Utah State University

DigitalCommons@USU

All Graduate Theses and Dissertations

Graduate Studies

5-2022

Optimal Relative Path Planning for Constrained Stochastic Space Systems

Nathan Bohus Stastny
Utah State University

Follow this and additional works at: https://digitalcommons.usu.edu/etd

Part of the Aerospace Engineering Commons

Recommended Citation

Stastny, Nathan Bohus, "Optimal Relative Path Planning for Constrained Stochastic Space Systems" (2022). *All Graduate Theses and Dissertations*. 8441.
https://digitalcommons.usu.edu/etd/8441

This Dissertation is brought to you for free and open access by the Graduate Studies at DigitalCommons@USU. It has been accepted for inclusion in All Graduate Theses and Dissertations by an authorized administrator of DigitalCommons@USU. For more information, please contact digitalcommons@usu.edu.

UtahStateUniversity
MERRILL-CAZIER LIBRARY

OPTIMAL RELATIVE PATH PLANNING FOR CONSTRAINED STOCHASTIC SPACE SYSTEMS

by

Nathan Bohus Stastny

A dissertation submitted in partial fulfillment
of the requirements for the degree

of

DOCTOR OF PHILOSOPHY

in

Aerospace Engineering

Approved:

David K. Geller, Ph.D. Major Professor	Stephen A. Whitmore, Ph.D. Committee Member
Matthew W. Harris, Ph.D. Committee Member	Geordie Richards, Ph.D. Committee Member
Randall Christensen, Ph.D. Committee Member	D. Richard Cutler, Ph.D. Interim Vice Provost of Graduate Studies

UTAH STATE UNIVERSITY
Logan, Utah

2022

Copyright © Nathan Bohus Stastny 2022

All Rights Reserved

ABSTRACT

Optimal Relative Path Planning for Constrained Stochastic Space Systems

by

Nathan Bohus Stastny, Doctor of Philosophy

Utah State University, 2022

Major Professor: David K. Geller, Ph.D.
Department: Mechanical and Aerospace Engineering

Rendezvous and proximity operations for automated spacecraft systems requires advanced path planning techniques that are capable of generating optimal paths. Real-world constraints, such as sensor noise and actuator errors, complicate the planning process. Operations also require flight safety considerations in order to prevent the spacecraft from potentially colliding with the associated companion spacecraft. This work proposes a new, ground-based trajectory planning approach that seeks an optimal trajectory while meeting all mission constraints and accounting for vehicle performance and safety requirements. This approach uses a closed-loop linear covariance simulation of the relative trajectory coupled with a genetic algorithm to determine fuel optimal trajectories. Spacecraft safety is addressed using statistical data from the linear covariance model to bound the probability of collision.

(289 pages)

PUBLIC ABSTRACT

Optimal Relative Path Planning for Constrained Stochastic Space Systems

Nathan Bohus Stastny

Rendezvous and proximity operations for automated spacecraft systems requires advanced path planning techniques that are capable of generating optimal paths. Real-world constraints, such as sensor noise and actuator errors, complicate the planning process. Operations also require flight safety considerations in order to prevent the spacecraft from potentially colliding with the associated companion spacecraft. This work proposes a new, ground-based trajectory planning approach that seeks an optimal trajectory while meeting all mission constraints and accounting for vehicle performance and safety requirements. This approach uses a closed-loop linear covariance simulation of the relative trajectory coupled with a genetic algorithm to determine fuel optimal trajectories. Spacecraft safety is addressed using statistical data from the linear covariance model to bound the probability of collision.

To my wife, Canda. *Je t'aimais, je t'aime, je t'aimerai...*

...and to the memory of my father and brother. *Toujours dans mon cœur.*

ACKNOWLEDGMENTS

For millennia, navigators used the stars to fix their location and chart a safe path. My life too has been fixed and charted by the stars in the form of family, friends and coworkers, each helping to bring me to this point. I express my deepest appreciation to all those who have made this research possible and given me a wonderful journey along the way.

I first wish to thank my parents for teaching me the value of education and hard work. Their love and examples gave my life an unchanging north star. Thank you.

I count myself blessed to have worked with and learned from the amazingly capable engineers at the Air Force Research Laboratory. My thanks go to Josue Munoz, Richard Zappulla, Andy Bustillos, Christina Straight, HP Dumm, David Meyer, Tom Ortiz, Alan Lovell and Rob Howard. Special thanks go to Frank Chavez, Jud McCarty, and Kevin Schum who's contributions and support were critical to my success. Additional thanks go to all of the government, contractor, and flight operations teams supporting the ANGELS and Mycroft programs. These programs have been the highlight of my career. And finally to those whose names must be redacted but whose contributions have left their mark.

I wish to express my gratitude to the Space Dynamics Laboratory for its support of this research. None of this would have been possible without the support of Paul Oppenheimer, Amy Secrist, and Pat Patterson. Special thanks go to Skylar Cox, Simon Shuster, and Francisco Franquiz for their technical support, encouragement, and friendship.

It is almost impossible to express my appreciation for the support of my advisor, Dr. David Geller. His ability to teach and inspire brought the classroom to life and gave meaning to the math. His friendship and example have helped guide my career. I am forever grateful.

And finally, but most importantly, I wish to thank my wife and children. They are my guide stars. None of this would have been possible without their support, encouragement, patience, love, and sacrifices over the last decade. *Ayo ka.*

Nathan Bohus Stastny

CONTENTS

Page

ABSTRACT iii

PUBLIC ABSTRACT iv

ACKNOWLEDGMENTS vi

LIST OF TABLES xi

LIST OF FIGURES xiv

ACRONYMS xxiii

1 INTRODUCTION 1
  1.1 Relative Orbital Motion 2
  1.2 GNC Systems and Path Planning Techniques 4
  1.3 Planning Constraints 5
    1.3.1 ΔV Usage 5
    1.3.2 Lighting 6
    1.3.3 Collision Risk 7
    1.3.4 Navigation and Trajectory Dispersions 8
  1.4 Research Scope and Overview 9
    1.4.1 Constrained Path Planning Thesis 10

2 ORBITAL RENDEZVOUS AND PROXIMITY OPERATIONS 12
  2.1 Dynamics 12
    2.1.1 Orbital Motion Dynamics 12
    2.1.2 Relative Motion Dynamics 16
  2.2 Relative Maneuvers 23
  2.3 Rendezvous and Proximity Operations Missions 26
  2.4 Related Research 32
    2.4.1 Relative Motion Dynamic Models 32
    2.4.2 Deterministic Path Planning Techniques for Proximity Operations 33
    2.4.3 Safety of Flight and Probability of Collision Techniques 34
    2.4.4 Constrained Path Planning Techniques for Proximity Operations 36
    2.4.5 Relative Navigation Techniques for Proximity Operations 38
    2.4.6 Linear Covariance Analysis Applications 39
  2.5 Conclusion 41

3 STATISTICAL ANALYSIS OF CLOSED-LOOP GNC SYSTEMS 42
  3.1 Nonlinear Monte Carlo Analysis 43
    3.1.1 Nonlinear Truth Dynamics 44
    3.1.2 Nonlinear Truth Measurement Model 45
    3.1.3 Nonlinear Navigation Model 46
    3.1.4 Nonlinear Guidance Model 49
    3.1.5 Nonlinear Actuator Model 50
    3.1.6 Monte Carlo Performance Statistics 50
    3.1.7 Confidence Intervals 52
  3.2 Linear Covariance Analysis 53
    3.2.1 Linear Augmented State and State Dynamics 54
    3.2.2 Linear Augmented State Update 55
    3.2.3 Linear Augmented State Correction 56
    3.2.4 Augmented State Covariance 57
    3.2.5 Linear Covariance Model 57
    3.2.6 Linear Covariance Performance Statistics 58
  3.3 Simplified Rocket Ship Analysis 60
    3.3.1 Monte Carlo Analysis 60
    3.3.2 Linear Covariance Analysis 64
    3.3.3 Optimal Maneuver Location 66

4 LINEAR COVARIANCE MODEL FOR RENDEZVOUS AND PROXIMITY OPERATIONS 68
  4.1 Truth Model Dynamics 68
  4.2 Truth Sensor Models 69
    4.2.1 Lidar Model 70
    4.2.2 Lidar Constraints 70
    4.2.3 Camera Model 71
    4.2.4 Camera Constraints 73
  4.3 Truth Thruster Control 77
  4.4 Navigation Model Dynamics 78
  4.5 Relative Measurement Update Models 79
    4.5.1 Lidar-based Navigation 79
    4.5.2 Angles-only Navigation 79
  4.6 Guidance Models 80
    4.6.1 Waypoint Targeting 81
    4.6.2 Velocity Matching 81
  4.7 Augmented State and Covariance 83
  4.8 Linear Covariance Dynamic Model 83
  4.9 Linear Covariance Measurement Update Model 84
  4.10 Linear Covariance Maneuver Correction Model 84
  4.11 RPO Performance Statistics 85
    4.11.1 True Position Dispersion 85
    4.11.2 True Range Dispersion 88
    4.11.3 Navigation Error Covariance 88
    4.11.4 True ΔV Dispersion 88

4.12 Nominal Trajectory 89
4.13 Linear Covariance Summary 92
4.14 Example Scenario 93

5 DETERMINISTIC VS CONSTRAINED TWO-IMPULSE RELATIVE PATH PLANNING 99
5.1 NMC Resizing Scenario 99
5.1.1 LinCov Evaluation of Deterministic Results 104
5.1.2 LinCov Global Evaluation 105
5.2 V-bar Hop Scenario 112
5.2.1 LinCov Global Results 114
5.3 Conclusions 117

6 OPTIMAL RELATIVE PATH PLANNING WITH POSITION DISPERSION CONSTRAINTS 119
6.1 Stochastic Trajectory Optimization for RPO Missions (STORM) 119
6.1.1 Optimization Approach 120
6.1.2 Genetic Algorithm 120
6.1.3 MATLAB *fmincon* Algorithm 124
6.1.4 Fitness Function 125
6.1.5 Decision Variable Size and Constraints 127
6.1.6 Free Final Time 129
6.2 Optimal Two-Impulse Path Planning 129
6.2.1 Optimal Two-Impulse NMC Resizing Results 130
6.2.2 Optimal Two-Impulse V-bar Hop Results 134
6.3 Conclusions 137

7 MULTI-WAYPOINT RELATIVE PATH PLANNING WITH POSITION DISPERSION CONSTRAINTS 138
7.1 Three-Impulse NMC Resizing 139
7.1.1 Deterministic Optimal Solution 142
7.1.2 Constrained Optimal Solution - Good Lighting 145
7.1.3 Constrained Optimal Solution - Poor Lighting 147
7.2 Three-impulse Co-elliptic Drift to NMC 150
7.2.1 Deterministic Optimal Solution 155
7.2.2 Constrained Optimal Solution 157
7.2.3 Constrained Optimal Solution - Free Final Time 160
7.2.4 Constrained Optimal Solution - Tight Constraint 163
7.3 Multi-Waypoint Solutions 166
7.4 Conclusions 167

8 PROXIMITY OPERATIONS SAFETY ANALYSIS 168
8.1 RPO Safety 170
8.1.1 Keep-out Zone Analysis 170
8.1.2 Tree Analysis 170
8.1.3 Trajectory Assessment 171
8.2 Probabilistic Trajectory Safety 173

- 8.2.1 Probability of Collision 174
- 8.2.2 Simplified Probability of Collision 177
- 8.2.3 Range Uncertainty 180
- 8.2.4 Mahalanobis Distance 183
- 8.2.5 Bounded Probability of Collision 184
- 8.3 Keep-out Zone Intersection 187
- 8.4 Linear Covariance Tree Analysis 193
- 8.5 Conclusions 195

9 OPTIMAL RELATIVE PATH PLANNING WITH POSITION DISPERSION AND SAFETY CONSTRAINTS 197
- 9.1 Fitness Function 197
- 9.2 NMC Resizing 199
  - 9.2.1 Simply Constrained Results 201
  - 9.2.2 Safety Constrained Results 203
- 9.3 Co-elliptic to NMC 207
  - 9.3.1 Simply Constrained Results 208
  - 9.3.2 Safety Constrained Results 209
- 9.4 V-bar Hop 214
  - 9.4.1 Simply Constrained Results 217
  - 9.4.2 Safety Constrained Results 221
- 9.5 Conclusions 223

10 CONCLUSIONS 224
- 10.1 Constrained Path Planning Thesis 225
- 10.2 Future Work 229

REFERENCES 231

APPENDICES 240
- A CLOHESSY-WILTSHIRE STATE TRANSITION MATRIX 241
- B SUN VECTOR IN LVLH FRAME 246
- C ROCKET SHIP EQUATIONS 249
  - C.1 Monte Carlo Analysis 249
  - C.2 Linear Covariance Analysis 252
- D ANGLES-ONLY MEASUREMENT SENSITIVITY MATRIX 255
- E MATLAB GENETIC ALGORITHM PARAMETER SETTINGS 258

CURRICULUM VITAE 262

LIST OF TABLES

Table Page

4.1 Nominal trajectory inputs. 94

4.2 Waypoint and transfer time inputs. 94

4.3 Scenario noise parameters. 94

4.4 Camera constraint parameters. 95

5.1 Two-impulse NMC resizing parameters. 100

5.2 Optimal transfer data for the four global minima. 102

5.3 Scenario parameters for the two-impulse linear covariance analysis. 104

5.4 Maneuver error uncertainty for three scenarios. 104

5.5 LinCov dispersion results for the non-constrained global optimal transfers for each scenario. 104

5.6 Optimal results for the constrained global optimal transfers for the NMC resizing scenario. $\Delta\bar{V} = 0.21502$ m/s. 109

5.7 Two-impulse V-bar hop scenario parameters. 114

5.8 LinCov parameters for the V-bar hop scenario. 116

5.9 Optimal $\Delta V^{3\sigma}$ results for the multi-revolution ($\Delta t_1 > T$) constrained V-bar hop transfers. $\Delta\bar{V} = 0.0053$ m/s. 117

5.10 Optimal $\Delta V^{3\sigma}$ results for the single revolution ($\Delta t_1 < T$) constrained V-bar hop transfers. $\Delta\bar{V} = 0.0106$ m/s. 117

6.1 LinCov global search and optimization results for the single revolution NMC resizing scenario with final position requirement, $S_{lim} = 25$ m. $\Delta\bar{V} = 0.21502$ m/s. 130

6.2 LinCov global search and optimization results for the V-bar hop scenario with a final position requirement, $D_f \leq 5$ m. $\Delta\bar{V} = 0.0053$ m/s. 134

7.1 Nominal trajectory inputs for NMC resizing scenario. 139

7.2 NMC resizing scenario noise parameters and camera constraints. 140

7.3 Three-impulse NMC resizing scenario test cases. 140

7.4 LinCov results for the three-impulse NMC resizing scenario with final position requirement. $\Delta\bar{V} = 0.1250$ m/s. 141

7.5 Optimal time and waypoint results for the three-impulse NMC resizing scenario with final position requirement. 141

7.6 Nominal trajectory inputs for co-elliptic to NMC scenario. 151

7.7 Co-elliptic to NMC scenario noise parameters and camera constraints. . . . 151

7.8 Three-impulse co-elliptic to NMC scenario test cases. 152

7.9 LinCov results for the three-impulse co-elliptic to NMC scenario with final position requirement. $\Delta\bar{V} = 0.3750$ m/s. 153

7.10 Optimal time and waypoint results for the three-impulse co-elliptic to NMC scenario with final position requirement. T_c is an optimization parameter for the free final time case only. 153

8.1 Probability values that x lies within the hyperellipsoid boundary defined by the equation $x^T D^{-1} x = l^2$ for 1, 2, and 3-dimensional distributions. 185

8.2 Probability of collision bound for select 3-dimensional covariance bounds. . 186

9.1 Nominal trajectory inputs for NMC resizing scenario. 199

9.2 NMC resizing scenario noise parameters, camera constraints, and safety parameters. 200

9.3 LinCov results for the three-impulse NMC resizing scenario. $\Delta\bar{V} = 0.1250$ m/s. 203

9.4 Optimal time and waypoint results for the three-impulse NMC resizing scenario. 203

9.5 Nominal trajectory inputs for the co-elliptic to NMC scenario. 207

9.6 Co-elliptic to NMC scenario noise parameters, camera constraints, and safety parameters. 208

9.7 LinCov results for the three-impulse coelliptic to NMC scenario. $\Delta\bar{V} = 0.3750$ m/s. 210

9.8 Optimal time and waypoint results for the three-impulse co-elliptic to NMC scenario. 210

9.9 Nominal trajectory inputs for the V-bar hop scenario. 215

9.10 V-bar hop scenario noise parameters, camera constraints, and safety parameters. 216

9.11 LinCov results for the three-impulse V-bar hop scenario. $\Delta\bar{V} = 0.0106$ m/s. 216

9.12 Time and waypoint results for the three-impulse V-bar hop scenario. 216

E.1 Parameters and values used for MATLAB *optimoptions*. 258

LIST OF FIGURES

Figure Page

1.1 LVLH (RIC) frame attached to the inertial reference frame. 2

1.2 Sample relative path defined by the CW equations in the LVLH frame. The elliptical path is translating from left to right. 3

1.3 Typical lighting constraints for RPO include (a) solar phase angles, (b) eclipse, and (c) scene back-ground including the earth. 7

1.4 Passive abort safety ellipse commonly used to minimize the risk of collision. 9

2.1 Two-body problem in inertial reference frame. 14

2.2 Two orbits around the earth defined by the two-body problem. 15

2.3 Local-vertical, local-horizontal frame. 17

2.4 Relative geometry and LVLH coordinate frame of two satellites in proximity around earth. 18

2.5 Waypoint-defined relative trajectory about RSO 25

2.6 View of Gemini VII from Gemini VI as they perform the first on-orbit proximity operations on December 15, 1965. 27

2.7 View of Challenger from the SPAS-1 satellite during STS-7 in June 1983. . 27

2.8 XSS-10 satellite (left). Delta II launch vehicle upper stage images by XSS-10 (right). 28

2.9 Artist's depiction of the DART spacecraft as it approaches MUBLCOM. . . 29

2.10 Image of NEXTSat taken by the ASTRO satellite as part of the Orbital Express mission. 30

2.11 ANGELS operations crew patch. From the author's personal collection. . . 31

2.12 Artist's depiction of the PRISMA spacecraft Mango and Tango. 32

3.1 Closed-loop GNC system in the real world. 43

3.2 Monte Carlo analysis block diagram for a closed-loop GNC system. 44

3.3 Number of simulation runs required to achieve a desired confidence interval fraction for different probabilities. 54

3.4 Linear covariance analysis block diagram for a closed-loop GNC system. . . 58

3.5 Rocket ship scenario flying to distant planet. 60

3.6 Rocket ship true position dispersions with calculated 3σ bounds for 1000 runs without navigation or maneuvers. 61

3.7 Rocket ship true position dispersions (black) and filter position dispersions (blue) for 1000 runs with a single measurement update on day 80. 62

3.8 Rocket ship true navigation position errors for 1000 runs with a single measurement update on day 80. 62

3.9 Rocket ship true position dispersions (black) and filter position dispersions (blue) for 1000 runs with a single measurement update on day 80 and correction maneuver on day 85. 63

3.10 Rocket ship true navigation position errors for 1000 runs with a single measurement update on day 80 and correction maneuver on day 85. 63

3.11 Rocket ship ΔV dispersions histogram and probability density function for 1000 runs. $\mu = 0.0025$ and $\sigma = 0.0375$. 64

3.12 Rocket ship true position dispersions (black) and filter position dispersions (blue) for 1000 runs with a measurement update on day 80 and correction maneuver on day 85. $\pm 3\sigma$ bounds from the Monte Carlo (red) and LinCov (cyan) analyses. 65

3.13 ΔV dispersions probability density function for the Monte Carlo and LinCov analyses. 65

3.14 Final true position and ΔV dispersions for all possible days to perform the measurement update as calculated by the linear covariance model. 67

4.1 RSO location and angles-only measurements in the camera frame (left) and camera focal plane (right). 72

4.2 Earth exclusion angles. 74

4.3 Example of the sun's conical path in the LVLH frame. 75

4.4 Beta angle between the reference orbit plane and the sun. 76

4.5 Solar exclusion angles. 76

4.6 True position dispersion ellipsoid with orthogonal ellipse projections. 86

4.7 True position dispersion ellipses in the Intrack-Crosstrack plane. 86

4.8 Example nominal trajectory for a single waypoint transfer in the Intrack-Crosstrack plane. 91

4.9 Nominal trajectory with maneuver directions. Initial trajectory is green. Final trajectory is red. 95

4.10 True position and velocity dispersions. Maneuvers times are marked by the triangles. Dashed lines represent the RSS of the three values. 96

4.11 True range dispersions. Top plot shows the nominal range with the 3σ range dispersion overlay. The bottom plot shows the range dispersion. 96

4.12 True navigation position and velocity error covariance. Maneuvers times are marked by the triangles. Dashed lines represent the RSS of the three values. 97

4.13 Sun and earth angles. The system system is unable to take images due to both sun and earth angular violations. Dashed lines are the exclusion angle limits $\theta_{\oplus}^{lim}$ and $\theta_{\odot}^{lim}$. 97

4.14 Nominal ΔV (blue) and 3σ dispersions (orange) for each maneuver. Total $\Delta\bar{V}^{3\sigma} = 0.5221$ m/s. 98

5.1 Representative two-impulse transfer between planar NMC orbits. 101

5.2 Scenario 1: ΔV surface map for all transfer scenarios where $\Delta t_0 + \Delta t_1 \leq t_f$. Contours range from 0.2 m/s to 1.0 m/s. Numbered markers indicate repeating/non-unique global minima. 101

5.3 Transfer orbits corresponding to the four global minima. Numbered markers indicate the initial burn locations. 103

5.4 Case A: Final position dispersion for all combinations of Δt_0 and Δt_1. . . . 106

5.5 Case A: ΔV dispersion for all combinations of Δt_0 and Δt_1 with overlayed final position dispersion contours. 106

5.6 Case B: Final position dispersion for all combinations of Δt_0 and Δt_1. . . . 107

5.7 Case B: ΔV dispersion for all combinations of Δt_0 and Δt_1 with overlayed final position dispersion contours. 107

5.8 Case C: Final position dispersion for all combinations of Δt_0 and Δt_1. . . . 108

5.9 Case C: ΔV dispersion for all combinations of Δt_0 and Δt_1 with overlayed final position dispersion contours. 108

5.10 Comparison between deterministic optimal transfer #1 and the constrained, single revolution optimal transfer for Case A ($\Delta t_0 = 0.08117T$ and $\Delta t_1 = 0.61117T$). 110

5.11 Case A: Final position dispersion for all combinations of Δt_0 and Δt_1 using logarithmic scaling. 111

5.12 Case B: Final position dispersion for all combinations of Δt_0 and Δt_1 using logarithmic scaling. 111

5.13 Case C: Final position dispersion for all combinations of Δt_0 and Δt_1 using logarithmic scaling. 112

5.14 Representative two-impulse V-bar hop transfer between two stationary in-track offsets. 113

5.15 Scenario 2: ΔV surface map for all transfer scenarios where $\Delta t_0 + \Delta t_1 \leq t_f$. Contours range between 0.0053 m/s and 1.0 m/s. 113

5.16 Scenario 2: Final position dispersion for all combinations of Δt_0 and Δt_1. . 115

5.17 Scenario 2: Final true ΔV dispersion for all combinations of Δt_0 and Δt_1 with overlayed final position dispersion contours. 115

6.1 STORM analysis tool implementing a hybrid optimization approach. 121

6.2 Genetic Algorithm flow diagram. 122

6.3 Single-point crossover function where the chromosomes from parents are used to generate children for the next generation. 123

6.4 Sample mutation processes for binary and real valued solution space chromosomes. 123

6.5 Selection, crossover, mutation, and elitism mechanics of a generic Genetic Algorithm. 124

6.6 Penalty shape as a function of S_{D_f} used to aid convergence. 126

6.7 Optimal NMC resizing trajectory and maneuvers. 131

6.8 Final true position dispersion for the optimal NMC resizing trajectory. . . . 132

6.9 Trajectory solutions for the GA final population. Optimal maneuvers are green. Optimal trajectory is red. 132

6.10 Locations of the GA initial and final populations and the *fmincon* optimal solution overlayed on the NMC resizing ΔV dispersion surface map. 133

6.11 Fitness function values for the initial and final GA populations of the NMC resizing scenario. 133

6.12 Optimal V-bar hop trajectory and maneuvers determined by the hybrid optimization method. 135

6.13 Final true position dispersion for the optimal V-bar hop trajectory. 135

6.14 Locations of the GA initial and final populations and the optimal solution overlayed on the V-bar hop ΔV dispersion surface map. 136

6.15 Fitness function values for the initial and final GA populations of the V-bar hop scenario. 137

7.1 Optimal trajectory and maneuver locations for the three-impulse NMC resizing scenarios. 142

7.2 Three-impulse NMC resizing scenario for the deterministic optimal solution. 143

7.3 Sun and earth angles for deterministic optimal solution of the three-impulse NMC resizing scenario. 143

7.4 Final true position dispersion for the deterministic optimal solution of the three-impulse NMC resizing scenario. Blue circle represents a desired final position dispersion constraint. 144

7.5 ΔV and 3σ ΔV dispersion values for the deterministic optimal solution. . . 144

7.6 Three-impulse NMC resizing scenario for the constrained optimal solution with good lighting. Navigation outage is due to earth angle violation. . . . 145

7.7 Sun and earth angles for constrained optimal solution of the three-impulse NMC resizing scenario with good lighting. 146

7.8 Final true position dispersion for the deterministic optimal solution of the three-impulse NMC resizing scenario with good lighting. Blue circle represents a final position dispersion constraint. 146

7.9 ΔV and 3σ ΔV dispersion values for the constrained optimal solution with good lighting. 147

7.10 Three-impulse NMC resizing scenario for the constrained optimal solution with poor lighting. Navigation outage due to earth and sun angle violations. 148

7.11 Sun and earth angles for constrained optimal solution of the three-impulse NMC resizing scenario with poor lighting. 148

7.12 Final true position dispersion for the deterministic optimal solution of the three-impulse NMC resizing scenario with poor lighting. Blue circle represents a final position dispersion constraint. 149

7.13 ΔV and 3σ ΔV dispersion values for the constrained optimal solution with poor lighting. 149

7.14 Co-elliptic and NMC trajectories. 151

7.15 Optimal trajectory and maneuver locations for the three-impulse co-elliptic to NMC scenarios in the Radial-Intrack plane. Axes are not equal to help clarify the trajectories. 153

