Transfer</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a^*(km)$</td>
<td>6667.32</td>
<td>22,835.4</td>
<td>29,243.5</td>
</tr>
<tr>
<td>$e^*$</td>
<td>0</td>
<td>0.7080</td>
<td>0.6785</td>
</tr>
<tr>
<td>$i^*(^\circ)$</td>
<td>37.40</td>
<td>35.78</td>
<td>57.88</td>
</tr>
<tr>
<td>$\omega^*(^\circ)$</td>
<td>N/A</td>
<td>254.0</td>
<td>259.6</td>
</tr>
</tbody>
</table>

Table 3.3. Test Case 1: Nominal Transfer Orbit Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>1st Burn</th>
<th>2nd Burn</th>
<th>Final Position</th>
</tr>
</thead>
<tbody>
<tr>
<td>$</td>
<td>\Delta v^*</td>
<td>(m/s)$</td>
<td>2,383.5</td>
</tr>
<tr>
<td>$\theta^*(^\circ)$</td>
<td>-1.36</td>
<td>7.21</td>
<td>N/A</td>
</tr>
<tr>
<td>$\psi^*(^\circ)$</td>
<td>5.99</td>
<td>74.7</td>
<td>N/A</td>
</tr>
<tr>
<td>$\eta^*(^\circ)$</td>
<td>255</td>
<td>37.7</td>
<td>121.7</td>
</tr>
<tr>
<td>$t(s)$</td>
<td>0</td>
<td>6,078.2</td>
<td>35,609.0</td>
</tr>
</tbody>
</table>

Table 3.4. Test Case 1: Nominal Burn Parameters

The deviation from nominal orbit due to the initial burn.

The sensitivity matrices corresponding to the reachable set at $t_f = 35,609.0(s)$ are computed for approximation orders 1-4. Additionally, 4th order sensitivity matrices corresponding to the reachable set at intermediate times are computed and used to depict the evolution of the reachability set in Figure 3.7.

Table 3.7 lists the statistical moments of the final position $r_2$. It is evident from the moments of $r_2$, and particularly the 3rd order moment, that the distribution is strongly non-Gaussian. This fact coupled with the large covariance indicates that it is likely necessary to include higher-order terms to achieve any reasonable accuracy.

10,000 points $\zeta^{(i)}$ are randomly sampled and propagated through the full simulation, as well as approximated using the sensitivity matrices. The percent error in the magnitude of final position $\%\epsilon_{r_2}$ for each sample is computed and the RMS values for each sensitivity order are shown in Table 3.6. The colorbars in

<table>
<thead>
<tr>
<th>Variable</th>
<th>Standard Deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\sigma_{\Delta V_1}$</td>
<td>5 (m/s)</td>
</tr>
<tr>
<td>$\sigma_{\theta_1}$</td>
<td>$2^\circ$</td>
</tr>
<tr>
<td>$\sigma_{\psi_1}$</td>
<td>$2^\circ$</td>
</tr>
<tr>
<td>$\sigma_{\Delta V_2}$</td>
<td>5 (m/s)</td>
</tr>
<tr>
<td>$\sigma_{\theta_2}$</td>
<td>$2^\circ$</td>
</tr>
<tr>
<td>$\sigma_{\psi_2}$</td>
<td>$2^\circ$</td>
</tr>
</tbody>
</table>

Table 3.5. Test Case 1: Nominal Maneuver Uncertainties
Figure 3.7. Test Case 1: Evolution of 4th Order Reachability Set

<table>
<thead>
<tr>
<th>Approximation Order</th>
<th>RMS($\epsilon_{r_2}$) (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1st Order</td>
<td>1.2227e0</td>
</tr>
<tr>
<td>2nd Order</td>
<td>1.1255e-1</td>
</tr>
<tr>
<td>3rd Order</td>
<td>2.3399e-2</td>
</tr>
<tr>
<td>4th Order</td>
<td>8.3191e-3</td>
</tr>
</tbody>
</table>

Table 3.6. Test Case 1: $|r_2|$ Position Error

Figure 3.8 show the percent error in final position $\%\epsilon_{r_2}$, and the axes show how the errors are spatially distributed. The axes of the plots in the left column show the final ECI positions, and the plots in the right column show Mahalanobis distances of final position and velocity on the x and y axes respectively.

It is evident from inspection of Figure 3.8 and Table 3.6 that the error in the final position decreases as the sensitivity matrix order is increased. It is intuitive that including higher-order Taylor series expansion terms will help account for nonlinearities introduced through the relatively high levels of uncertainty prescribed to the maneuvers and the length of time the over which the simulation is propagated.
Figure 3.8. Test Case 1: $|r_2|$ Approximation % Error. The color of the particle corresponds to the logarithm of % error, i.e., $\log \% \epsilon$. The cold blue color represents smaller values ($\approx O(10^{-5})$) for %$\epsilon$ while red color represents higher values ($\approx O(10)$) for %$\epsilon$. 
<table>
<thead>
<tr>
<th>Variable</th>
<th>(Mean)</th>
<th>(Variance)</th>
<th>(Skewness)</th>
<th>(Kurtosis)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_2$</td>
<td>-1.2804E+07</td>
<td>1.4277E+12</td>
<td>-1.1368E+17</td>
<td>6.1968E+24</td>
</tr>
<tr>
<td>$y_2$</td>
<td>1.9892E+07</td>
<td>4.1975E+12</td>
<td>-7.9525E+17</td>
<td>5.3445E+25</td>
</tr>
<tr>
<td>$z_2$</td>
<td>3.559E+07</td>
<td>5.5135E+12</td>
<td>-3.1677E+18</td>
<td>9.5403E+25</td>
</tr>
</tbody>
</table>

Table 3.7. Test Case 1: $r_2$ Statistical Moments

It is visually apparent from Figure 3.7 that the domain of the reachability set expands quite rapidly over time leading to a breakdown of the validity of the Taylor series expansion at lower order approximations. To continue to improve the approximation accuracy, even higher-orders sensitivities would need to be computed; however, there is a point where the computational cost of computing the next sensitivity matrix order may not be worth the diminishing return on accuracy.

### 3.6.1.2 Test Case 2: Lunar Orbit Insertion

This test case simulates a Lunar Orbit Insertion (LOI) maneuver based on 1st orbit insertion maneuver of the Korea Pathfinder Lunar Orbiter mission described in Song et al. [94]. A similar example can be found in Houghton et al. 2007 [95]. It is assumed that the orbiter approaches the moon on a hyperbolic arrival trajectory with $i = 90^\circ$ such that the satellite can enter into a polar orbit. The nominal LOI maneuver is planned to execute at the periapse of the arrival orbit such that the target insertion orbit has a period of 12 hours. A diagram showing the geometry of the arrival trajectory and insertion maneuver is shown in Figure 3.9.

The semi-major axis ($a_0$) and eccentricity ($e_0$) of the target orbit are defined by the relationships:

\[
a_0 = \left( \frac{(P/2\pi)^2}{\mu} \right)
\]

\[
e_0 = 1 - \frac{R_m + h_p}{a_0}
\]

Where $P$ is the orbital period, $\mu$ is the standard gravitational parameter of the Moon, $R_m$ is the radius of the Moon, and $h_p$ is the altitude above the lunar surface at periapse. It is assumed that the parameters $i_0, \omega_0, \Omega_0, \nu_0, h_p$ and hyperbolic arrival velocity at periapse $v_{hyp}$ are design parameters that can be specified based on mission objectives. The current analysis will use the same target orbit as in Song et al. [94]: $h_p = 200$ km, $|v_{hyp}| = 2.4$ km/s, time of maneuver $t_1 = 0$ s and all other targeted orbit parameters shown in Table 3.8. The values given represent...
Table 3.8. Test Case 2: Nominal Orbit Elements

<table>
<thead>
<tr>
<th>Orbital Element</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Semi-Major Axis ( (a_0) )</td>
<td>6143.4 (km)</td>
</tr>
<tr>
<td>Eccentricity ( (e_0) )</td>
<td>0.6847</td>
</tr>
<tr>
<td>Inclination ( (i_0) )</td>
<td>90°</td>
</tr>
<tr>
<td>Argument of Periapse ( (\omega_0) )</td>
<td>0°</td>
</tr>
<tr>
<td>RAAN ( (\Omega_0) )</td>
<td>0°</td>
</tr>
<tr>
<td>True Anomaly ( (\nu_0) )</td>
<td>0°</td>
</tr>
<tr>
<td>Period ( (P) )</td>
<td>12 hr</td>
</tr>
</tbody>
</table>

The nominal orbit of the spacecraft immediately after the burn. Given these target orbit specifications and the hyperbolic arrival velocity, the nominal maneuver is given by

\[
\mathbf{\mu}_{u,1} = [\Delta v]^* = 334.568 \text{m/s \( \theta^* = 180^\circ \ \psi^* = 0^\circ \)]
\]

(3.74)

Uncertainty will be included in the \((3 \times 1)\) inertial Cartesian position and velocity vectors \( \mathbf{r}_{hyp} \) and \( \mathbf{v}_{hyp} \), as well as in the \((3 \times 1)\) nominal maneuver vector \( \mathbf{\mu}_{u,1} \). Figure 3.6 illustrates the geometry of the orbit insertion maneuver in the
Variable & Direction & Standard Deviation \\
\hline
Position & \(N\) & 1,000 \((m)\) \\
Position & \(T\) & 10,000 \((m)\) \\
Position & \(W\) & 1,000 \((m)\) \\
Velocity & \(N\) & 1 \(\left(\frac{m}{s}\right)\) \\
Velocity & \(T\) & 10 \(\left(\frac{m}{s}\right)\) \\
Velocity & \(W\) & 1 \(\left(\frac{m}{s}\right)\) \\
Pitch & N/A & 0.5° \\
Yaw & N/A & 0.5° \\
Burn Magnitude & N/A & 5 \(\left(\frac{m}{s}\right)\) \\
\hline
\end{tabular}

Table 3.9. Test Case 4: Input Uncertainty

\begin{tabular}{l|c|c|c|c}
Variable & (Mean) & (Variance) & (Skewness) & (Kurtosis) \\
\hline
\(x_2\) & -5.1419E+06 & 8.4121E+08 & -3.6223E+10 & 2.1234E+18 \\
\(y_2\) & -4.4100E-12 & 5.0806E+07 & -8.1546E-04 & 7.7845E+15 \\
\(z_2\) & 4.4240E+06 & 4.0729E+10 & -3.4876E+15 & 4.9823E+21 \\
\hline
\end{tabular}

Table 3.10. Test Case 2: \(r_2\) Statistical Moments

NTW frame. The input variables and their associated standard deviations are listed in Table 3.9. The final simulation time is \(t_f = 2\) hours. The sensitivity coefficient matrices corresponding to the final time are computed up to fourth order. Additionally, fourth order sensitivity matrices for some intermediate times are computed and used to depict the evolution of the reachability set in Figure 3.10. The statistical moments of \(r_2\) are shown in Table 3.10.

![Figure 3.10. Test Case 2: Evolution of 4th Order Reachability Set](image_url)
Approximation Order | RMS(\epsilon_{r_2}) (%) |
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1st Order</td>
<td>4.1859e-2</td>
</tr>
<tr>
<td>2nd Order</td>
<td>1.4326e-3</td>
</tr>
<tr>
<td>3rd Order</td>
<td>6.0057e-5</td>
</tr>
<tr>
<td>4th Order</td>
<td>2.8592e-6</td>
</tr>
</tbody>
</table>

Table 3.11. Test Case 4: |r_2| Position Error

10,000 points \(\zeta^{(i)}\) are sampled from \(\pi(\zeta^{(i)})\) and both propagated through the full simulation as well as approximated using sensitivity matrices orders 1-4. Table 3.11 shows the RMS of percent error for each sensitivity order, and Figure 3.14 depicts the distribution of errors over the samples. The plots on the left of Figure 3.14 show a scatter plot of \(r_2\), and the plots on the right show error contours vs the Mahalanobis distances of the initial position and velocity on the x and y axes. The color scale of Figure 3.14 is a log scale of percent error. Similar to test case 1, the RMS errors given in Table 3.11 show that including higher-order sensitivity matrices in the polynomial model decreases the approximation error. Furthermore, the Mahalanobis distance plots show that even samples far from the mean have improved accuracy using higher-order sensitivities, implying that the approximation is valid for a larger domain of uncertainty. The following sections will explore reachability set computation using non-symmetric inputs for non-cooperative targets.

3.6.2 Noncooperative Targets

This section presents the example of a noncooperative maneuvering target with mixed input distribution. This example validates how the DQ method can be used to construct non-product quadrature sets in n-D space which are far more efficient than traditional tensor product-based methods.

3.6.2.1 Test Case 3: Single Unknown Maneuver

Consider the case of a noncooperative target in a Low Earth Orbit (LEO) making a single unknown maneuver at an unknown time. The dynamic model used is the
Figure 3.11. Test Case 2: $|r_2|$ Approximation % Error. The color of the particle corresponds to the logarithm of % error, i.e., $\log \%\epsilon$. The cold blue color represents smaller values ($\approx O(10^{-7})$) for $\%\epsilon$ while red color represents higher values ($\approx O(10^{-1})$) for $\%\epsilon$. 
nonlinear J2-perturbed relative equations of motion given in [96].

\[
f(\mathbf{x}, t) = \begin{pmatrix}
\dot{x} \\
\dot{y} \\
\dot{z}
\end{pmatrix} = 
\begin{cases}
2\dot{y}\omega_z + x\omega_z^2 + y\alpha_z - z\omega_x\omega_z - r\eta^2 + \zeta s_i s_\theta + \eta^2(r + x) + \zeta s_i s_\theta
\\
-2\dot{x}\omega_z + 2\dot{z}\omega_x - x_k \alpha_z + y(\omega_z^2 + \omega_x^2) + z\alpha_x + \zeta s_i c_\theta + \eta^2 y + \zeta s_i c_\theta
\\
-2\dot{y}\omega_x - x\omega_x\omega_z - y\alpha_x + z\omega_x^2 + \zeta c_i + \eta^2 z + \zeta c_i
\end{cases}
\]

(3.75)

where $\omega_i$ terms are angular velocities of the rotating frame, $\alpha_i$ terms are the angular accelerations of the rotating frame, and $\zeta, \eta$ are constants related to the $J_2$ acceleration. The coordinate frame for this model is the relative RSW frame defined in [92], where $\hat{R}$ is the radial direction, $\hat{S}$ is the in-track direction and $\hat{W}$ is the orbit normal direction. The origin of the RSW frame is located at the nominal orbit of the target at $t_0$.

The input vector for this case can be represented as $\mathbf{z}_t = [u_{t_1}, t_1]$. Using the $(15 \times 3)$ 5th order spherically uniform CUT set found in Section 3.4, $\mathbf{X}_1 \in \mathcal{U}_{s}(0,1)$, and a 5th order $(3 \times 1)$ uniform Gauss-Legendre set $\mathbf{X}_2 \in \mathcal{U}(-1,1)$, a tensor product set can be computed as

\[
\mathbf{X}_{tens} = \mathbf{X}_1 \otimes \mathbf{X}_2
\]

(3.76)

where $\mathbf{X}_{tens}$ is a $(45 \times 4)$ set which replicates up to 5th order moments. Although this method is valid, the tensor product set does not contain the minimal number of points. Using the DQ algorithm outlined in Section 3.5, a 5th order set of quadrature points $\mathbf{X}_{DQ}$ can be found with only 21 points constrained to lie within the support of the mixed spherically uniform, linearly uniform input $\mathbf{z}_t$.

Assume the target is initially in an inclined LEO orbit with orbital elements given in Table 3.12, and is assumed to make a maneuver bounded by a magnitude of $5 \text{m/s}$ $u_{t_1} \in \mathcal{U}_s(0,5 \text{m/s})$ and bounded maneuver time $t_1 \in \mathcal{U}(0,P/2)$ where $P$ is the orbital period ($P = 5309.6s$). Reachability set coefficients with for a second order polynomial basis are computed in intervals of 30 seconds throughout the first orbital period using each quadrature set $\mathbf{X}_{tens}$ and $\mathbf{X}_{DQ}$. 10,000 random Monte Carlo samples are then drawn from the input vector $\zeta_{MC}^{(i)}$ and exactly integrated through the dynamic model to target states $\mathbf{x}_{mc,k}^{(i)} = \mathbf{X}(\zeta_{mc}^{(i)}, k)$ in the same 30 second intervals. The error between the exact MC sample evaluations and the polynomial
<table>
<thead>
<tr>
<th>Orbital Element</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Semi-Major Axis</td>
<td>6,578.1 (km)</td>
</tr>
<tr>
<td>Eccentricity</td>
<td>0</td>
</tr>
<tr>
<td>Inclination</td>
<td>45°</td>
</tr>
<tr>
<td>Right Ascension of Ascending Node</td>
<td>0°</td>
</tr>
<tr>
<td>Argument of Periapse</td>
<td>0°</td>
</tr>
<tr>
<td>True Anomaly</td>
<td>0°</td>
</tr>
</tbody>
</table>

Table 3.12. LEO Initial Orbital Elements

![Graph](image.png)

Figure 3.12. Test Case 3: Approximation Errors for Reachability Sets Computed Via Tensor Product and DQ Methods

The norm of position and velocity errors, $||\epsilon_r||_2$, $||\epsilon_v||_2$ for each sample are computed and the RMS error value is computed over all samples

$$\text{rms}(||\epsilon||_2) = \sqrt{\frac{1}{10000} \sum ||\epsilon^{(i)}||_2}$$

The RMS error of position and velocity is plotted for each method vs. time in Figure 3.12. It is immediately discernible that the errors for the DQ method and the tensor product method are nearly identical throughout the entirety of the simulation. This result demonstrates, that although the non-product DQ method has computational savings of 53% compared to the product method ($N = 21$ vs.