7.16 Optimal trajectory and maneuver locations for the three-impulse co-elliptic to NMC scenarios in the Crosstrack-Intrack plane. Axes are not equal to help clarify the trajectories. 154

7.17 Three-impulse co-elliptic to NMC scenario for the deterministic optimal solution. 155

7.18 Sun and earth angles for deterministic optimal solution of the three-impulse co-elliptic to NMC scenario. 156

7.19 Final true position dispersion for the deterministic optimal solution of the three-impulse co-elliptic to NMC scenario. Blue circle represents a 15 m dispersion. 156

7.20 ΔV and 3σ ΔV dispersion values for the deterministic optimal solution. . . 157

7.21 Three-impulse co-elliptic to NMC scenario path for the constrained optimal solution. 158

7.22 Sun and earth angles for deterministic optimal solution of the three-impulse co-elliptic to NMC scenario. 158

7.23 Final true position dispersion for the constrained optimal solution of the three-impulse co-elliptic to NMC scenario. Blue circle represents a 15 m dispersion. 159

7.24 ΔV and 3σ ΔV dispersion values for the constrained optimal solution. . . . 159

7.25 Three-impulse co-elliptic to NMC scenario for the free final time optimal solution. 161

7.26 Sun and earth angles for the free final time optimal solution of the three-impulse co-elliptic to NMC scenario. 161

7.27 Final true position dispersion for the free final time optimal solution of the three-impulse co-elliptic to NMC scenario. Blue circle represents a 15 m dispersion. 162

7.28 ΔV and 3σ ΔV dispersion values for the free final time optimal solution. . 162

7.29 Three-impulse co-elliptic to NMC scenario for the tightly constrained optimal solution. 164

7.30 Sun and earth angles for tightly constrained optimal solution of the three-impulse co-elliptic to NMC scenario. 164

7.31 Final true position dispersion for the tightly constrained optimal solution of the three-impulse co-elliptic to NMC scenario. Blue circle represents a 3 m dispersion. 165

7.32 ΔV and 3σ ΔV dispersions for the tightly constrained optimal solution. . . 165

7.33 Comparison of mean ΔV, total $\Delta V^{3\sigma}$, and D_f solutions of the NMC resizing scenario using multiple waypoints. Red line represents the 5 m position dispersion requirement. 166

8.1 Approach Ellipsoid and Keep-out Zones for the International Space Station represent method for keep-out zone safety during proximity operations. Figure is not to scale. 171

8.2 Tree analysis of the three-impulse NMC resizing scenario showing the free-drifting trajectory branches. All units are in meters. 172

8.3 Tree analysis of the three-impulse NMC resizing scenario showing the range to the RSO for each free-drifting trajectory branch. 172

8.4 1σ covariance boundary in the local and principal axes for a 2-dimensional system. 176

8.5 Probability of collision values for tree analysis of the NMC resizing scenario. 178

8.6 Maximum probability of collision values for tree analysis of the NMC resizing scenario. 178

8.7 Relative range for tree analysis of the NMC resizing scenario with overlayed P_c values. 179

8.8 Position dispersion ellipse and the resulting range uncertainties. The 3σ range uncertainty intersects the keep-out zone while the 3σ ellipse does not. 182

8.9 Relative range for tree analysis of the NMC resizing scenario with overlayed range uncertainty bounds. 182

8.10 Position dispersion ellipse and resulting Mahalanobis distance to the RSO (4.17σ) and KOZ along the line-of-sight vector (3.67σ). 184

8.11 Probability that x lies within the hyperellipsoid boundary defined by the equation $x^T D^{-1} x = l^2$. 186

8.12 Calculated distance between scaled ellipse and the origin. 191

8.13 Minimum distance between the 1, 2, and 3σ ellipses and the origin. 191

8.14 Location of the point on the ellipse corresponding to the minimum distance to the origin as l increases from 0. The 1, 2, and 3σ ellipses are shown for reference. 192

8.15 Relative range for tree analysis of the NMC resizing scenario with overlayed minimum distance for the 4σ bounded probability covariance. 194

8.16 Relative range for tree analysis of the NMC resizing scenario with overlayed minimum distance for the 4σ bounded probability covariance. Results are zoomed in to show the keep-out zone violation. 195

9.1 Range and 4σ minimum range projection for simply constrained NMC resizing scenario. 202

9.2 Probability of collision for simply constrained NMC resizing scenario. 202

9.3 NMC resizing scenario trajectory path for the safety constrained optimal solution. 204

9.4 Tree analysis of safety constrained NMC resizing scenario. 204

9.5 Range and 4σ minimum range for safety constrained NMC resizing scenario. 205

9.6 Probability of collision for safety constrained NMC resizing scenario. 206

9.7 ΔV and 3σ ΔV dispersion values for the safety constrained NMC resizing scenario. 206

9.8 Range and 4σ minimum range for co-elliptic to NMC scenario without safety constraint. Minimum range violates RSO keep-out zone. 209

9.9 Co-elliptic to NMC scenario trajectory for the safety constrained optimal solution. 210

9.10 Tree analysis of the safety constrained co-elliptic to NMC scenario. Radial-Intrack plane. Branch 1 not shown in full for clarity. 211

9.11 Tree analysis of safety constrained co-elliptic to NMC scenario. Crosstrack-Intrack plane. Branch 1 not shown in full for clarity. 211

9.12 Range and 4σ minimum range for safety constrained co-elliptic to NMC scenario. 212

9.13 True position dispersion ellipsoid for tree analysis branch 2 at the time of minimum distance relative to the RSO keep-out zone. 213

9.14 True position dispersion ellipsoid for tree analysis branch 3 at the time of minimum distance relative to the RSO keep-out zone. 213

9.15 ΔV and 3σ ΔV dispersion values for the safety constrained co-elliptic to NMC scenario. 214

9.16 ΔV and 3σ ΔV dispersions for the simply constrained V-bar hop scenario. 218

9.17 Sun and earth angles for the simply constrained V-bar hop scenario. 218

9.18 Final true position dispersion for the simply constrained V-bar hop scenario. Blue circle represents the 2 m dispersion. 219

9.19 Tree analysis of simply constrained V-bar hop scenario. 219

9.20 Range and 4σ minimum range projection for simply constrained V-bar hop scenario. 220

9.21 ΔV and 3σ ΔV dispersions for the simply constrained V-bar hop scenario. 220

9.22 V-bar hop scenario trajectory for the safety constrained optimal solution. . 221

9.23 Tree analysis of the V-bar hop scenario with safety constraint. 222

9.24 Range and 4σ minimum range projection for the V-bar hop scenario with safety constraint. 222

9.25 ΔV and 3σ ΔV dispersion values for the V-bar hop scenario with safety constraint. 223

ACRONYMS

AAPF	Adaptive Artificial Potential Function
ADCS	Attitude Determination and Control System
AFRL	Air Force Research Laboratory
ANGELS	Automated Navigation and Guidance Experiment for Local Space
AON	Angles-only Navigation
APF	Artificial Potential Function
ASAT	Anti-satellite
ATV	Automated Transfer Vehicle
AVGS	Advanced Video Guidance System
CW	Clohessy-Wiltshire
DART	Demonstration for Autonomous Rendezvous Technology
EAGLE	ESPA Augmented Geostationary Laboratory Experiment
FFT	Free final time
GA	Genetic Algorithm
GEO	Geosynchronous Earth Orbit
GNC	Guidance, Navigation, and Control
GSSAP	Geosynchronous Space Situational Awareness Program
IR	Infrared
ISS	International Space Station
ITAR	International Traffic in Arms Regulations
Lidar	Light Detection and Ranging
LinCov	Linear Covariance
LEO	Low Earth Orbit
LP	Linear Programming
LROE	Linear Relative Orbital Elementes
LTI	Linear, time-invariant
LVLH	Local-vertical, local-horizontal

MC	Monte Carlo
MPC	Model Predictive Control
MUBLCOM	Multiple Paths, Beyond-Line-of-Sight Communications
NMC	Natural Motion Circumnavigation
OE	Orbital Express
OSAM	On-orbit servicing, assembly and manufacturing
PSD	Power Spectral Density
P_c	Probability of collision
pdf	Probability Density Function
PSO	Particle Swarm Optimization
RHC	Receding Horizon Control
RIC	Radial, In-track, Cross-track
RPO	Rendezvous and Proximity Operations
RPOD	Rendezvous, Proximity Operations, and Docking
RSO	Resident Space Object
STORM	Stochastic Trajectory Optimization for RPO Missions
XSS	Experimental Small Satellite

CHAPTER 1

INTRODUCTION

On January 29, 2003 the US Air Force Research Laboratory (AFRL) launched the Experimental Small Satellite (XSS)-10 satellite into a low earth orbit aboard a Delta II rocket [1]. While the mission only lasted 24 hours, the 28 kg maneuverable technology demonstration satellite generated a lasting impact on the field of Rendezvous and Proximity Operations (RPO). Space rendezvous is the process of intentionally maneuvering one spacecraft into the vicinity of a second spacecraft, typically through the use of a propulsion system. Proximity operations are the range of activities that the maneuvering spacecraft may perform while in the vicinity (i.e. proximity) of the target satellite. RPO activities may include inspection, servicing, or docking. XSS-10 provided the initial lessons learned for several AFRL follow-on RPO missions including XSS-11 (2005) [2], Automated Navigation and Guidance Experiment for Local Space (ANGELS) (2014) [3], and Mycroft (2018) [4, 5] that would enable autonomous, safe proximity operations. AFRL was joined by several other organizations in the time since XSS-10, to design and deploy advanced RPO spacecraft including NASA's Demonstration for Autonomous Rendezvous Technology (DART) (2005) [6], DARPA's Orbital Express (2007) [7], the Swedish Space Corporation's Prisma (2010) [8], and the Air Force's Geostationary Space Situational Awareness Program (GSSAP) (2014) [9], with many other missions expected in the future.

A key aspect of all of these missions is the interaction between the spacecraft's Guidance, Navigation, and Control (GNC) subsystem and the path planning tools and techniques implemented to achieve mission success in the presence of the vehicle's specific capabilities and constraints. This dissertation proposes a new RPO path planning technique, based on closed-loop linear covariance techniques, that takes into account the spacecraft's GNC system performance and limitations to generate safe and robust trajectories in the presence of system errors.

This chapter will introduce the concepts of relative orbital motion and trajectory planning for closed-loop GNC systems. Current trajectory planning techniques will be discussed briefly before exploring some of the real-world limitation that must be, but frequently are not, accounted for in the trajectory planning process. Special attention will focus on collision risks and trajectory dispersions.

1.1 Relative Orbital Motion

Critical to all RPO missions are the underlying relative orbital mechanics of one spacecraft relative to another. These models define the dynamics of an active, maneuvering vehicle relative to a passive vehicle. The passive and active vehicles may be referred to as the target and chaser, deputy and chief, or Resident Space Object (RSO) and simply the vehicle. This relative motion is commonly expressed in an RSO-centric frame known as the local-vertical, local-horizontal (LVLH) frame that rotates with the RSO's orbit. One version of the LVLH frame is the slightly more descriptive Radial, Intrack, Crosstrack (RIC) frame as depicted in Figure 1.1.

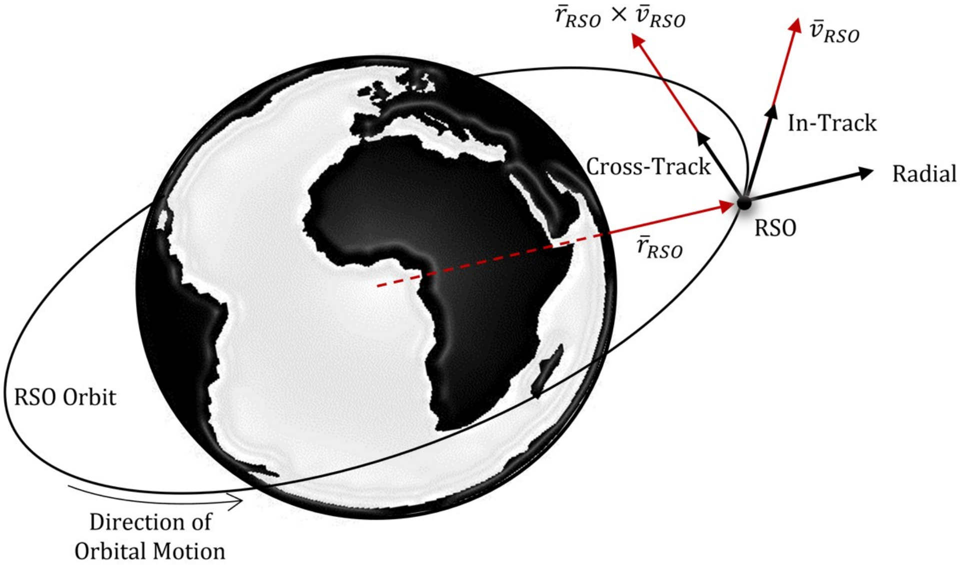

Fig. 1.1: LVLH (RIC) frame attached to the inertial reference frame.

The origin of the RIC frame is the RSO's inertial position, $\bar{r}_{RSO}$, and its orthogonal basis aligns with the RSO's radial position vector ($\hat{i}^{LVLH}$), the inertial angular momentum vector ($\hat{k}^{LVLH}$), and the completed triad ($\hat{j}^{LVLH}$). For an RSO in a circular reference orbit, the final basis vector ($\hat{j}^{LVLH}$) aligns with the RSO's velocity vector, $\bar{v}_{RSO}$, which is tangential to the reference orbit.

The most common (and simple) model for relative orbital motion is the Clohessy-Wiltshire (CW) equations [10]. For circular orbits, the CW equations describe two harmonic oscillators that are decoupled in the in-plane ($\hat{i}^{LVLH}$ and $\hat{j}^{LVLH}$) and out-of plane ($\hat{k}^{LVLH}$) components. The in-plane oscillator defines a 2x1 elliptical path who's center moves along the intrack direction at a rate proportional to its offset in the radial direction, all while the spacecraft moves around the translating elliptical path. A sample of this relative motion in the RIC frame is depicted in Figure 1.2. The detailed derivation of the CW equations is provided in Section 2.1,

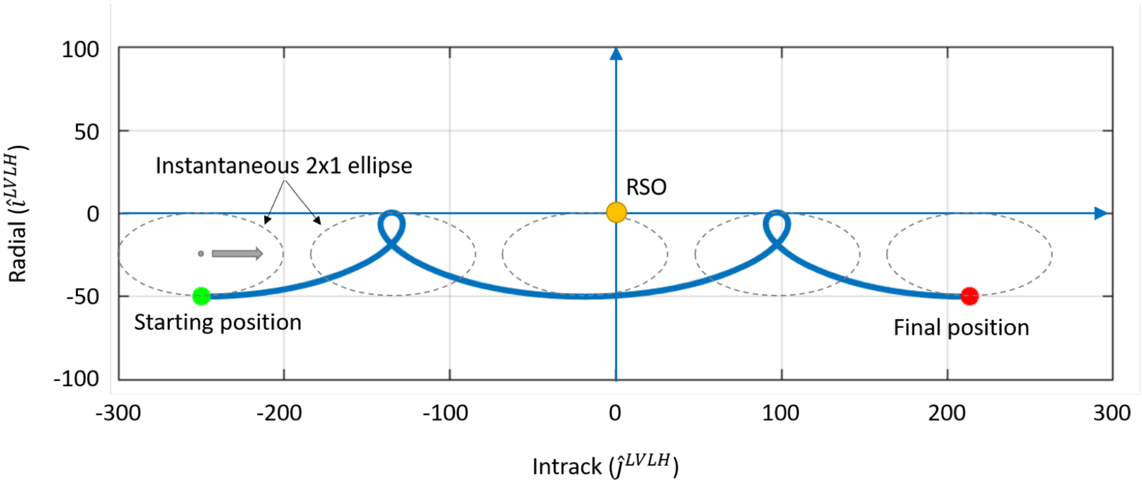

Fig. 1.2: Sample relative path defined by the CW equations in the LVLH frame. The elliptical path is translating from left to right.

1.2 GNC Systems and Path Planning Techniques

The execution of an RPO mission consists of several interconnected components, including the onboard GNC algorithms, and ground-based planning software used to generate a desired path for the spacecraft to follow. Many early RPO missions operated as an open-loop system where the ground would perform all key navigation and planning steps, uploading timed maneuvers for spacecraft execution. While such methods reduce spacecraft complexity, the overall system is susceptible to data latency and error build-up. As a result, second-generation RPO systems relied more heavily on onboard processing for GNC. While bringing additional capability to a mission this approach also required increased planning and scrutiny to ensure mission success lest the vehicle's software have issues and create unnecessary risk.

The fundamental component of current RPO systems is a robust relative navigation system (sometimes referred to as an RPO package) that can accurately estimate the relative position and velocity of the vehicle with respect to the RSO. Popular navigation methods include passive angles-only navigation [11], active lidar-based navigation [12], or advanced computer vision techniques that produces full pose (relative position and orientation) information [13]. The second key component of an RPO mission is the onboard guidance algorithms capable of producing automated maneuver commands derived from the relative navigation and ground provided goals (e.g. station-keeping or waypoint following). While onboard guidance algorithms can be complex and tailored for specific activities, many current missions are pushing toward simplified waypoint following techniques that relies on advanced ground planning to feed the spacecraft a robust and safe trajectory to be followed over an extended planning period.

With a spacecraft capable of following arbitrary, waypoint-defined trajectories loaded from the ground, mission execution becomes more reliant on the ground planning tools to generate safe and optimal trajectories. Ground planning tools can be generic or custom built for specific activities. These tools can also be connected to purpose-built or industry standard optimization routines where the outputs are analyzed against critical spacecraft constraints and performance metrics before commands can be uplinked to the spacecraft. Additional details on these existing guidance techniques are provided in Sections 2.4.2 and 2.4.4.

Examples of purpose-built algorithms that can be used for ground planning include linear methods with safety constraints, fuel-optimal methods with moving terminal constraints, or methods designed to improve navigation performance. While the list of potential algorithms is extensive, most suffer from the same downfall in that they do not account for all mission constraints or objectives in the problem formulation. Additionally, safety evaluations are commonly performed after the trajectories are generated resulting in the need to modify trajectories (often performed manually) in order to meet mission requirements. Ground planning software that can account for the numerous planning constraints would improve overall trajectory design, reduce fuel consumption and improve vehicle safety.

1.3 Planning Constraints

Calculating the maneuvers required to transfer between relative waypoints is a straight forward exercise executed by countless students of orbital mechanics [14]. What this simple approach fails to account for are real world limitations and constraints inherent to satellite operations. This section briefly describes some of the most common limitations and constraints and their impact on the planning process.

1.3.1 ΔV Usage

All RPO spacecraft are fuel limited. Minimizing fuel usage during operations can significantly extend a vehicle's mission lifetime or expand the operations than can be accomplished during a certain event. Numerous fuel optimization techniques have been developed for RPO missions and are popular in ground planning software. These approaches typically employ standard or novel optimization techniques to find a nominal, minimum fuel, trajectory between the given boundary conditions. Navigation and control errors, however, can lead to off-nominal trajectories that result in increased fuel consumption compared to the calculated optimal reference. Large jumps in fuel usage compared to the optimal may occur when the navigation solution jumps as a result of new or erroneous measurements. Onboard trajectory maintenance may also increase fuel consumption due to tight constraints about the nominal trajectory. Some error conditions, such as last minute maintenance maneuvers, can result in significant fuel usage that is orders of magnitude larger than the calculated optimal. Discrepancies between the predicted fuel consumption and actual fuel expenditure can lead to overly conservative fuel budgets or artificial limitations on mission operations.

RPO mission planners typically assign an arbitrary scale factor to the nominally calculated ΔV values to account for these errors. Scale factors can be between 1.1 and 2.0 depending on the planned scenario, thruster configuration, navigation uncertainty, and familiarity with the vehicle.

1.3.2 Lighting

After ΔV constraints, the most common constraints placed on the trajectory design process are lighting conditions. Missions that rely on passive electro-optical sensors can be blinded when the angle between the sensor and the sun drops below a specified threshold preventing precious navigation updates and causing the navigation errors to increase throughout the outage [15]. If acted upon, the resultant navigation errors may produce off-nominal trajectories and increase fuel usage. This condition is analogous to a spacecraft's attitude determination and control system (ADCS) relying on gyro data only to determine attitude whenever the star tracker is blinded. If an outage lasts longer than a specified amount of time, fault detection monitors may force the spacecraft into an associated safe mode halting the planned mission.

The relative lighting angle of an RSO can also have a significant effect on sensor data processing. These off-nominal lighting conditions can produce sensor measurement errors outside nominally anticipated values. In electro-optical systems, this dependency commonly presents itself as either an offset between the true RSO center-of-mass and the measured center-of-brightness or as increased measurement noise. Eclipse conditions are similar to low solar phase angles and must be tracked depending on the navigation sensors used. Sensors using optical wavelengths will experience navigation outages during eclipse while infrared (IR)-based sensors may be able to operate through eclipse depending on their wavelength and the RSO's thermal properties. Eclipse durations for geosynchronous earth orbits (GEO) last upwards of 72 minutes but are rare, occurring only near the equinoxes. For low earth orbit (LEO), eclipses can last up to 41% of the orbital period. For all cases, eclipse constraints must be accounted for in the planning process.

Similar to the lighting constraints, the background scene presented to the navigation sensor can result in outages or increased errors. An RSO against an earth background may result in larger tracking errors if the image processing is capable of tracking at all.

Typical lighting constraints are depicted in Figure 1.3.

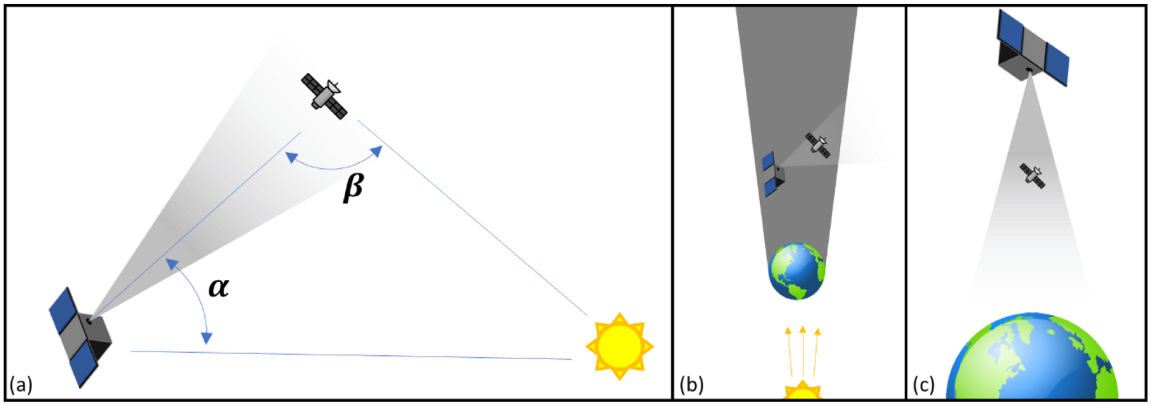

Fig. 1.3: Typical lighting constraints for RPO include (a) solar phase angles, (b) eclipse, and (c) scene back-ground including the earth.

1.3.3 Collision Risk

Ever present in all RPO missions is the risk of collision between the two objects. On 15 April 2005, the Demonstration of Autonomous Rendezvous Technology (DART) satellite was launched from a Pegasus-XL rocket into a low-earth polar orbit to rendezvous shortly thereafter with the previously launched Multiple Paths, Beyond-Line-of-Sight Communications (MUBLCOM) satellite. During the proximity operations phase of the short mission, DART experienced an anomaly and collided with MUBLCOM, knocking it into a slightly higher orbit [16]. As a result of the DART collision, increased scrutiny has been placed on all subsequent RPO missions.

Collision risks can be managed through either *passive* or *active* techniques. Passive collision avoidance indicates that the spacecraft motion is safe over a given time horizon even if the spacecraft fails to perform future maneuvers. This is primarily achieved through trajectories that do not pass directly through the RSO's orbit, such as the so-called natural motion circumnavigation (NMC) orbits. While generally useful for prolonged operations, passively safe orbits are limited in utility necessitating active safety techniques. An 2x1 elliptical NMC centered on the RSO is depicted in Figure 1.4 where the ellipse center's radial offset is zero resulting on no motion in the intrack direction. Passive safety is critical in the unlikely event that any given maneuver does not execute (as might occur if the vehicle suffers a reboot or other unexpected anomaly).

Active collision avoidance is simply the process of quantifying collision risks when designing a relative trajectory and employing techniques to mitigate that risk. A common method is to consider the predicted closest approach between the vehicle and the RSO. In other words, don't plan a path that runs through or close to the RSO. Flight operations may dictate the minimum allowable closest approach based on the vehicle performance, the RSO size or organizational policies.

Probability-based risk assessments can also be implemented to determine the risk of collision between objects. Probability of collision assessments are commonly evaluated for spacecraft conjunctions based on ground tracking data. These same risk assessments are not commonly performed for proximity operations due to ever-changing conditions and computational difficulties.

1.3.4 Navigation and Trajectory Dispersions

An often overlooked constraint in mission planning is the limitation of a spacecraft's relative navigation system. An RPO-class spacecraft can only follow its commanded trajectory as well as it knows and is able to accurately adjust its current trajectory. Trajectory errors or dispersions can result in off-nominal performance, increased fuel usage, or increased risk of collision. Navigation constraints are particularly important in an angles-only navigation

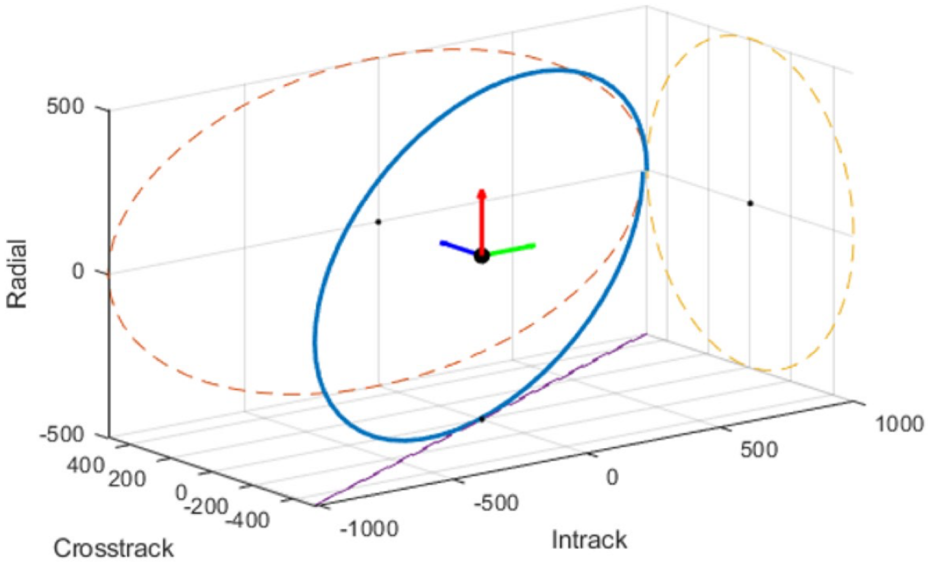

Fig. 1.4: Passive abort safety ellipse commonly used to minimize the risk of collision.

(AON) scheme due to the so-called range observability problem [17]. Due to the limited information in angle measurements and the potential for multiple relative trajectories to generate the same measurements, AON systems struggle to accurately estimate the range to an RSO. While significant research has been put into the range observability problem, AON systems are still inherently limited in range accuracy. Trajectory optimization techniques have been developed and implemented to account for this limitation by creating optimal observability maneuvers that attempt to reduce the overall range uncertainty. These methods, however, typically do not account for potential trajectory dispersions resulting from the combined navigation and control loop.

1.4 Research Scope and Overview

The combination of these constraints, coupled with the vehicle's actual performance, makes for a difficult mission planning problem. Tools that optimize over one parameter (e.g. fuel) often fail to account for other constraints (e.g. lighting). Unfortunately, a common approach to this difficult problem is to start with a preferred optimization technique and to adjust the output (i.e. nominal trajectory) manually until all constraints are met. While all constraints may eventually be met in this approach, an optimal trajectory is not rigorously computed. Additionally, these off-nominal trajectories may create unsafe conditions due to the limitation in the evaluation process, particularly related to trajectory dispersions.

1.4.1 Constrained Path Planning Thesis

The thesis of this proposed dissertation is that *closed-loop linear covariance techniques can be coupled with stochastic optimization techniques to develop optimal relative spacecraft trajectories that meet safety constraints while simultaneously meeting performance requirements in the presence of sensor noise, actuator errors, and dynamics modeling errors.*

This research will focus on the three key areas. First will be the development of stochastic models describing the relative motion and GNC algorithms of an RPO system. These models will also include considerations for all key system limitations and constraints. Second, a novel collision risk assessment criteria derived from the stochastic models will be developed to provide reliable safety assessments of planned trajectories and dispersions. Finally, an optimization technique will be paired with the stochastic models to perform optimal relative path planning that includes all performance considerations and constraints while minimizing the statistical fuel consumption.

Chapter 2 of this dissertation will derive of the underlying CW equations used to describe the relative orbital motion common to proximity operations. A thorough survey of RPO missions and related technologies is then presented. Chapter 3 develops the Monte Carlo and linear covariance techniques necessary for analyzing closed-loop GNC systems. This is followed in Chapter 4 by the full derivation of the RPO linear covariance model that addresses the key constraints and performance objectives considered for this research. Chapter 5 will then explore the differences between this RPO linear covariance model and deterministic path planning methods for simple two-burn trajectories. Given its performance capabilities, the linear covariance model is paired with a Genetic Algorithm and local optimization solver in Chapter 6 to determine optimal path planning. This optimal planning tool is then used to evaluate several operationally relevant, multi-burn scenarios in Chapter 7. Chapter 8 considers spacecraft safety and presents several approaches for evaluating spacecraft safety. A novel bounded probability of collision method is developed and integrated into the linear covariance model. Chapter 9 evaluates the optimal multi-burn path planning technique with this new safety criteria. Finally, Chapter 10 provides a summary and general conclusions.

CHAPTER 2

ORBITAL RENDEZVOUS AND PROXIMITY OPERATIONS

Just a short 5 years after launching Sputnik, the first satellite into orbit, the Soviet Union made a second leap forward with their Vostok 3 and 4 satellites. Launched a day apart, the satellites trajectories were designed so that they passed within 6.5 km of each other [18]. While this first step may seem simple by modern space system standards, the mission showed the feasibility of performing orbital rendezvous and proximity operations. Many missions have followed, each building on the technologies of their predecessors. This chapter develops the key dynamics for orbital motion followed by the dynamics describing the motion of one satellite relative to a reference orbit. A detailed survey of previous RPO missions is presented. Finally, related research is presented in the areas of relative motion modeling, deterministic and constrained path planning techniques, flight safety and collision avoidance techniques, followed by relative navigation and linear covariance analyses.

2.1 Dynamics

This section discusses the the basics of classical orbital mechanics and relative orbital motion between two satellites in proximity. The Clohessy-Wiltshire equations are derived from first principles for both continuous and discrete dynamics.

2.1.1 Orbital Motion Dynamics

In his 1687 work *Principia*, Isaac Newton published three laws of motion for particles in an inertial (nonrotating and nonaccelerating) reference frame [19]. The three laws of motion are stated as follows:

1. A particle remains in its state of rest or uniform, straight-line motion unless it is acted upon by forces to change that state.

2. The force acting on a particle equals the mass of the particle times its inertial acceleration.
3. For every force acting on a particle, there is an equal and opposite reaction force.

The second laws is commonly written as:

$$\boldsymbol{F} = m\boldsymbol{a} \tag{2.1}$$

where $\boldsymbol{F}$ is the force acting on the particle, m is the mass of the particle, and $\boldsymbol{a}$ is the inertial acceleration of the particle resulting from the applied force.

Principia also presented Newton's law of universal gravitation:

- Every particle attracts every other particle in the universe with a force that is directly proportional to the product of their masses and inversely proportional to the square of the distance between their centers

This law is commonly written in vector form as:

$$\boldsymbol{F} = -\frac{Gm_1m_2}{r^3}\boldsymbol{r} \tag{2.2}$$

where F is the gravitational force acting on the masses m_1 and m_2, $\boldsymbol{r}$ is the position vector of m_2 relative to m_1 and r is the magnitude of the vector $\boldsymbol{r}$. G is the universal gravitational constant and equal to 6.67259×10^{-11} $\mathrm{Nm^2/kg^2}$.

For the case of two objects of masses m_1 and m_2 with position vectors $\boldsymbol{R_1}$ and $\boldsymbol{R_2}$ in an inertial frame (as shown in Figure 2.1), the force on each object can be written as:

$$m_1\ddot{\boldsymbol{R}}_{\boldsymbol{1}} = +\frac{Gm_1m_2}{r^3}\boldsymbol{r} \tag{2.3}$$

$$m_2\ddot{\boldsymbol{R}}_{\boldsymbol{2}} = -\frac{Gm_1m_2}{r^3}\boldsymbol{r} \tag{2.4}$$

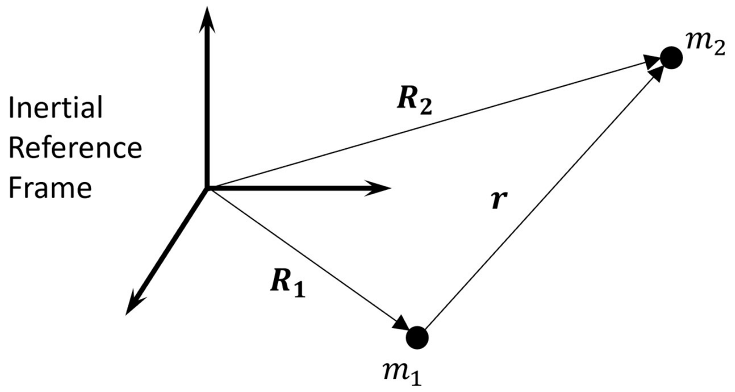

Fig. 2.1: Two-body problem in inertial reference frame.

where $\ddot{\boldsymbol{R}} = d^2\boldsymbol{R}/dt^2$ is the acceleration of the particle in the inertial reference frame. Note that the force acting on each object are equal and in opposite directions as stated by Newton's third law.

Subtracting Equation (2.3) from Equation (2.4) yields:

$$\ddot{\boldsymbol{r}} + \frac{G(m_1 + m_2)}{r^3}\boldsymbol{r} = 0 \tag{2.5}$$

where $\ddot{\boldsymbol{r}} = \ddot{\boldsymbol{R}}_{\boldsymbol{2}} - \ddot{\boldsymbol{R}}_{\boldsymbol{1}}$ is the acceleration of m_2 relative to m_1 in the inertial reference frame. For a small object (e.g. a satellite) in proximity to a larger object (e.g. a planet) where the mass of the smaller object is negligible, Equation (2.5) can be simplified as:

$$\ddot{\boldsymbol{r}} + \frac{\mu}{r^3}\boldsymbol{r} = 0 \tag{2.6}$$

where μ is called the planet's *gravitational parameter*. For the earth, $\mu_\oplus = Gm_\oplus = 398,600\ \text{km}^3/\text{s}^2$. Equation (2.6) provides the dynamics for the classic two-body problem, defining the orbital motion of a small object relative to a single larger object. This second-order differential equation describes elliptical orbits as shown in Figure 2.2.

The beauty of Equation (2.6) is that it, through Newton's laws, provides a dynamical model for orbital motion that perfectly matches Johannes Kepler's empirical laws of planetary motion [20].

1. The orbit of each planet is an ellipse, with the sun at a focus.
2. The line joining the planet to the sun sweeps out equal areas in equal times.
3. The square of the period of a planet is proportional to the cube of its mean distance from the sun.

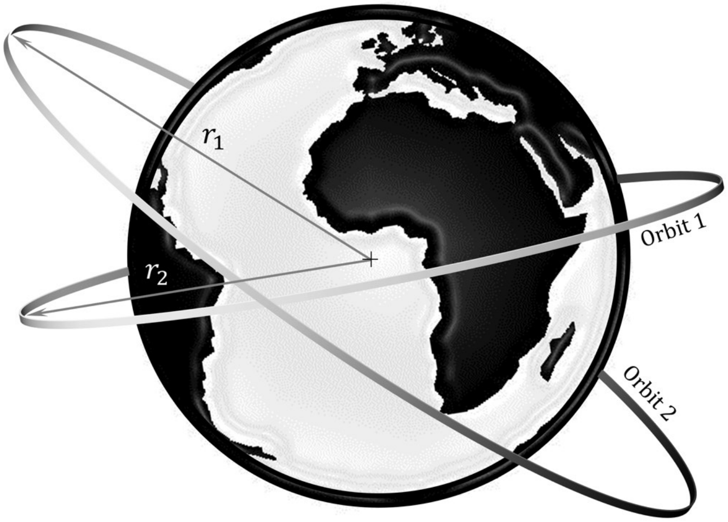

Fig. 2.2: Two orbits around the earth defined by the two-body problem.

It is important to note that there are many additional forces acting on a satellite's motion beyond those described in the two-body problem. Satellites orbiting the earth are also affected by gravitational forces from the sun, moon and the other planets of the solar system. Atmospheric drag can greatly affect low earth orbit (LEO) satellites while solar radiation pressure can be a major perturbation to geosynchronous orbits. Equation (2.6) can be rewritten to account for these perturbations [21]:

$$\ddot{\boldsymbol{r}} + \frac{\mu}{r^3}\boldsymbol{r} = \boldsymbol{a}_p \tag{2.7}$$

where $\boldsymbol{a}_p$ is the summation of perturbing accelerations acting on the satellite.

For this work we are principally concerned with circular (or near circular) orbits where the magnitude of the orbital position, r, is constant. In this unique case the time required to complete one orbit, known as the orbital period, is calculated as:

$$T = 2\pi\sqrt{\frac{r^3}{\mu}} \tag{2.8}$$

as defined by Kepler's third law. Orbital period can also be written in terms of the orbit's mean motion, ω:

$$T = \frac{2\pi}{\omega} \tag{2.9}$$

where:

$$\omega = \sqrt{\frac{\mu}{r^3}} \tag{2.10}$$

For orbits near the earth surface ($r \approx 7000$ km), $T = 5828$ sec and ω is on the order of 1.1×10^{-3} rad/s. For geostationary orbits ($r = 42164$ km), $T = 86164$ sec and $\omega = 7.29 \times 10^{-5}$ rad/s.

For additional details on orbital mechanics and astrodynamics the reader is directed to the works by Curtis [14], Montenbruck and Gill [21], or Bond and Allman [22].

2.1.2 Relative Motion Dynamics

For the case of two satellites in proximity to one another (orbiting a common celestial body) we can further expand on the orbital mechanics to describe the motion of one satellite relative to the other. In defining this relative motion it is convenient to first define a new coordinate frame, the local-vertical local-horizontal (LVLH) frame (first introduced in Section 1.1).

The origin of the LVLH frame is the reference satellite's inertial position, $\boldsymbol{r}$ and moves with the satellite along its orbital path. The x-axis, also called the Radial vector, points straight up from the earth. The z-axis is aligned with the satellite's angular momentum vector:

$$\boldsymbol{h} = \boldsymbol{r} \times \dot{\boldsymbol{r}} \tag{2.11}$$

Finally, the y-axis completes the triad. For a circular orbit the y-axis is aligned with the satellite's velocity vector. The LVLH frame is shown in Figure 2.3 and its basis vectors are defined as:

$$x^{LVLH} = \frac{\boldsymbol{r}}{||\boldsymbol{r}||} \tag{2.12a}$$

$$z^{LVLH} = \frac{\boldsymbol{r} \times \dot{\boldsymbol{r}}}{||\boldsymbol{r} \times \dot{\boldsymbol{r}}||} \tag{2.12b}$$

$$y^{LVLH} = \frac{z^{LVLH} \times x^{LVLH}}{||z^{LVLH} \times x^{LVLH}||} \tag{2.12c}$$

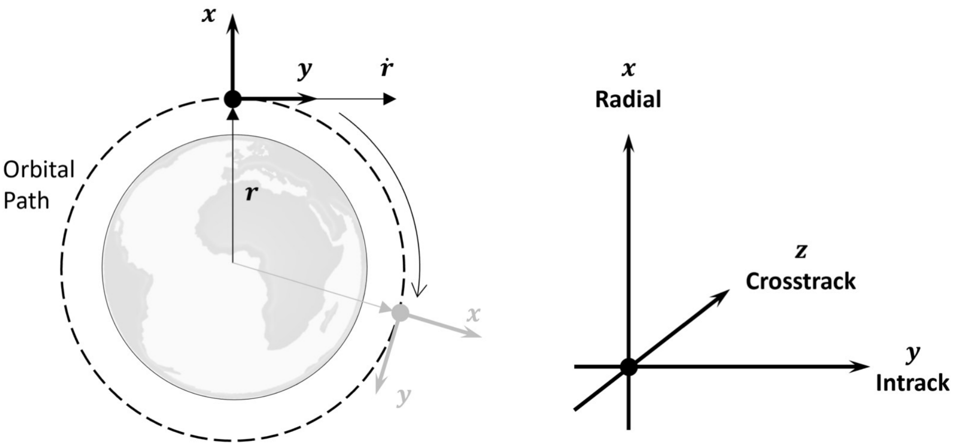

Fig. 2.3: Local-vertical, local-horizontal frame.

For the derivation of the relative motion dynamics, we will use the approach proposed by Prussing and Conway [23]. Additionally, we will follow the terminology proffered by Lovell and refer to the primary vehicle (defining the LVLH frame) as the chief and the second vehicle as the deputy [24]. The relative geometry of the chief and deputy are depicted in Figure 2.4 where $\boldsymbol{r}_0$ and $\boldsymbol{r}$ are the inertial position vectors of the chief and deputy, respectively, and $\boldsymbol{r}_{rel}$ is the position vector of the deputy relative to the chief. These vectors provide the following relation:

$$\boldsymbol{r} = \boldsymbol{r}_0 + \boldsymbol{r}_{rel} \tag{2.13}$$

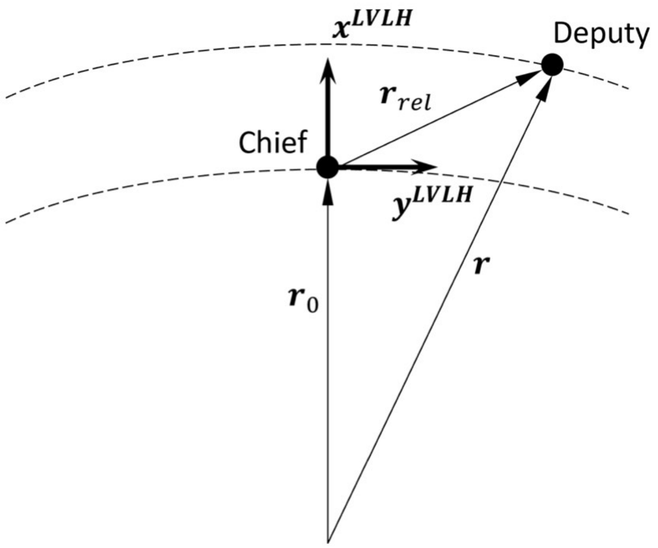

Fig. 2.4: Relative geometry and LVLH coordinate frame of two satellites in proximity around earth.

Through kinematic relationships we can differentiate Equation (2.13) to provide the dynamics of the deputy satellite relative to the chief:

$$\ddot{\boldsymbol{r}} = \ddot{\boldsymbol{r}}_{\boldsymbol{0}} + \boldsymbol{a}_{rel} + 2(\boldsymbol{\Omega} \times \boldsymbol{v}_{rel}) + \dot{\boldsymbol{\Omega}} \times \boldsymbol{r}_{rel} + \boldsymbol{\Omega} \times (\boldsymbol{\Omega} \times \boldsymbol{r}_{rel}) \tag{2.14}$$

where:

$\ddot{\boldsymbol{r}}$ = the inertial acceleration of the deputy satellite

$\ddot{\boldsymbol{r}}_0$ = the inertial acceleration of the chief satellite

$\boldsymbol{a}_{rel}$ = the acceleration of the deputy relative to the chief

$\boldsymbol{v}_{rel}$ = the velocity of the deputy relative to the chief

$\boldsymbol{\Omega}$ = the angular velocity of the LVLH frame

$2(\boldsymbol{\Omega} \times \boldsymbol{v}_{rel})$ = the Coriolis acceleration

$\dot{\boldsymbol{\Omega}} \times \boldsymbol{r}_{rel}$ = the Euler acceleration

$\boldsymbol{\Omega} \times (\boldsymbol{\Omega} \times \boldsymbol{r}_{rel})$ = the centripetal acceleration

The magnitude of the angular velocity is given by Equation (2.10) and aligned with the satellite's orbital momentum vector defined in Equation (2.11). When coordinatized in the LVLH frame this vector coincides with the LVLH z-axis and can be written as:

$$\boldsymbol{\Omega} = \begin{bmatrix} 0 \\ 0 \\ \omega \end{bmatrix} \tag{2.15}$$

Equation (2.14) can be rewritten in terms of the relative position, velocity, and acceleration of the deputy:

$$\boldsymbol{a}_{rel} = \ddot{\boldsymbol{r}}_{rel} - 2(\boldsymbol{\Omega} \times \boldsymbol{v}_{rel}) - \dot{\boldsymbol{\Omega}} \times \boldsymbol{r}_{rel} - \boldsymbol{\Omega} \times (\boldsymbol{\Omega} \times \boldsymbol{r}_{rel}) \tag{2.16}$$

where $\ddot{\boldsymbol{r}}_{rel} = \ddot{\boldsymbol{r}} - \ddot{\boldsymbol{r}}_0$ is the inertial acceleration of the deputy relative to the chief. Consistent with our assumptions in Equations (2.8) and (2.10), we will again assume that the chief satellite is in a circular orbit such that $\dot{\Omega} = 0$ and that Equation (2.16) becomes:

$$\boldsymbol{a}_{rel} = \ddot{\boldsymbol{r}}_{rel} - 2(\boldsymbol{\Omega} \times \boldsymbol{v}_{rel}) - \boldsymbol{\Omega} \times (\boldsymbol{\Omega} \times \boldsymbol{r}_{rel}) \tag{2.17}$$

With the kinematics defined, we will next look to simplify the relative acceleration term, $\ddot{\boldsymbol{r}}_{rel}$, in order to make Equation (2.17) more useful. The two-body gravitational field in Equation (2.6) can be rewritten in the general form:

$$\ddot{\boldsymbol{r}} = g(\boldsymbol{r}) \tag{2.18}$$

where $g(\boldsymbol{r})$ is the nonlinear function describing the gravitational acceleration at location $\boldsymbol{r}$. Substituting Equation (2.13) into Equation (2.18) gives:

$$\ddot{\boldsymbol{r}} = \ddot{\boldsymbol{r}}_0 + \ddot{\boldsymbol{r}}_{rel} = g(\boldsymbol{r}_0 + \boldsymbol{r}_{rel}) \tag{2.19}$$

The nonlinear function is then expanded in a Taylor series expansion about the *known* reference orbit $\boldsymbol{r}_0$:

$$g(\boldsymbol{r}_0 + \boldsymbol{r}_{rel}) = g(\boldsymbol{r}_0) + \frac{\partial g(\boldsymbol{r}_0)}{\partial \boldsymbol{r}_0}\boldsymbol{r}_{rel} + \boldsymbol{b} \tag{2.20}$$

where $\partial g(\boldsymbol{r}_0)/\partial \boldsymbol{r}_0$ is the gravity gradient matrix and $\boldsymbol{b}$ represents the second and higher-order terms of the expansion.

For the case of orbital proximity operations, it can safely be assumed that the magnitude of $\boldsymbol{r}_{rel}$ is small compared to the magnitude of the reference orbit's inertial position $\boldsymbol{r}_0$ such that $|r_{rel}/r_0 \ll 1|$. Since the gravity gradient matrix is evaluated at the chief's reference orbit, this assumption allows us to neglect the second-order and higher terms of the expansion with minimal impact to accuracy. For a LEO mission ($r_0 \approx 7000$ km), proximity operations may begin at a relative distance of 100 km, such that $r_{rel}/r_0 = 0.014$. For a GEO mission, starting proximity operations at the same 100 km yields $r_{rel}/r_0 = 0.002$.

Substitute Equation (2.20) into Equation (2.19), ignoring the higher-order terms of the expansion, gives a linear differential equation for the relative acceleration $\ddot{\boldsymbol{r}}_{\boldsymbol{rel}}$:

$$\ddot{\boldsymbol{r}}_{rel} = \boldsymbol{G}(\boldsymbol{r}_0)\boldsymbol{r}_{rel} \tag{2.21}$$

where $\boldsymbol{G}(\boldsymbol{r}_0) = \partial g(\boldsymbol{r}_0)/\partial \boldsymbol{r}_0$ is the 3×3 gravity gradient matrix. Evaluating the partial derivative for the reference orbit, the linear function $\boldsymbol{G}$ is calculated to be:

$$\boldsymbol{G}(\boldsymbol{r}_0) = \frac{\mu}{r_0^5}(3\boldsymbol{r_0}\boldsymbol{r}_0^T - r_0^2\boldsymbol{I}_3) \tag{2.22}$$

where $\boldsymbol{I}_3$ is a 3×3 identity matrix.

In the LVLH frame, the chief's inertial position vector is reduced to the following vector form:

$$\boldsymbol{r}_0^{LVLH} = \begin{bmatrix} r_0 \\ 0 \\ 0 \end{bmatrix} \tag{2.23}$$

Applying this coordinitization to Equation (2.22) produces the following diagonal gravity gradient matrix:

$$\boldsymbol{G}^{LVLH}(\boldsymbol{r}_0) = \frac{\mu}{r_0^3}\begin{bmatrix} 2 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \tag{2.24}$$

Now that we have both the kinematics and linearized gravitational forces, we can combine the two to determine the dynamics of the deputy relative to the chief in the rotating LVLH frame. The relative position vector $\boldsymbol{r}_{rel}$ in the LVLH frame has the following components and derivatives:

$$\boldsymbol{r}_{rel} = \begin{bmatrix} x \\ y \\ z \end{bmatrix}, \quad \boldsymbol{v}_{rel} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix}, \quad \boldsymbol{a}_{rel} = \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} \tag{2.25}$$

Combining Equations (2.15), (2.24), (2.21), and (2.25) into Equation (2.17) yields the following system of linear differential equations:

$$\ddot{x} - 3\omega^2 x - 2\omega\dot{y} = 0 \tag{2.26a}$$

$$\ddot{y} + 2\omega\dot{x} = 0 \tag{2.26b}$$

$$\ddot{z} + \omega^2 z = 0 \tag{2.26c}$$

The equations are linear second-order differential equations with constant coefficients. The x and y equations are coupled while the crosstrack motion defined in z is decoupled.

Equation (2.26) is commonly known as the Clohessy-Wiltshire (CW) equations [10]. First proposed by George Hill in 1878 to describe the relative motion of the moon around the earth [25], the equations were rediscovered in 1960 by Clohessy and Wiltsire to describe the relative motion of two earth-orbiting satellites. Because of this connection, the equations are sometimes referred to as the CWH or HCW equations or even just Hill's equations to properly recognize the American astronomer.

To produce a more convenient formulation, we can write the CW equations as a continuous state space problem with a state vector in terms of the relative position and velocity:

$$\boldsymbol{X} = \begin{bmatrix} x & y & z & \dot{x} & \dot{y} & \dot{z} \end{bmatrix}^T \tag{2.27}$$

$$\dot{\boldsymbol{X}} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 3\omega^2 & 0 & 0 & 0 & 2\omega & 0 \\ 0 & 0 & 0 & -2\omega & 0 & 0 \\ 0 & 0 & -\omega^2 & 0 & 0 & 0 \end{bmatrix} \boldsymbol{X} \tag{2.28}$$

Equation (2.26) can also be solved to generate a discrete state space solution where the state at a given time, $\boldsymbol{X}(t)$, can be determined from the initial state, $\boldsymbol{X}(t_0)$ and the elapsed time.

$$\boldsymbol{X}(t) = \boldsymbol{\Phi}(t, t_0)\boldsymbol{X}(t_0) \tag{2.29}$$

where the initial state is defined as:

$$\boldsymbol{X}(t_0) = \begin{bmatrix} \boldsymbol{r}_0 \\ \boldsymbol{v}_0 \end{bmatrix} \tag{2.30}$$

or

$$\boldsymbol{X}(t_0) = \begin{bmatrix} x_0 & y_0 & z_0 & \dot{x}_0 & \dot{y}_0 & \dot{z}_0 \end{bmatrix}^T \tag{2.31}$$

and the state transition matrix (assuming $t_0 = 0$) is defined as:

$$\mathbf{\Phi}(t) = \begin{bmatrix} 4 - 3\cos\omega t & 0 & 0 & \frac{1}{\omega}(\sin\omega t) & \frac{2}{\omega}(1 - \cos\omega t) & 0 \\ 6(\sin\omega t - \omega t) & 1 & 0 & \frac{2}{\omega}(\cos\omega t - 1) & \frac{1}{\omega}(4\sin\omega t - 3\omega t) & 0 \\ 0 & 0 & \cos\omega t & 0 & 0 & \frac{1}{\omega}\sin\omega t \\ 3\omega\sin\omega t & 0 & 0 & \cos\omega t & 2\sin\omega t & 0 \\ 6\omega(\cos\omega t - 1) & 0 & 0 & -2\sin\omega t & 4\cos\omega t - 3 & 0 \\ 0 & 0 & -\omega\sin\omega t & 0 & 0 & \cos\omega t \end{bmatrix} \tag{2.32}$$

The full derivation of Equation (2.32) is available in Appendix A.