75
Figure 3.13. Test Case 3: Scatterplot of Reachability Set Position Errors

\( N = 45 \), there is no loss in accuracy. This is due to the fact that the same order of MCEs were satisfied for both sets regardless of the number of points used.

Figure 3.15 shows a position scatter plot of all 10,000 MC samples \( x^{(i)}_{\text{mc}} \) at the final timestep, and the colorbar denotes the distribution of position errors throughout the reachability set. It has been demonstrated here that the non-product DQ quadrature method has equivalent accuracy to a tensor product method, with significantly improved efficiency.

Now consider reachability sets computed using polynomial model orders \( D = 2, 3, 4, 5 \) and DQ method with MCE orders \( d = 4, 6, 8, 10 \) respectively. Figure 6.8 shows scatterplots of reachability sets of varying order after one full period \( t_f = P \) where the colorbar depicts position error on a log scale. It is apparent that the approximation error decreases significantly with approximation error. Figure 6.8 shows the RMS log position error vs time for all polynomial models. A similar trend can be seen where the approximation error is lower for higher-order polynomial models throughout the entire propagation times. Interestingly, in this case the error is approximation drops dramatically after the upper bound on maneuver time \( P/2 \). This is because any samples which have not maneuvered yet are clustered around the nominal trajectory (i.e. the origin) for which a polynomial approximation is a poor fit. After all samples have maneuvers, these points disperse and a smooth polynomial fit is a much better approximation.
Figure 3.14. Test Case 3: Position Approximation Error for Varying Order $D$

Figure 3.15. Test Case 3: RMS Position Error (log scale) vs. Time
3.7 Conclusion

This chapter has developed the HOSM method for nonlinear reachability set propagation via numerical evaluation of Taylor series expansion coefficients. The efficiency and accuracy of the HOSM method is highly dependent on the particular quadrature method used to evaluate the polynomial coefficients. For high dimensional inputs, evaluation of these coefficients may become infeasible altogether; therefore, the HOSM method is strongly limited by the dimension of the input. Furthermore, the accuracy of HOSM is shown to increase when higher-order sensitivity terms are included; but computing these higher-order sensitivities comes at the computational cost of requiring quadrature sets that satisfy higher-order MCEs.

One major difference between HOSM method, and traditional uncertainty propagation is that the input variables considered may or may not be mixed distributions. Typical uncertainty propagation only considers Gaussian uncertainty in initial conditions; however, examples have been presented here for both purely Gaussian inputs for known maneuvers and mixed inputs for unknown maneuvers. Unknown maneuvers have been uniquely defined by bounding maneuver magnitude to be spherically uniform, and maneuver time to be linearly uniform. A generic moment matching quadrature method has been presented which enables direct construction of quadrature sets for these mixed inputs in $n$-D space, enabling highly efficient computation.

The reachable space of a maneuvering target as defined in this chapter serves as the search area for the target. If the reachability set is small, then the target may be easily found using classical tracking techniques; however, if the reachability set is physically larger than the field of view of the sensors tasked for tracking, classical methods may fail. The following Chapter presents a novel framework for detection and tracking of maneuvering satellites which cannot be located using classical techniques.
Chapter 4  |  Reachability Set Search

4.1 Introduction

The previous chapter developed the HOSM uncertainty propagation method which enables the computation of the reachable space of a target. This chapter will build on this reachability set computation technique to develop a novel framework for tracking maneuvering satellites which cannot be found using classical tracking techniques.

Maneuvering target tracking is a well-studied problem in the literature, with a history dating back to the 1970’s [35, 97, 98]. A comprehensive introduction to maneuvering target tracking is presented in a six part paper series [28–33] which covers literature up to the early 2000’s. The core objective of maneuvering target tracking is to extract meaningful information about the trajectory of the target based on observational data. In most tracking applications, this involves using real-time measurements to inform the dynamic model, as well as reconstruct or estimate maneuvers as they occur. Methods for maneuvering target tracking in this category include decision-based methods [34–36] and multiple-model methods [37–42]. The authors emphasize here that the literature cited above assume data-richness, such that measurements of the target are acquired during the target maneuver.

Methods using real-time measurements are useful in air and ground target tracking scenarios where it is reasonable to assume that measurements of the target are available throughout the entire trajectory. Unfortunately, due to the limited coverage and availability of sensor resources, satellite tracking applications frequently have large time delays between observations on the order of hours or days.
In these data-sparse situations, unobserved maneuvers can drastically change the target trajectory to the point where a tasked sensor loses custody of the intended target entirely.

Detecting and reconstructing maneuvers in data-sparse situations has received some, albeit limited, coverage in the literature. Patera [43] addresses the problem of detecting maneuvers and other events (collisions, reentry, etc) in terms of statistically significant charges in orbital energy. An optimal control-based method has also been developed to reconstruct finite maneuvers [44–46] connecting two measurements. The underlying technique for this method was first formulated in 1988 as the minimum model error method [47]. The minimum model error method treats the control as an unmodeled deviation from the dynamics, and minimizes this deviation such that the state estimate is statistically consistent with the observations. When applied to the satellite tracking problem, this method formulates the maneuver reconstruction process as a two point boundary value problem under the assumption of a minimum fuel control policy. Whenever the optimal control profile rises above the level of system noise, it is assumed that a maneuver has occurred. Although the minimum fuel maneuver is not necessarily a bad assumption, this method does not account for the many sub-optimal trajectories that can explain the same observational data. Furthermore, the orbital energy method as well as the minimum model error method make the assumption that the target can be observed after making a maneuver. The problem addressed in this work is fundamentally different. The problem currently considered is to use sensor data in conjunction with the dynamical model for target motion and bounds on maneuver parameters to seek and locate a target satellite which has been lost due to an unknown maneuver.

In this respect, the objective of this chapter is to determine a set of sensing parameters to detect the target given the search area determined using the HOSM method. The search area, reachability set, and target state pdf are synonymous terms defined by the mapping of target initial state, maneuver, and parametric uncertainties to state at future times. The pdf represents the set of all possible target states given a-priori knowledge on input uncertainty bounds and measurement data, and the true target state will always lie within this set. Sensor data is used to update the target pdf regardless of whether the target is detected or not. If the tasked sensor is able to detect the target, then the associated measurement noise is
used to update the each region; whereas, in the case of an unsuccessful detection, the detection likelihood is used to update the search region. An important feature of this approach is that unsuccessful detection is also exploited to improve the future search and tracking of noncooperative satellites. The method presented in this chapter creates a unified framework for search, detection, tracking, and maneuver estimation of a noncooperative target.

Assuming a target reachability set can be computed via the HOSM method described in the previous chapter, a method for tasking sensors to search the set must be developed. A greedy in time maximum detection likelihood policy is implemented for this purpose. In practice, the maximum likelihood cost function for highly nonlinear dynamic and measurement models are evaluated using MC samples efficiently propagated via the reachability set model. The final component to the proposed framework is the measurement update step, implemented as a classical Bayesian particle filter update. A key contribution of the proposed method is the utilization of measurements of the reachability set rather than the target itself to provide better information on the remaining possible target locations. If the tasked sensor does not observe the target, the detection likelihood function is used to update the target search area. Conversely, if the target is located, an importance sampling with progressive correction (ISPC) procedure is used to accurately define the posterior.

The organization of the chapter is as follows. Section 4.2 provides a description of the data-sparse maneuvering target search problem. Section 4.3 and Section 4.4 detail the sensor tasking and filtering/estimation components of the proposed framework respectively. Lastly, Section 4.5 provides simulations and analysis to validate the method numerically.

### 4.2 Problem Description

This section provides a description of the reachability set search problem using a controllable observer. Assume that a maneuvering target with state $\mathbf{x}_t$ has dynamic flow $\mathbf{X}_t$, and that polynomial model coefficients $\mathbf{C}$ up to the $d^{th}$ order expansion have been computed via HOSM method

$$\mathbf{x}_{t,k} = \mathbf{X}_t(\zeta_t, k) \approx \mathbf{C}_k \phi(\zeta_t)$$  \hspace{1cm} (4.1)
where $\zeta_t$ is the normalized target input consisting of all initial condition, control, and parametric uncertainties, and $\phi$ is a $d^{th}$ order polynomial basis. Also assume that the observer state $x_{ob}$ with dynamic flow $\chi_{ob}$ is given by

$$x_{ob,k} = \chi_{ob}(z_{ob}, k) \quad (4.2)$$

where $z_{ob}$ is the observer input consisting of deterministic initial conditions, control, and model parameters. Since $z_{ob}$ is deterministic, normalization to $\zeta_{ob}$ is mathematically meaningless, so it will be left unnormalized. The combined system state is defined as the target state augmented by the observer state $x^T = [x^T \ x_{ob}^T]$.

Assume the observer has field of view (FOV) constrained to a region defined by $C_s(x_k, \theta_k) \leq 0$ where $\theta_k$ is an ($l \times 1$) vector of sensor parameters. The concept of detection likelihood can now be introduced, which describes the probability that a sensor reaches some statistical threshold to make a binary decision on whether or not the target has been measured. A piecewise detection likelihood function $\pi_d(x_k, \theta_k)$ can be defined with respect to the FOV constraints

$$\pi_d(x_k, \theta_k) = \begin{cases} 
\pi_d'(x_k, \theta_k) & \text{if } C_s(x_k, \theta_k) < 0 \\
0 & \text{otherwise} 
\end{cases} \quad (4.3)$$

such that the target has a probability $\pi_d'(x_k, \theta_k) \in [0, 1]$ of being detected within the FOV. Given known bounds on the target uncertainties $\zeta_t$, FOV constraints, and detection likelihood function, the objective is to determine sensor parameters $\theta_k$ which maximize some target detection metric $J_d$

$$\max_{\theta_k} J_d \quad (4.4)$$

Assuming selection of $\theta_k$ provides a successful detection, the target measurement can be modeled by

$$y_k = h(x_k) + \nu_k \quad (4.5)$$

where $y$ is an ($m \times 1$) measurement vector, $h$ is the measurement model, and $\nu_k \in N(0, R_k)$ is zero mean Gaussian measurement noise. If there is not a detection, then $y_k$ is defined as a null set. One of the novel aspects of the proposed method is to utilize both actual measurements as well as null measurements to update the target estimate. The proposed framework depicted in Figure 4.1, can
be split into three main components

1. Reachability Set Computation

2. Sensor Tasking

3. Tacking/Estimation

The first component, reachability set computation, is discussed thoroughly in Chapter 3, and it is assumed that polynomial coefficients $C_k$ have already been computed. The second and third components will be discussed in detail in the following sections.

### 4.3 Sensor Tasking

Assume that polynomial coefficients which map $\pi(\zeta)$ to $\pi(x)$ have already been computed. The question now becomes how to select sensor parameters to search the reachability set. This chapter considers scenarios under the assumptions of a single-observer, single-target, and greedy in time tasking approach. Under these assumptions, it is sufficient to define criteria $J_d$ as the expectation value of the detection likelihood function $\pi_d(x_k, \theta_k)$

$$J_d = E[\pi_d(x, \theta)] = \int_{\Omega} \pi_d(x, \theta)\pi(x)dx$$  \hspace{1cm} (4.6)
Simulations in this chapter define the observer as a space-based sensor with parameters $\theta$ given by attitude angles with associated attitude unit vector $\hat{a}$. The detection likelihood function in equation (4.3) can be defined in a number of ways. Perhaps the most simplistic detection likelihood function is $\pi'_d(x, \theta) = 1$ which implies a 100% probability of detecting targets within the observer FOV. Other options include geometric models which adjust the likelihood based on the position within the observer FOV, or physics-based models which use basic principals to model the detection likelihood. Since the purpose of this chapter is to provide a methodological framework rather than to provide realistic simulations, the detection likelihood function used in this paper is a toy geometric model with conical FOV constraints

$$\pi'_d(x, \theta) = \left(1 - \left(\frac{\gamma}{\gamma^*}\right)^2\right) \exp\left[-\frac{|\rho|}{|\rho_s|}\right]$$

(4.7)

$$C_s(x, \theta) = \gamma(x, \theta) - \gamma^*$$

where $\gamma^*$ is the FOV half-angle, $|\rho_s|$ is a scale distance, $\rho = x_t - x_{ob}$ is the range vector, and $\gamma$ is the angle between attitude vector and range vector. Angle $\gamma$ can be computed by

$$\gamma(x, \theta) = \cos^{-1}\left(\frac{\rho \cdot \hat{a}}{|\rho|}\right)$$

(4.8)

A schematic of the geometry between the observer FOV, target, and detection likelihood function is shown in Figure 4.2. The problem now becomes how to evaluate the expected detection likelihood, i.e. cost function.

Due to the FOV constraints imposed on the detection probability function and
the generic nonlinear pdf $\pi(x)$, evaluation via quadrature methods is not a practical option. Thus, a MC evaluation is necessary. Typically this would require the explicit propagation of random samples throughout the dynamics; however, the polynomial model computed in the previous chapter enables accurate approximation of samples to represent $\pi(x)$ several orders of magnitude faster than explicit evaluation of the dynamics. If $N$ samples are randomly drawn from the normalized input vector $\zeta^{(i)} \in \pi(\zeta)$ and assigned equal weights $w^{(i)} = \frac{1}{N}$, then target state is approximated using the reachability coefficients

$$x^{(i)} \approx C\phi(\zeta^{(i)}) \quad (4.9)$$

and the reachability set, i.e. pdf, can be approximated as a finite sum

$$\pi(x) \approx \sum_{i=1}^{N} w^{(i)} \delta(x^{(i)} - x) \quad (4.10)$$

where $\delta(\cdot)$ is the Dirac delta function. Substituting (4.10) into (4.6) yields the MC approximated cost function

$$J_d = \sum_{i=1}^{N} w^{(i)} \pi_d(x^{(i)}, \theta) \quad (4.11)$$

This approximate evaluation of the expected detection likelihood, enables a feasible implementation of the sensor parameter selection. In practice, the objective function is maximized by performing a grid search of $\theta$ and selecting the attitude which maximizes $J_d$. Due to the axial symmetry of the conic FOV and detection probability, only two attitude angles, are required to describe the observer attitude. Note that there is a trade-off here between computational efficiency and the optimal of the grid search.

An adaptive grid is used to avoid unnecessary computational costs. The grid is defined at each timestep by

$$\theta_{grid,1} = \min(\theta(x^{(i)})) : \gamma^* : \max(\theta(x^{(i)}))$$

$$\theta_{grid,2} = \min(\phi(x^{(i)})) : \gamma^* : \max(\phi(x^{(i)}))$$

$$\Phi = \theta_{grid,1} \otimes \theta_{grid,2}$$

85
where $\Phi$ is the grid of angles to search, and $\theta, \phi$ are the azimuth and colatitudes angles respectively which correspond to the reachability samples. In this manner the minimum and maximum azimuth and colatitude angles of the current target reachability samples are used to bound the grid search and set nodes in increments of the FOV half-angle rather than fully grid searching all possible angles. After the optimal sensor parameters are selected, a measurement is taken and must be used to update the reachability set. The following section discusses how to utilize the measurement to update the estimated target state and maneuver sequence using a classical Bayesian particle filter update.

### 4.4 Filtering and Estimation

The traditional filtering problem involves fusion of target measurement data with propagated state uncertainty to most accurately describe the posterior target distribution. In contrast, the problem considered here is to use propagated state uncertainty to task sensors to detect the target. If the target is not detected, observations of vacant regions of the search area are used to update the search area at future times; however, if the target is detected, the objective is synonymous with the traditional filtering problem. Here, the prior and posterior update states will be denoted $x^-$ and $x^+$ respectively.

In data-rich applications, linearization such as the Extended Kalman Filter (EKF) are frequently sufficient to approximate nonlinear systems. The state can often be approximated as Gaussian in these applications; however, linear assumptions quickly break down in data-sparse applications due to the large interval between measurements. As a result, a particle filtering framework will be used to consider full density estimation of the target pdf. Particle filters are a subset of sequential Monte Carlo (SMC) techniques which are well suited for this purpose. SMC techniques are a popular class of methods which approximate a full pdf $\pi(x)$ as an ensemble of discrete samples. A thorough introduction to SMC methods can be found in [99–101].

Assuming that sensor parameters $\theta$ have been computed via maximum likelihood procedure given in Section 4.3, a measurement $\tilde{y}$ can be taken and used to update the target reachability set. Since the reachability set is already approximated by the MC ensemble representation given by equation (4.10), it is natural to update
the prior pdf $\pi(x^-)$ using the process of Bayesian inference. Bayes rule can be written as

$$\pi(x^+) = \frac{\pi(x^-)\pi(\tilde{y}|x^-)}{\int \pi(x^-)\pi(\tilde{y}|x^-)dx^-}$$  \hspace{1cm} (4.13)$$

where $\pi(\tilde{y}|x^-)$ is the measurement likelihood function. Notice that the denominator of Bayes rule is simply a normalization factor, so the update can instead be written as a proportionality

$$\pi(x^+) \propto \pi(x^-)\pi(\tilde{y}|x^-)$$  \hspace{1cm} (4.14)$$

Substituting (4.10) into (4.14), the measurement update can be written as a point-wise weight update

$$\hat{w}^{(i)+} \propto w^{(i)-}\pi(\tilde{y}|x^{(i)-})$$  \hspace{1cm} (4.15)$$

where $\hat{w}^{(i)+}$ is an intermediate weight proportional to the prior weight. Recognizing that the denominator of Bayes rule is given by $\sum \hat{w}^{(i)+}$, the posterior weights are given by

$$w^{(i)+} = \frac{\hat{w}^{(i)+}}{\sum_{i=1}^N \hat{w}^{(i)+}}$$  \hspace{1cm} (4.16)$$

The prior samples can now shifted to the posterior samples $\zeta^{(i)+} \rightarrow \zeta^{(i)-}$ and propagated to the next timestep.