2.2 Relative Maneuvers

A key distinction of proximity operations is the ability to adjust and control the relative trajectory to achieve a desired mission. In this section we will use the CW equations to develop a simplified targeting method that can be used to transfer a satellite from its current relative state to a desired state. This method, commonly referred to as CW targeting, is analogous to to Lambert's problem in inertial space [23].

To simplify the derivation we will partition the state transition matrix defined in Equation (2.32) into four 3×3 submatrices:

$$\mathbf{\Phi}(t) = \begin{bmatrix} \mathbf{\Phi}_{rr}(t) & \mathbf{\Phi}_{rv}(t) \\ \mathbf{\Phi}_{vr}(t) & \mathbf{\Phi}_{vv}(t) \end{bmatrix} \tag{2.33}$$

such that:

$$\boldsymbol{r}(t) = \mathbf{\Phi}_{rr}(t)\boldsymbol{r}_0 + \mathbf{\Phi}_{rv}(t)\boldsymbol{v}_0 \tag{2.34a}$$

$$\boldsymbol{v}(t) = \mathbf{\Phi}_{vr}(t)\boldsymbol{r}_0 + \mathbf{\Phi}_{vv}(t)\boldsymbol{v}_0 \tag{2.34b}$$

Note that the *rel* subscript has been dropped to simplify the notation.

To achieve a desired position at a specified time, $\boldsymbol{r}^*(t)$, we simply need to determine the requisite velocity, $\boldsymbol{v}_0^*$, at our current position that intersects that desired position and adjust our velocity accordingly. This is achieved by rearranging Equation (2.34a):

$$\boldsymbol{v}_0^* = (\boldsymbol{\Phi}_{rv})^{-1} \left[\boldsymbol{r}^*(t) - \boldsymbol{\Phi}_{rr}\boldsymbol{r}_0\right] \tag{2.35}$$

where $\boldsymbol{r}^*$ is three-dimensional desired position in the LVLH frame and is referred to as the target *waypoint*.

In order to reach this waypoint, the spacecraft must impart a change in its current velocity to match the calculated desired velocity.

$$\boldsymbol{v}_0^* = \boldsymbol{v}_0 + \Delta V \tag{2.36}$$

This change in velocity, ΔV, is referred to as a *delta velocity* or *delta v* for short. While the ΔV shown here is impulsive, such maneuvers are not truly achievable through spacecraft propulsion systems. For short maneuvers relative to the satellite's orbital period, the impulsive approximation is adequate for modeling discrete thruster forces. The impulsive approximation is also frequently applied to mission planning activities while letting spacecraft flight algorithms manage the actual maneuver execution.

Using this CW targeting approach, satellite operators can string together a series of waypoints (along with their associated transfer times) to follow any desired path relative to the chief as shown in Figure 2.5.

Path planning, as shown in Chapter 1, is a complex process with significant constraints that must be followed for mission success. Additionally, fuel resources are limited and must be preserved. Spacecraft fuel capacity, while generally a commodity measured in kilograms, is commonly tracked as a ΔV capacity measured in meters per second. ΔV capacity can be calculated using the Tsiolkovsky (ideal) rocket equation [26]:

$$\Delta V = I_{\mathrm{sp}}\ g_0 \ln\left(\frac{m_0 + m_{\mathrm{p}}}{m_0}\right) \tag{2.37}$$

where:

$$I_{\text{sp}} = \text{the specific impulse of the rocket propellant}$$

$$g_0 = \text{the gravitational constant at sea level } (9.81 \text{ m/s}^2)$$

$$m_0 = \text{the spacecraft mass without propellant } (\textit{dry mass})$$

$$m_{\text{p}} = \text{the propellant mass}$$

The combined mass $(m_0 + m_p)$ is referred to as a satellite's *wet mass*. Hydrazine, a common spacecraft propellant, has a specific impulse of ∼230 seconds. A 100 kg small satellite (dry mass) with 20 kg of hydrazine propellant would have a total ΔV capacity of ∼410 m/s for maneuvering.

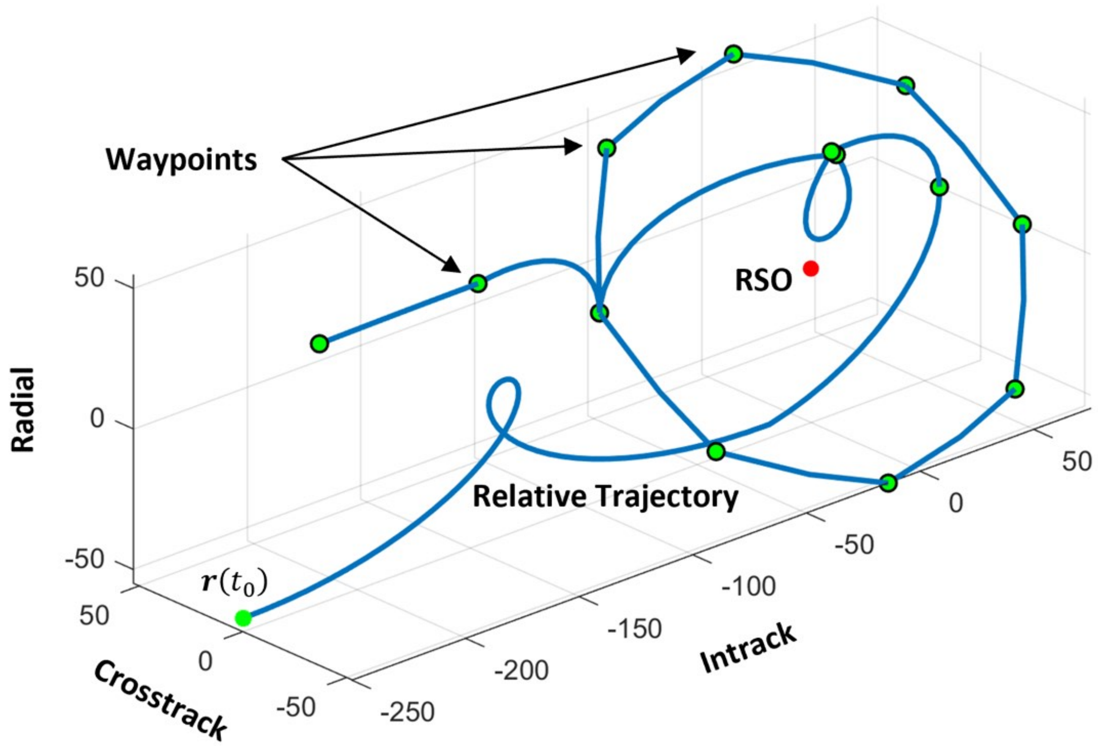

Fig. 2.5: Waypoint-defined relative trajectory about RSO

2.3 Rendezvous and Proximity Operations Missions

Rendezvous and proximity operations missions have their origin in the dawn of the space age when Gordon Cooper and Pete Conrad maneuvered their Gemini V capsule around a phantom reference orbit in 1965. This feat was surpassed only four months later when Gemini VI and VII maneuvered to within 30 cm of each other in December 1965. A detailed history of Gemini mission planning for rendezvous and proximity operations is given by Parten and Mayer [27]. It should be noted that the Soviet Union had previously launched several pairs of their Vostok vehicles where the two were able to pass within several kilometers of each other and establish radio communication. These missions, however, did not have the ability to to maneuver or adjust their orbits preventing true proximity operations. The more capable Gemini maneuvers were preparation for the upcoming Apollo missions that would required RPO operations about both the earth and moon to achieve the nation's goal of landing men on the moon. Young and Alexander provide a history of lunar rendezvous [28]. Maneuvering for these mission were extensively executed by astronauts with ground-based planning techniques. Buzz Aldrin's 1963 MIT PhD dissertation addressed path planning techniques given visual line-of-sight measurements for manned spaceflight [29]. This work led the other astronauts to nickname him "Dr. Rendezvous" when he joined the astronaut corp later that year.

NASA continued performing RPO missions with the advent of the space shuttle. The first such mission occurring on STS-7 in 1983 with the Challenger orbiter deploying and eventually grappling the Shuttle Pallet Satellite (SPAS-1). Both Pearson [30] and Goodman [31] provide detailed histories of shuttle rendezvous and proximity operations.

2003 marked a new phase in RPO as AFRL launched it XSS-10 satellite on a 24 hour autonomous mission [1]. This mission marked a drastic switch away from manned RPO missions to more automated approaches. AFRL followed XSS-10 with the yearlong XSS-11 mission [2]. Details of the activities performed during the XSS-10 and XSS-11 missions are not publicly available.

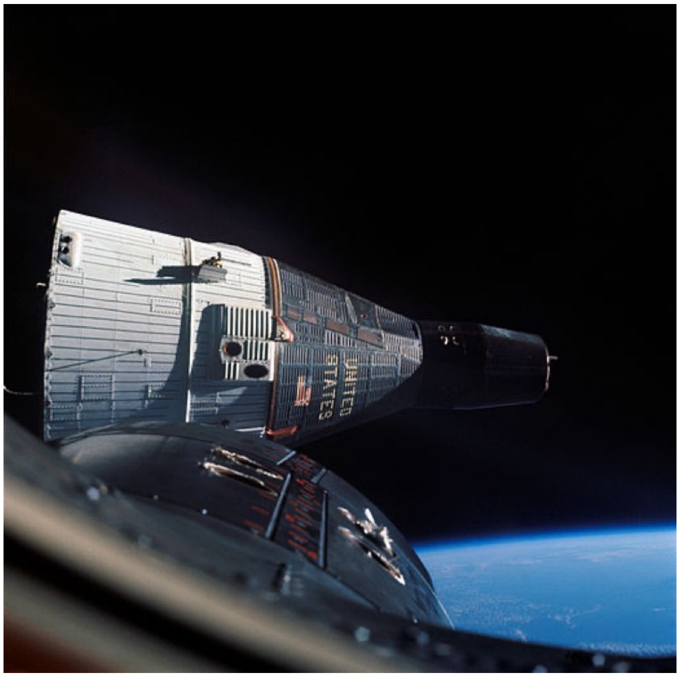

Fig. 2.6: View of Gemini VII from Gemini VI as they perform the first on-orbit proximity operations on December 15, 1965.

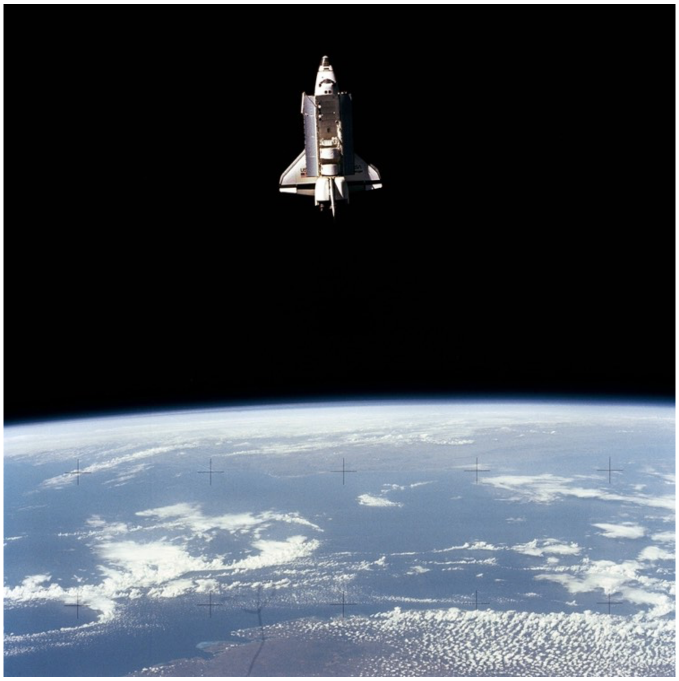

Fig. 2.7: View of Challenger from the SPAS-1 satellite during STS-7 in June 1983.

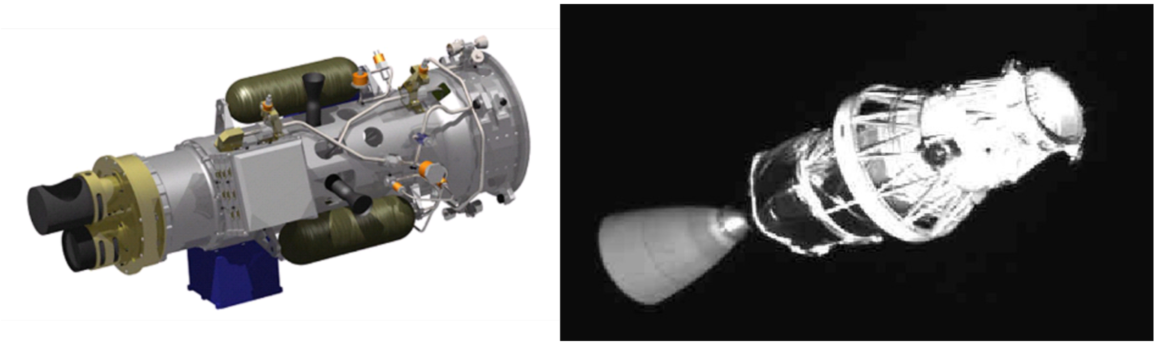

Fig. 2.8: XSS-10 satellite (left). Delta II launch vehicle upper stage images by XSS-10 (right).

NASA continued to explore RPO technologies with the launch of the unfortunately named DART mission in 2005 (just four days after the launch of XSS-11). DART, similar to XSS-10, was a 24 hour autonomous mission that ended prematurely when the vehicle expended more fuel than planned. Rumford [32] provides a overview of the DART mission. It was later discovered that DART had collided with its RSO when ground observers noted a change in the MUBLCOM satellite's orbit. NASA convened a Mishap Investigation Board to determine the cause of the collision and later released a summary of the findings [16]. The impacts of the mishap were felt widely through the RPO community as national leadership demanded more information on safety practices from both mission designers and operators.

The Defense Advanced Research and Projects Agency (DARPA) further pushed the RPO envelop in 2007 with the launch of the Orbital Express (OE) mission which included docking and berthing. Both Dennehy [33] and Weismuller [34] provide details of the the OE mission related to RPO and docking (RPOD) activities.

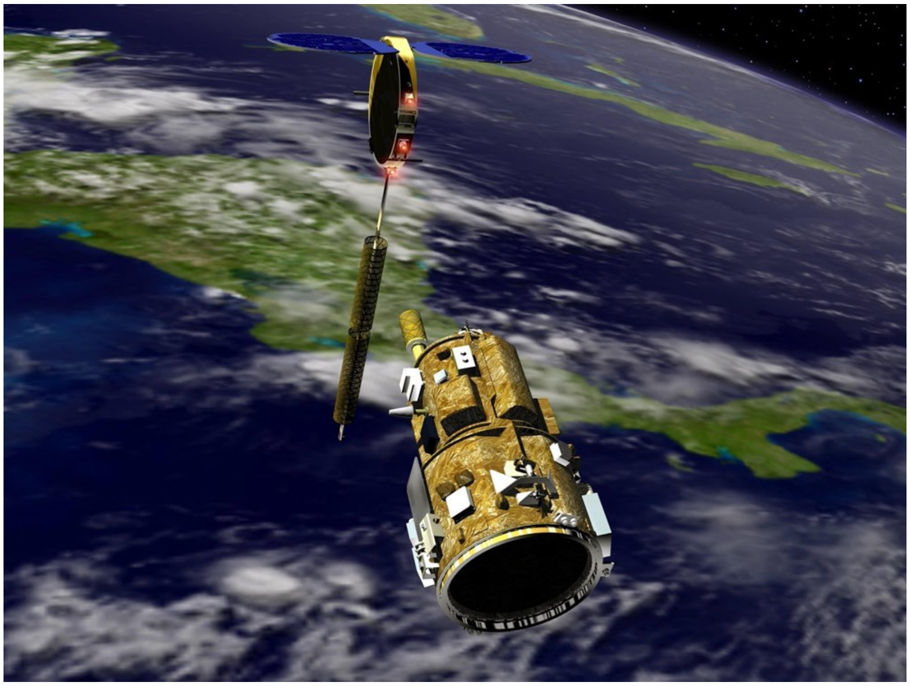

Fig. 2.9: Artist's depiction of the DART spacecraft as it approaches MUBLCOM.

AFRL continued its RPO work with its third and fourth generation systems, ANGELS (2014) [3] and Mycroft (2018) [4], both expanding guidance and navigation capabilities for vehicles in the GEO regime. Similar to the XSS series, details of these missions are not publicly available. It was reported in 2019 that Mycroft and it's launch host, EAGLE, were tasked to inspect the disabled S5 satellite [35] demonstrating that these technologies would require high level performance capable of supporting operational demands and not just research objectives. These high performance requirements are especially apparent in the Air Force Space Command's GEO Space Situational Awareness Program (GSSAP) that performs operational satellite inspection at GEO through a fleet of RPO-capable satellites [9].

Prior to the launch of the first two GSSAP satellites, Dr. Gene H. McCall and John H. Darrah, who had both previously served as the Chief Scientist for Air Force Space Command, published an article discussing the need for Space Situational Awareness and the expertise and care needed to perform missions such as GSSAP. They make a particularly strong argument for accurate and safe trajectory (i.e. maneuver) planning that serves, in part, as a motivation for the work of this dissertation research.

> *"Assuredly, the Air Force and its contractors well understand that the GSSAP vehicles must possess unprecedented accuracy in terms of propulsion and positioning. A collision will result in significant political and financial problems; moreover, it could produce debris capable of contaminating a large portion of the geosynchronous orbit. Certainly, maneuvering operations will generate very tense times at the satellite control center at Schriever AFB, Colorado."* [36]

The ANGELS mission was the author's first experience with on-orbit RPO flight operations where they served as both the Lead Mission Planner (responsible for all maneuver planning) and as the GNC Operations Lead (responsible for on-console GNC activities) for this mission. The author can personally attest to the "tense times" that can occur prior to maneuvering operations. It is this work (and a follow-on role as the Mycroft Principal Investigator) that informs and inspires much of the author's research in (and love for) RPO.

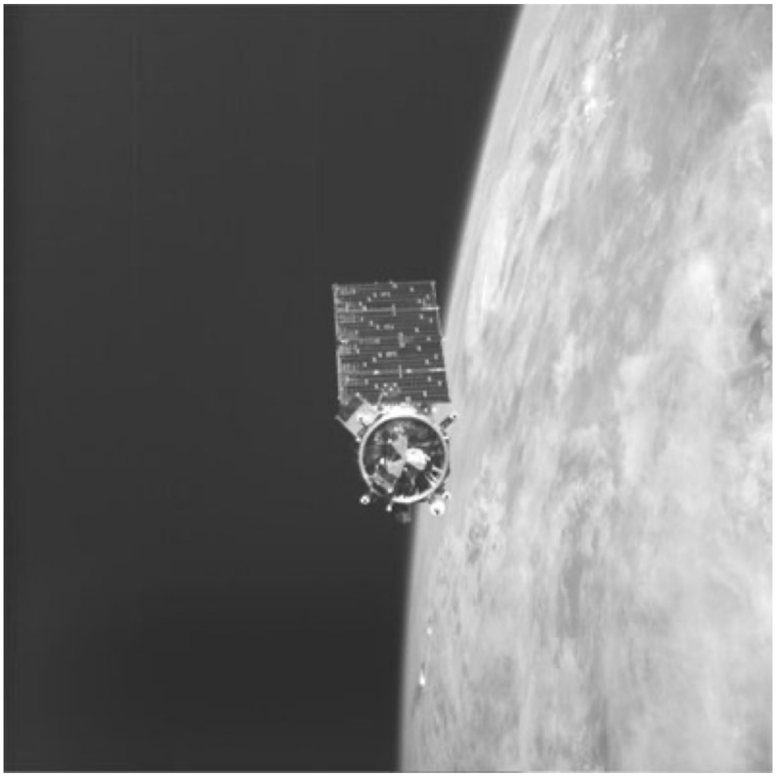

Fig. 2.10: Image of NEXTSat taken by the ASTRO satellite as part of the Orbital Express mission.

The ANGELS mission operations patch is shown in Figure 2.11.

Fig. 2.11: ANGELS operations crew patch. From the author's personal collection.

The United States is, of course, not the only country to perform RPO demonstrations. The European Space Agency has its Automated Transfer Vehicle (ATV) whose RPO capabilities are detailed by Pinard [37]. The Swedish Space Corporation (currently OHB-Sweden) launched a pair of satellites, named Mango and Tango, as part of the 2010 Prisma mission. Larsson provides flight results for the Prisma GNC experiments [38] in one of many publications based on the Prisma performance and collected data. Woffinden and Geller provide a history of relative navigation for proximity operations [18].

Fig. 2.12: Artist's depiction of the PRISMA spacecraft Mango and Tango.

2.4 Related Research

This section provides references for research related to the stated thesis.

2.4.1 Relative Motion Dynamic Models

Though not a major component of this research, it is important to note the advances in relative motion modeling that have been developed to support RPO missions. The most widely used method was developed by Clohessy and Wiltshire in 1960 and consists of a set of linear, time-invariant (LTI) equations describing the motion of a vehicle relative to an RSO in the LVLH frame (Section 2.1.2). The CW equations have limited accuracy due to their assumption of a circular reference orbit and simple two-body (i.e. unperturbed) dynamics.

Additional models have been developed to account for higher-order gravitational effects, atmospheric drag, and orbital eccentricity. Yamanaka and Ankerson [39] expanded relative motion modeling to include arbitrary elliptical reference orbits. Schweighart and Sedwick [40] developed a high fidelity linearized model for relative motion that incorporates

J_2, the dominant earth gravitational perturbation resulting from the earth's oblateness. In addition to the J_2 perturbation, Gaias et al. [41] developed a relative motion model that includes time-varying differential drag perturbations.

Sullivan et al. [42] provide a thorough survey and analysis of several relative motion models.

2.4.2 Deterministic Path Planning Techniques for Proximity Operations

Given a spacecraft's current relative state and a desired final relative state, optimization techniques can be implemented to determine the optimal path (or sequence of time-tagged maneuvers) that meet these boundary conditions. The process of determining this optimal path, commonly referred to as *guidance*, often relies on relative motion dynamics such as the efficient CW equations mentioned in Section 1.1. The majority of these guidance methods are deterministic, based on the relative motion dynamics and common performance limitations such as limited thrust capability.

Optimal RPO path planning is a popular research topic and has been explored from a number of directions. Most method look to optimize fuel consumption while addressing various constraints. Digiroloamo et al. [43] developed an offline path planner for performing autonomous inspections of the International Space Station (ISS). The method generates fuel optimal waypoints given an inspection vehicle's dynamics, thruster models and any obstacles that may exist in the field. Muñoz [44] develops an Adaptive Artificial Potential Function (AAPF) to perform rapid path planning for proximity operations that can be implemented in flight software. The AAPF method is an extension of standard Artificial Potential Functions (APF) that incorporates the relative motion dynamics to better solve for the optimal path. Bennet et al. [45] use waypoints to generate faster-than-natural circumnavigations of an RSO in a method that shows the general applicability of waypoints to RPO path planning.

Various optimization techniques have also been applied to the path planning problem. Lu [46] uses a lossless relaxation technique to generate a simplified problem in order to generate an optimal RPO solution. The relaxed problem is then discretized and solved using a second-order cone problem solver. Ortolano [47] uses a set of linear relative orbital elements (LROE) in order to apply a convex optimization technique to calculate optimal transfers. Weiss et al. [48, 49] apply Model Predictive Control (MPC) techniques to solve for impulsive velocity changes in three-dimensional relative space to perform general RPO transfers and then use nonlinear modeling to validate the results through Monte Carlo simulations.

Though solving a deterministic problem, several researchers have implemented stochastic optimization techniques to address the optimal path planning problem. Pontani [50, 51] implements a Particle Swarm Optimization (PSO) technique to solve for time-optimal, finite-thrust, multiple-maneuver rendezvous trajectories. Similarly, Li et al. [52] implement a Genetic Algorithm (GA) to perform path planning for multi-spacecraft formations.

2.4.3 Safety of Flight and Probability of Collision Techniques

When applying simplified techniques, it is not improbable or uncommon for the optimal path to generate unsafe trajectories. Safety of flight for spacecraft operations is a broad research field that includes approaches for both conjuncting objects and objects is proximity.

The most common approaches to flight safety address the concept of two vehicles in similar orbits that are predicted to pass close to one another based on ground-based orbit determination (OD). OD solutions provide state uncertainty covariances that lend themselves to probabilistic calculations. In these cases the risk of collision can be identified as either a probability of collision (P_c) or a related risk index based on the covariance data. Patera [53] developed a general method for calculating the probability of collision between such objects in orbit through several simplifying assumptions allowing the problem to be reduced to a one-dimensional path integral.

This approach of simplifying the three-dimensional problem to a one-dimensional problem is common to ground-based collision assessments. Chan [54–56] developed a series of collision probability calculations focused on improving accuracy while minimizing computational requirements for calculating P_c. His methods include general cross-sectional areas (as opposed to simple circular or spherical areas), and low and high-velocity encounters. Most of Chan's research was published in his book *Spacecraft Collision Probability* [57].

Not all conjunctions are conducive to the one-dimensional simplifying assumptions and are typically distinguished by low-velocity encounters. Patera [58] developed a method for calculating collision probabilities during these low-velocity encounters that involve nonlinear dynamics and a contour integration methodology that transforms the error covariance matrix into the primary reference frame where the errors are symmetric. Coppola [59] evaluates the effects of cross correlations between spacecraft when calculating collision probability. While Coppola's work was not focused on proximity operations, cross-correlation terms are common in the RPO scenarios.

Alfano [60] addresses nonlinear relative motion for spacecraft collision probability by breaking the problem into near linear sections and summing the collision probability across the segments. Carpenter [61] identifies issues with existing P_c methods for low-velocity encounters, such as might exist during proximity operations. Due to the complexity of the problem, Carpenter recommends using Monte Carlo trials with accurate process noise models to better estimate probability.

Limited work has been performed in determining P_c specifically for RPO or formation flying missions. Slater [62] developed a path planning technique for formation flying that uses probability-based collision calculations to assess risk of objects passing through the formation. Uncertainty covariances are propagated using linear theory to assess a "collision metric" over time. However, the research concludes that dynamics uncertainty and rapidly changing probabilities make calculating efficient ΔV solutions difficult. Gaylor and Barbee [63] developed several passive methodologies for safe proximity operations where the risk of collision is evaluated after optimal paths are generated. The collision risk is determined using a conservative (but computationally efficient) n-sigma approach that ensures the risk of collision is less than a specified value (without having to directly calculate the probability) by checking the number of standard deviations the vehicle path is away from the hard-body radius of the object. Sun [64] attempts to addresses the issue of collision risk during proximity operations by applying a monotonically increasing index related to the active vehicle's position error. Frey [65] implements a graph search to determine feasible solutions for relative motion, including non-periodic orbits, that satisfy collision avoidance risks. His approach also focuses on avoiding numerical errors due to covariance dilution. Philips [66] makes a significant contribution to the proximity operations collision assessment in his work by generating and evaluating several collision probability metrics, including instantaneous P_c, total P_c, and Monte Carlo techniques.

2.4.4 Constrained Path Planning Techniques for Proximity Operations

Given the importance of flight safety, extensive research has focused on generating constrained optimal paths that prevent potential collisions. Perhaps the two most significant works (in relation to this proposed research) include the PhD dissertations of Breger [67] (MIT, 2007) and Prince [68] (AFIT, 2018).

Breger develops a linear programming (LP) guidance method that minimizes fuel while also enforcing both active and passive safety constraints. In Breger's other published works include [69, 70] he and How develop a general convex formulation for fuel-optimal transfers that guarantees collision avoidance and performs trade-offs between active and passive approaches to safety. These results show the benefit of addressing safety over traditional path planning approaches.

Prince [68] takes a more operational approach to the problem and generates solutions for both the fuel and time-optimal transfers including for mission specific trajectories such as the so-called teardrop orbit. The proposed solution accounts for both collision safety and evaluates sun and moon conditions for mission planning considerations. In [71], Prince generates a multi-stage optimization technique for generating fuel optimal proximity maneuvers with either hard or soft lighting constraints where a low fidelity model is used with particle swarm optimization (PSO) to feed a high fidelity model implementing psuedospectral optimization techniques.

The inclusion of constraints similar to those used by How or Prince are not necessarily unique. Richards [72] generates a fuel optimal guidance scheme that accounts for collision avoidance constraints using a mixed-integer linear programming (LP) techniques. Roger [73] developed a path planning tool for planning an ISS separation, inspection, and docking mission where safety is achieved through the use of safety ellipses and constrained velocity profiles. Schlanbusch and Oland [74] developed three different approaches to spacecraft formation reconfiguration with collision avoidance constraints based on Null-Space Based behavioral control. The first approach uses constant sized spheres with repulsive gains. The second approach uses variable state dependent gains to adjust the repulsive forces required to avoid collisions. And the third approach uses variable sized spheres to initiate early avoidance while allowing easier passage at safe distances.

Sauter and Palmer [75] focuses on a semi-analytic approach that can be implemented in flight software that generates a fuel-optimal, collision-free path. Collision avoidance is achieved through sequential optimization where constraints are examined in each iteration. Holzinger et al. [76] develop a receding horizon control (RHC) approach to path planning with passive safety requirements. Results show that even small amounts of cross-track motion significantly reduce the risk of collision. Conversely, navigation uncertainty and process noise are shown to be significant drivers in ΔV usage for passive safety.

Limited research has focused on maintaining specified relative locations. Irvin [77, 78] explores optimal solutions for maintaining an arbitrary hover locations relative to an RSO. Such hover orbits are beneficial for spacecraft inspection from potential safe hold locations.

Not all constrained methods are focused on fuel-optimization. Dannemiller [79] developed a multi-maneuver version of CW targeting in order to provide a simplified method for meeting real-world operational constraints such as lighting and ground visibility.

Franquiz [80] merges trajectory optimization with angles-only range observability to generate two-burn transfers that also produce improved AON performance. Navigation is not modeled directly, but accounted for in a range observability criteria. He implements a two step approach that uses a grid search followed by a nonlinear program solver to determine the optimal solution. The approach also includes keep-out-zone constraints on the trajectory, eclipse lighting, and field-of-view considerations.

Huang et al. [81] implement a PSO to solve the nonlinear path planning problem for the formation reconfiguration problem with collision avoidance. Collision avoidance is enforced by spot checking regions where collisions are most likely to occur.

It should be noted the majority of the above methods implement a deterministic view of collision where safety is achieved by maintaining a minimum distance between the maneuvering vehicle and the RSO. Relative navigation uncertainties, if addressed at all, are based on performance indices that indirectly model navigation performance rather than modeling navigation performance directly. None of these methods consider trajectory dispersion in their formulations.

2.4.5 Relative Navigation Techniques for Proximity Operations

Critical to proximity operations is the spacecraft's ability to estimate its relative position and velocity relative to the RSO. Relative navigation may consist of active sensors, passive optics, or even data shared between spacecraft. RPO spacecraft may also implement multiple relative navigation schemes and sensors in order to operate over varied ranges from the RSO. Measurements are most commonly processed by an onboard Kalman filter.

Angles-only navigation is a popular approach for its simplicity where a camera is used to generate line-of-sight measurements to the RSO. Angles-only is, however, noted for its inability to accurately estimate range. Woffinden and Geller [11] provide a widely cited treatise of AON for proximity operations. They also provide metrics for assessing and generating range observability through the concept of observability maneuvers [17]. Additional AON analysis was performed by Chari [82], Schmidt [83] and Tombasco [84].

Numerous other navigation techniques generate full relative position data. These methods are typically characterized as either cooperative, where data or *a priori* information is shared between the vehicles, or non-cooperative where the active vehicle performs its relative navigation without any help from the RSO. Allen [12] evaluates a lidar system for the terminal phase of rendezvous and docking in one example of a non-cooperative system.

Lidar-based systems are the most common non-cooperative sensor used to generate full relative position data for relative navigation. Numerous techniques also exist for cooperatively generating relative position data. Tweddle [85] evaluates a computer vision technique where a small fiducial on the RSO is used to determine relative data. A similar capability is shown by Howard and Book [86]. Junkins et al. [87] proposed a novel technique using position sensing diode photodetectors that can provide either line-of-sight or relative position data. Petit et al. [88] demonstrated vision-based navigation using optics with satellite mockups, demonstrating that acceptable performance can be achieved. Others developing advanced techniques can be found in [89–91].

Another popular relative navigation technique, particularly for formation flying missions, is to cooperatively share data between the vehicles involved. Montenbruck [92] shows the use of shared GPS signals for relative navigation. The Russian Soyuz and Progress vehicles use active radio frequency (RF) methods to determine relative position data for rendezvous and docking [18, 93].

Relative navigation for proximity operations is almost exclusively processed using a Kalman filter or one of its variations. The majority of the papers cited above include details of their respective navigation filters. Details on optimal state estimation, including the Kalman filter and various nonlinear filters, can be found in the books by Maybeck [94], Stengel [95], Zarchan [96], Crassidis [97], Simon [98], and Ristic [99]. The book *Statistical Orbit Determination* by Tapley, Schutz, and Born [100] even provides detailed implementations specific to orbital mechanics.

2.4.6 Linear Covariance Analysis Applications

The final component to this research is a linear covariance analysis. At its core, a linear covariance (LinCov) analysis is a sensitivity study that seeks to analyze the statistical performance of a GNC system, about a nominal reference, in a single run. Originally developed by Maybeck [94] for linear systems, the method is capable of analyzing a closed-loop GNC system with a truth model, a navigation scheme and sensors, and a guidance system that drives actuators feeding back into the truth model (see Chapter 3).

Geller [101] expanded on the simple linear models to generate a closed-loop linear covariance technique that's applicable to nonlinear systems where linearization adequately models the system performance. Geller's specific application was for spacecraft rendezvous scenarios. This work has continually been expanded to include LinCov for powered lunar descent [102], powered ascent [103], and atmospheric entry [104].

While these works have focused on aerospace applications, a general tutorial for the approach was published by Christensen and Geller [105] showing the applicability of the technique to complex, closed-loop systems. Further expanding this generalized approach, Christensen, Droge and Leishman [106] developed a closed-loop linear covariance framework for path planning through uncertain obstacle fields. In this work, the primary vehicle's probability of collision is determined for static obstacles of know size and shape but with uncertain locations. Collision probabilities are calculated directly from the true Gaussian probability density function for the primary vehicle and multiple obstacles across the planning field assuming independent Gaussian random variables. This work also provides references to multiple approaches for determining collision probabilities using chance-constraints for motion planning.

Jin and Geller [107] first proposed the merging of a closed-loop linear covariance analysis into an optimization scheme that seeks to minimize statistical fuel consumption (i.e. the maximum likely fuel consumption). Their work modeled high accuracy relative position measurements for a multi-maneuver CW targeting guidance scheme that also required a maximum position dispersion at the final time. Stastny and Geller [108] expanded on the work of Jin by including AON and lighting constraints where the measurements are inhibited whenever the sun is too close to the camera boresight. The approach includes dynamic modeling error, navigation errors, and maneuver execution errors. This work showed that the approach can be used to achieve arbitrarily small final dispersions even when accounting for the range observability limitations of AON (and ignoring some noise sources). Results also showed that achieving small final dispersion may cause unsafe trajectories.

2.5 Conclusion

This chapter derived from first principles the equations necessary to understand spacecraft rendezvous and proximity operations. Through a few simplifying assumptions, the CW equations provide trajectory planners an accurate and manageable method for modeling relative motion between two spacecraft and for calculating trajectory shaping maneuvers critical to on-orbit operations. From this launching point, researchers and practicing engineers have been able to expand the knowledge base for motion modeling and trajectory planning. Extensive research has also sought to infuse elements of trajectory safety to identify and (potentially) prevent collisions. The fusion of these safety elements with trajectory optimization techniques is an expanding field supporting trajectory planners for the ever growing number of RPO missions. This research seeks to add to this field by fusing novel safety metrics with closed-loop linear covariance techniques to provide safe and optimal trajectories.

CHAPTER 3

STATISTICAL ANALYSIS OF CLOSED-LOOP GNC SYSTEMS

The rendezvous and proximity operations models presented in Chapter 2 adequately represent the real-world dynamics of an RPO mission, however, these are still only the deterministic dynamics. The deterministic dynamics are the results of complex dynamics including higher-order gravitational effects, gravitational perturbations from other celestial objects, and even atmospheric drag forces. An accurate understanding of precise relative motion requires high-fidelity models that include all of these effects. In addition to high-fidelity astrodynamics, RPO missions are characterized by maneuvers that alter the spacecraft's orbit and relative navigation algorithms that feed guidance algorithms used to calculate those maneuvers. This sort of system is referred to as a closed-loop GNC system since the spacecraft's flight computer is connected to the real world environment through sensor inputs and actuator outputs as depicted in Figure 3.1.

Due to the stochastic nature of the real world (including sensor and actuator hardware) it is critical to analyze and tune the flight computer algorithms to achieve reliable and stable performance when placed into real world operations. The statistics from such an analysis can help designers select appropriate components, identify necessary modifications or improvements, and verify system performance against requirements.

There are two common methods for analyzing closed-loop GNC systems. The first method is known as a Monte Carlo analysis. Named after the Monte Carlo Casino in Monaco, this method relies on random sampling of the system's random processes to generate numerical results. While Monte Carlo results are typically quite accurate (to the level of the system models), they generally require a significant number of runs to generate meaningful statistics. The second methods is a linear covariance (LinCov) analysis. This form of analysis relies on statistical properties rather than random samples and can gener-

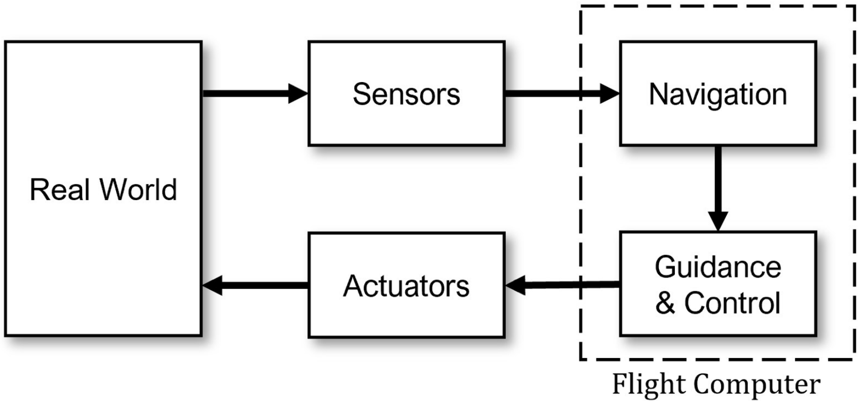

Fig. 3.1: Closed-loop GNC system in the real world.

ate meaningful statistics in a single run. However, the results may be of limited accuracy if the system is highly nonlinear and the linear approximations are inadequate. Sections 3.1 and 3.2 will develop the equations used for the nonlinear Monte Carlo and linear covariance analyses, respectively. Finally, Section 3.3 will analyze a simplified closed-loop GNC system using both the Monte Carlo and linear covariance methods to demonstrate the effectiveness of LinCov for statistical analysis.

For background on the stochastic processes implemented here, the reader is directed to the seminal work by Maybeck [94].

3.1 Nonlinear Monte Carlo Analysis

This section develops the general equations for a nonlinear Monte Carlo analysis as depicted in Figure 3.2. For this approach, the real world will be replaced by truth dynamics (a high fidelity model of the real world) driven by random process noise that represents variations in the real world (or equivalently, limitations in the deterministic truth models to accurately model the real world). Similarly, sensor and actuator models replace physical hardware and are again driven by stochastic processes to represent true measurements and actuation interactions with the real world. Finally, the flight computer GNC algorithms are developed.

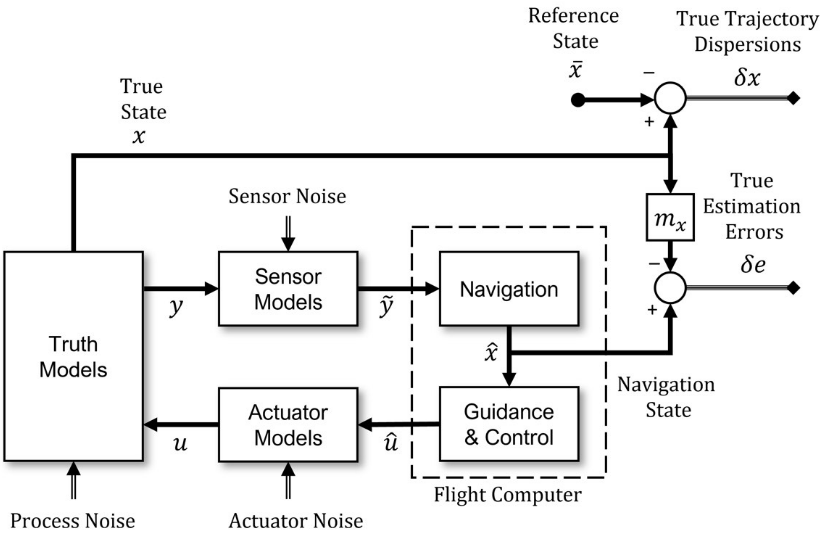

Fig. 3.2: Monte Carlo analysis block diagram for a closed-loop GNC system.

3.1.1 Nonlinear Truth Dynamics

The dynamics of the true state vector are given *formally* by a nonlinear vector differential equation:

$$\dot{\mathrm{x}} = f(\mathrm{x}, \mathrm{u}) + \mathrm{W}w \tag{3.1}$$

where $\mathrm{x} \in \mathbb{R}^n$ is the true state vector, $u \in \mathbb{R}^{n_u}$ is the true control vector from the actuator, and $w \in \mathbb{R}^{n_w}$ is the process noise that is used to account for errors in the dynamic models. W is an $n \times n_w$ matrix that maps the process noise into the state vector channels. It is important to note that this form cannot be used rigorously, since its solution cannot be generated. A more rigorous approach to the stochastic differential equation would be:

$$\mathrm{dx} = f(\mathrm{x}, \mathrm{u})dt + \mathrm{Wd}\beta(t) \tag{3.2}$$

where β is a Brownian motion process of diffusion $Q(t)$. Equation (3.1), however, is often used heuristically in engineering applications to describe a stochastic system model.

The process noise is an uncorrelated, zero-mean Gaussian process:

$$E\left[w\right] = 0_{n_w \times 1} \tag{3.3}$$

$$E\left[w(t)w^T(t')\right] = \left[Q_w\delta(t-t')\right]_{n_w \times n_w} \tag{3.4}$$

where $E[\cdot]$ is the expectation operator, Q_w is the process noise strength, and $\delta(t)$ is the Dirac delta. The Dirac delta can be viewed from a (nonrigourous) engineering approach as a function which is zero everywhere except at the origin, where it is infinite:

$$\delta(t) = \begin{cases} \infty, & (t = 0) \\ 0, & (\text{otherwise}) \end{cases} \tag{3.5}$$

The reader is directed to the work by Greenberg for a more rigorous view of the Dirac delta [109].

3.1.2 Nonlinear Truth Measurement Model

Discrete nonlinear truth measurements, as produced by sensor hardware, are generated from the true state using the following function:

$$\tilde{\mathrm{y}}_j = h(\mathrm{x}_j) + \nu_j \tag{3.6}$$

where $\tilde{y} \in \mathbb{R}^{n_{\tilde{y}}}$ is the true sensor measurement, and $\nu \in \mathbb{R}^{n_{\tilde{y}}}$ is the measurement noise and is a zero-mean Gaussian process:

$$E\left[\nu_j\right] = 0_{n_{\tilde{y}} \times 1} \tag{3.7}$$

$$E\left[\nu_j \nu_{j'}^T\right] = \left[R_\nu \delta_{jj'}\right]_{n_{\tilde{y}} \times n_{\tilde{y}}} \tag{3.8}$$

where R_ν is the measurement covariance and δ_{ij} is the Kronecker delta.

$$\delta_{ij} = \begin{cases} 1, & (i = j) \\ 0, & (\text{otherwise}) \end{cases} \tag{3.9}$$

3.1.3 Nonlinear Navigation Model

Sensor measurements are processed by an onboard navigation system in order to estimate the true flight dynamics. Embedded processors on these spacecraft flight avionics, however, are regularly years behind their ground processing counterparts due to radiation, mechanical, thermal and other environmental concerns for space avionics. This limitation frequently requires mission designers to develop simplified dynamics models for implementation in an onboard navigation filter. The nonlinear navigation dynamics model is given as:

$$\dot{\hat{\mathrm{x}}} = \hat{f}(\hat{\mathrm{x}}, \hat{\mathrm{u}}) \tag{3.10}$$

where $\hat{\mathrm{x}} \in \mathbb{R}^{\hat{n}}$ is the estimated state vector and will likely contain fewer elements than the truth state vector, x, and $\hat{\mathrm{u}} \in \mathbb{R}^{n_{\hat{u}}}$ is the commanded control vector as calculated by the guidance and control algorithms. It is important to note that all elements of the navigation model use the hat symbol ($\hat{}$) to distinguish them from the true dynamics in Equation (3.1). Additionally, the navigation dynamics do not include process noise in the state propagation equations as it is meant to represent the mean or best estimate of the true state. The navigation filter, however, is based on a design model that does include process noise terms whose strength is used to tune the navigation filter. The design model (dm) dynamics are given as:

$$\dot{\hat{\mathrm{x}}}^{\mathrm{dm}} = \hat{f}(\hat{\mathrm{x}}^{\mathrm{dm}}, \hat{\mathrm{u}}) + \hat{\mathrm{W}} w^{\mathrm{dm}} \tag{3.11}$$

where $w^{\mathrm{dm}} \in \mathbb{R}^{n_{w^{\mathrm{dm}}}}$ is the uncorrelated, zero-mean Gaussian design model process noise and $\hat{W}$ is an $\hat{n} \times n_{w^{dm}}$ matrix that maps the process noise into the state vector.

$$E\left[w^{\mathrm{dm}}\right] = 0_{n_{w^{dm}} \times 1} \tag{3.12}$$

$$E\left[\left(w^{\mathrm{dm}}(t)\right)\left(w^{\mathrm{dm}}(t')\right)^{T}\right] = \left[\hat{Q}_{w^{dm}}\delta(t - t')\right]_{n_{w^{dm}} \times n_{w^{dm}}} \tag{3.13}$$

For simplicity, we will drop the *dm* indicator from the remainder of the derivation unless required for clarity.

The navigation filter state covariance matrix, $\hat{P}$, is propagated according to the continuous time model:

$$\dot{\hat{P}}(t) = \hat{F}_{\hat{\mathrm{x}}}\hat{P}(t) + \hat{P}(t)\hat{F}_{\hat{\mathrm{x}}}^{T} + \hat{G}\hat{Q}_{w}\hat{G}^{T} \tag{3.14}$$

where $\hat{F}_{\hat{\mathrm{x}}}$ is the Jacobian of the nonlinear dynamics function, $\hat{f}$, evaluated at time t.

$$\hat{F}_{\hat{\mathrm{x}}} = \left.\frac{\partial \hat{f}(\hat{\mathrm{x}}, \hat{\mathrm{u}})}{\partial \hat{\mathrm{x}}}\right|_{\hat{\mathrm{x}}(t)} \tag{3.15}$$

The state covariance matrix can also be propagated according to a discrete-time model:

$$\hat{P}(t_{i+1}) = \hat{\Phi}(t_{i+1}, t_i)\hat{P}(t_i)\hat{\Phi}^{T}(t_{i+1}, t_i) + \hat{Q}_{d,w} \tag{3.16}$$

where $\hat{\Phi}(t_{i+1}, t_i)$ is the state transition matrix from t_i to t_{i+1} and $\hat{Q}_{d,w}$ is the discrete process noise covariance for that time interval.

$$\hat{\Phi}(t_{i+1}, t_i) = e^{\hat{F}(t_{i+1}-t_i)} = e^{\hat{F}\Delta t} \tag{3.17}$$

$$\hat{Q}_{d,w} = \int_{t_i}^{t_{i+1}} \hat{\Phi}(t_{i+1}, \tau)\hat{W}\hat{Q}_{\nu}\hat{W}^{T}\hat{\Phi}^{T}(t_{i+1}, \tau)d\tau \tag{3.18}$$

Discrete measurements are predicted by the navigation filter through the design models nonlinear measurement equation:

$$\hat{\mathrm{y}}_j = \hat{h}(\hat{\mathrm{x}}_j) \tag{3.19}$$

where $\hat{\mathrm{y}}_j \in \mathbb{R}^{n_{\hat{y}}}$ is the estimated measurement input at time t_j.

The measurement noise covariance, derived from the design model, has the following characteristics:

$$E\left[\nu_j^{dm}\right] = 0_{n_{\hat{y}}\times 1} \tag{3.20}$$

$$E\left[\nu_j^{dm}\nu_{j'}^{dm^T}\right] = \left[\hat{R}_{\nu}\delta_{jj'}\right]_{n_{\hat{y}}\times n_{\hat{y}}} \tag{3.21}$$

The modeled measurement noise covariance matrix, $\hat{R}_{\nu}$, has the same dimensions as the true measurement noise covariance, R_{ν}, (assuming $n_{\tilde{y}} = n_{\hat{y}}$) but the two do not necessarily take on the same values due to potential simplification of the nonlinear measurement model, $\hat{h}$ or due to differences between the a priori sensor spec and actual sensor performance.

The actual and predicted measurements can now be used to update the filter's estimated state vector:

$$\hat{\mathrm{x}}_j^+ = \hat{\mathrm{x}}_j^- + \hat{\mathrm{K}}_j \Big[\tilde{\mathrm{y}}_j - \hat{\mathrm{y}}_j\Big] \tag{3.22}$$

where $\hat{\mathrm{x}}_j^-$ and $\hat{\mathrm{x}}_j^+$ are the estimated states at time t_j immediately before and after the measurement processing, respectively, and $\hat{K}$ is the Kalman gain. The Kalman gain is determined by:

$$\hat{K}_j = \hat{P}_j^- \hat{H}_j^T \left[\hat{H}_j \hat{P}_j^- \hat{H}_j^T + \hat{R}_\nu\right]^{-1} \tag{3.23}$$

where $\hat{H}$ is the Jacobian of the nonlinear navigation measurement model, $\hat{h}$, evaluated at time t_j:

$$\hat{H}_j = \frac{\partial \hat{h}(\hat{\mathrm{x}})}{\partial \hat{\mathrm{x}}}\bigg|_{\hat{\mathrm{x}}_j^-} \tag{3.24}$$

and $\hat{P}_j^-$ is the state covariance matrix prior to processing the measurement.

The state covariance is also updated using the Joseph formulation of the state covariance update equation [94]:

$$\hat{P}_j^+ = \left[I - \hat{K}_j \hat{H}_j\right] \hat{P}_j^- \left[I - \hat{K}_j \hat{H}_j\right]^T + \hat{K}_j \hat{R}_\nu \hat{K}_j^T \tag{3.25}$$

where $\hat{P}_j^+$ is the state covariance matrix at time t_j after the measurement is updated, and I is an $\hat{n} \times \hat{n}$ identity matrix.

Since the true and estimated state vectors do not necessary contain the same states or number of elements, a mapping between the two states exists that can be used to determine the true values of the navigation states.

$$\hat{\mathrm{x}}_{\mathrm{true}} = m_{\mathrm{x}}(\mathrm{x}) \tag{3.26}$$

where $\mathrm{x}_{\mathrm{nav}}$ is the true navigation state. If the true and navigation states contain the same elements then m_{x} simplifies to an identity matrix.

3.1.4 Nonlinear Guidance Model

Using the estimated states, the flight computer executes a nonlinear guidance algorithm to determine the desired control vector:

$$\hat{\mathrm{u}}_k = \hat{g}(\hat{\mathrm{x}}_k) \tag{3.27}$$

where $\hat{\mathrm{u}} \in \mathbb{R}^{n_{\hat{u}}}$ is the discrete actuator command and $\hat{g}$ the nonlinear guidance algorithm. As with sensor measurement inputs, actuator commands can be modeled as continuous outputs. Only the discrete case is considered here.