The measurement likelihood function $\pi(\tilde{y}|x^{(i)-})$ plays an important role in the proposed search procedure and is closely related to the detection likelihood function. For simulation purposes, a random number is drawn on the interval $[0, 1]$ and if it is less than the detection likelihood of the true target state $\pi_d(x^*, \theta)$, the target is considered to have been located. If the target is detected, then the measurement likelihood is defined using the zero mean Gaussian sensor noise $\nu$ associated with the measurement model (4.5)

$$\pi(\nu) = \pi(\tilde{y}|x^-) = \frac{1}{(2\pi)^{m/2}\sqrt{|R|}}exp\left(\frac{-(\tilde{y} - h(x^-))^TR^{-1}(\tilde{y} - h(x^-))}{2}\right)$$  \hspace{1cm} (4.17)$$

Conversely, if the target is not detected, the measurement likelihood is defined to be the complement of the detection probability

$$\pi(\tilde{y}|x^-) = 1 - \pi_d(x^-, \theta)$$  \hspace{1cm} (4.18)$$

In this manner, samples inside the observer FOV will have their weight reduced...
proportional to the likelihood that the target would have been detected if it were truly located at the sample.

Notice that any particle states with very low measurement likelihood \( \pi(\tilde{y}|\mathbf{x}^{(i)-}) \) will have a posterior weight \( w^{(i)+} \) near zero. In this situation, the particle \( \mathbf{x}^{(i)+} \) contributes very little to the approximation of the posterior, and as the particle filter update is repeated over multiple timesteps, the weight becomes concentrated in very few or even a single particle. This phenomenon is known as sample impoverishment or particle degeneracy and leads to a poor approximation of the target pdf. Resampling is an effective way to alleviate this issue. [102] provides a concise summary and comparison of several popular resampling algorithms. The systematic resampling algorithm will be used in this paper for its efficiency and easy implementation. Note that since the system is propagated using the HOSM method, the reachability samples \( \zeta^{(i)} \) are redrawn from \( \pi(\zeta^+) \) rather than directly from \( \pi(\mathbf{x}^+) \). Typically, a resampling condition is used to avoid costly resampling procedures at every timestep. [103] characterizes this resampling condition using the Effective Sample Size (ESS) criterion

\[
\text{ESS} = \left( \sum_{i=1}^{N} \left( w^{(i)} \right)^2 \right)^{-1}
\]  

The resampling condition is to resample only when ESS is less than a specific threshold \( N_t \). The threshold \( N_t = N/2 \) is used in implementation of the examples in this chapter.

Although resampling is an effective method for preventing particle degeneracy over multiple timesteps, if the prior distribution is very diffuse with respect to the likelihood function, problems with particle degeneracy may arise at a single timestep. This is frequently the case in the current application when a target with a large reachability set is detected by an observer with low sensor noise. For example, if a sensor has a range measurement with noise standard deviation on the order of meters, and the closest particle in the reachability set has a range on the order of kilometers from the measured range, then the closest particle is thousands of standard deviations away from the expected measurement. The likelihood of such a particle is numerically zero, and thus, all of the particle weights will be set to zero and the filter will become singular.
This phenomenon is well understood in particle filters, and several methods have been developed to alleviate this practical implementation issue. This paper uses the importance sampling with progressive correction (ISPC) technique outlined in [104] to update the reachability set when the target is detected. The idea behind ISPC is to include an expansion factor $\lambda_k$ in the likelihood function, and iteratively resample intermediate posterior distributions until the expansion factor converges to the true posterior. The modified likelihood function is given by

$$
\pi'(\tilde{y}|x) = \frac{1}{(2\pi)^{m/2}\sqrt{|R|}} \exp\left(\frac{-((\tilde{y} - h(x^-))^T R^{-1} (\tilde{y} - h(x^-)))}{2\lambda_k}\right)
$$

such that for $\lambda_k > 1$ the standard deviation of the likelihood function is artificially inflated. [104] provides an adaptive choice for parameter $\lambda_k$ given by

$$
\lambda_k = \frac{\max_{1\leq i\leq N} -(\tilde{y} - h(x^-))^T R^{-1} (\tilde{y} - h(x^-))}{2\log(\delta_{\text{max}})}
$$

where $\delta_{\text{max}}$ is a tunable parameter which influences the size of the progressive correction steps.

The final step in the maneuvering satellite search procedure is to estimate the target state and maneuver sequence. In fact, the HOSM method framework unifies the estimation of the target state and maneuver sequence into the same procedure. Since the samples in the maneuver space $\zeta(i)$ are mapped to samples in the state space $x(i)$ by the polynomial approximation (4.9), estimates can be computed simultaneously using

$$
\begin{bmatrix}
E[x+]
E[\zeta+]
\end{bmatrix} = \sum_{i=1}^{N} w(i) \begin{bmatrix}
C\phi(\zeta(i)^+) \\
\zeta(i)^+
\end{bmatrix}
$$

Note that during the search phase, the point estimate of the target given above may be a poor estimate of the true target location due to the very diffuse target pdf. Typically, a point estimate of mean target state only becomes meaningful after the first target detection when the pdf contracts to the levels of sensor noise around the measurement. Additionally, it is important to note that since measurements are taken in the $x$ space, the reliability of maneuver estimates in the $\zeta$ space are highly dependent on the quality of the polynomial approximation (4.9).
Once the update and estimation steps are complete, the reachability samples are propagated to the next timestep and the cycle is repeated until the target is detected. The reachability set search method can be summarized as following:

1. **Initialize Particles and Quadrature Points:**
   
   Randomly sample $N$ reachability set particles from the normalized maneuver distribution: $\zeta_{rs}^{(i)} \in \pi(\zeta)$ and assign equal weights $w_{i}^{(i)} = \frac{1}{N}$, and initialize quadrature points computed via non-product quadrature method: $\zeta_{q}^{(i)}$

2. **Reachability Set Propagation**
   
   Directly propagate quadrature points: $x_{q,k+1}^{(i)} = \chi(\zeta_{q}^{(i)}, z_{ob}, k + 1)$ to next measurement at $t_{k+1}$. Use $x_{q,k+1}^{(i)}$ to evaluate polynomial model coefficients. Propagate reachability samples to $t_{k+1}$ using polynomial approximation:
   
   $x_{rs,k+1}^{(i)} \approx C_{k+1} \phi(\zeta_{rs}^{(i)})$. Since the quadrature points are designed to be minimal, approximating $x_{rs,k+1}^{(i)}$ in this manner can offer orders of magnitude of computational savings.

3. **Sensor Parameter selection**
   
   Create a grid of sensor parameters $\Phi$ using the adaptive grid criteria in (4.12). For each parameter combination, evaluate the cost function (4.11) using reachability samples $x_{rs,k+1}^{(i)}$ and select the parameters which maximize detection likelihood $\theta^{*} = \max_{\Phi} J_{d}(x, \Phi)$

4. **Reachability Set Update**
   
   Update the reachability set particle weights using the point-wise update in (4.15) and (4.16). If target is detected, use likelihood function (4.17), if target is not detected, use likelihood function (4.18).

5. **Resample**
   
   If the target is detected for the first time, resample using ISPC algorithm. Otherwise, if resampling criterion $ESS < \frac{N}{2}$ is satisfied, resample $\zeta_{rs}^{(i)}$ using systematic resampling algorithm.

6. **Estimate State and Maneuver**
   
   Compute the current target state and maneuver estimates using (4.22). If desired, covariance estimates can be computed as well. Keep in mind that
estimate covariance will be very large if the reachability set is still being searched. Return to step 2.

The above framework enables a feasible closed-cycle search procedure for locating a noncooperative target satellite that has been lost due to unknown maneuvers. Figure 4.3 summarizes each component of the proposed method, and the following section will present numerical simulations and discuss results.

4.5 Numerical Simulations

This section provides simulation results for three examples of the reachability set search method. The three simulation test cases are 1) a single maneuver LEO case, 2) a single maneuver GEO case, and 3) a two maneuver GEO case. All examples consider a space-based observer in an orbit similar to the target such that the initial conditions are provided in relative coordinates. The dynamic model $f(x, t)$ used for all results is the nonlinear $J_2$-perturbed relative motion model defined in [96]. The equations of motion for this problem are the nonlinear $J_2$ perturbed model used in Chapter 3 test case 3 (3.75).

The observer is prescribed range $|\rho|$, range-rate $|\dot{\rho}|$, azimuth $\hat{\theta}_1$, and colatitude
\( \hat{\theta}_2 \) measurements given by

\[
\mathbf{h}(\mathbf{x}) = \begin{bmatrix}
|\rho| \\
|\dot{\rho}| \\
\hat{\theta}_1 \\
\hat{\theta}_2
\end{bmatrix} = \begin{cases}
\sqrt{\rho_1^2 + \rho_2^2 + \rho_3^2} \\
\frac{\rho \cdot \rho}{\sqrt{\rho_1^2 + \rho_2^2 + \rho_3^2}} \\
\tan^{-1}(\frac{\rho_2}{\rho_1}) \\
\cos^{-1}(\rho_3)
\end{cases}
\tag{4.23}
\]

where all relative position and velocity coordinates are given in the RSW frame. The sensor parameters \( \theta = [\theta_1 \, \theta_2]^T \) specify the observer attitude, which can be converted into a unit vector \( \mathbf{a} \) given by

\[
\mathbf{a} = \begin{bmatrix}
sin(\theta_2) \cos(\theta_1) \\
\sin(\theta_2) \sin(\theta_1) \\
\cos(\theta_1)
\end{bmatrix}
\tag{4.24}
\]
such that \( \gamma \) is the angle between \( \mathbf{a} \) and \( \rho \) and can be computed using (4.8). Note that the outcome of the simulation is dependent on the overall observability of the reachability set with respect to the observer detection zone. If the volume of the observer FOV is significantly less than the volume of the reachability set, then the observer may never be able to locate the target within the set; and conversely, if the observer FOV is larger than the reachable set, then the target will be detected within the first few timesteps. The parameters selected for all examples correspond to a sensor with long range, but narrow FOV, and were chosen for moderate observability to demonstrate a range of outcomes.
<table>
<thead>
<tr>
<th>Orbital Element</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Semi-Major Axis</td>
<td>7364.6 (km)</td>
</tr>
<tr>
<td>Eccentricity</td>
<td>0.1</td>
</tr>
<tr>
<td>Inclination</td>
<td>23.5°</td>
</tr>
<tr>
<td>RAAN</td>
<td>0°</td>
</tr>
<tr>
<td>Argument of Periapse</td>
<td>0°</td>
</tr>
<tr>
<td>True Anomaly</td>
<td>0°</td>
</tr>
</tbody>
</table>

Table 4.1. Test Case 1: Initial Target Nominal Orbital Elements

4.5.1 Test Case 1: Single Maneuver LEO

This section presents results for a reachability set search in LEO. The initial target state for this test case is an inclined and slightly eccentric orbit, with initial orbital elements given by Table 4.1. For this test case, the target maneuvers are defined as

\[ z_t = [u_{t,1}, t_1], \quad u_{t,1} \in U_0(0, 10m/s), \quad t_1 \in U(0, P) \quad (4.25) \]

where \( P \) is the orbital period. Assume the true target maneuver is given by

\[ u_t^* = \left[ -1(\text{m/s}) - 7(\text{m/s}) 4(\text{m/s}) 12(\text{mins}) \right] \]

The observer is assigned an initial state of

\[ x_{ob} = \left[ -5000(\text{m}) -100000(\text{m}) 5000(\text{m}) 0(\text{m/s}) 0(\text{m/s}) 0(\text{m/s}) \right]^T \quad (4.26) \]

where \( x_{ob} \) is in RSW relative coordinates. The observation zone half angle is \( \gamma^* = 5^\circ \), and the scale range is \( |\rho_s| = 85 \text{ km} \). Assume that measurements are taken every \( \Delta t = \text{mins} \), and the measurements have noise standard deviations \( \sigma_r = 100m \) in range, \( \sigma_r = 0.1m/s \) in range rate, and \( \sigma_\theta = 0.1^\circ \) in attitude angles. The reachability sets in this problem were generated using a second order polynomial basis, and fourth order quadrature points for the spherically uniform maneuver as well as the uniformly distributed maneuver time. The reachability set for this scenario after one hour using \( N = 10,000 \) reachability samples is shown in Figure 4.5, where the color bar indicates the maneuver time.

Figure 4.6 depicts the error in estimated state on a logarithmic scale vs time. The left-hand side of Figure 4.6 shows the error in position estimate and the right-hand side shows the error in velocity. The true target maneuver time and initial detection time are shown as a vertical red and blue line respectively. Figure 4.8 gives the error in estimated maneuver time over the course of the simulation.
Inspecting Figure 4.6, three distinct regions of the simulation can be seen. The first region is before the target has maneuvered, where error can be seen to gradually increase. This is due to the observer satellite taking asymmetric measurements of nearest region of the reachability set. Since the satellite is not found in this region, the target state pdf is updated asymmetrically and the error increases because the true target is still at the origin. The second region is after the target has maneuvered. In this region, the error rapidly increases to the order of tens of kilometers in position and around $10 \frac{\text{m}}{\text{s}}$ in velocity. The third region occurs after the target has been detected, where the error in the state estimate is dependent on the prescribed sensor error.

Using the prescribed simulation parameters, the target is located approximately 28 minutes after it maneuvered and the error in position drops about 3 orders of magnitude after the first target detection. This example was analyzed for only a single realization of the target maneuver, but the following GEO examples will use MC simulations to assess the performance of the method for various true target maneuver sequences.

### 4.5.2 Test Case 2: Single Maneuver GEO

This test case presents results for the reachability set search method applied to a target satellite making a single unknown maneuver in GEO. The nominal target
Figure 4.6. Test Case 1: State Estimate Error
Figure 4.7. Test Case 1: Final State Distribution

satellite orbit at \( t = 0 \) is defined by the orbital elements and corresponding Earth Centered Inertial (ECI) coordinate state vector given in Table 4.2 and Table 4.3 respectively.

Assume that the target maneuver capability is \( 5m/s \) and that half an orbital period elapses without observing the target such that maneuver time is defined as
uniformly distributed between 0 and 12 hours

\[ \mathbf{z}_t = [\mathbf{u}_{t,1}, t_1], \quad \mathbf{u}_{t,1} \in \mathcal{U}(0, 5m/s), \quad t_1 \in \mathcal{U}(0, 12hr) \quad (4.27) \]

Assume measurements are taken every 5 minutes (\( \Delta t = 300sec \)) starting from \( t = 12hr \) through the end of the first orbital period \( t_f = 24hr \). The measurements are assigned Gaussian noise with covariance matrix \( \mathbf{R} = \text{diag}([25m, 0.1m/s, 0.1^\circ, 0.1^\circ]^2) \). The observer is assigned a FOV with half angle \( \gamma^\star = 7.5^\circ \) and a scale distance \( |\rho_s| = 1000km \) for computing detection probability. The nominal observer orbit at the start of the search \( t = 12hr \) is given by the relative RSW coordinate state vector given in Table 4.4.
<table>
<thead>
<tr>
<th>RSW State Vector</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x$</td>
<td>200 (km)</td>
</tr>
<tr>
<td>$y$</td>
<td>0 (km)</td>
</tr>
<tr>
<td>$z$</td>
<td>0.5 (km)</td>
</tr>
<tr>
<td>$\dot{x}$</td>
<td>0 (m/s)</td>
</tr>
<tr>
<td>$\dot{y}$</td>
<td>$-22$ (m/s)</td>
</tr>
<tr>
<td>$\dot{z}$</td>
<td>0 (m/s)</td>
</tr>
</tbody>
</table>

Table 4.4. Test Case 2: Observer State at t=12hr (RSW Coordinates)

Figure 4.9. Test Case 2: 5th Order Polynomial Reachability Set at t=12hr

Quadrature sets which satisfy moment constraint equations up to $d = 4, 6, 8, 10$ order are computed using DQ and used to evaluate reachability set coefficients for polynomial models of $D = 2, 3, 4, 5$ order respectively. The number of quadrature points and basis functions required for each model is given in Table 4.5. Once reachability set coefficients are computed, 10,000 samples $\zeta^{(i)}$ are drawn and evaluated using the polynomial model. The position of these samples in RSW coordinates at $t = 12hr$ is shown in Figure 4.9 for the 5th order polynomial model, where the colorbar depicts the maneuver time for a given sample.