For the case of a discrete control input (e.g. impulsive maneuver model), the control vector can be used to correct the navigation filter's estimated state vector.

$$\hat{\mathrm{x}}_k^{+c} = \hat{\mathrm{x}}_k^{-c} + \hat{b}(\hat{\mathrm{u}}_k) = \hat{\mathrm{x}}_k^{-c} + \hat{b}\left[\hat{g}(\hat{\mathrm{x}}_k^{-c})\right] \tag{3.28}$$

$$P_k^{+c} = P_k^{-c} + \hat{B}\hat{S}_{\Delta w}\hat{B}^T \tag{3.29}$$

where the superscripts $-c$ and $+c$ indicate the state or covariance before and after the control correction, respectively. $\hat{b}$ is the mapping of the control vector into the navigation states and $\hat{B}$ is its Jacobian:

$$\hat{B} = \left.\frac{\partial \hat{b}(\hat{x})}{\partial \hat{\mathrm{u}}}\right|_{\hat{\mathrm{x}}_{\mathrm{k}}^{-\mathrm{c}}} \tag{3.30}$$

The term $\hat{S}_{\Delta w}$ is the discrete actuator noise covariance and is derived from the navigation design model:

$$\hat{\mathrm{u}}_k^{\mathrm{dm}} = \hat{g}(\hat{\mathrm{x}}_k^{\mathrm{dm}}) + \eta_k^{dm} \tag{3.31}$$

$$E\left[\eta_k^{dm}\right] = 0_{n_{\hat{u}} \times 1} \tag{3.32}$$

$$E\left[\eta_k^{dm}\eta_{k'}^{dm^T}\right] = \left[\hat{S}_{\Delta w}\delta_{kk'}\right]_{n_{\hat{u}} \times n_{\hat{u}}} \tag{3.33}$$

3.1.5 Nonlinear Actuator Model

The final piece of the model is the nonlinear actuator model that converts the desired control vector, $\hat{\mathrm{u}}$, into true actuator output, u. The truth model is analogous to Equation (3.31):

$$\mathrm{u}_k = \hat{\mathrm{u}}_k + \eta_k \tag{3.34}$$

where u is the true actuator output feeding the true dynamics in Equation (3.1) and η is the true actuator noise given as an uncorrelated, zero-mean Gaussian process:

$$E\left[\eta_k\right] = 0_{n_{\mathrm{u}}\times 1} \tag{3.35}$$

$$E\left[\eta_k \eta_{k'}^T\right] = \left[S_\eta \delta_{kk'}\right]_{n_{\mathrm{u}}\times n_{\mathrm{u}}} \tag{3.36}$$

where S_η is the discrete control covariance matrix.

We again note that the true actuator noise covariance in Equation (3.36) does not necessarily equal the design model actuator noise covariance in Equation (3.33).

3.1.6 Monte Carlo Performance Statistics

With the full nonlinear closed-loop simulation equations in place, it is now possible to analyze the performance of the GNC system by executing multiple simulation runs to generate statistical data. Each run of the simulation is driven by randomly generated samples of the w, ν_k, and η_j processes and random initial conditions.

Reference State

Consider n runs of the Monte Carlo simulation such that $\mathrm{x}^i(t)$, $i = 1, 2, ..., n$ are the individual truth states. At each time step, the mean of the states is calculated as:

$$\bar{\mathrm{x}}(t) = E[\mathrm{x}(t)] \approx \frac{1}{n}\sum_{i=1}^{n} \mathrm{x}^i(t) \tag{3.37}$$

where x^i is a single run of true state within the nonlinear simulation.

The average true state vector, $\bar{\mathrm{x}}(t)$, represents the nominal trajectory and serves as the reference against which the other runs are evaluated. In some cases, the reference trajectory may also be generated by performing a single run of the closed-loop simulation where all of the simulation errors are set to zero; $w = \nu = \eta = 0$. This approach is not always valid for nonlinear systems.

True State Dispersion

The true state dispersion, $\delta \mathrm{x}$, represent the difference between the true state and the reference state and demonstrates the overall ability of the closed-loop system to track the desired reference state.

$$\delta \mathrm{x}^i(t) = \mathrm{x}^i(t) - \bar{\mathrm{x}}(t) \tag{3.38}$$

The individual simulation runs can also be used determine the true state dispersion covariance, D_{true}, which provides the variance and correlation statistics of the true state dispersions. D_{true} is a key performance metric of the system analysis.

$$D_{\text{true}}(t) = E\Big[\delta \mathrm{x}\ \delta \mathrm{x}^T\Big] \approx \frac{1}{n-1} \sum_{i=1}^{n} \Big(\mathrm{x}^i(t) - \bar{\mathrm{x}}(t)\Big)\Big(\mathrm{x}^i(t) - \bar{\mathrm{x}}(t)\Big)^T \tag{3.39}$$

True Navigation Error

Similarly, the true navigation error, $\delta \mathrm{e}$, represents the difference between the individual true and estimated states, providing an indication of how well the navigation filter estimates the true state values.

$$\delta \mathrm{e}^i(t) = \hat{\mathrm{x}}^i(t) - m_x \mathrm{x}^i(t) \tag{3.40}$$

where $\hat{\mathrm{x}}^i$ is a single run of the estimated state within the navigation portion of the nonlinear simulation.

The true navigation error covariance, P_{nav}, at each time step is calculated from the combined runs:

$$P_{\text{nav}}(t) = E\Big[\delta \mathrm{e}\ \delta \mathrm{e}^T\Big] \approx \frac{1}{n-1} \sum_{i=1}^{n} \Big(\hat{\mathrm{x}}^i(t) - m_x\big(\mathrm{x}(t)\big)\Big)\Big(\hat{\mathrm{x}}^i(t) - m_x\big(\mathrm{x}(t)\big)\Big)^T \tag{3.41}$$

Control Errors

Another key metric in the Monte Carlo analysis is the control error resulting from both the navigation errors and the actuator errors. Control error statistics are used to properly size actuators or estimate consumables (such as fuel) to ensure mission success. The discrete nominal controls values can be calculated as:

$$\bar{\mathrm{u}}_k = E[\mathrm{u}_k] \approx \frac{1}{n}\sum_{i=1}^{n} \mathrm{u}_k^i(t) \tag{3.42}$$

The control dispersions and dispersions covariance for each discrete control are then calculated as:

$$\delta\mathrm{u}_k^i = \mathrm{u}_k^i - \bar{\mathrm{u}}_k \tag{3.43}$$

$$D_{\mathrm{ctrl},k} = E\Big[\delta\mathrm{u}_k \ \delta\mathrm{u}_k^T\Big] \approx \frac{1}{n-1}\sum_{i=1}^{n}\Big(\mathrm{u}_k^i - \bar{\mathrm{u}}_k\Big)\Big(\mathrm{u}_k^i - \bar{\mathrm{u}}_k\Big)^T \tag{3.44}$$

3.1.7 Confidence Intervals

The accuracy of the calculated covariance matrices in Equations (3.39), (3.41), and (3.44) are determined by the number of simulation runs, n, used to generate the statistics. Small data sets will produce values with a low confidence of accuracy. Conversely, increased confidence comes from extensive data sets which accurately represent the overall statistics. For many high-fidelity closed-loop simulations, however, increased simulation runs comes at the expense of the total required run time. Confidence intervals provide a mathematical framework for assessing the reliability of the calculated values [110]. Inversely, confidence intervals can also be used to determine the number of simulation runs required to achieve a desired confidence level.

For a statistical distribution, the maximum likelihood estimates of μ and σ^2 for a normal distribution are [111]:

$$\hat{\mu} = \bar{X} \tag{3.45}$$

$$\hat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n}\big(X_i - \bar{X}\big)^2 \tag{3.46}$$

The probability that the true mean, μ, lies in the interval $\bar{X} \pm \frac{S}{\sqrt{n}}\mathcal{T}_{n-1}(\alpha/2)$ is $1-\alpha$:

$$P\left(\bar{X} - \frac{S}{\sqrt{n}}\mathcal{T}_{n-1}(\alpha/2) \leq \mu \leq \bar{X} + \frac{S}{\sqrt{n}}\mathcal{T}_{n-1}(\alpha/2)\right) = 1-\alpha \tag{3.47}$$

where:

$$\frac{\sqrt{n}\left(\bar{X}-\mu\right)}{S} \sim \mathcal{T}_{n-1} \tag{3.48}$$

$$S^2 = \frac{1}{n-1}\sum_{i=1}^{n}\left(X_i - \bar{X}\right)^2 \tag{3.49}$$

and $\mathcal{T}_{n-1}$ is the t-distribution with $n-1$ degrees of freedom. The $\sim$ symbol used here indicates "is distributed as". $\mathcal{T}_{n-1}(\alpha/2)$ is the point beyond which the t-distribution has a probability of $\alpha/2$.

Similarly, the probability that the true variance, σ^2, lies in the calculated interval is:

$$P\left(\frac{n\hat{\sigma}^2}{\chi^2_{n-1}(\alpha/2)} \leq \sigma^2 \leq \frac{n\hat{\sigma}^2}{\chi^2_{n-1}(1-\alpha/2)}\right) = 1-\alpha \tag{3.50}$$

where:

$$\frac{n\hat{\sigma}^2}{\sigma^2} \sim \chi^2_{n-1} \tag{3.51}$$

and χ^2_{n-1} is the chi-squared distribution with $n-1$ degrees of freedom. It should be noted that, unlike Equation (3.47), the interval in Equation (3.50) is not symmetric about σ^2.

The number of Monte Carlo simulation runs required to achieve a desired confidence interval is shown in Figure 3.3. To have a 95% confidence that the calculated standard deviation is within 10% of the true standard deviation, a total of 850 Monte Carlo runs must be completed. 600, 1500, and 2450 runs must be completed for a confidence of 90%, 99%, and 99.9%, respectively [105].

3.2 Linear Covariance Analysis

This section develops the linear covariance models used to evaluate closed-loop GNC systems. The LinCov analysis tool can generate the the same statistical data as the Monte Carlo simulation but in a single run.

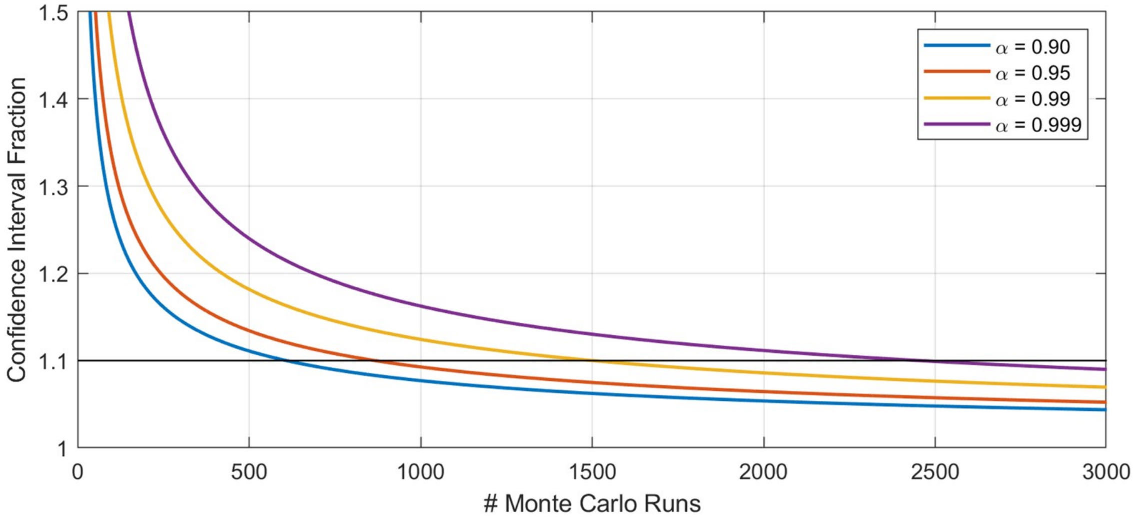

Fig. 3.3: Number of simulation runs required to achieve a desired confidence interval fraction for different probabilities.

3.2.1 Linear Augmented State and State Dynamics

To generate a LinCov model we will first define and augmented state vector containing the true and navigation filter error states:

$$\mathrm{X} = \begin{bmatrix} \mathrm{x} - \bar{\mathrm{x}} \\ \hat{\mathrm{x}} - m_x(\bar{\mathrm{x}}) \end{bmatrix} = \begin{bmatrix} \delta\mathrm{x} \\ \delta\hat{\mathrm{x}} \end{bmatrix} \tag{3.52}$$

where $\mathrm{X} \in \mathbb{R}^{n+\hat{n}}$ and $\delta\mathrm{x}$ and $\delta\hat{\mathrm{x}}$ are the true and navigation error states and based on the states defined in Equations (3.38) and (3.40), respectively.

The linear augmented state dynamics are given as:

$$\dot{\mathrm{X}} = \mathcal{F}\mathrm{X} + \mathcal{W}w \tag{3.53}$$

where

$$\mathcal{F} = \begin{bmatrix} F_{\mathrm{x}} & 0_{n\times\hat{n}} \\ 0_{\hat{n}\times n} & \hat{F}_{\hat{\mathrm{x}}} \end{bmatrix} \tag{3.54}$$

$$\mathcal{W} = \begin{bmatrix} W \\ 0_{\hat{n}\times n_w} \end{bmatrix} \tag{3.55}$$

F_{x} is the Jacobian of the truth model dynamics in Equation (3.1):

$$F_{\mathrm{x}} = \left.\frac{\partial f(\mathrm{x},\mathrm{u})}{\partial \mathrm{x}}\right|_{\mathrm{x}(t)} \tag{3.56}$$

and $\hat{F}_{\hat{\mathrm{x}}}$ is the Jacobian of the navigation model dynamics in Equation (3.10) as previously shown in Equation (3.15).

The critical aspect of Equation (3.53) is the validity of the linearization of the true and navigation dynamics. This step is generally applicable for nonlinear GNC systems and can be controlled by reducing the propagation step size.

3.2.2 Linear Augmented State Update

The augmented state update equation is given as:

$$\mathrm{X}_j^+ = \mathcal{A}_j \mathrm{X}_j^- + \mathcal{B}_j \nu \tag{3.57}$$

where X_j^- and X_j^+ are the augmented state at time t_j before and after the update, and:

$$\mathcal{A}_j = \begin{bmatrix} I_{n\times n} & 0_{n\times\hat{n}} \\ \hat{K}_j H_j & I_{\hat{n}\times\hat{n}} - \hat{K}_j \hat{H}_j \end{bmatrix} \tag{3.58}$$

$$\mathcal{B}_j = \begin{bmatrix} 0_{n\times\hat{n}} \\ \hat{K}_j \end{bmatrix} \tag{3.59}$$

$\hat{K}_j$ is the navigation filter Kalman gain defined in Equation (3.23) and $\hat{H}_j$ is the navigation filter measurement sensitivity matrix defined in Equation (3.24). H_j is the linearized truth measurement from Equation (3.6):

$$H_j = \left. \frac{\partial h(\mathrm{x})}{\partial \mathrm{x}} \right|_{\mathrm{x}_j^-} \tag{3.60}$$

Note that the true measurement, $\tilde{y}$, used in the navigation update (Equation (3.22)) is generating using a linearized measurement model plus the measurement noise according to the following approximation:

$$\tilde{\mathrm{y}}_j \approx \mathrm{H}_j \mathrm{x}_j + \nu_j \tag{3.61}$$

With this approximation, Equation (3.22) is rewritten as:

$$\delta\hat{\mathrm{x}}_j^+ = \left[I - \hat{K}_j \hat{H}_j\right] \delta\hat{x}_j^- + \hat{K}_j H_j \delta\mathrm{x}_j^- + \hat{K}_j \nu \tag{3.62}$$

to yield the form in Equation (3.57). The true state dispersions are unaffected by measurements.

3.2.3 Linear Augmented State Correction

Augmented state corrections, according to discrete control events, are corrected as:

$$\mathrm{X}_k^{+c} = \mathcal{D}_k \mathrm{X}_k^{-c} + \mathcal{N}_k \eta \tag{3.63}$$

where X_k^{-c} and X_k^{+c} are the augmented states at time t_j before and after the control correction, respectively, and:

$$\mathcal{D}_k = \begin{bmatrix} I_{n\times n} & \hat{B}_k \hat{G}_k \\ 0_{\hat{n}\times n} & I_{\hat{n}\times\hat{n}} + \hat{B}_k \hat{G}_k \end{bmatrix} \tag{3.64}$$

$$\mathcal{N}_k = \begin{bmatrix} \hat{B} \\ 0_{\hat{n}\times\hat{n}} \end{bmatrix} \tag{3.65}$$

$\hat{B}$ is the Jacobian defined in Equation (3.30) and $\hat{G}$ is the Jacobian of the nonlinear guidance algorithm in Equation (3.27):

$$\hat{G}_k = \left. \frac{\partial \hat{g}(\hat{\mathrm{x}})}{\partial \hat{\mathrm{x}}} \right|_{\hat{\mathrm{x}}_k^{-c}} \tag{3.66}$$

3.2.4 Augmented State Covariance

Equations (3.53), (3.57), and (3.63) represent the linearized state equations for the augmented state vector. Because the augmented state vector represents the dispersions from the true and navigation states, it has the following statistics:

$$E\left[\mathrm{X}\right] = 0_{(n+\hat{n})\times 1} \tag{3.67}$$

$$E\left[\mathrm{X}\mathrm{X}^T\right] = \left[C\right]_{(n+\hat{n})\times(n+\hat{n})} \tag{3.68}$$

where C is the augmented state covariance, the key to the linear covariance analysis!

3.2.5 Linear Covariance Model

The linear covariance model simply requires that we propagate, update, and correct this augmented state covariance along the nominal trajectory $\bar{\mathrm{x}}(t)$.

$$\dot{C} = \mathcal{F}C + C\mathcal{F}^T + \mathcal{W}Q_w\mathcal{W}^T \tag{3.69}$$

$$C_j^+ = \mathcal{A}_j C_j^- \mathcal{A}_j^T + \mathcal{B}_j R_\nu \mathcal{B}_j^T \tag{3.70}$$

$$C_k^{+c} = \mathcal{D}_k C_k^{-c} \mathcal{D}_k^T + \mathcal{N}_k S_\eta \mathcal{N}_k^T \tag{3.71}$$

Since $\mathcal{A}$ requires the Kalman gain for the update step, the model also requires that we propagate, update, and correct the navigation covariance (Equations (3.14), (3.25), and (3.29), respectively) in order to calculate the Kalman gain (Equation (3.23)).

For discrete time propagation, Equation (3.69) can be written as:

$$C_{i+1} = \Phi_A(t_{i+1}, t_i) C_i \Phi_A^T(t_{i+1}, t_i) + Q_{d,w}^A \tag{3.72}$$

where:

$$\Phi_A(t_{i+1}, t_i) = e^{\mathcal{F}(t_{i+1}-t_i)} = e^{\mathcal{F}\Delta t} \tag{3.73}$$

$$Q_{d,w}^A = \int_{t_i}^{t_{i+1}} \Phi_A(t_i, \tau)\mathcal{W}Q_w\mathcal{W}^T\Phi_A^T(t_i, \tau)d\tau \tag{3.74}$$

The linear covariance model is initialized by the reference state, x̄, and the initial covariance:

$$C_0 = \begin{bmatrix} C_{\text{true}}(t_0)_{n\times n} & 0_{n\times\hat{n}} \\ 0_{\hat{n}\times n} & 0_{\hat{n}\times\hat{n}} \end{bmatrix} \tag{3.75}$$

where $C_{\text{true}}(t_0)$ is the dispersion of the true state x.

The linear covariance analysis block diagram is shown in Figure 3.4.

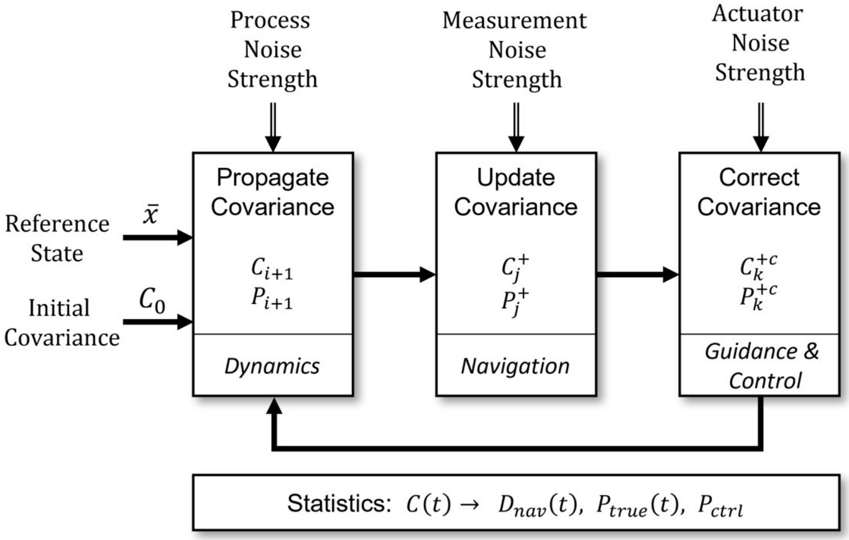

Fig. 3.4: Linear covariance analysis block diagram for a closed-loop GNC system.

3.2.6 Linear Covariance Performance Statistics

Similar to the Monte Carol performance statistics in Section 3.1.6, the linear covariance model provides the true state dispersion covariance, the true navigation error covariance, and all control error covariances.

True State Dispersion

The true state dispersion is contained directly in the augmented state covariance and is extracted as:

$$D_{\text{true}}(t) = E\Big[\delta\text{x}\ \delta\text{x}^T\Big] = \Big[I_{n\times n} \quad 0_{n\times\hat{n}}\Big]\, C(t)\, \Big[I_{n\times n} \quad 0_{n\times\hat{n}}\Big]^T \tag{3.76}$$

Navigation State Dispersion

The navigation state dispersion is also contained directly in the augmented state covariance and is extracted as:

$$D_{\text{nav}}(t) = E\Big[\delta\hat{\text{x}}\ \delta\hat{\text{x}}^T\Big] = \Big[0_{\hat{n}\times n} \quad I_{\hat{n}\times\hat{n}}\Big]\, C(t)\, \Big[0_{\hat{n}\times n} \quad I_{\hat{n}\times\hat{n}}\Big]^T \tag{3.77}$$

This error between the navigation state and the reference state isn't particularly useful for analysis as it is more convenient and informative to compare the error between the navigation state and the true state it is trying to estimate.

True Navigation Error Covariance

The true navigation error covariance is derived from the augmented state covariance as:

$$P_{\text{true}}(t) = E\Big[\delta e\ \delta e^T\Big] = \Big[-M_{\text{x}} \quad I_{\hat{n}\times\hat{n}}\Big]\, C(t)\, \Big[-M_{\text{x}} \quad I_{\hat{n}\times\hat{n}}\Big]^T \tag{3.78}$$

where M_{x} is the Jacobian of the state mapping function in Equation (3.26) evaluated at the reference state:

$$M_{\text{x}} = \left.\frac{\partial m_x(\text{x})}{\partial \text{x}}\right|_{\bar{\text{x}}} \tag{3.79}$$

Control Error Dispersion

The control error covariance for each discrete control event is also derived from the augmented state covariance as:

$$D_{\text{ctrl},k} = E\Big[\delta\text{u}_k\ \delta\text{u}_k^T\Big] = \Big[0_{n_u\times n} \quad \hat{G}_k\Big]\, C_k^{-c}\, \Big[0_{n_u\times n} \quad \hat{G}_k\Big]^T + \Delta S_\eta \tag{3.80}$$

with $\hat{G}$ defined in Equation (3.66).

3.3 Simplified Rocket Ship Analysis

To illustrate the the concepts of the Monte Carlo and linear covariance analysis techniques, consider a simplified scenario of an interstellar rocket ship cruising towards a distant planet as shown in Figure 3.5. The rocket ship is travelling at 1 astronomical unit (au) per day towards the planet that is exactly 100 au away. In order to meet a narrow landing window, the rocket ship must be within ±0.5 au at the 100 day mark. The rocket ship's speed, however, is affected by random perturbations and the exact velocity is unknown during the approach. Given this growing uncertainty in the rocket ship's speed, the pilot wants to know the likelihood of meeting the tight landing window. The equations for both analyses are derived in Appendix C to simplify the discussion here.

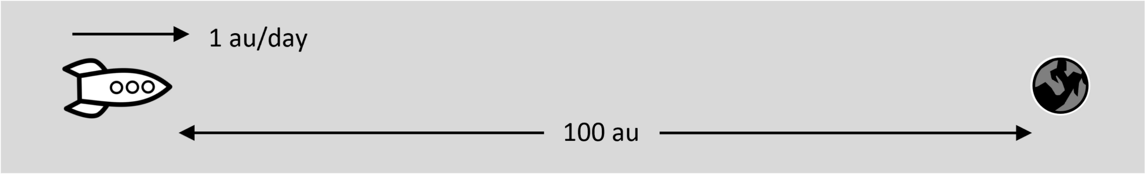

Fig. 3.5: Rocket ship scenario flying to distant planet.

3.3.1 Monte Carlo Analysis

The true position dispersions from 1000 runs of the Monte Carlo analysis, without performing any measurements or maneuvers, are shown in Figure 3.6. The calculated 3σ dispersion bounds are also shown. The final 3σ position dispersion is ±1.64 au, well outside the landing window requirement.

To assist in the approach the pilot is able to make a single range measurement to the planet. Based on her experience, the pilot chooses to take the measurement on day 80. With just the range measurement added to the analysis the true position dispersions are shown in Figure 3.7 and the true navigation position errors (the difference between the true position and the estimated position) is shown in Figure 3.8. It can be seen that the navigation measurement improves the pilot's knowledge of her ship's position (for some limited time), but does not improve the true final dispersion.

Luckily, the ship has a thruster capable of performing a single correction maneuver during the approach. Given the preparation time required for the maneuver, the pilot always performs the maneuver 5 days after receiving the measurement (i.e. day 85). The results with this added maneuver are shown in Figures 3.9 and 3.10.

Performing the correction maneuver reduces the final 3σ position dispersion from ± 1.64 au down to ± 0.31 au, which, to the the pilots pleasure, meets the landing requirement. Given the limited fuel onboard the rocket ship, the pilot is also concerned with the amount of fuel that may be required to execute the maneuver. The 3σ dispersion on the thruster ΔV is ± 0.112 au/day as shown on the control histogram in Figure 3.11.

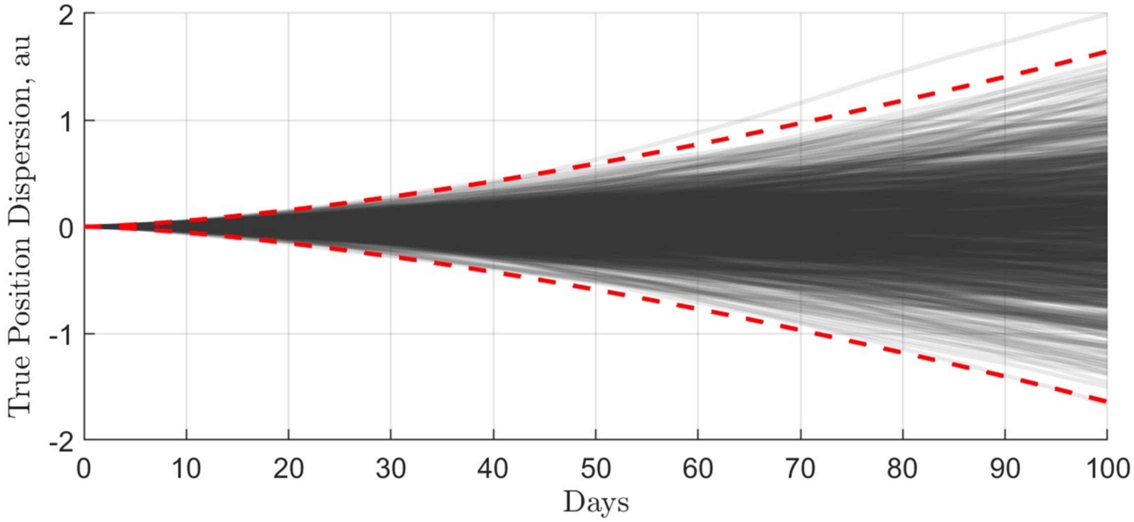

Fig. 3.6: Rocket ship true position dispersions with calculated 3σ bounds for 1000 runs without navigation or maneuvers.

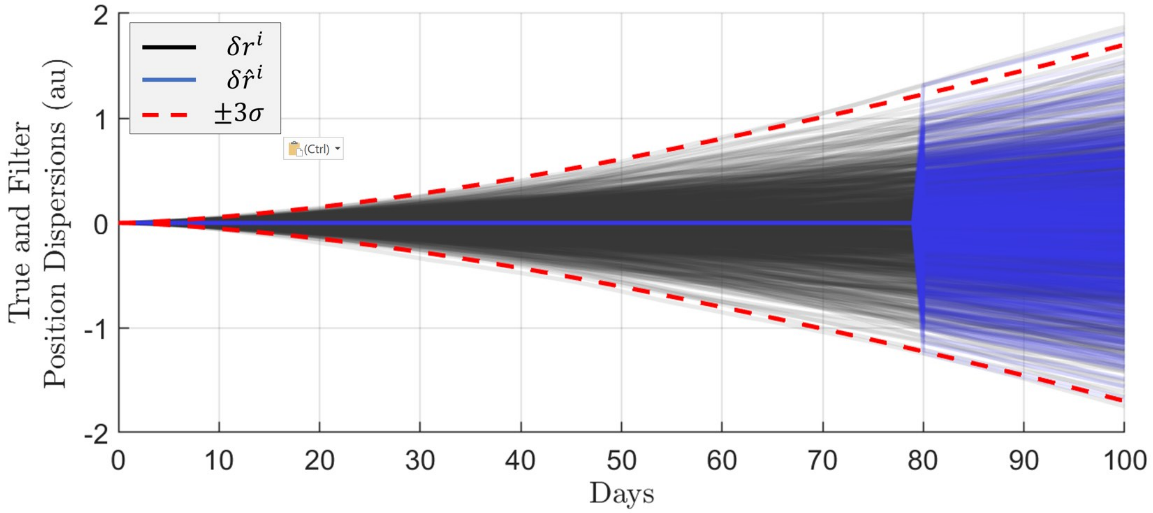

Fig. 3.7: Rocket ship true position dispersions (black) and filter position dispersions (blue) for 1000 runs with a single measurement update on day 80.

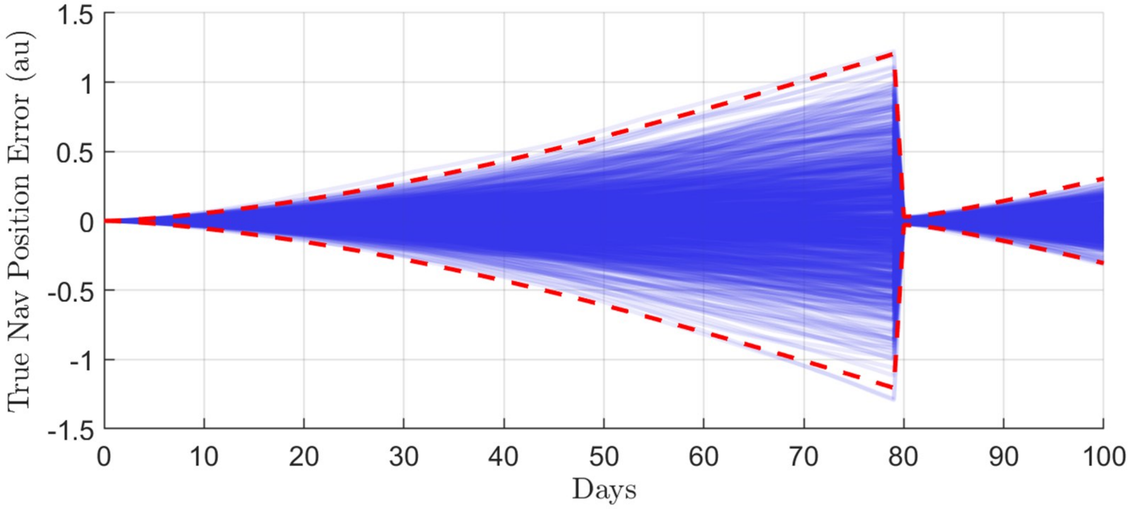

Fig. 3.8: Rocket ship true navigation position errors for 1000 runs with a single measurement update on day 80.

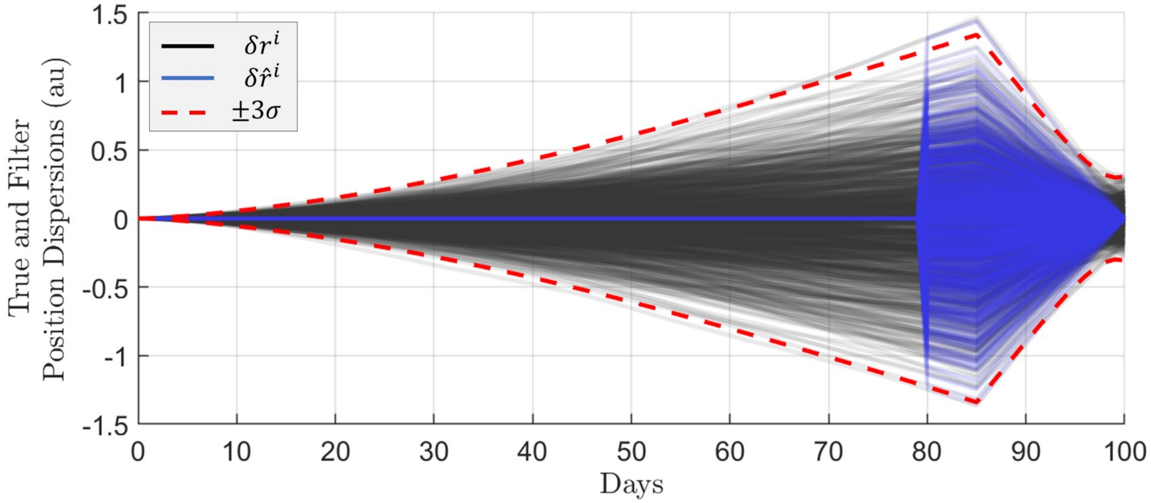

Fig. 3.9: Rocket ship true position dispersions (black) and filter position dispersions (blue) for 1000 runs with a single measurement update on day 80 and correction maneuver on day 85.

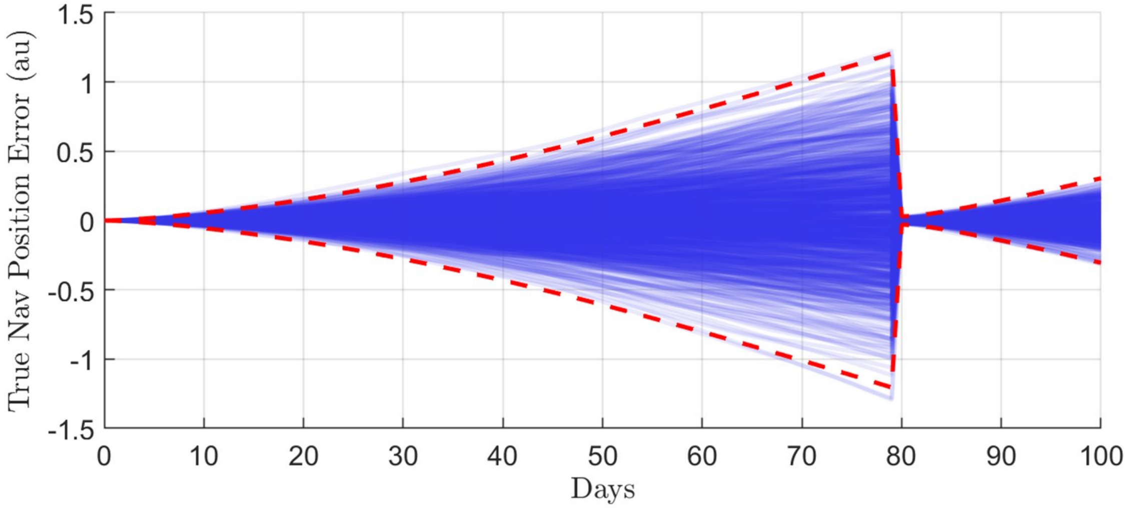

Fig. 3.10: Rocket ship true navigation position errors for 1000 runs with a single measurement update on day 80 and correction maneuver on day 85.

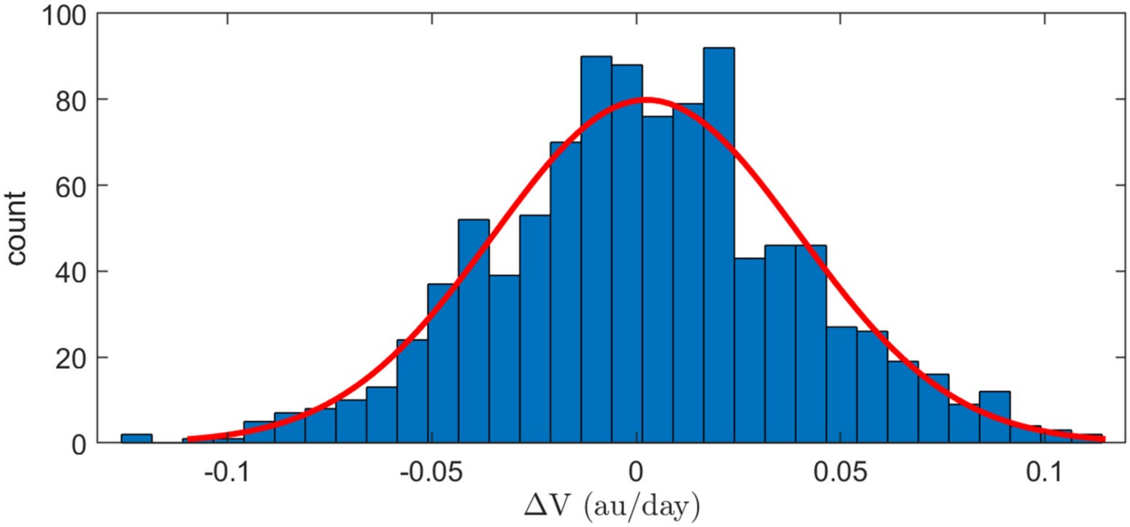

Fig. 3.11: Rocket ship ΔV dispersions histogram and probability density function for 1000 runs. $\mu = 0.0025$ and $\sigma = 0.0375$.

3.3.2 Linear Covariance Analysis

We can perform the same analysis in a fraction of the time using the linear covariance equations derived in Appendix C. The true position dispersion calculated by the LinCov model (including the measurement update and correction maneuver) is shown in Figure 3.12 against the results calculated using the Monte Carlo analysis. The linear covariance bounds are nearly identical to those generated by the Monte Carlo analysis.

The ΔV dispersion from the LinCov analysis is ± 0.112 (3σ), the same as calculated by the Monte Carlo simulation. The probability distribution functions (pdf) for the Monte Carlo and Linear Covariance analyses are shown in Figure 3.13.

For this example, the linear covariance simulation takes about the same amount of time to run as a single run of the Monte Carlo analysis. Since 1000 runs were required for the analysis, the LinCov took 0.1% of the time to complete and generated nearly identical results.

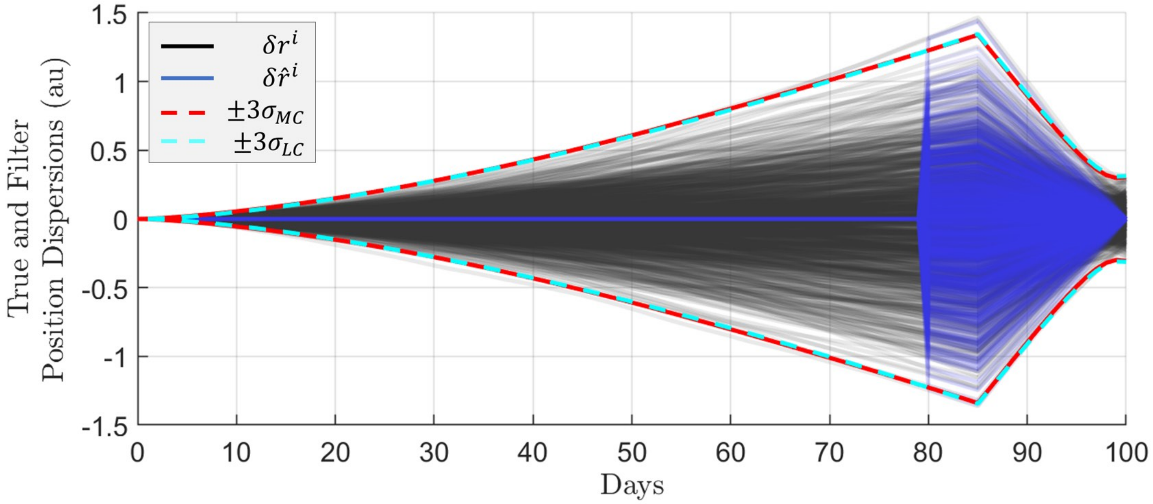

Fig. 3.12: Rocket ship true position dispersions (black) and filter position dispersions (blue) for 1000 runs with a measurement update on day 80 and correction maneuver on day 85. ±3σ bounds from the Monte Carlo (red) and LinCov (cyan) analyses.

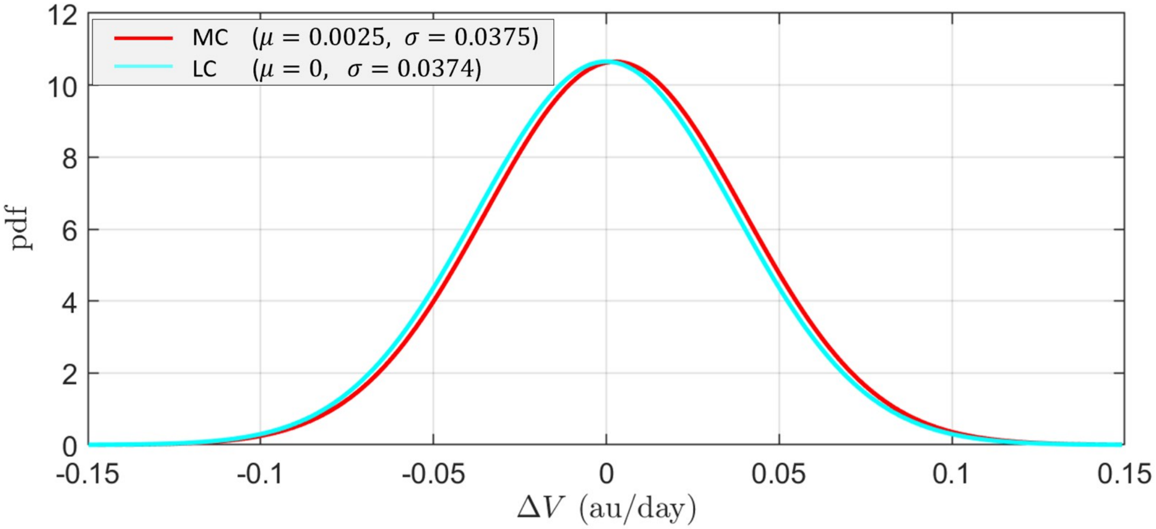

Fig. 3.13: ΔV dispersions probability density function for the Monte Carlo and LinCov analyses.

3.3.3 Optimal Maneuver Location

The pilot feels content that her intuition to perform the range measurement on day 80 met the landing window requirement. However, in an effort to minimize fuel, she would like to know the optimal time to perform the update and the maneuver 5 days later.

To determine the optimal time, we can use either the Monte Carlo or linear covariance simulations to evaluate all possible options from day 1 to 94. Performing this analysis via Monte Carlo would require 94,000 evaluation (assuming we stay with 1000 runs per evaluation to maintain the same statistics). The linear covariance analysis, by comparison, only requires 94 evaluations leading to a significant reduction in processing time. The linear covariance results for the final position and ΔV dispersions are shown in Figure 3.14. After running the analysis, the pilot decides to perform the navigation update on day 69 resulting in a final position dispersion of ± 0.490 au (3σ), just under the landing requirement, and a ΔV dispersion of ± 0.063 au/day (3σ), a reduction by almost a factor of 2.

It should be pointed out that this scenario is intentionally simple and does not address the full range of possibility for Monte Carlo or LinCov analyses. It is, however, simple enough to quickly demonstrate the analyses and provide instructional visualization. The analysis also demonstrates the power of linear covariance techniques to generate optimal solutions for stochastic systems.

The reader is directed to the tutorial by Christensen and Geller for a more detailed example [105] of a closed-loop linear covariance analysis.

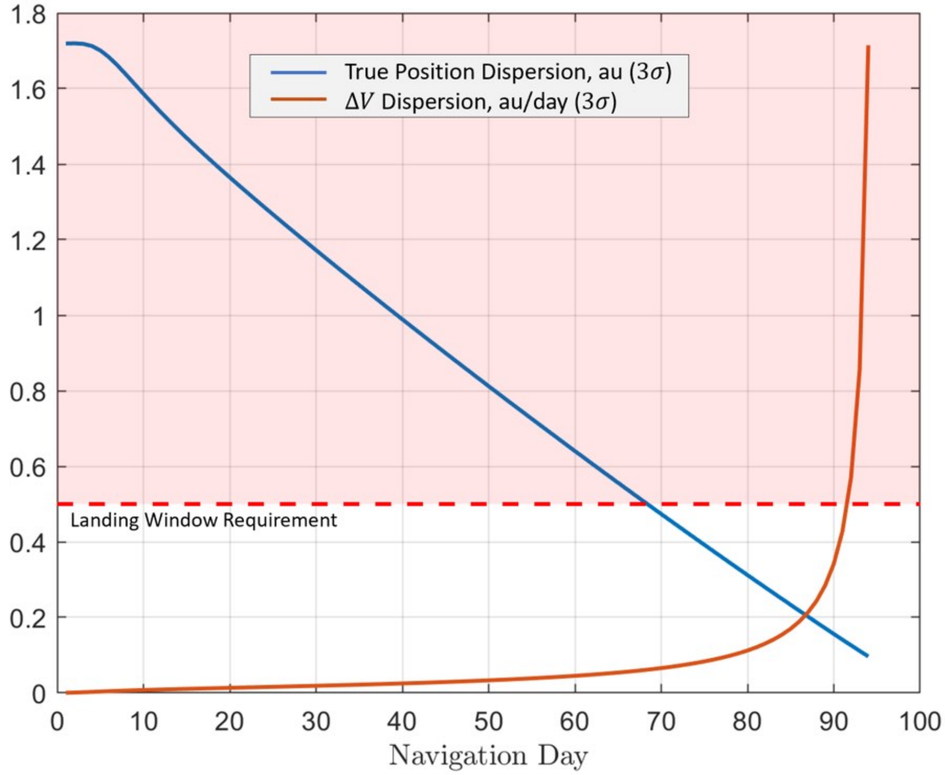

Fig. 3.14: Final true position and ΔV dispersions for all possible days to perform the measurement update as calculated by the linear covariance model.

CHAPTER 4

LINEAR COVARIANCE MODEL FOR RENDEZVOUS AND PROXIMITY OPERATIONS

In this chapter we will develop the linear covariance equations for a generic rendezvous and proximity operations simulation. Since this model is intended as a generic analysis tool to be paired later with an optimization routine, we will develop a simplified model with limited states in order to minimize run time. Geller provides a detailed development of the linear covariance equations more suitable for specific mission analysis [101].

4.1 Truth Model Dynamics

The RPO truth model consists of the true position and velocity of the spacecraft in the LVLH frame:

$$\mathrm{x} = \begin{bmatrix} \mathrm{r} \\ \mathrm{v} \end{bmatrix}_{6\times 1}^{\mathrm{LVLH}} \tag{4.1}$$

which can be propagated using the continuous Clohessy-Wiltshire (CW) equations with additional random acceleration noise to represent the limitations in those equations.

$$\dot{\mathrm{x}} = F\mathrm{x} + Ww \tag{4.2}$$

$$F = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 3\omega^2 & 0 & 0 & 0 & 2\omega & 0 \\ 0 & 0 & 0 & -2\omega & 0 & 0 \\ 0 & 0 & \omega^2 & 0 & 0 & 0 \end{bmatrix}, \quad W = \begin{bmatrix} 0_{3\times 3} \\ I_{3\times 3} \end{bmatrix}, \quad w = \begin{bmatrix} w_x \\ w_y \\ w_z \end{bmatrix} \tag{4.3}$$

The acceleration noise, w, is an uncorrelated, zero-mean Gaussian process:

$$E[w] = 0_{3\times1} \tag{4.4}$$

$$E\left[w(t)w^T(t')\right] = \left[Q_w\delta(t-t')\right]_{3\times3} \tag{4.5}$$

where Q_w is the process noise power spectral density (PSD).

In order to simplify the propagation process and reduce dependencies on time steps, we will choose to implement the truth model using the discrete form of the CW equations:

$$\mathrm{x}_{i+1} = \Phi(t_{i+1}, t_i)\mathrm{x}_i + w_{d,i} = \Phi(\Delta t)\mathrm{x}_i + w_{d,i} \tag{4.6}$$

where:

$$\Phi(\Delta t) = e^{F\Delta t} = \begin{bmatrix} \Phi_{rr} & \Phi_{rv} \\ \Phi_{vr} & \Phi_{vv} \end{bmatrix} \tag{4.7}$$

is the CW state transition matrix defined in Equation (2.32) and the discrete noise is integrated from the continuous noise terms [94].

$$w_{d,i} = \int_{t_i}^{t_{i+1}} \Phi(t_{i+1}, \tau)W(\tau)d\beta(\tau) \tag{4.8}$$

$$E\left[w_{d,i}w_{d,j}^T\right] = \left[Q_d\delta_{ij}\right]_{6\times6} \tag{4.9}$$

where β is a Brownian motion process.

4.2 Truth Sensor Models

This simulation will use two potential sensors for relative navigation. The first sensor is a lidar that produces a full, three-dimensional position measurement of the RSO relative to the RPO spacecraft. Similar measurements can be achieved through other sensor phenomenology and processing, but the general measurement models are the same. The second sensor is a camera that produces a line-of-sight measurement, in the form of angles or measured pixel locations, from the RPO spacecraft to the RSO.

4.2.1 Lidar Model

Light detection and ranging (lidar) is an active sensor that uses either light or laser sources to measure the distance (i.e. range) to an object. A lidar can detect range from reflected light using either the measured time of flight from short pulses of light (called direct time-of-flight) or from the the change in time and phase from a continuous, modulated signal (called indirect time-of-flight). Space-based lidars used for proximity operations generally have narrow beam widths and require precision pointing to generate a reflected signal for measurement.

After calibration (to remove mounting misalignment and any range measurement bias) the lidar measurement at time t_j is accurately modeled as:

$$\tilde{\mathrm{y}}_j^{lidar} = \mathrm{r}_j + \nu_j^{lidar} = \begin{bmatrix} I_{3\times3} & 0_{3\times3} \end{bmatrix} \mathrm{x}_j + \nu_j^{lidar} \tag{4.10}$$

where $\tilde{\mathrm{y}}^{lidar}$ is the lidar measurement output, r_j is the true relative position vector of the spacecraft relative to the RSO and ν^{lidar} is an uncorrelated, zero-mean Guassian noise process.

$$E[\nu_j^{lidar}] = 0_{3\mathrm{x}1} \tag{4.11}$$

$$E[(\nu_j^{lidar})(\nu_{j'}^{lidar})^T] = \begin{bmatrix} R_\nu^{lidar}\delta_{jj'} \end{bmatrix}_{3\times3} \tag{4.12}$$

In reality, many lidar sensors produce a single range measurement value that, when coupled with the spacecraft attitude, is able to produce a full three-dimensional estimated position vector in the desired reference frame.

4.2.2 Lidar Constraints

Lidar measurments are generally limited to a maximum range based on the laser power and the spacecraft's pointing accuracy. Small errors in attitude control or sensor alignment will prevent the sensor for hitting its target at longer ranges. Even with perfect pointing knowledge and control, the lidar's return signal strength is reduced the farther away the object is from the laser source preventing the sensor from detecting and producing a valid range measurement. Valid lidar measurements are only produced when $\|\mathrm{r}\| \leq r^{max}$ where r^{max} is the specified maximum lidar range.

4.2.3 Camera Model

Cameras are an extremely popular choice for RPO relative navigation due to their simplicity and low power consumption compared to active sensors such as lidar. Navigation cameras can either operate in the visible or infrared spectrum. Visible sensors offer increased resolution but rely on reflected light for RSO detection. Infrared sensors typically employ lower resolution detectors but can detect radiated heat allowing operations during orbital eclipse periods.