Consider a single realization of the true target maneuver $\mathbf{u}_{t_1}^*$ and maneuver time $t_{1}^*$ given by

$$\mathbf{u}_{t_1}^* = [1.84, -2.54, 2.30]^T (m/s), \quad t_{1}^* = 5396(s)(+12hr)$$ (4.28)
Table 4.5. Test Case 2: Summary of Quadrature Sets for Single Maneuver Case (n=4)

<table>
<thead>
<tr>
<th>MCE Order</th>
<th>Number of Points</th>
<th>Model Basis Order</th>
<th>Number of Basis Functions</th>
</tr>
</thead>
<tbody>
<tr>
<td>($d$)</td>
<td>($N$)</td>
<td>($D$)</td>
<td>($L$)</td>
</tr>
<tr>
<td>4</td>
<td>16</td>
<td>2</td>
<td>15</td>
</tr>
<tr>
<td>6</td>
<td>47</td>
<td>3</td>
<td>35</td>
</tr>
<tr>
<td>8</td>
<td>110</td>
<td>4</td>
<td>70</td>
</tr>
<tr>
<td>10</td>
<td>244</td>
<td>5</td>
<td>126</td>
</tr>
</tbody>
</table>

Figure 4.10 depicts the evolution of the target pdf in the initial target orbital plane ($\hat{R}, \hat{S}$) using a $5^{th}$ order polynomial basis in 1 hour intervals. A subset of 100 samples are randomly selected and plotted over the contours of Figure 4.10. The initial observer state and true target maneuver were specifically selected for this test case to demonstrate a few key takeaways. First and foremost, notice that the true target state is captured within the contours of the pdf at all times. The pdf contours represent all possible states and maneuvers histories that could explain the measurements up to the current time. Secondly, the geometry of the observer and the reachability set means that the target will not be in the observer’s FOV when initially searching the most likely region for the target (near the origin). This is done to demonstrate how the contours of the target pdf become more concentrated around the true target state as vacant regions of the reachability set are searched, and by 17 hours into the simulation the true target location is centered in the most probable region of the reachability set. The largest reduction in uncertainty during the search procedure is, by far, at the time of first detection $t_d$. Therefore, $t_d$, as well as state and maneuver estimate errors are important performance metrics. The detection time for the $5^{th}$ order case is 18 hours and 25 minutes.

The state and maneuver estimates at both the time of first detection $t_d$ as well as the final simulation time $t_f$ are computed for all polynomial models using (4.22). The target position and velocity estimate errors $\epsilon_r$ and $\epsilon_v$, respectively, are defined as

$$\epsilon_r = \frac{||E[r_t] - r_t^*||_2}{||r_0||_2}, \quad \epsilon_v = \frac{||E[v_t] - v_t^*||_2}{||v_0||_2}$$  \hspace{1cm} (4.29)

where $r_0$ and $v_0$ are used to normalize the errors by the initial Cartesian position and velocity. The maneuver estimate errors are given by

$$\epsilon_u = ||E[u_{t,1}] - u_{t,1}^*||_2, \quad \epsilon_t = |E[t_1] - t_1^*|$$  \hspace{1cm} (4.30)
These errors are tabulated for the initial detection time as well as the final time $t_f = 24 hrs$ in Table 4.6 and Table 4.7, respectively. The error values for both the target state and maneuver estimate appear to decrease as the order of the polynomial model increases. Additionally, notice that the additional measurements after the initial detection help to improve the estimate at $t_f$ when compared to $t_d$.

Examining these results too closely is somewhat moot, as only looking at a single realization of the true target maneuver doesn’t encompass the full domain of the reachability set.
Figure 4.11. Test Case 2: Detection Time for 250 Simulations and Varying $D$

Rather than considering a single realization of the true target maneuver, now consider the full simulation run 250 times with a new true target maneuver randomly sampled for each simulation. The simulation is run using the same 250 realizations for all polynomial models. Figure 4.11 shows histograms of the time of first detection $t_d$ for varying polynomial model orders. From Figure 4.11, it can be observed that target is detected quickly in the majority of the simulations; however some target maneuvers cause the target to be undetected for longer. The simulations are run long enough that 100% of targets are detected, however increasing the polynomial order slightly reduces the time required to locate the target in these extreme cases. The target was detected in 95% of simulations by $t = 20.17hr$ for the model order $D = 2$, $t = 19.58hr$ for $D = 3$, $t = 19.17hr$ for $D = 4$, and $t = 18.58hr$ for $D = 5$.

Figure 4.12, and Figure 4.13 show histograms of log estimate error in maneuver magnitude $\epsilon_u$, and maneuver time $\epsilon_t$ respectively for varying polynomial model orders. It can be readily seen that the maneuver magnitude and maneuver time estimate errors are reduced by increasing polynomial order. Since reduction of detection time is only very modestly improved by increased polynomial order, selection of the polynomial order to be used is dependent on the application considered. If the only requirement is to locate the target, perhaps a low order reachability set can be used to define the reachable space more efficiently; however,
if maneuver reconstruction is required, higher accuracy reachability sets may be necessary to accurately estimate the maneuver. Table 4.8 summarizes the results of the 250 simulations by giving the mean error of the target state and maneuver estimates at the final time along with a $1\sigma$ plus or minus bounds.
Table 4.8. Test Case 2: Normalized Target State and Maneuver Estimate Error Summary at \( t_f \): Mean Error ± 1σ

<table>
<thead>
<tr>
<th>MCE Order ((d))</th>
<th>Number of Points ((N))</th>
<th>Model Basis Order ((D))</th>
<th>Number of Basis Functions ((L))</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>106</td>
<td>2</td>
<td>45</td>
</tr>
<tr>
<td>6</td>
<td>1,892</td>
<td>3</td>
<td>165</td>
</tr>
<tr>
<td>8</td>
<td>10,260</td>
<td>4</td>
<td>495</td>
</tr>
</tbody>
</table>

Table 4.9. Test Case 3: Summary of Quadrature Sets for Two Maneuver Case \((n=8)\)

4.5.3 Test Case 3: Two Maneuvers GEO

This section presents the results for a GEO case with two unknown maneuvers. All simulation parameters for this test case other than the number of maneuvers are identical to test case 2, including the initial target state and maneuver capabilities, and the observer state and observation zone parameters. For this test case, the random target input vector is given by

\[
z_t = [u_{t,1}, t_1, u_{t,2}, t_2,] \tag{4.31}
\]

Two major differences in this test case are the dimension of the input and the total \( \Delta V \) the target is allowed to make. Due to the curse of dimensionality, the number of points required to evaluate the reachability coefficients greatly exceeds that required for a 4-D input. In fact, 6\textsuperscript{th} and 8\textsuperscript{th} order quadrature points were unable to be found directly in eight dimensional space using the DQ algorithm, so a tensor product of lower dimensional sets had to be used. Table 4.9 lists the number of basis functions and number of quadrature points required to compute reachability sets for varying polynomial model order.

For brevity, only the results for the 4\textsuperscript{th} order polynomial model will be shown. Figure 4.14 depicts the target reachability set in RSW coordinates using a 4\textsuperscript{th} order polynomial model, where the colorbar indicates the average maneuver time. As is intuitively obvious, the larger total \( \Delta V \) available to the target makes the
Figure 4.14. Test Case 3: 4th Order Polynomial Reachability Set at t=12hr

Figure 4.15. Test Case 3: Evolution of Target PDF in Orbital Plane During Search (4th Order Polynomial Approximation)

physical size of the two maneuver reachability set considerably larger than the single maneuver set shown in Figure 4.9. The evolution of the pdf for a single realization of the two maneuvers is shown in Figure 4.15. The detection time for this realization was 15hr 45min.

The 4th order polynomial model is used to run 250 full simulations with a new true target maneuver sequence randomly sampled for each iteration. The error
in estimated maneuver magnitude \( \epsilon_u \) and maneuver time \( \epsilon_t \) for both unknown maneuvers at the final simulation time is given in Figure 4.16. It is apparent from Figure 4.16 that the maneuver estimates have greater error than the respective 4th order models in the single maneuver test case (Figure 4.12 and Figure 4.13). This is to be expected for several reasons. Firstly, the physically larger size of the reachability set will cause errors in the polynomial approximation to increase, thereby increasing error in the mapping between maneuvers and target states. Furthermore, the second maneuver creates the possibility for multiple maneuver sequences to achieve the same target trajectory. In this situation, it is impossible for the particle filter to disambiguate between possible maneuver histories and the filter may converge around an incorrect solution.

Despite the increased error in maneuver estimate and larger search region, using the proposed method the observer satellite was able to locate the target satellite in 96% of the 250 simulations run (i.e. 10 unfound targets). There are several reasons why some targets were not detected in this simulation. Firstly, the parameters of the observation zone and detection probability were kept the same as the single maneuver case, and due to the larger search region, portions of the reachability set further away from the observer were unable to be searched adequately. Secondly, the final simulation time was kept constant at 24hr so it may be the case that the observer simply didn’t have enough time to find the target in extreme cases.

Simulations where the target was not detected were removed from state and maneuver estimate calculations. The error in position \( \epsilon_r \) and velocity \( \epsilon_v \) estimate at
Table 4.10. Test Case 3: Normalized Target State and Maneuver Estimate Error Summary at $t_f$ for $4^{th}$ Order Polynomial Model: Mean Error±$1\sigma$

<table>
<thead>
<tr>
<th>Error</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\epsilon_{u1}$ (m/s)</td>
<td>1.4918±2.052</td>
</tr>
<tr>
<td>$\epsilon_{t1}$ (min)</td>
<td>173.36±175.05</td>
</tr>
<tr>
<td>$\epsilon_{u2}$ (m/s)</td>
<td>5.236±1.809</td>
</tr>
<tr>
<td>$\epsilon_{t2}$ (min)</td>
<td>170.63±176.24</td>
</tr>
<tr>
<td>$\epsilon_r$</td>
<td>3.388E-4±8.995E-4</td>
</tr>
<tr>
<td>$\epsilon_v$</td>
<td>4.370E-4±7.6567E-4</td>
</tr>
</tbody>
</table>

the final time are computed using the normalization given by (4.29). The average state and maneuver estimate errors at the final simulation time plus or minus $1\sigma$ is shown in Table 4.10. The average error in both maneuver and state estimates were higher for this case than the comparable $4^{th}$ order estimate errors in test case 1. This is to be expected due to the higher dimensionality and larger physical size of the reachability set.

### 4.6 Conclusions

This chapter presents a systematic search method for detecting a satellite that has been lost in space due to unknown maneuvers. A novel characterization of unknown maneuver bounds define the target search space, and a maximum detection likelihood approach is used to task sensors. Sensor observations of vacant regions of the search space are used to reduce uncertainty in target state, and inform future sensor tasking. In fact, the reachability set represents the set of all possible states and maneuver histories that can be explained by the measurements up to a given time. The true target state will always be a subset of this search space given that the bounds on input uncertainty are not violated. The proposed method provides a unified framework for search, detection, tracking, and maneuver estimation of a noncooperative target.

The success of the proposed search method, i.e. first detection time and percent of detected targets, is largely dependent on the observability of the problem. Observability is determined by many factors including the sensor FOV, detection likelihood, maneuver capabilities of the target, and the geometry of the observer with respect to the reachability set. Sensors with large FOV relative to the target
reachability set will easily locate the target, whereas a sensor with small FOV relative to a physically large reachable set may only be able to partially observe the reachability set. One method to increase observability of the problem is to consider a distributed network of sensors. Generalization of the single-sensor, single-target sensor tasking cases examined here to a distributed sensor network will almost certainly prove beneficial for increasing target observability.

Polynomial model order of the reachability set approximation is a flexible design parameter left to the discretion of the user. Higher-order polynomial models offer better target state and maneuver estimation accuracy at the expense of increased computational cost. If the only objective is to locate the target without regard for accurate maneuver reconstruction, it has been demonstrated that using low order polynomials to model the reachable space can provide similar detection time performance with lower computational cost. As discussed in Chapter 3, a limiting factor of the reachability set computation method is the assumed number of target maneuvers. Increasing the number of maneuvers, i.e. input dimension, causes exponential growth in the number of quadrature points required to compute reachability coefficients. This exponential growth in computational expense can be curbed using the non-product methods, however, even these methods have limitations. The following chapter will discuss a method for resampling reachability set quadrature points so that the input dimension does not increase with sequential maneuvers.
Chapter 5
Higher-Order Moment Matching

5.1 Introduction

A significant focus of this dissertation has been on uncertainty propagation for the computation of reachability sets. The HOSM method developed in Chapter 3 enables highly efficient and accurate propagation of initial condition, maneuver, and parametric uncertainty via sensitivity tensors; however, input dimensionality increases for every sequential control action. The numerical sampling methods, i.e. quadrature methods, used to compute the sensitivity coefficients for HOSM method drastically increase in computational complexity as input dimension increases. Tensor product methods incur an exponential growth as illustrated in Figure 5.1, and even the non-product methods discussed, such as DQ and CUT have limitations on curtailing this growth. This computational burden severely limits the applicability of the HOSM method to compute reachability sets with multiple or continuous maneuvers. Several alternative approaches to computing reachability sets, including convolution, principal of maximum entropy, and direct moment method are given in [105].

The higher-order moment matching (HOMM) method presented here, is equivalent to the direct moment method in [105] with an additional critical resampling step. This method was developed with the explicit purpose of circumventing the growth in input dimension with sequential timesteps for reachability set computation, and is composed of two parts 1) Direct computation of higher-order moments,
and 2) computation of quadrature points which match these moments. The first component of this method is straightforward. The higher-order moments of any generic stochastic system can be computed directly by separating the expectation values of independent variables and evaluating numerically. The second component of this method is to resample a set of quadrature points and weights which match the computed higher-order moments. The designed quadrature (DQ) method developed in Section 3.5 will be implemented for this purpose.

In addition to propagating maneuver uncertainty for reachability set computation, the HOMM method can be applied to the measurement update step in a Kalman filter. Typically, the post update mean and covariance can be computed (using either linear or nonlinear methods) and the posterior is redefined as Gaussian. Redefining the posterior in this manner effectively loses any information about the higher-order moments of the posterior, which in highly nonlinear or data-sparse applications can be significant.

This chapter is organized in the following manner. Section 5.2 provides a
description of the problem addressed in this chapter, and Section 5.3 discusses how higher-order moments for a generic system can be computed via separation of expectation values. Section 5.4.1 and Section 5.4.2 numerically validate the method for reachability set computation, and measurement update test cases, respectively. Section 5.4.3 combines the reachability set propagation and the measurement update steps to provide a framework for a fully automated numerical higher-order filtering procedure.

5.2 Problem Description

This section defines the scope and objective of the higher-order moment matching problem. This method applies to any stochastic system with independent random vectors, but in the context of this dissertation, the applications considered are uncertainty propagation and measurement update for satellite tracking.

Consider the unforced continuous-time dynamic model

\[ \dot{x} = f(x, t) \]  

where \( x \) is the \((n \times 1)\) target state vector and \( f \) are the dynamic equations of motion. The system flow \( \chi \) is given by the integrating the dynamics over the timestep considered.

\[ \chi(x_k, k + 1) = \int_{t_k}^{t_{k+1}} f(x, t) dt \]  

Assume that at each timestep a discrete impulsive control is applied such that the state at \( t_{k+1} \) is given by

\[ x_{k+1} = \chi(x_k, k + 1) + g(u_k) \]  

where \( u \) is the \((l \times 1)\) impulsive control vector, and \( g \) is the control model. The HOSM method developed in Chapter 3 can be used to compute the desired order statistical moments of \( x_{k+1} \) and future states by explicitly accounting for uncertainties in \( z_k = [x_k^T, u_k^T] \). Non-product quadrature methods such as CUT and DQ are used to efficiently sample the random variable space defined by \( z_k \), and these minimal quadrature sets are also used to derive a surrogate polynomial model based on a Taylor series expansion.
Now consider that instead it is desired to map uncertainties to $x_{k+2}$. In this case, the Taylor series expansion must be computed with respect to the input vector $z_{k+1}^T = [x_k^T, u_k^T, u_{k+1}^T]$, which increases the dimension of the sampling space to be $(n + 2l)$. Proceeding in this manner causes the input dimension to increase linearly with every timestep considered.

Augmenting sequential maneuver vectors onto the input means that higher-order information of the target state can be automatically preserved; however, the curse of dimensionality quickly makes sampling this large input space infeasible. The traditional solution to this problem is to resample the $n$-D state at each timestep rather than sequentially augmenting maneuvers and sampling the $(n + kl)$-D input at the initial time. This is typically accomplished by computing the post-maneuver mean and covariance and redefining the distribution as Gaussian. Normalized Gaussian quadrature points can be linearly scaled to any posterior mean and covariance very efficiently without the need to resolve quadrature points online; however, resampling in this manner means that higher-order information about the pdf is lost between timesteps thereby reducing the fidelity of the estimate.

Given quadrature set $X_{x,k}$ which evaluates the moments of $x_k$, and quadrature set $X_{u,k}$ which evaluates the moments of $u_k$, the objective of this chapter is to develop a method to numerically resample quadrature set $X_{x,k+1}$ which retains higher-order moment information of $x_{k+1}$. This new quadrature set can then be used to propagate to the next timestep, and repeat the procedure while maintaining a constant state dimensionality. The HOMM method presented here consists of evaluation of arbitrary order moments of the post maneuver state, coupled with the DQ algorithm to resample quadrature sets which replicate the computed moments. The following section will discuss how to compute arbitrary higher-order moments of an additive stochastic system.

### 5.3 Higher-Order Moment Computation for Additive Systems

This section discusses computation of the higher-order moments of the post-maneuver target state. Given the dynamic system in (5.2) and quadrature sets in $x_k$ and $u_k$, the objective is to evaluate the moments of $x_{k+1}$ up to some arbitrary
order $d$

$$E[x_{\alpha_1,k+1}], \ E[x_{\alpha_1,k+1}x_{\alpha_2,k+1}], \ldots \ E[x_{\alpha_1,k+1}x_{\alpha_2,k+1} \ldots x_{\alpha_d,k+1}] \quad (5.4)$$

where $\alpha_1, \alpha_2, \ldots, \alpha_d = 1, 2, \ldots n$. Using quadrature sets $X_{x,k}, X_{u,k}$, the expectation values $E[\chi_{\alpha_1} \chi_{\alpha_1} \ldots \chi_{\alpha_d}]$ and $E[g_{\alpha_1} g_{\alpha_1} \ldots g_{\alpha_d}]$ can be evaluated numerically. The time indices on the propagated prior state $x_{\alpha_1,k+1}$ will be dropped since expectation values in terms of the prior $x_{\alpha_1,k}$ may be distinguished by the system flow $\chi_{\alpha_1}$. The first two post-maneuver moments in $x$, i.e. mean and covariance, can be computed

$$\mu_{\alpha_1} = E[x_{\alpha_1}] = E[\chi_{\alpha_1}] + E[g_{\alpha_1}]$$

$$\Sigma_{\alpha_1 \alpha_2} = E[(x_{\alpha_1} - E[x_{\alpha_1}]) (x_{\alpha_2} - E[x_{\alpha_2}])]
\quad = (E[\chi_{\alpha_1} \chi_{\alpha_2}] - E[\chi_{\alpha_1}] E[\chi_{\alpha_2}]) + (E[g_{\alpha_1} g_{\alpha_2}] - E[g_{\alpha_1}] E[g_{\alpha_2}]) \quad (5.5)$$

To avoid numerical errors due to vastly different orders of magnitude when computing higher-order moments, the post-maneuver state is normalized to the zero-mean, unit variance variable $\zeta$

$$\zeta = (\sqrt{\Sigma})^{-1} (x - \mu)$$

$$= (\sqrt{\Sigma})^{-1} (\chi - E[\chi]) + (\sqrt{\Sigma})^{-1} (g - E[g])$$

$$= r + s \quad (5.6)$$

where $r$ and $s$ are normalized functions of the prior state and maneuver respectively. Any normalized moment up to arbitrary order $d$ can now be expressed as

$$E[\zeta_{\alpha_1} \zeta_{\alpha_2} \ldots \zeta_{\alpha_d}] = E(r_{\alpha_1} + s_{\alpha_1})(r_{\alpha_2} + s_{\alpha_2}) \ldots (r_{\alpha_d} + s_{\alpha_d}) \quad (5.7)$$

Since the functions $r$ and $s$ are independent, any expectation values containing both terms can be separated and evaluated separately. Using this fact, higher-order
moments up to 4th order are given by

$$
E[\zeta_1] = E[r_1] + E[s_1]
E[\zeta_1 \zeta_2] = E[r_1 r_2] + (E[r_1] E[s_2] + E[s_1] E[r_2]) + E[s_1 s_2]
E[\zeta_1 \zeta_2 \zeta_3] = E[r_1 r_2 r_3] + (E[r_1 r_2] E[s_3] + E[r_1 r_3] E[s_2] + E[r_2 r_3] E[s_1]) + (E[s_1 s_2] E[r_3] + E[s_1 s_3] E[r_2] + E[s_2 s_3] E[r_1]) + E[s_1 s_2 s_3]
$$

Writing out the higher-order moments explicitly can quickly become quite burdensome. The process can be automated by noticing that the expanded form of the higher-order moments has the form of a binomial expansion in multidimensional space. The expansion for a scalar binomial is given by

$$
(r + s)^d = \sum_{i=0}^{d} \binom{d}{i} r^{d-i} s^i
$$

where the combinatorial operator

$$
\binom{d}{i} = \frac{d!}{(d-i)i!}
$$

is a constant factor multiplying each monomial term. For example:

$$
(r + s)^4 = \binom{4}{4} r^4 + \binom{4}{3} r^3 s + \binom{4}{2} r^2 s^2 + \binom{4}{1} rs^3 + \binom{4}{0} s^4
$$

In multidimensional space this operator describes the number of dimension permutations for a given term of the binomial series. Take the 4th order moment
expansion given in (5.8) as an example. The term corresponding to \( r^3 s \) contains the following permutations:

\[
(E[r_{a_1}r_{a_2}r_{a_3}]E[s_{a_4}] + E[r_{a_1}r_{a_3}r_{a_4}]E[s_{a_2}] + E[r_{a_1}r_{a_3}r_{a_4}]E[s_{a_2}] + E[r_{a_2}r_{a_3}r_{a_4}]E[s_{a_1}])
\]

This term can be compactly written as

\[
(4)^*E[r_{a_1}r_{a_2}r_{a_3}]E[s_{a_4}]
\]

(5.12)

where the \((\cdot)^*\) operator denotes the sum of multidimensional permutations for that binomial term. Using this notation, the higher-order moments up to \( d \)th order can be concisely written as

\[
E[\zeta_{a_1}] = E[r_{a_1}] + E[s_{a_1}]
\]

\[
E[\zeta_{a_1}\zeta_{a_2}] = E[r_{a_1}r_{a_2}] + (2)^*(E[r_{a_1}]E[s_{a_2}]) + E[s_{a_1}s_{a_2}]
\]

\[
E[\zeta_{a_1}\zeta_{a_2}\zeta_{a_3}] = E[r_{a_1}r_{a_2}r_{a_3}] + (3)^*(E[r_{a_1}r_{a_2}]E[s_{a_3}]) + (3)^*E[s_{a_1}s_{a_2}]E[r_{a_3}] + E[s_{a_1}s_{a_2}s_{a_3}]
\]

\[
E[\zeta_{a_1}\zeta_{a_2}\zeta_{a_3}\zeta_{a_4}] = E[r_{a_1}r_{a_2}r_{a_3}r_{a_4}] + (4)^*E[r_{a_1}r_{a_2}r_{a_3}]E[s_{a_4}] + (6)^*E[r_{a_1}r_{a_2}]E[s_{a_3}s_{a_4}] + (4)^*E[s_{a_1}s_{a_2}s_{a_3}]E[r_{a_4}] + E[s_{a_1}s_{a_2}s_{a_3}s_{a_4}]
\]

\[
\vdots
\]

\[
E[\zeta_{a_1}\zeta_{a_2} \ldots \zeta_{a_d}] = \sum_{i=0}^{d} \binom{d}{i}^* E[r^{d-i}]E[s^i]
\]

(5.14)

Using this binomial structure, the process of evaluating the normalized higher-order moments can be automated up to any arbitrary order. It is very important to note the order of quadrature sets \( X_{x,k} \) and \( X_{u,k} \) when evaluating moments in \( \zeta \). For example, evaluation of the term \( E[r_{a_1}r_{a_2}r_{a_3}] \) up to he accuracy of a second order Taylor series requires that the quadrature set \( X_{x,k} \) evaluate up to \( 8 \)th order moments in \( x_k \).

As a direct result of this fact, if the quadrature set \( X_{x,k} \) evaluates up to \( d \)th order moments, then the \( d \)th moment of post maneuver state \( E[x_{a_1}x_{a_2} \ldots x_{a_d}] \) may only be evaluated up to \( 1 \)st order accuracy. This can introduce significant errors into higher-order moments, especially if there is long propagation times that make a first order approximation particularly poor.
This provides a framework for computing the higher-order moments of \( \zeta \); however, a new quadrature set \( \zeta^{(i)}, w^{(i)} \) which match these moments are required to propagate the state pdf to the next timestep. Unfortunately, since these moments are generic and non-symmetric, classical quadrature methods such as sparse grid, CUT and GH, are not applicable. The DQ method presented in Section 3.4 provides a framework for generic moment matching, and will be summarized here.

Assume a set of post-maneuver state moments are computed, and the objective is to compute a set of \( N \) points and weights which satisfy these MCEs

\[
\begin{bmatrix}
E[1] \\
E[\zeta_\alpha] \\
E[\zeta_\alpha \zeta_\beta] \\
\vdots \\
E[\zeta_\alpha \zeta_\beta \ldots \zeta_\gamma]
\end{bmatrix}
= 
\begin{bmatrix}
1 & 1 & \ldots & 1 \\
\zeta^{(1)}_\alpha & \zeta^{(2)}_\alpha & \ldots & \zeta^{(N)}_\alpha \\
\zeta^{(1)}_\alpha \zeta^{(1)}_\beta & \zeta^{(2)}_\alpha \zeta^{(2)}_\beta & \ldots & \zeta^{(N)}_\alpha \zeta^{(N)}_\beta \\
\vdots & \vdots & \ddots & \vdots \\
\zeta^{(1)}_\alpha \zeta^{(1)}_\beta \ldots \zeta^{(1)}_\gamma & \zeta^{(2)}_\alpha \zeta^{(2)}_\beta \ldots \zeta^{(2)}_\gamma & \ldots & \zeta^{(N)}_\alpha \zeta^{(N)}_\beta \ldots \zeta^{(N)}_\gamma
\end{bmatrix}
\begin{bmatrix}
w_1 \\
w_2 \\
\vdots \\
w_N
\end{bmatrix}
\tag{5.15}
\]

which can be expressed in compact form as

\[
R = Z(X)w - E
\tag{5.16}
\]

where \( E \) is the \((L \times 1)\) vector of moments, \( Z \) is an \((L \times N)\) matrix of polynomials evaluated at all quadrature points, \( w \) is an \((N \times 1)\) vector of weights, and \( R \) is the \((L \times 1)\) error vector. First, a cost function is defined as

\[
G = ||\hat{R}_k||_2^2 = ||R_k||_2^2 + c_k^2 \sum_{i=1}^{N_i} P_i(d)
\tag{5.17}
\]

where \( c_k \) is a scalar factor, \( d \) is a vector of all decision variables, i.e. points and weights, \( P_i \) are error terms for either weight positivity constraints or point support constraints, and \( G \) is the augmented cost function. A standard unconstrained Gauss-Newton minimization approach can be used to update the decision variables such that iteratively solving a series of these minimization problems will drive the MCEs and constraints to be satisfied. Typically, once a converged set is found, points with the lowest weights are removed and the set is re-solved until a set with fewer points \( N \) than the current best set cannot be found.

DQ was previously used to directly construct mixed distribution quadrature
sets with known moments directly in $n$-D space; however, it can also be used to determine quadrature sets which match normalized higher-order moments of a generic system $\zeta^{(i)}$. Once normalized points have been computed, they can be re-scaled to the true post-maneuver state using

$$x_{k+1}^{(i)} = \mu + \sqrt{\Sigma} \zeta^{(i)}$$

(5.18)

These points can be used to propagate the post-maneuver state to the next timestep, as well as compute a polynomial surrogate model using HOSM method. Unfortunately, standard Hermite polynomials will no longer be orthogonal with respect to this distribution. The examples in this dissertation use simple monomial basis functions; however, in general the Gram-Schmidt method can be used to evaluate an orthogonal polynomial basis. This procedure can be repeated to compute the pdf after an arbitrary number of maneuvers with no increase in state dimension. A similar procedure can be used in a Kalman filtering context to update the higher-order moments of the posterior.

### 5.3.1 HOMM for Measurement Update

The higher-order moment computation method described above can be used for other applications as well. Consider the Kalman measurement update given by

$$y_{k+1} = h(x_{k+1}^-) + \nu_{k+1}$$
$$x_{k+1}^+ = x_{k+1}^- + K_{k+1}(\hat{y} - y_{k+1})$$

(5.19)

where $y$ is the measurement, $h$ is measurement model, $\nu$ is the measurement noise, and $x^-, x^+$ are the prior and posterior states respectively. Higher-order moment information can be automatically preserved between measurement updates by augmenting measurement noise onto the input vector for each timestep $z = [x_{k+1}^- T \nu_{k+1} T \nu_{k+2} T \ldots]^T$ and directly mapping input to output similar to how multiple maneuvers were augmented to the input using the HOSM method. Alternatively, the posterior state can be resampled at each timestep. Recall from Chapter 2 that
the posterior mean and covariance for a Kalman update is given by

\[
\begin{align*}
\mu_{k+1}^+ &= \mu_{k+1}^- + K_{k+1}(\hat{y}_{k+1} - E[h(x_{k+1}^-)]) \\
\Sigma_{k+1}^+ &= \Sigma_{k+1}^- - K_{k+1}\Sigma_{k+1}^{(xy)}
\end{align*}
\]  

where Kalman gain and variance matrices are given by

\[
\begin{align*}
K_{k+1} &= \Sigma_{k+1}^{(xy)}\left(\Sigma_{k+1}^{(yy)}\right)^{-1} \\
\Sigma_{k+1}^{(xy)} &= E[x_{k+1}^- h(x_{k+1}^-)^T] - E[x_{k+1}^-]E[h(x_{k+1}^-)^T] \\
\Sigma_{k+1}^{(yy)} &= E[h(x_{k+1}^-)h(x_{k+1}^-)^T] - E[h(x_{k+1}^-)]E[h(x_{k+1}^-)^T] + R_{k+1}
\end{align*}
\]  

The traditional resampling procedure, like that of the unscented and conjugate unscented Kalman filters, is to use the mean and covariance to linearly re-scale normalized Gaussian quadrature points. A natural extension of this procedure is to evaluate higher-order moments of the posterior state and resample points which match these moments. The third order moment, i.e. the skewness tensor \( \Phi^+ \), can be written by dropping the time indices \( k + 1 \) and using dimension index notation as

\[
\begin{align*}
\Phi_{<\alpha\beta\gamma}^+ &= \Phi_{a\beta\gamma}^- - K_{\alpha\theta}\Phi_{a\beta\gamma}^{(yxx)} - K_{\beta\theta}\Phi_{a\beta\gamma}^{(yx)} - K_{\gamma\theta}\Phi_{a\beta\gamma}^{(yy)} + K_{\alpha\theta}K_{\beta\theta}\Phi_{a\beta\gamma}^{(yy)} + K_{\alpha\theta}K_{\gamma\theta}\Phi_{a\beta\gamma}^{(yy)} \\
&+ K_{\beta\theta}K_{\gamma\theta}\Phi_{a\beta\gamma}^{(yy)} + K_{\alpha\theta}K_{\beta\theta}K_{\gamma\theta}\Phi_{a\beta\gamma}^{(yy)}
\end{align*}
\]  

where the third order variational tensors can be written as

\[
\begin{align*}
\Phi_{a\beta\gamma}^{(yxx)} &= E[h_a(x^-)x_\beta^-x_\gamma^-] - E[h_a(x^-)x_\beta^-]E[x_\gamma^-] - E[h_a(x^-)]E[x_\beta^-]x_\gamma^-] + 2E[h_a(x^-)]E[x_\beta^-]E[x_\gamma^-] \\
\Phi_{ab\gamma}^{(yyx)} &= E[h_a(x^-)h_b(x^-)x_\gamma^-] - E[h_a(x^-)h_b(x^-)]E[x_\gamma^-] - E[h_a(x^-)]E[x_\gamma^-]x_\gamma^-] + 2E[h_a(x^-)]E[h_b(x^-)]E[x_\gamma^-] \\
\Phi_{abc}^{(yyy)} &= E[h_a(x^-)h_b(x^-)h_c(x^-)] - E[h_a(x^-)h_b(x^-)]E[h_c(x^-)] - E[h_a(x^-)]E[h_b(x^-)]E[h_c(x^-)] + 2E[h_a(x^-)]E[h_b(x^-)]E[h_c(x^-)]
\end{align*}
\]  

(5.23)
Note that since sensor noise is Gaussian, all odd-order noise terms are equal to zero. This formulation can be extended generically to the desired order moment, however, the notation quickly becomes clunky and difficult to work with. Using the notation in Section 5.3, the Kalman update can be normalized and re-written in the same form as the impulsive satellite maneuver in (5.2)

\[
x_{k+1}^+ = \left[ x_{k+1}^- + K_{k+1}(\tilde{y} - h(x_{k+1}^-)) \right] + K_{k+1}\nu_{k+1} - \chi(x_{k+1}^-) - g(\nu_{k+1})
\]

where \(\chi(x_{k+1}^-)\) and \(g(\nu_{k+1})\) are analogous to the system flow and the maneuver model from the previous section respectively.

The following sections will present examples of this method applied to both reachability set propagation for a multiple maneuver trajectory, as well as higher-order Kalman filter update. Taken together, these two problems constitute a fully nonlinear, constant dimension filter, termed the higher-order filter (HOF). The HOF can be used for accurate estimation in highly nonlinear and data-sparse applications. A HOF proof of concept will be shown using a simple duffing oscillator system, as well as a discussion of the limitations and unresolved numerical problems.

### 5.4 Numerical Simulations

This section presents numerical simulations for three test cases. The first test case considers application of the HOMM method to reachability set propagation for a cooperative three burn transfer to an inclined Earth orbit. The second test case considers HOMM method for measurement update of a target in an eccentric equatorial orbit. The third test case combines both reachability set propagation and measurement update for a toy duffing oscillator problem.

#### 5.4.1 Test Case 1: Multiple Maneuver Reachability Set

This section demonstrates how higher-order information can be maintained in a reachability set between maneuvers via application of the HOMM method. The example used here is that of a cooperative maneuvering satellite, and is based on the three-burn optimal transfer problem in Section 3.6.1, i.e. the 63.4° inclination transfer orbit given in [93]. For this example, target state \(\mathbf{x}\) of dimension \(n = 6\) is
given by the Cartesian position and velocity in ECI coordinates, and maneuver of
dimension \( m = 3 \) is given by the maneuver magnitude \(|\Delta v|\), pitch angle \( \theta \), and yaw
angle \( \psi \). Since the maneuvers are cooperative, maneuver times are considered to
be known deterministically and Gaussian uncertainty is prescribed in the target
maneuver magnitude and attitude angles. The nominal maneuver sequence is given
in Table 5.1, and the standard deviations assigned to each maneuver magnitude and
angles are \( \sigma_{|\Delta v|} = 2.5 \text{ m/s} \), \( \sigma_{\theta} = 0.5^\circ \), \( \sigma_{\psi} = 0.5^\circ \). The initial state is given by the
orbital elements in Table 5.2, and the full nominal three burn transfer trajectory is
depicted in Figure 5.2. The uncertainty in initial state is Gaussian with standard
deviation of \( \sigma_r = 50 \text{ m} \) and \( \sigma_v = 1 \text{ m/s} \) in Cartesian coordinates.

The system flow is given by two-body dynamics in \( (3.69) \), and the maneuver
model (depicted in Figure 3.6) is given by \( (3.70) \) and \( (3.71) \). In Chapter 3 only
the first two maneuvers were considered because the input dimension increased by
3 for every maneuver considered, and all CUT points are not available for \( n = 9 \)
inputs. Using HOMM method all three maneuvers can be included because the
system dimensionality does not increase as the number of maneuvers increases.

The following results depict the evolution of the target state using four reachabil-
ity set propagation methods 1) the unscented transform, 2) the 5\textsuperscript{th} order conjugate
unscented transform (CUT), 3) the 5\textsuperscript{th} order HOMM method, and 4) Monte Carlo
(MC) method using \( N = 1000 \). Both the UT and CUT methods use traditional
Gaussian resampling after each maneuver, the HOMM method resamples using
the method proposed in this chapter, and the MC method randomly samples the
entire maneuver sequence at the initial time and exactly evaluates the system

| Maneuver \( u_i \) | \(|\Delta v| \text{ (m/s)}\) | \( \theta \text{ (°)}\) | \( \psi \text{ (°)}\) | \( t_{m,i} \text{ (sec)}\) |
|-----------------|-----------------|--------|--------|-----------------|
| 1               | 2383.5          | -1.36  | 5.99   | 0               |
| 2               | 1235.7          | 7.21   | 74.7   | 6056.1          |
| 3               | 461.5           | -21.8  | 121.2  | 37370.0         |

Table 5.1. Test Case 1: Nominal Maneuver Sequence

<table>
<thead>
<tr>
<th>Semi-Major Axis</th>
<th>( a \text{ (m)})</th>
<th>6667320</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eccentricity</td>
<td>( e )</td>
<td>0</td>
</tr>
<tr>
<td>Inclination</td>
<td>( i \text{ (°)})</td>
<td>37.4</td>
</tr>
<tr>
<td>RAAN</td>
<td>( \Omega \text{ (°)})</td>
<td>0</td>
</tr>
<tr>
<td>Argument of Latitude</td>
<td>( \eta \text{ (°)})</td>
<td>255</td>
</tr>
</tbody>
</table>

Table 5.2. Test Case 1: Initial Orbital Elements
Figure 5.2. Test Case 1: Nominal Three-Burn Transfer

flow. The initial state quadrature set is initialized using Gaussian 5th order CUT points ($N_x = 76$), and the maneuver quadrature set is initialized as Gaussian 9th order CUT points ($N_u = 59$). Since $u_i$ is Gaussian at each maneuver, the more accurate 9th order set may be used for moment computation; however, only 5th order moments will be matched for the non-Gaussian state at each timestep.

Assume the first maneuver occurs $t = 0$. The post-maneuver mean and covariance are computed with (5.5) and can be used to normalize $x_0^{(i)}$ and $g(u_1^{(i)})$ with (5.6). These normalized functions are then used to compute up to 5th order moments of the post maneuver state, and the DQ algorithm is applied to resample a quadrature set which matches these moments. The post maneuver state quadrature set has $N_x = 71$ points, which is fewer points than the original 76 points required for the 5th order Gaussian CUT set. Figure 5.3 shows the normalized initial velocity quadrature points $v_0^{(i)}$, the normalized maneuver quadrature points $g(u_1^{(i)})$ and the resulting normalized post maneuver velocity quadrature points accurate up to 5th order MCEs.

Figure 5.4 depicts the resampled post-maneuver quadrature sets for all filters after each maneuver. The figures in the left column depict the UT and CUT methods which are resampled as Gaussian, and the figures on the right depict the HOMM quadrature points. The target pdf (PF samples) appear to be relatively
Figure 5.3. Test Case 1: Normalized Velocity Quadrature Sets for the First Maneuver

Gaussian after the first two maneuvers so the traditional UT and CUT resampling procedures still capture the posterior fairly well; however, due to long propagation times between the second and third maneuver the PF samples become noticeably non-Gaussian. Here, it can be clearly seen that the post-maneuver quadrature set using the HOMM method more accurately captures the true posterior pdf compared to the Gaussian resampled UT and CUT methods. Also note that as mentioned previously, the CUT method requires 76 points, whereas the HOMM method only has 71 points (after the 1st maneuver). The number of quadrature points required for the CUT method appear deceivingly small because of the symmetry of the CUT points and the plot is shown in only two out of the six state dimensions. After each maneuver the quadrature sets are propagated through the dynamic model under only gravitational forces. During this time no resampling is required and the state pdf may be approximated using the HOSM method to construct a polynomial model. Figure 5.5 shows the contours of the pdf approximated using a 2nd order polynomial model at two intermediate timesteps using both CUT and HOMM quadrature sets. Note that both quadrature sets match up to 5th order MCEs, with the only difference being that the CUT set is resampled as Gaussian after every maneuver. The contour plots depict the approximated pdf at $t = 7hr$, (between the second and third maneuver) and at $t = 13.5hr$ (after the third maneuver).
Figure 5.4. Test Case 1: Post Maneuver Reachability Sets at Each Maneuver
difference between the two approximations at $t = 7hr$ is indiscernible; however, there is a subtle difference between the two at $t = 13.5hr$. It may be surmised that the PF samples, i.e. reachability samples, at the extremes of the pdf are more closely captured by the contours approximated via HOMM resampling method. This difference would likely become even more noticeable as more maneuvers are added which would compound the error from Gaussian resampling.

This section has demonstrated how nonlinear reachability set propagation for sequential maneuvers can be accomplished using the HOMM method. The next section will discuss application of the HOMM method to the complimentary problem in target tracking, the measurement update.
5.4.2 Test Case 2: Higher Measurement Update

This section will provide an example of the HOMM method for nonlinear measurement update. Consider the problem of tracking a non-maneuvering target satellite with state $x_t$ using sensor measurements from an Earth-based sensor with state $x_{ob}$. The target dynamics are given by planar two-body equations of motion $f_t$ and system flow $\chi_t$

$$\dot{x}_t = f_t(x_t, t), \quad f(x_t, t) = \begin{bmatrix} \mathbf{v} \\ -\frac{\mu r}{|r|^3} \end{bmatrix}$$

$$x_{t,k+1} = \chi_t(x_{t,k}, k+1) = \int_{t_k}^{t_{k+1}} f_t(x_t, t) \, dt$$ (5.25)

where $\mu$ is the standard gravitational parameter of the Earth, and target state is given by planar position and velocity $x_T^T = [\mathbf{r}^T, \mathbf{v}^T]$ ($n = 4$) in ECI coordinates. Also assume that the Earth-based sensor has discrete-time dynamic model governing position $r_{ob}$ given by

$$x_{ob,k+1} = \chi_{ob}(x_{ob,k}, k+1) = \begin{bmatrix} R_e \cos(\theta_{k+1}) \\ R_e \sin(\theta_{k+1}) \end{bmatrix}, \quad \theta_{k+1} = \theta_k + \omega_e(t_{k+1} - t_k)$$ (5.26)

where $R_e$ is the radius of the Earth, $\theta_k$ is the current angle between the ground station and $\hat{I}$ direction, and $\omega_e$ is the rotation rate of the Earth. The ground station initial position is given by $\theta_0 = -15^\circ$. Assume a scalar ($m = 1$) angle measurement

$$y_{k+1} = h(\rho_{k+1}) + \nu_{k+1}, \quad h(\rho) = \tan^{-1}(\rho_2/\rho_1)$$ (5.27)

where $\rho$ is the relative state vector $x_t - x_{ob}$, and $\nu$ is zero-mean Gaussian measurement noise with standard deviation $\sigma = 0.1^\circ$. Since the measurement is scalar, $9^{th}$ order Gauss-Legendre points with be used for all moment computations with respect to the measurement noise. Assume that the initial target is in an elliptical orbit with period $P = 11.590\, \text{hrs}$ and nominal orbital parameters given in Table 5.3. These orbital elements correspond to a nominal initial state $x_0^T = [0(km), 11762(km), -5.8213(km/s), 4.3078(km/s)]$. Also assume initial state is prescribed a Gaussian uncertainty with covariance matrix $\Sigma_0 = \text{diag}([100(m), 100(m), 0.1(m/s), 0.1(m/s)]^2)$. The true deviation in initial state due
<table>
<thead>
<tr>
<th>Element</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Semi-Major Axis ($a$)</td>
<td>26000 (km)</td>
</tr>
<tr>
<td>Eccentricity ($e$)</td>
<td>0.74</td>
</tr>
<tr>
<td>Inclination ($i$)</td>
<td>0 (rad)</td>
</tr>
<tr>
<td>True Anomaly ($\theta$)</td>
<td>$\pi/2$ (rad)</td>
</tr>
<tr>
<td>Right Ascension of Ascending Node ($\Omega$)</td>
<td>0 (rad)</td>
</tr>
<tr>
<td>Argument of Periapse ($\omega$)</td>
<td>0 (rad)</td>
</tr>
</tbody>
</table>

Table 5.3. Test Case 2: Target Initial Orbital Elements

to uncertainty is $\Delta x^*_0 = [-37.7(m) - 10.49(m) 0.2668(m/s) - 0.0637(m/s)]^T$ which corresponds to an extreme realization of the initial condition for the prescribed $\Sigma_0$. This extreme initial condition uncertainty is selected to illustrate some of the issues associated with linear approximations and Gaussian resampling.

This simulation will be run with four filters for comparison: 1) the unscented Kalman filter (UKF), 2) 5th order conjugate unscented Kalman filter (CUKF), 3) 5th order higher order moment matching filter (HOMMF), and 4) a Bayesian particle filter (PF) with $N = 1000$ samples. The initial state is sampled using 3rd order Gaussian UT points for the UKF, 5th order Gaussian CUT points for both the CUKF and HOMMF, and randomly selecting 1000 MC points for the PF. The particle filter is updated using the point-wise Bayesian weight update given in Section 4.4 and all other filters are updated under the assumption of a best linear unbiased estimator, i.e. Kalman update, given in Section 2.3.4. Updating higher-order statistical moments of the posterior is most beneficial in data-sparse situations because nonlinearities have a greater effect on the higher-order moments of the state pdf. To illustrate this, assume the first measurement is taken at $t = 22.75 hr$ (approximately 2 full periods after the initial time) and another measurement is taken roughly one period later at $t = 35 hr$. At each of these measurements both the UKF and CUKF are traditionally resampled as Gaussian; however, the HOMMF is resampled by solving for quadrature points which match up to 5th order moments of the posterior using DQ.

Consider the first period of dynamic uncertainty propagation between the start of the simulation and the first measurement. The quadrature sets for each filter are normalized to zero-mean, unit variance variables every 15 minutes, and the moments up to 5th order are computed. Figure 5.6 shows these moments in x-position during the first two periods of propagation for all filters. The first and
second order normalized moments are, by definition, zero and one respectively (give or take error due to numerical precision). Notably, the UKF is not able to replicate the moments of the PF (considered ground truth) as accurately as the CUKF and HOMMF. This is a direct result of the UKF only matching up to $3^{rd}$ order moments whereas both CUKF and HOMMF match up to $5^{th}$ order moments. The propagated prior distribution at the time of the first measurement is illustrated.
in Figure 5.7 where a subset of 100 particle filter samples are plotted over the pdfs. Figure 5.7(a) depicts the contours of the prior pdf using the 3rd order UT points to compute a linear (Gaussian) model, and Figure 5.7(b) depicts the contours of the prior pdf using the 5th order CUT points to compute a 2nd order polynomial model. It is apparent that the explicitly propagated PF samples are not accurately captured by the Gaussian approximation of the propagated prior computed using UT points in Figure 5.7(a); however the contours from the 2nd order polynomial model computed using 5th order CUT points are able to capture the nonlinear pdf curvature much more accurately.

Now consider the 1st measurement update, where the true measurement is simulated by randomly selecting a realization of sensor noise. Figure 5.8 illustrates the measurement updated quadrature sets for all filters, where the ground truth, i.e. particle filter samples, are shown are in both plots for comparison. Figure 5.8(a) shows both the posterior UKF and CUKF resampled as Gaussian where the covariance ellipses indicate a 97% confidence interval, and Figure 5.8(b) shows the posterior HOMMF resampled up to 5th order accuracy. Note that since the true state is an extreme realization, even the more accurately propagated prior (from 5th order CUT points) indicates that the true target state lies in a unlikely region of the pdf. As a result, the traditional Gaussian resampling procedure used for UKF and CUKF does not accurately capture the true target state; however, the higher-order information maintained in the HOMM procedure enables the posterior HOMMF quadrature set to capture the true target state.
Figure 5.8. Test Case 2: Posterior Quadrature Sets at 1st Measurement

Note that in Figure 5.8 the particle filter samples do not appear to produce the same posterior pdf as the filters using a linear Kalman update. This is because the Bayesian particle filter update is a fundamentally different type of update from the approximate linearized Kalman update. Consequently, the spread of the posterior PF is determined by resampling particles only near prior samples with the highest measurement likelihood. In contrast, the Kalman updated filters have a covariance that is a direct function of the both the prior and measurement covariances.

All filter quadrature sets are propagated to the second measurement at $t = 35\text{hr}$, and undergo identical update procedures. Figure 5.9 depicts the post update quadrature points after the second measurement update for all filters. It is apparent that the UKF and CUKF resampled as Gaussian after the first measurement are beginning to diverge even further from the true target state; however since the HOMMF was resampled as non-Gaussian at the first measurement, the propagated pdf and the posterior at the second measurement are able to converge towards the true target state.

This test case demonstrates the benefit of applying the HOMM method to a Kalman filter measurement update for sparsely taken measurements. Although there is a benefit to the HOMM method applied to data-sparse problems, this comes at the cost of computational speed. Since the traditional method of resampling simply requires re-scaling Gaussian quadrature sets, they are far more efficient than running the DQ algorithm to directly solve for quadrature points at each timestep. For example, computing the posterior quadrature points for the CUKF at the first timestep took 0.003623 seconds whereas computing the posterior quadrature points
using the HOMM method took 8.78344 seconds. This increase in computational time is several orders of magnitude and is a result of the iterative DQ algorithm; however, considering that this method is only recommended for data-sparse applications, the increase in computational time may be acceptable. The following section will combine the measurement update shown here with the reachability set propagation application presented in the previous section. When combined, these two applications comprise a fully nonlinear numerical filtering cycle.

### 5.4.3 Test Case 3: Higher-Order Filter

This section will combine the previous two applications, i.e., reachability set propagation and measurement update, to demonstrate implementation of a fully nonlinear HOF. This case involves matching higher-order moments twice in every filter cycle: once to account for the control during dynamic propagation, and the second to account for measurement noise during the update. The example shown here will serve as a simple proof of concept case using an \( n = 2 \) duffing oscillator system. Duffing oscillators are notoriously chaotic systems and is thus used as a test case for high dynamic nonlinearity.

Consider the differential equation governing the motion of a forced duffing oscillator

\[
\ddot{x} + \delta \dot{x} + \beta x + \alpha x^3 = u \tag{5.28}
\]

where \( \alpha, \beta, \) and \( \delta \) are model parameters, and \( u \) is the control. This model can be written in state space form where the state is given by dimensionless position and
velocity coordinates $x^T = [x\ x_3]^T$

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

$$f(x, t) = \begin{bmatrix} \dot{x} \\ -\delta \dot{x} - \beta x - \alpha x^3 \end{bmatrix}, \quad g(u, t) = \begin{bmatrix} 0 \\ u \end{bmatrix}$$ (5.29)

Assume that the initial state is Gaussian $x_0 \in \mathcal{N} (\mu_x, \Sigma_x)$ with mean $\mu_x^T = [0, 1]^T$, covariance $\Sigma_x = diag([0.05, 0.05]^2)$, and that the control is zero-mean Gaussian $u \in \mathcal{N} (0, \sigma_u^2)$ with covariance $\sigma_u^2 = 0.05^2$. Using the continuous system discretization procedure found in [106], a discrete time dynamic model can be written as

$$x_{k+1} = \chi(x_k, k) + g(u_k, k)$$

$$\chi(x_k, k) = \int_{t_k}^{t_{k+1}} f(x, t)dt, \quad g(x_k, k) = \begin{bmatrix} 0 \\ u_d \end{bmatrix}$$ (5.30)

where the discrete-time control is Gaussian $u_d \in \mathcal{N}(0, \sigma_u^2 \Delta t)$. Also assume that $m = 2$ measurements $y_{k+1}$ are available at times $t_{k+1}$, given by the nonlinear measurement model

$$y_{k+1} = h(x_{k+1}) + \nu_{k+1}, \quad h(x_{k+1}) = \begin{bmatrix} x_3 \\ \dot{x}_3 \end{bmatrix}$$ (5.31)

where $\nu$ is Gaussian measurement noise $\nu \in \mathcal{N}(0, \Sigma_\nu)$. Nominal parameter values for this simulation are given by $\Delta t = 2s$, $\alpha = 1$, $\beta = -1$, $\delta = 0.2$, and measurement noise is assigned covariance $\Sigma_\nu = diag([0.01, 0.01]^2)$. The true initial state $x_0^* = [2.5058, 1.4860]^T$ is selected intentionally to represent an extreme realization of the initial state uncertainty. Similar to test case 2, several different filters will be run in parallel and compared. The filters used in this example are a particle filter with $N = 1000$, a UKF, a $4^{th}$ order CUKF, and a $5^{th}$ HOF. The CUKF and HOF both replicate $5^{th}$ order moments during dynamic propagation and measurement update; however, the estimates for these filters will diverge over multiple timesteps due to the cumulative effect of including $5^{th}$ order moments in the resampling process of the HOF.

Figure 5.10 shows the propagated prior and posterior sample points for all filters over the first three timesteps. The ellipses represent 97% confidence intervals based
on the covariance of each filter. Since the true state was intentionally selected as an extreme realization, none of the filters are able to accurately capture the true target within the 97% confidence interval at the propagated prior. In fact, after the first measurement update at \( t_1 \), the posteriors still are not able to accurately estimate the true state; however, it can be seen that the HOF quadrature points skew towards the true state. Over the next several timesteps the HOF and PF are the only filters which are able to accurately capture the true state within the covariance bounds.

Error \( \epsilon \) is defined as the true state minus the estimated mean for a given filter

\[
\epsilon_{\text{filter}, k} = x^* - \sum_{i=1}^{N} w^{(i)} x^{(i)}_{\text{filter}, k} \tag{5.32}
\]

The simulation is run for a total of 50 seconds, and Figure 5.11 shows the absolute errors for all filters on a logarithm scale vs time. Notice that in the beginning of the simulation when there is larger uncertainty due to the initial state, the PF and HOF outperform the UKF and CUKF which are resampled as Gaussian. This is unsurprising since both the dynamic and measurement models are nonlinear which means that statistical information is lost every time the propagated prior or posterior are resampled as Gaussian. Table 5.4 and Table 5.5 give the RMSE values of the propagated prior and posterior distributions for all filters over all timesteps. It is apparent that the HOF performed noticeably better, and closer to the PF (considered ground truth) than either the CUKF or the UKF for this problem.

There are some notable practical considerations when implementing this method. The first is the order of moments being matched. The duffing oscillator examined here is only two dimensional, so using the DQ algorithm to solve for quadrature points is feasible (even up to higher than the 5th order considered here); however, increasing state dimension dramatically increases the computational burden of

\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|}
\hline
 & Prior & Posterior \\
\hline
PF & 0.024817 & 0.024279 \\
HOF & 0.066035 & 0.039163 \\
UKF & 0.093023 & 0.055648 \\
CUKF & 0.096939 & 0.051507 \\
\hline
\end{tabular}
\caption{Position RMSE}
\end{table}

\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|}
\hline
 & Prior & Posterior \\
\hline
PF & 0.119950 & 0.067344 \\
HOF & 0.133259 & 0.074859 \\
UKF & 0.173250 & 0.085635 \\
CUKF & 0.169689 & 0.127731 \\
\hline
\end{tabular}
\caption{Velocity RMSE}
\end{table}
Figure 5.10. Test Case 3: Prior and Posterior Distributions for the First Three Timesteps.
Figure 5.11. Test Case 3: Prior and Posterior Estimation Errors for All Filters vs Time

solving for higher-order quadrature points. Therefore, depending on the system, real time implementation of this method up to the desired order may be infeasible.

The second consideration is numerical precision, and errors accumulating from moment evaluation. As higher-order moments of the posterior are included, the order of magnitude of moments (even when normalized) can vary considerably. This can cause numerical errors to accumulate both when computing higher-order moments, and when solving for DQ points which match these moments. In has been observed through numerical experimentation that continuing to use the HOMM method can cause numerical errors to accumulate and over-exaggerate the influence of non-Gaussian moments. In some cases this can lead to filter divergence.
The last consideration is that depending on the nature of the system considered, non-Gaussian effects may not always be important. In fact, for many astrodynamics applications, non-Gaussian effects (i.e. non-zero odd-order moments) are only significant in a fraction of the orbital period. In cases where non-Gaussian effects are insignificant it may be beneficial to resample as Gaussian, both for speed and to avoid accumulating numerical errors in higher-order moments as mentioned above.

5.5 Conclusion

This chapter has developed the HOMM method for nonlinear filtering applications. Traditionally, maintaining higher-order information in stochastic systems with sequential random variables requires appending the state vector with these variables at each timestep. Alternatively, the state can be redefined as Gaussian at each timestep, at the expense of losing higher-order information. Resampling the state density as Gaussian can cause errors in state estimate for extreme realizations of uncertainty especially when coupled with high model nonlinearity and long propagation times. The HOMM method can be applied individually to uncertainty propagation or measurement update problems, or applied to both to comprise a closed-loop HOF. The numerical examples shown in this chapter validate the utility of this method for highly nonlinear dynamic and measurement models; however, there are some important practical considerations. Considerations worth noting include the practicality of real-time implementation based on moment matching order and state dimension, as well as the numerical stability of moment computations. Additional research to improve the speed of the DQ and to determine conditions which indicate whether to match HOMs or to resample as Gaussian may prove beneficial.
Chapter 6  
Cislunar Applications

6.1 Introduction

The last 5 years or so has seen an explosion of interest in lunar and cislunar missions. NASA’s Artemis program is a planned series of missions aimed at returning the US to the moon and enabling a sustained human-robotic presence for research and exploration purposes [107]. The Artemis program relies on complex mission architecture including joint government and industry partnerships the space launch system (SLS) and Orion capsule, construction of a lunar gateway for staging and logistics, and the human landing system (HLS) for enabling surface missions [108–110].

Other countries have also either already begun, or announced lunar missions for the coming decade including China [13,14], Russia [15,16], India [17], and Japan [18]. The myriad of missions being launched in the near future, as well as the potential for military and intelligence utilization of cislunar space has created an urgent need for expanded space domain awareness (SDA) in the cislunar domain [19].

Unfortunately, there are significant challenges which limit current capabilities including 1) Data-sparsity from limited coverage and availability of sensor resources, 2) Low sensor signal to noise ratio (SNR) due to vast distances and atmospheric blurring effects of ground-based sensors, and 3) The sheer volume of the cislunar domain [20] (roughly 1000 times the volume of the traditional SDA domain out to geosynchronous orbit). All of the aforementioned difficulties are greatly exacerbated, when considering the maneuver capabilities of noncooperative satellites.

Recently work has been done to apply some traditional target tracking methods
discussed in chapter 3 to the problem of maneuvering satellites in the cislunar domain. One such example is the application of the Optimal Control Based Estimator (OCBE), i.e. minimum model error method, to identify unmodelled maneuvers between measurements for targets in cislunar space [111]. This method assumes that custody of the target can be maintained between measurements; however, issues associated with data-sparsity and the vast volume of cislunar space create an acute potential for noncooperative maneuvering satellites to be lost, thereby invalidating this assumption. This gap in the literature provides a rich potential for application of the methods developed in this dissertation to the cislunar domain.

Many of the planned lunar missions cited above make use of well-known dynamical structures known as Lagrange point orbits. Lagrange points represent dynamical points of equilibrium, and extensive research has been done into designing stable periodic orbits by leveraging these special structures [12,112–115]. This chapter seeks to address the issue of tracking a noncooperative satellite in cislunar space with a specific focus on orbital regimes near Lagrange point orbits. The examples presented specifically consider the problem of tracking a noncooperative maneuvering satellite initially in a stable L1 halo orbit.

The organization of Chapter 6 is as follows. Section 6.2 provides equations of motion for cislunar space and defines an observer and measurement model. Section 6.4 provides examples of both pure reachability set propagation as well as a reachability set search scenario for a cislunar target. Section 6.5 provides concluding remarks.

### 6.2 The Circular Restricted Three Body Problem

All scenarios discussed in previous chapters either used two body or J2-perturbed two body dynamical models because the satellites operated close to Earth. When satellite orbits extend past the GEO belt into the cislunar domain, the dominant perturbation becomes third body effects from the moon. This section discusses the circular restricted three body problem (CRTBP) and some important features of the new dynamical model.

In this problem, two primary masses $M_1$ and $M_2$ are in circular orbits about their common barycenter, and that a third much smaller object with negligible
mass, i.e. a satellite, moves under the influence of these two bodies. Characteristic units are defined to non-dimensionalize the problem such that the distance unit is the semimajor axis of the orbit of the primaries $DU = a$, the mass unit is the total mass of the two primaries $MU = M_1 + M_2$ and the time unit is selected such that the gravitational constant equal to one $TU = \sqrt{\frac{DU^3}{GMU}}$. Typically, the smaller mass is assigned as $M_2, M_2 \leq M_1$ and a mass ratio $\mu$ can be be defined

$$\mu = \frac{M_2}{MU} \quad (6.1)$$

A rotating reference frame is defined where the barycenter of the system is defined as the origin, and both primaries lie on the x-axis as depicted in Figure 6.1. $r_1$ and $r_2$ are the relative position vectors between the third body and masses $M_1$ and $M_2$ respectively, and $r_s$ is the position of the satellite with respect to the barycenter.

The equations of motion derived in [116] are given as

$$\ddot{x} = x + 2\dot{y} - \frac{1 - \mu}{r_1^3}(x + \mu) - \frac{\mu}{r_2^3}(x - (1 - \mu))$$

$$\ddot{y} = y - 2\dot{x} - \frac{1 - \mu}{r_1^3}y - \frac{\mu}{r_2^3}y$$

$$\ddot{z} = -\frac{1 - \mu}{r_1^3}z - \frac{\mu}{r_2^3}z \quad (6.2)$$
where state is given by $\mathbf{x} = [x\ y\ z\ \dot{x}\ \dot{y}\ \dot{z}]^T$, and $r_1, r_2$ are the magnitudes of $\mathbf{r}_1, \mathbf{r}_2$ respectively.

Lagrange points, i.e. equilibrium points, exist where all velocity and acceleration terms are equal to zero. It can be immediately observed that equilibrium points may only occur in the x-y plane, because out of plane positions will always have non-zero acceleration in the $z$ direction. If $y$ is also set as zero, the equation for $\ddot{x}$ provides a quintic function in $x$ which can be reformulated into equations for three unique colinear Lagrange points, $L_1$, $L_2$, and $L_3$ at distances $\gamma_1, \gamma_2$ and $\gamma_3$ from their nearest primaries.

$$
1 - \mu - \gamma_1 = \frac{1 - \mu}{(1 - \gamma_1)^2} - \frac{\mu}{\gamma_1^2} \\
1 - \mu + \gamma_2 = \frac{1 - \mu}{(1 + \gamma_2)^2} + \frac{\mu}{\gamma_2^2} \\
\mu + \gamma_3 = \frac{1 - \mu}{\gamma_3^2} + \frac{\mu}{(\gamma_3 + 1)^2}
$$

These equations can be solved numerically to get the colinear Lagrange points. For the Earth-Moon system colinear Lagrange points occur at $x_{L_1} = 0.8369$, $x_{L_2} = 1.1557$, and $x_{L_3} = -1.0051$. Examining the equation for $\ddot{y}$, it can be seen that other equilibrium points occurs when all velocities are zero, and $r_1 = r_2 = 1$. These are the $L_4$ and $L_5$ points and are sometimes referred to as the equilateral points. Figure 6.2 depicts all Lagrange points for the CRTBP. The following section
will discuss periodic orbits around these Lagrange points with a particular focus on halo orbits.

6.3 Periodic Orbits in the CRTBP

There has been extensive research performed into determining periodic orbits in the CRTBP, which would enable space stations to be placed in cislunar space for logistical or refueling purposes. Since closed-form analytical solutions for the CRTBP don’t exist, typically they are found via numerical integration. It can be quite difficult to determine initial conditions which lead to periodic orbits, and they are typically found using a combination of analytical approximation and numerical correction method. Some well-known families of these orbits are the Lyapunov orbits, distant retrograde orbits (DROs) [117–122], and halo orbits which will be examined here.

Halo orbits are a family of three dimensional out-of-plane orbits which exist around either the $L_1$ or $L_2$ points. Halo orbits are symmetric about the x-z plane, which is a key characteristic often used to solve for their initial conditions. Approximation of the initial conditions of a halo orbit can be done effectively by linearizing the CRTBP about the Lagrange point considered, and computing a third order analytical expansion using the method in [112]. Once reasonable initial conditions are found, a differential correction (DC) method can be used to refine the guess. The following will outline the basic equations for the DC method used.

Consider a linearized model of the CRTBP dynamics

\[ \delta \dot{x} = A(x^*, t) \delta x \tag{6.4} \]

where $x^*$ is the linearization point, and $A$ is the time-varying Jacobian matrix of the dynamics. The Jacobian is given by

\[
A(x^*, t) = \begin{bmatrix}
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 1 \\
a_{xx} & a_{xy} & a_{xz} & 0 & 2 & 0 \\
a_{xy} & a_{yy} & a_{yz} & -2 & 0 & 0 \\
a_{xz} & a_{yz} & a_{zz} & 0 & 0 & 0
\end{bmatrix} \tag{6.5}
\]
where the $a_{ij}$ terms are partial derivatives of acceleration with respect to position

\[
\begin{align*}
    a_{xx} &= 1 - \frac{(1 - \mu)}{r_1^3} - \frac{\mu}{r_2^3} + \frac{3(1 - \mu)(x + \mu)^2}{r_1^5} + \frac{3\mu(x - (1 - \mu))^2}{r_2^5} \\
    a_{yy} &= 1 - \frac{(1 - \mu)}{r_1^3} - \frac{\mu}{r_2^3} + \frac{3(1 - \mu)y^2}{r_1^5} + \frac{3\mu y^2}{r_2^5} \\
    a_{zz} &= -\frac{(1 - \mu)}{r_1^3} - \frac{\mu}{r_2^3} + \frac{3(1 - \mu)z^2}{r_1^5} + \frac{3\mu z^2}{r_2^5} \\
    a_{xy} &= \frac{3(1 - \mu)(x + \mu)y}{r_1^5} + \frac{3\mu(x - (1 - \mu))y}{r_2^5} \\
    a_{xz} &= \frac{3(1 - \mu)(x + \mu)z}{r_1^5} + \frac{3\mu(x - (1 - \mu))z}{r_2^5} \\
    a_{yz} &= \frac{3(1 - \mu)yz}{r_1^5} + \frac{3\mu yz}{r_2^5}
\end{align*}
\]  

(6.6)

Since $A(x^*, t)$ is time varying, (6.4) cannot be solved analytically; however, a general solution to the linearized system is given by

\[\delta x(t) = \Phi(t, t_0)\delta x(t_0)\]  

(6.7)

where $\Phi(t, t_0)$ is a $(6 \times 6)$ matrix known as the state transition matrix (STM). The STM maps the initial deviation in state to the final deviation and satisfies the following properties

\[\dot{\Phi}(t, t_0) = A(x^*, t)\Phi(t, t_0), \quad \Phi(t_0, t_0) = I\]  

(6.8)

Therefore, using the initial condition $\Phi(t_0, t_0) = I$, the STM can be computed by explicitly integrating along with the dynamic equations of motion. This gives a direct expression for determining how much to update the initial conditions of a halo orbit $\delta x(t_0)$ based on the deviation from the desired orbit at time $t$.

The DC procedure can be simplified by leveraging some characteristics of halo orbits. Firstly, the fact that these orbits are symmetric about the x-z axis means that when the satellite is passing through this plane, the velocity is orthogonal to the plane. This means that the state can be initialized at the plane crossing by $x(t_0) = [x_0 \ 0 \ z_0 \ 0 \ \dot{y}_0 \ 0]^T$, and that at the half-period plane crossing the desired state will be $x(P/2) = [x \ 0 \ z \ 0 \ \dot{y} \ 0]^T$. Using the method in [123], the halo orbit initial conditions can be corrected by fixing the $z$ amplitude $A_z$ and differentially correcting...
the initial $x$ position and $y$ velocity. The final equations for this correction are given by

\[
\begin{bmatrix}
\delta x_0 \\
\delta y_0 \\
\end{bmatrix} = \begin{bmatrix}
\phi_{41} - \phi_{21} \frac{\ddot{x}(P/2)}{\dot{y}(P/2)} & \phi_{45} - \phi_{25} \frac{\ddot{z}(P/2)}{\dot{y}(P/2)} \\
\phi_{61} - \phi_{21} \frac{\ddot{y}(P/2)}{\dot{y}(P/2)} & \phi_{65} - \phi_{25} \frac{\ddot{z}(P/2)}{\dot{y}(P/2)} \\
\end{bmatrix} \begin{bmatrix}
-\dot{x}(P/2) \\
-\dot{z}(P/2) \\
\end{bmatrix}
\]  

(6.9)

Figure 6.3 depicts the family of $L_1$ halo orbits, and the following section will provide numerical simulations to validate the HOSM method and reachability set search method for a target in one of these orbits.

### 6.4 Numerical Simulations

This section serves to validate the SDA framework proposed in this dissertation via numerical simulations, and demonstrates their applicability to real-world cis-lunar problems. Consider a target initially in an $L_1$ halo orbit of amplitude $A_z = 60,000 (km)$, depicted in Figure 6.4. The first test case demonstrates reachability set computation for this target using the HOSM method, and the second test case shows how a space-based GEO sensor can be used to search for this target.
6.4.1 Test Case 1: Reachability Set Propagation

Assume there is a target initially in an $L_1$ halo orbit with $z$ amplitude $A_z = 60,000km$ and initial state given by $x_0 = [0.8395 0 0.1561 0 0.2607 0]^T$ where all units are dimensionless. This orbit corresponds to an orbital period of $P = 2.7139$ in dimensionless units, or roughly 11.78 days. It is known that the target has maneuver capabilities of $\Delta v_{\text{max}} = 0.01$ which corresponds to roughly $10.25(m/s)$ of delta-v. The objective of this simulation is to compute the reachability set for an unknown target maneuver magnitude defined as spherically uniform $u_{t,1} \in U(0, \Delta v_{\text{max}})$ and maneuver time defined as uniform over the first orbital period $t_m \in U(0, P)$.

$$u = [u_{t,1} \ t_{m,1}] \quad (6.10)$$

Using the HOSM method and monomial basis functions, various polynomial models of orders $D = 2, 3, 4, 5$ are computed at various timesteps throughout the first orbital period. The same quadrature sets computed in Section 4.5.1 can be used since the maneuvers are defined using the same distributions for magnitude and maneuver time. The number of basis functions $L$ and number of quadrature points required for these sets can be found in Table 4.5.

10,000 random samples are drawn from the maneuver space and both propagated directly using the CRTBP dynamics and approximated at each timestep using all polynomial models. A depiction of the target position reachability set at $t = P$ is
Figure 6.5. Test Case 1: Target Reachability Set at $t = P$

given in Figure 6.5. The error in position is computed at each timestep as

$$
\epsilon_r(t) = ||r_{mc}(t) - r_{approx}(t)||_2
$$

(6.11)

where $r_{mc}$ is the sample position evaluated via direct propagation and $r_{approx}$ is the sample position approximated using a polynomial model. Figure 6.6 shows scatterplots of the position reachability set on the $x$, $y$, and $z$ axes with log error displayed on the colorbar for each polynomial at the half period $t = P/2$. Figure 6.7 displays the same information at the end of the first period $t = P$. It is apparent from Figure 6.6 and Figure 6.7 that increasing polynomial approximation order $D$, decreases the approximation error over the domain of the reachability set. The two norm of position error over all samples is shown vs time for varying approximation order in Figure 6.8.

Both representation of the computed reachability set error clearly show the advantage of including higher-order statistical moments of the pdf when computing polynomial coefficients. This is consistent with results shown for other applications including two-body dynamics in ECI frame, and J2-perturbed Two-body dynamics in the relative RSW frame. The strongly non-Gaussian effects in the CRTBP, especially near the Lagrange points, make the HOSM method particularly useful in the cislunar domain. The following section uses the reachability sets computed
6.4.2 Test Case 2: Reachability Set Search

This test case demonstrates how the reachability set search method may be used to locate lost targets in the cislunar domain. Assume the target has the same initial state, and maneuver capability as that defined in the previous test case, where the true target maneuver is given by.

$$u^* = [-0.4980 \ 0.5285 \ -0.4415 \ -0.9500] \quad (6.12)$$

Also assume there is a controllable space-based sensor in a GEO orbit and the objective is to task the sensor to locate the target. The initial state of this sensor
Figure 6.7. Test Case 1: Reachability Set Position Error at $t = P$

can be found by transformation of an orbit in ECI coordinates to rotating CRTBP coordinates. Assume that the inclination of the Moon’s orbit with respect to Earth’s equator is 23.28° (the average between major and minor lunar standstill [124]). Also assume that the sensor is in an equatorial orbit with an initial position on the x-axis of the CRTBP coordinate frame between the Earth and the Moon. The initial position is given by

$$r_{rot} = r_{eci} + r_{eci/rot}$$  \hspace{1cm} (6.13)

where $r_{eci} = [0.1098885 \ 0 \ 0]^T$ represents the semimajor axis of a 24hr orbital period, and $r_{eci/rot} = [-0.012150 \ 0 \ 0]^T$ represents the position shift from the ECI frame to the barycentric CRTBP frame in non-dimensionalize coordinates. The velocity in the CRTBP frame can be computed as

$$v_{rot} = v_{eci} + \Omega \times r_{rot}$$  \hspace{1cm} (6.14)
where $v_{eci}$ is the circular velocity of the sensor in a GEO orbit and $\Omega$ is the angular velocity of the Earth-Moon system ($\Omega = 1$ in non-dimensional units). The initial observer state in non-dimensional units is given as $x_{ob,0} = [0.0977, 0, 0, 0, 2.6443, -1.1850]^T$. Figure 6.9 depicts the observer orbit, and nominal target orbit in rotating CRTBP coordinates. Note that the observer orbit has zero inclination with respect to Earth; however, the inclination of the CRTBP rotation frame causes the observer motion to take a toroidal shape.

Assume that the observer is equipped with an optical sensor where angle information is derived from pixel position. Angle measurements are modeled with the relative position vector in CRTBP $\rho$

$$h(x_{k+1}) = \begin{bmatrix} \tan^{-1}(\rho_{2}/\rho_{1}) \\ \cos^{-1}(\rho_{3}) \end{bmatrix}$$

and are assigned Gaussian noise $\nu$ with standard deviation of one arcsecond, i.e. $\sigma_\theta = 2.908 \times 10^{-4} rad$. The same detection probability function used in Chapter 4 given by (4.7) is used here. The FOV half angle is $\gamma^* = 1^\circ$ and the scale distance is $|\rho_s| = 0.25$ or one quarter of a distance unit. These parameters are selected to provide relatively low detection probabilities to reflect low SNR inherent in cislunar
Assume that the target reachability set is propagated using a 5\textsuperscript{th} order polynomial model for half an orbital period without any measurements, then measurements are taken every half hour $\Delta t = 30\text{min}$ starting at $P/2$. 10,000 reachability set samples are propagated using HOSM method and used to select sensor parameters which maximize the expected value of detection probability. The evolution of the reachability set contours in the y-z plane and in one day intervals is shown in Figure 6.10. It can be seen that as the simulation progresses and repeated null observations of the most likely region of the reachability set are taken, the pdf contours begin to disperse into the regions of the set further away from the nominal trajectory. This process is slow to occur because each observation has a very low chance of detecting the target, so it takes several days of accumulated measurements for the search method to identify fringe regions of the pdf as likely target locations. The target is detected at $t_d = 2.3844$ (approximately 10 days 8 hours after beginning of simulation), after which it is apparent from Figure 6.10 that the target pdf collapses around the true target state.

Figure 6.11 shows the value of the sensor tasking cost function, i.e. maximum expected detection likelihood, as a function of time. The expected detection likelihood oscillates as a function of the distance between observer and the target,
Figure 6.10. Test Case 2: Evolution of Target PDF in y-z Plane During Search ($5^{th}$ order approximation)
Expected Detection Probability vs. Time

Figure 6.11. Expected Detection Probability vs. Time

i.e. position of sensor in GEO orbit, and noticeably increases after the first detection. This is because after the first detection all of the samples in the reachability set become concentrated in a small enough region to observe the all simultaneously. As mentioned previously, the detection likelihood is quite low (between 1% - 3%) for any single measurement, but given a long enough search time and enough measurements, the sensor is eventually able to get a successful detection.

Now instead of a single realization, consider 250 full simulations with randomly selected true maneuvers $u^*$. All simulations are run only using the $5^{th}$ order polynomial model. The estimation error in maneuver magnitude $\epsilon_u$ and maneuver time $\epsilon_t$ are computed as

$$
\epsilon_u = ||E[u_{t,1}] - u^*_{t,1}||_2 \quad \epsilon_t = |E[t_1] - t^*_1|
$$  \hspace{1cm} (6.16)

and the estimation error in target position $\epsilon_r$ and velocity $\epsilon_v$ are computed as

$$
\epsilon_r = ||E[r_t] - r^*_t||_2 \quad \epsilon_v = ||E[v_t] - v^*_t||_2
$$  \hspace{1cm} (6.17)

The error for both maneuver and state estimates are computed only at the final time $t_f = P$ for all simulations. Out of the 250 simulations, 3.2% did not successfully detect the target by the end of the simulation $t_f = P$. Simulations where the
target was not detected were removed from error computations to remove outlier effects. A histogram of the detection times for all 250 simulations is shown in Figure 6.12, where the detection time denotes how long it took to detect the target \( t = P/2 \). Histograms of the maneuver estimation error at \( t_f \) for both maneuver magnitude and maneuver time are shown in Figure 6.13, and histograms of position and velocity estimation error at \( t_f \) is shown in Figure 6.14. Note all error values are given based on dimensionless distance units \( DU \) and time units \( TU \) for the CRTBP. Table 6.1 shows the final time RMSE over all simulations for maneuver estimate errors as well as position and velocity state estimate errors.

Overall, the performance characteristics of the reachability set search method show similar trends to those in the test cases in Chapter 4. The detection time for the majority of target maneuver realizations is towards the beginning of the start of measurements because most maneuver realizations have not departed significantly from the nominal orbit at this time. The detection time is later only in some extreme realizations where a large maneuver was executed relatively early in the simulation.
Figure 6.13. Test Case 2: Histograms of Maneuver Estimation Error at $t_f$ for 250 Simulations

Figure 6.14. Test Case 2: Histograms of State Estimation Error at $t_f$ for 250 Simulations

### 6.5 Conclusion

This chapter has provided examples of the probabilistic methods developed throughout this dissertation to target tracking applications in cislunar space. SDA in cislunar space is an area of significant interest for current-day and future missions which plan on utilizing this domain for logistical staging and refueling for missions to the Moon and beyond. Unfortunately, tracking targets over the vast distances

<table>
<thead>
<tr>
<th>$RMS(\epsilon_u)$</th>
<th>$RMS(\epsilon_t)$</th>
<th>$RMS(\epsilon_r)$</th>
<th>$RMS(\epsilon_v)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.729E-3</td>
<td>2.4041E-1</td>
<td>1.3338E-2</td>
<td>1.2993E-2</td>
</tr>
</tbody>
</table>

Table 6.1. RMSE of Maneuver and State Estimates at $t_f$
to cislunar space is very difficult, especially when coupled with unknown target maneuvers. Due to long propagation times between measurements and low SNR it is very easy to lose tracking custody for cislunar targets. Furthermore, the strongly nonlinear dynamics in this domain renders many existing methods for predicting target location inadequate.

The methods developed in this dissertation enable urgently needed target tracking capabilities in this domain, even in the presence of very low detection probabilities and unknown maneuvers. The HOSM method has been demonstrated to accurately capture the reachability set of a target making an unknown maneuver at an unknown time from an initial halo orbit. The polynomial models used to approximate these reachability sets provide orders of magnitude of computational saving compared to explicit Monte Carlo methods, and offer flexibility in the trade-off between computational cost and accuracy via selection of basis function order.

Reachability sets computed via HOSM method can then be used for sensor tasking to search for and locate the target. The search method presented in Chapter 4 is well-suited for this application as it enables information to be gained from null measurements, as is frequently the case with low SNR target in cislunar space. In fact, this method provides a natural link between SNR and the measurement update, facilitated by the detection probability. Further work on realistically modeling detection likelihood may prove beneficial for enabling real-world implementation.
Chapter 7
Conclusion

This dissertation develops novel uncertainty characterization and propagation methods to address unsolved problems in the area of space domain awareness (SDA) with a particular focus on tracking noncooperative maneuvering satellites. The main contribution of this dissertation is a probability-based framework for tracking satellites where target satellite state and maneuvers are defined stochastically, and sensor data is acquired by exploiting target reachability sets. This framework uses state-of-the-art uncertainty propagation techniques to compute reachability sets which define the search space for sensors, then use sensor data as feedback to update the reachability set. The reachability set-based approach proposed here has unique applications to problems where the target satellite has been lost and must be located.

The first contribution of this dissertation is development of an efficient method for reachability set computation, termed the higher-order sensitivity matrix (HOSM) method. This method computes a Taylor series-based polynomial approximation model to map uncertainty from target state, maneuvers, and model parameters to future target states. The coefficients of the polynomial model corresponding to partial derivatives of the system flow are computed via least squares minimization, and use highly efficient non-product quadrature methods to evaluate the required integrals. The Conjugate Unscented Transform (CUT) method can be used when all inputs have symmetric probability density functions (pdfs), i.e. Gaussian or Uniform; however, the HOSM method is also applicable to inputs with mixed pdfs. This dissertation uniquely characterizes uncertainty in unknown target maneuvers as spherically uniform random variables and maneuver time as uniformly distributed. This characterization of uncertainty imposes no assumptions on target attitude...
or maneuver time, and instead bounds the maneuver based on target maneuver capabilities and a feasible maneuver window. Quadrature sets for this unique stochastic maneuver definition are computed using a generic moment matching method known as designed quadrature (DQ).

The HOSM method offers several orders of magnitude of computational savings compared to explicit propagation of Monte Carlo (MC) samples through the dynamic model; however, there are limitations to be considered. Depending on the magnitude of uncertainties, the nonlinearity of system dynamics, and the length of propagation times, it may become infeasible to compute a high enough order polynomial model to capture the target reachability set to the desired accuracy. This problem is greatly exacerbated if the dimension of the stochastic input vector becomes too large from multiple maneuvers, model parameters, or initial conditions.

The next contribution is development of a systematic sensor tasking method using reachability sets. The target reachability set defines a search region over which sensors are tasked, which is particularly useful in applications where the target has been lost and tracking custody must be re-acquired. Selection of sensor parameters is based on optimization of a maximum detection likelihood criteria. Sensor tasking for single-target single-sensor applications have been considered here; however, extension to multi-sensor tasking scenarios is likely to increase the observability of reachability set, particularly those with low detection probability such as distant targets in cislunar space.

Another contribution of this dissertation is the utilization of sensor data to update the target reachability set in a statistically consistent manner. The reachability set is updated using Bayesian inference and is intimately related to the detection probability. When the target is undetected by a given measurement, the reachability set is updated to reflect a lower probability that the true target is in the observed region. A critical feature of this update strategy is that sensor data from observations of vacant regions of space are used to enhance knowledge of the remaining possible target states, as well as influencing sensor tasking at future times. Once the target is detected, more traditional tracking methods may be employed.

The methods in this dissertation have been extensively tested and validated via numerical simulations. Depending on the application considered, the accuracy of reachability set computation increases roughly one order of magnitude for
every order increase in polynomial basis functions. Furthermore, the non-product quadrature methods used to compute these sets have been shown to be far more efficient than comparable tensor product methods, with a 53% improvement in computational benefit for a single maneuver case being a modest benefit compared with higher dimensional cases. Using the simulation parameters defined in each test case, the reachability set search method has also been demonstrated to locate the target within the defined time window for the large majority of cases. Randomly sampling 250 realizations of true maneuver parameters, the target was successfully detected within half an orbital period for 100% of trials for a target in GEO making a single unknown maneuver, 96% of trials for a target in GEO making two unknown maneuvers, and 96.8% of trials for a target in an $L_1$ halo orbit making a single unknown maneuver.

The final contribution is to develop a systematic method to extend the computation of reachability sets to sequential maneuvers while maintaining constant system dimensionality. This procedure, termed the higher-order moment matching (HOMM) method, computes the higher-order moments of the target state after every maneuver and solves for quadrature points which match this generic set of moments. The higher-order moments are computed using a quadrature set in the current state and a quadrature set in maneuver variables, and evaluating the expectation values of independent variables separately. After the post-maneuver state higher-order moments are known, the DQ method is used to determine a set of quadrature points and weights which can be used to continue reachability set propagation to the next maneuver. This method also has applications for maintaining higher-order information after the measurement update in a Kalman filter.

### 7.1 Future Work

The contributions developed in this dissertation extend the envelope of modern-day SDA capabilities; however, several notable avenues exist for further enhancing the developed framework. One such research direction is to integrate realistic sensor modeling into the framework presented. Realistic, physics-based sensor models will enable more rigorous definition of detection probability using metrics like signal-to-noise ratio, and provide more concrete information about implementation
Another future research direction is to extend the reachability set search method to multi-target, multi-sensor scenarios. The simulations provided in this dissertation only considered single sensors; however, tasking of distributed sensor networks is a problem of significant interest. This sensor tasking problem can be accomplished using information theoretics like the mutual information metric to task multiple sensors over a finite time horizon. This has promise for tasking of space-based sensor networks for surveillance of cislunar space that is not achievable by current ground-based tracking networks.

Alternative methods for preserving higher-order information during reachability set propagation and measurement update are also of great interest. One such idea is to use the method of particle flow, to define the post-maneuver or post-update distribution. Particle flow utilizes a pseudo-time homotopy method to directly migrate prior quadrature points to posterior quadrature points using the Fokker-Planck-Kolmogorov equation. If this method were able to be successfully implemented in the proposed framework, it may supplant the computationally intensive direct moment matching method via DQ used in the current framework.

A final, and perhaps most easily achievable area of improvement is to improve the efficiency of the existing framework. One example is to simply parallelize algorithms such as the evaluation of quadrature points in the higher-order sensitivity matrix method or evaluation of cost function for the reachability set search method. This can be easily accomplished by splitting these tasks and running simultaneously on multiple computing cores. Another avenue for improving the existing methods is to perform a detailed tuning of the parameters in the DQ method. For example, overall computation time for the HOMM method can likely be reduced by accepting a DQ solution with suboptimal number of points \( N \) and directly propagating additional samples to the next timestep, rather than wasting time on costly singular value decompositions by repeatedly pruning and enriching to find a set with minimum \( N \).
References


157


URL https://doi.org/10.2514/1.43963


[83] Lee, T., M. Majji, and P. Singla “Conjugate Unscented Transformation Based Approach To Compute Higher Order State Transition Matrix For Nonlinear Dynamic Systems: Applications To Estimation And Control,”.


Vita

Zachary Hall

Zachary Hall was born in Madison, Wisconsin, on December 7th, 1994. He received his bachelors degree in mechanical engineering from Lehigh University in Bethlehem, Pennsylvania in May 2013. In August of the same year, he joined the Control and Analysis of Stochastic Systems (CASS) Lab at Penn State University to begin graduate work. He received his masters thesis in May 2019, which focused on reachability set computation and the solution to the uncertain Lambert problem, and formed the foundation of the maneuvering target tracking framework developed in this dissertation. All graduate work was performed under the supervision of Dr. Puneet Singla.