A key aspect of camera-based navigation is the image processing algorithm used to detect and centroid the RSO in the sensor's field-of-view. The measured output from the algorithms may consist of the calculated centroid pixel location (p_{row}, p_{col}) or the derived angles (α, ε) based on the camera field-of-view and optical model.

For this model we will assume a simplified pin-hole camera (i.e. no distortions) that generates angle measurements with simple uncorrelated, zero-mean Gaussian noise at time t_j:

$$\tilde{\mathrm{y}}_j^{cam} = h^{cam}(\mathrm{x}_j) + \nu_j^{cam} \tag{4.13}$$

$$\tilde{\mathrm{y}}_j^{cam} = \begin{bmatrix} \alpha \\ \varepsilon \end{bmatrix} + \nu_j^{cam} = \begin{bmatrix} r_x^{cam}/r_z^{cam} \\ r_y^{cam}/r_z^{cam} \end{bmatrix} + \nu_j^{cam} \tag{4.14}$$

where:

$$\mathrm{r}_{RSO}^{cam} = \begin{bmatrix} r_x^{cam} \\ r_y^{cam} \\ r_z^{cam} \end{bmatrix} = -T_{lvlh}^{cam}\mathrm{r} \tag{4.15}$$

r_{RSO}^{cam} is the position of the RSO in the camera coordinate frame and T_{lvlh}^{cam} is the direction cosine matrix (DCM) that defines the rotation from the LVLH coordinate frame to the camera coordinate frame. The camera frame is shown in Figure 4.1.

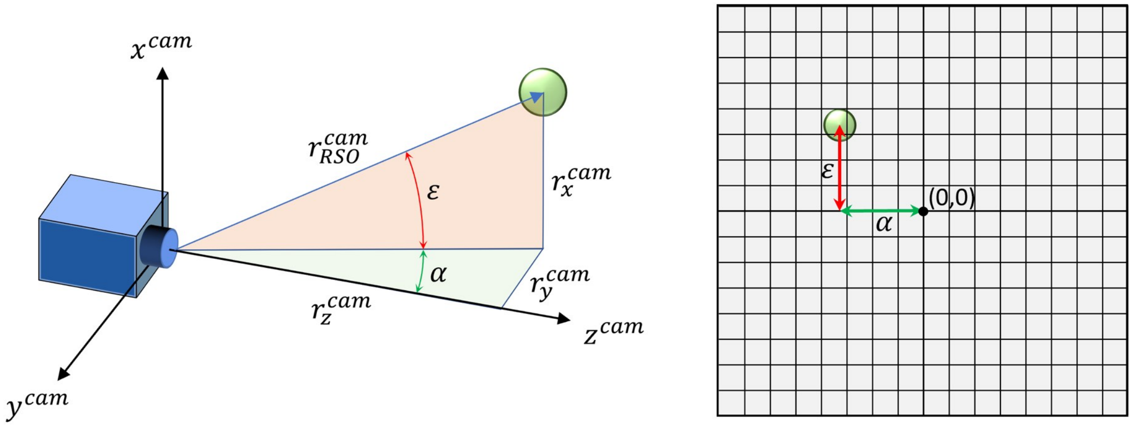

Fig. 4.1: RSO location and angles-only measurements in the camera frame (left) and camera focal plane (right).

The nominal spacecraft attitude for the simulation points the camera's boresight, $\hat{k}^{cam}$, directly at the estimated RSO location and the camera's x-axis, $\hat{\imath}^{cam}$, is then constrained in the cross-track direction.

$$\hat{z}_{cam}^{lvlh} = -\frac{\hat{\mathrm{r}}}{\|\hat{\mathrm{r}}\|} \tag{4.16}$$

$$\hat{x}_{cam}^{lvlh} = \hat{k}^{lvlh} \times \hat{z}_{cam}^{lvlh} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \times \hat{z}_{cam}^{lvlh} \tag{4.17}$$

$$\hat{\jmath}^{cam} = \hat{k}^{cam} \times \hat{\imath}^{cam} \tag{4.18}$$

$$T_{lvlh}^{cam} = \begin{bmatrix} \hat{\imath}^{cam} & \hat{\jmath}^{cam} & \hat{k}^{cam} \end{bmatrix}^T \tag{4.19}$$

The true measurement is a complex function of multiple variables including the camera optics, focal plane and misalignment, the observed RSO's shape, attitude, and surface properties, the observer's attitude errors and jitter, and the incidence angle of the sun. Observations may even be affected by slow thermal distortions in the spacecraft's structure causing time varying bias effects. Based on all of these variables it is the job of the image processing algorithm to accurately identify the centroid/pixel location that corresponds to the RSO's center-of-gravity (rather than the center-of-brightness). These complex effects have been ignored for this analysis and we will assume the measurement error is a zero-mean, Gaussian noise process:

$$E[\nu_j^{cam}] = 0_{2\times 1} \tag{4.20}$$

$$E[(\nu_j^{cam})(\nu_{j'}^{cam})^T] = \left[R_\nu^{cam}\delta_{jj'}\right]_{2\times 2} \tag{4.21}$$

4.2.4 Camera Constraints

Camera measurements are subject to outages when external factors prevent the image processing algorithm from identifying the RSO and producing a valid centroid. For this model we will include two key outages due to the sun and the earth.

Camera Measurement Outage - Earth

Image processing algorithm are typically unable to identify an RSO against the earth background, thus preventing sensor measurements.

The angle between the sensor boresight and the center of the earth is given as:

$$\phi_\oplus = \cos^{-1}\left(-\hat{i}^{lvlh}\cdot\hat{k}^{cam}\right) = \begin{bmatrix}-1 & 0 & 0\end{bmatrix}\hat{k}^{cam} \tag{4.22}$$

The angle subtended by the earth is a function of the orbit radius (assuming a circular orbit):

$$\psi_\oplus = \tan^{-1}\left(\frac{R_\oplus}{r}\right) \tag{4.23}$$

and the radius can be defined as a function of the orbital mean motion:

$$r = \left(\frac{\mu}{\omega^2}\right)^{1/3} \tag{4.24}$$

The angle between the sensor boresight and the earth limb is then given as:

$$\theta_\oplus = \phi_\oplus - \psi_\oplus \tag{4.25}$$

In cases where $\theta_\oplus < 0$ the camera boresight is pointed below the earth limb. Camera

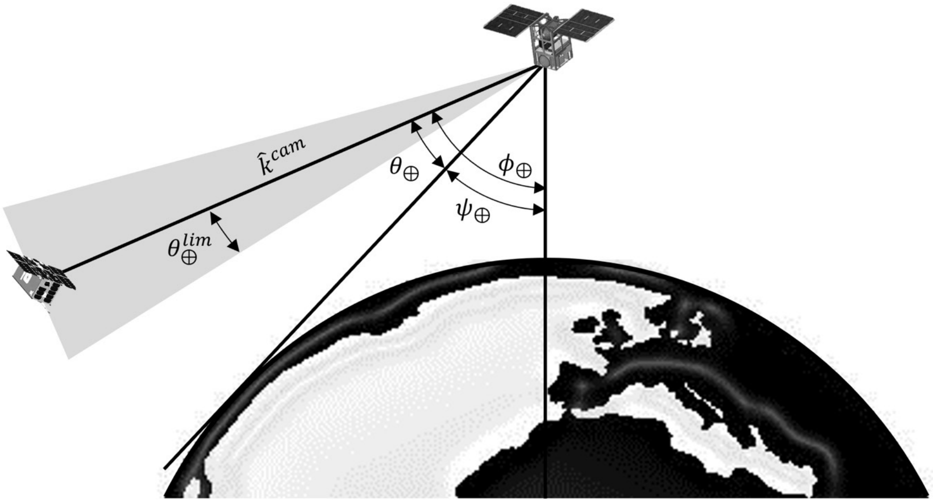

Fig. 4.2: Earth exclusion angles.

measurements are inhibited whenever $\theta_{\oplus}$ falls below a specified angle, $\theta_{\oplus}^{lim}$. This limiting angle is typically determined by the camera field-of-view to prevent any portion of the earth from entering the image and interfering with image processing. The earth exclusion angles are shown in Figure 4.2.

Sun Model in the LVLH Frame

To determine if the sun will impact camera measurements, we need to know the sun's location in the LVLH coordinate frame. Over a short period of time (several orbits), the sun vector, relative to the RSO location at the origin of the LVLH frame, traces out a conical shape with the apex at the origin and the axis aligned with the cross-track axis. The sun sweeps out one complete revolution of the cone each orbit.

Given the initial unit sun vector, the sun vector in the LVLH frame at any future time is given as:

$$\hat{s}^{lvlh}(t) = \begin{bmatrix} \cos\omega t & \sin\omega t & 0 \\ -\sin\omega t & \cos\omega t & 0 \\ 0 & 0 & 1 \end{bmatrix} \hat{s}^{lvlh}(t_0) \tag{4.26}$$

where $\hat{s}^{lvlh}(t_0)$ is the initial unit sun vector at time t_0, ω is the orbital mean motion, and $\hat{s}^{lvlh}(t)$ is the unit sun vector at time t. The derivation of Equation (4.26) is provided in Appendix B. It it is important to note that the sun rotates in the same direction as the vehicle when in an NMC, creating the possibility of having the sun continually at the observer's back, or persistently blinding the camera optics. The sun's conical path in the LVLH frame is shown in Figure 4.3.

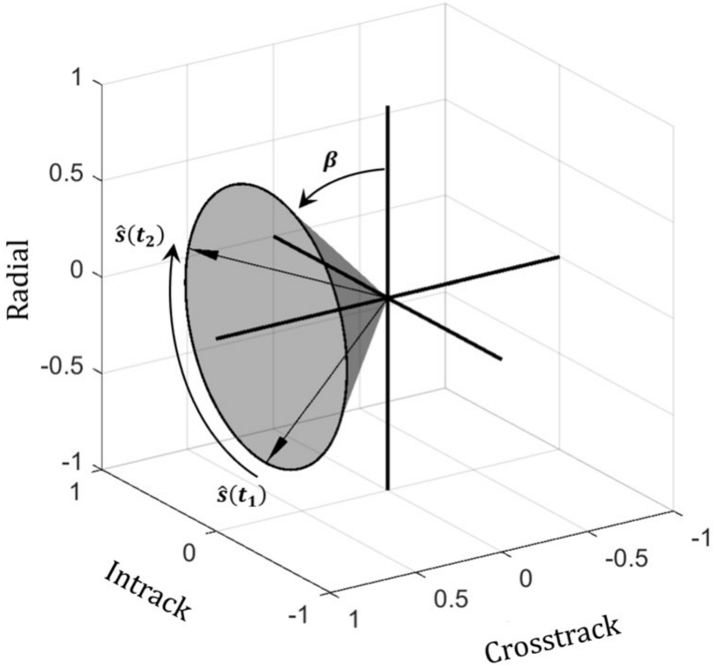

Fig. 4.3: Example of the sun's conical path in the LVLH frame.

The angle of the cone from the radial-intrack plane corresponds the the beta angle of the reference orbit where β is the angle between the orbit plane and the sun (Figure 4.4). For $\beta = 0$ the sun traces a path around the radial-intrack plane. For $\beta = 90°$ the sun maintains a constant position along the cross-track axis in the LVLH frame. Variations in the reference orbit's beta angle can significantly impact proximity operations.

Camera Measurement Outage - Sun

Navigation cameras are subject to stray light, particularly from the sun, which can saturate the image focal plane array and prevent measurements from being generated.

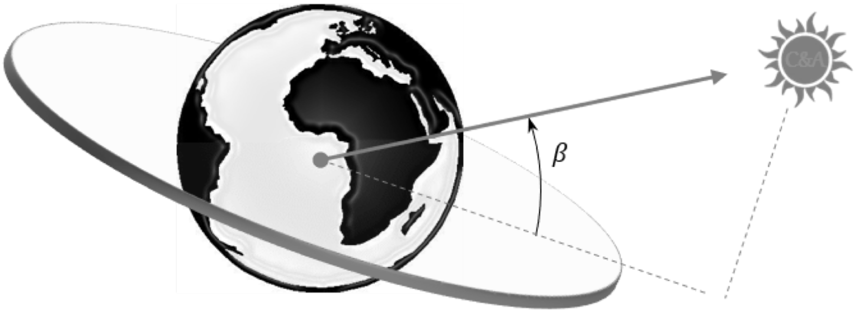

Fig. 4.4: Beta angle between the reference orbit plane and the sun.

The angle between the camera boresight and the sun is calculated as:

$$\theta_{\odot} = \cos^{-1}\left(\hat{k}^{cam} \cdot \hat{s}^{lvlh}\right) \tag{4.27}$$

Camera measurements are inhibited whenever $\theta_{\odot}$ falls below a specified angle, $\theta_{\odot}^{lim}$. This limiting angle is determined by the camera optics. Solar measurements are shown in Figure 4.5.

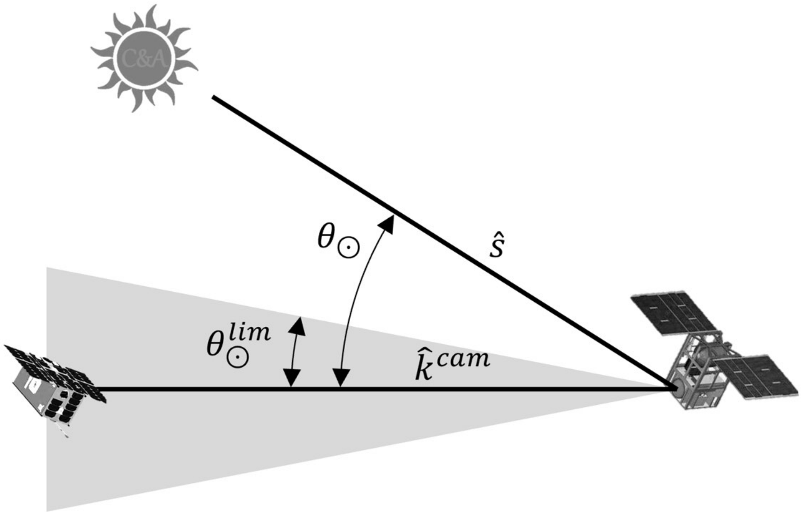

Fig. 4.5: Solar exclusion angles.

4.3 Truth Thruster Control

Key to most RPO missions is the propulsion system that enables the spacecraft to adjust its orbit to achieve a desired trajectory near the RSO. Common propulsion systems used for RPO include cold/warm gas, monopropellant, and bipropellant systems. Solid propellants and solar-electric thrusters can also be used but have their own unique control issues not addressed here. Cold and warm gas systems consist of a single pressurized gas, such as the refrigerant R-134a, that is expelled in gaseous form to create reaction forces. While simple and inexpensive, cold/warm gas systems do not typically provide a large total impulse (or ΔV) and are limited to smaller spacecraft with limited missions. Control accuracy is usually as good as the ability to control the thruster valve. Monopropellants, such as the popular but toxic hydrazine (N_2H_4), can provide a significantly larger total ΔV but with increased complexity. The increased complexity also increases modeling and control requirements to achieve accurate results.

For this simulation, we will assume a generic thruster control system applicable to many propulsion systems. Given a commanded ΔV at time t_k, the thrust controller and actuator executes the maneuver as:

$$\Delta V_k = \Delta \hat{V}_k + \eta_k \tag{4.28}$$

where $\Delta \hat{V}$ is the commanded ΔV and η is the actuator noise due to errors in either the thruster or thruster control. The actuator noise is a zero-mean, Gaussian noise process:

$$E[\eta_k] = 0_{3\text{x}1} \tag{4.29}$$

$$E[\eta_k \eta_{k'}^T] = S_{\Delta V_k} \delta_{kk'} \tag{4.30}$$

The maneuver error covariance, $S_{\Delta V}$, can be either a function of $\Delta\bar{V}$ or a fixed value. As a function of $\Delta\bar{V}$, the maneuver error covariance can be scaled as a percentage of the commanded value (e.g 5% of the $\Delta\bar{V}$ magnitude). Larger burns equate to larger errors. A constant value for $S_{\Delta V}$ would indicate that a propulsion system has an fixed error uncertainty (e.g. ± 2 cm/s) regardless of the commanded ΔV. This error model would correspond to systems with accurate thruster control but inaccurate thrust output (possibly due to a slow actuator valve).

4.4 Navigation Model Dynamics

The onboard design reference model is identical to the truth model described with states for the estimated position and velocity:

$$\hat{\mathrm{x}} = \begin{bmatrix} \hat{\mathrm{r}} \\ \\ \hat{\mathrm{v}} \end{bmatrix}^{\mathrm{LVLH}}_{6\times 1} \tag{4.31}$$

The estimated state is propagated according to the discrete CW dynamics:

$$\hat{\mathrm{x}}(t_{i+1}) = \hat{\Phi}(t_{i+1}, t_i)\hat{\mathrm{x}}(t_i) = \hat{\Phi}(\Delta t)\hat{\mathrm{x}}(t_i) \tag{4.32}$$

where $\hat{\Phi}$ is the CW state transition matrix defined in Equation (2.32).

The state covariance matrix is propagated using the discrete form:

$$\hat{P}(t_{i+1}) = \left[\hat{\Phi}(t_{i+1}, t_i)\right] \hat{P}(t_i) \left[\hat{\Phi}(t_{i+1}, t_i)\right]^T + \hat{Q}_d \tag{4.33}$$

where $\hat{Q}_d$ is the discrete process noise and equivalent to Equation (4.9) defined for the truth model process noise. The discrete process noise can be approximated as:

$$\hat{Q}_d \approx \hat{\Phi}\hat{W}\hat{Q}_w\hat{W}^T\hat{\Phi}^T\Delta t \tag{4.34}$$

4.5 Relative Measurement Update Models

The onboard navigation filter processes measurement from both the lidar and camera when available.

4.5.1 Lidar-based Navigation

Lidar measurements at time t_k are processed using the Kalman filter update equations for the state and covariance.

$$\hat{\mathrm{x}}_j^+ = \hat{\mathrm{x}}_j^- + \hat{K}_j \left[\tilde{y}_j^{lidar} - \hat{H}^{lidar}\hat{\mathrm{x}}_j^-\right] \tag{4.35}$$

$$\hat{P}_j^+ = \left[I - \hat{K}_j\hat{H}^{lidar}\right]\hat{P}_j^-\left[I - \hat{K}_j\hat{H}^{lidar}\right]^T + \hat{K}_j R_\nu^{lidar}\hat{K}_j^T \tag{4.36}$$

where:

$$\hat{K}_j = \hat{P}_j^-\left(\hat{H}_j^{lidar}\right)^T\left[\left(\hat{H}_j^{lidar}\right)\hat{P}_j^-\left(\hat{H}_j^{lidar}\right)^T + \hat{R}_\nu^{lidar}\right]^{-1} \tag{4.37}$$

$$\hat{H}^{lidar} = \begin{bmatrix} I_{3\times3} & 0_{3\times3} \end{bmatrix} \tag{4.38}$$

$\hat{K}_j$ is the Kalman gain and $\hat{H}^{lidar}$ is the lidar measurement sensitivity matrix. The true sensor measurement, $\tilde{y}^{lidar}$, is given in Equation (4.10).

4.5.2 Angles-only Navigation

Camera angle measurements at time t_j are also processed using the Kalman filter update equations for the state and covariance.

$$\hat{\mathrm{x}}_j^+ = \hat{\mathrm{x}}_j^- + \hat{K}_j \left[\tilde{y}_j^{cam} - \hat{h}^{cam}\left(\hat{\mathrm{x}}_j^-\right)\right] \tag{4.39}$$

$$\hat{P}_j^+ = \left[I - \hat{K}_j\hat{H}_j^{cam}\right]\hat{P}_j^-\left[I - \hat{K}_j\hat{H}_j^{cam}\right]^T + \hat{K}_j R_\nu^{cam}\hat{K}_j^T \tag{4.40}$$

where:

$$\hat{K}_j = \hat{P}_j^-\left(\hat{H}_j^{cam}\right)^T\left[\left(\hat{H}_j^{cam}\right)\hat{P}_j^-\left(\hat{H}_j^{cam}\right)^T + \hat{R}_\nu^{cam}\right]^{-1} \tag{4.41}$$

$$\hat{H}_j^{cam} = \begin{bmatrix} 1/r_z^{cam} & 0 & 0 \\ 0 & 1/r_z^{cam} & 0 \end{bmatrix} \begin{bmatrix} -T_{lvlh}^{cam} & 0_{3\times3} \end{bmatrix} \tag{4.42}$$

Again, $\hat{K}_j$ is the Kalman gain, $\hat{h}^{cam}$ is the nonlinear function predicting the camera measurement values, equivalent to the nonlinear truth model, and $\hat{H}_j^{cam}$ is the camera measurement sensitivity matrix derived in Appendix D. The true sensor measurement, $\tilde{y}^{cam}$, is given in Equation (4.13).

4.6 Guidance Models

Given the estimated state and desired trajectory (defined by waypoints and associated transfer times), the onboard guidance system calculates the necessary maneuvers to maintain the trajectory. The calculated ΔV at time t_k, indicated by $\Delta\hat{V}_k$ is then passed to the controller for thrust actuation.

For a waypoint following guidance system there are two maneuver types: waypoint targeting and velocity matching. Waypoint targeting uses the CW targeting algorithms described in Section 2.2 where the inputs include the current estimated position, the commanded waypoint location, and the corresponding transfer time. Velocity matching, typically applied when the vehicle has reached the final waypoint, simply corrects the vehicle's velocity to match the desired trajectory state. For either case, the state and covariance are corrected for a maneuver at time t_k as:

$$\hat{\mathrm{x}}_k^{+c} = \hat{\mathrm{x}}_k^{-c} + \hat{B}\Delta\hat{V}_k \tag{4.43}$$

$$\hat{P}_k^{+c} = \hat{P}_k^{-c} + \hat{B}S_{\Delta V}\hat{B}^T \tag{4.44}$$

where:

$$\hat{B} = \begin{bmatrix} 0_{3\times3} \\ I_{3\times3} \end{bmatrix} \tag{4.45}$$

Since $\Delta\hat{V}$ is a linear function of $\hat{\mathrm{x}}$, Equation (4.43) can be rewritten in the following form:

$$\hat{\mathrm{x}}_k^{+c} = \hat{\mathrm{x}}_k^{-c} + \hat{B}\left(\hat{G}\hat{\mathrm{x}}_k^{-c} + \hat{D}\right) \tag{4.46}$$

where the values $\hat{G}$ and $\hat{D}$ are dependent on the maneuver method.

4.6.1 Waypoint Targeting

From Equations (2.35) and (2.36), the calculated ΔV to reach a waypoint is calculated as:

$$\Delta\hat{V}_k = \hat{\Phi}_{rv}^{-1}(\Delta t)\left[R^* - \hat{\Phi}_{rr}(\Delta t)\,\hat{\mathrm{r}}_k^{-c}\right] - \hat{\mathrm{v}}_k^{-c} \tag{4.47}$$

where R^* is the commanded waypoint and Δt the commanded transfer time to reach the waypoint. Equation (4.47) can be rearranged into a linear function as:

$$\Delta\hat{V}_k = \begin{bmatrix} \hat{\Phi}_{rv}^{-1}(\Delta t)\;\; \hat{\Phi}_{rr}(\Delta t) \\ -I_{3\times3} \end{bmatrix} \hat{\mathrm{x}}_k^{-c} + \hat{\Phi}_{rv}(\Delta t)\,R^* \tag{4.48}$$

such that:

$$\hat{G}_{\mathrm{wpt}} = \begin{bmatrix} \hat{\Phi}_{rv}^{-1}(\Delta t)\;\; \hat{\Phi}_{rr}(\Delta t) \\ -I_{3\times3} \end{bmatrix} \tag{4.49}$$

$$\hat{D}_{\mathrm{wpt}} = \hat{\Phi}_{rv}(\Delta t)\,R^* \tag{4.50}$$

4.6.2 Velocity Matching

The objective of the guidance system to reach a final desired state, x_f at the final time, t_f. One way to achieve this is to make the final waypoint equal to the final desired state, $R_f^* = \mathrm{x}_f$, such that the final ΔV is calculated as:

$$\Delta\hat{V}_f = \mathrm{v}_f - \hat{\mathrm{v}}(t_f) \tag{4.51}$$

Another option is to let the spacecraft enter the final trajectory prior to t_f such that it will coast through to x_f at the specified time. In this case it is necessary to determine the final waypoint and its associated velocity based on the intended coast time, here referred to as t_c. The final state can be back propagated to this derived state, x^*, using the discrete CW equations and a negative transfer time.

$$\mathrm{x}^* = \hat{\Phi}\left(-\Delta t_c\right)\mathrm{x}_f \tag{4.52}$$

where:

$$\mathrm{x}^* = \begin{bmatrix} R^* \\ V^* \end{bmatrix}^{\mathrm{LVLH}} \tag{4.53}$$

The derived waypoint, R^*, is targeted using the waypoint targeting method described above. The derived velocity, V^*, is used to determine the final velocity matching maneuver.

$$\Delta\hat{V}_f = V^* - \hat{\mathrm{v}}\left(t_{f-c}\right) \tag{4.54}$$

where $t_{f-c} = t_f - t_c$.

The two velocity matching methods are equal for $t_c = 0$ and can be simplified to the following linear function of $\hat{\mathrm{x}}$:

$$\Delta\hat{V}_k = \begin{bmatrix} 0_{3\times3} \\ -I_{3\times3} \end{bmatrix} \hat{\mathrm{x}}_k^{-c} + V_k^* \tag{4.55}$$

such that:

$$\hat{G}_{\mathrm{vm}} = \begin{bmatrix} 0_{3\times3} \\ -I_{3\times3} \end{bmatrix} \tag{4.56}$$

$$\hat{D}_{\mathrm{vm}} = \begin{cases} \mathrm{v}_f, & (t_c = 0) \\ V^*, & (t_c \neq 0) \end{cases} \tag{4.57}$$

We have now derived all of the necessary equation to perform a nonlinear Monte Carlo analysis of the rendezvous problem. All of the equations were linear with the exception of the Kalman gain used in the navigation state update, and the camera-based angles-only sensor measurements which required us to linearize the measurement function to generate the measurement sensitivity matrix, $\hat{H}_j^{cam}$. We will now take these equations and develop the LinCov model used for subsequent analysis.

4.7 Augmented State and Covariance

To develop the linear covariance model we begin by defining an augmented state vector consisting of both the true and estimated state deviations from the nominal reference trajectory:

$$\mathrm{X} = \begin{bmatrix} \mathrm{x} - \bar{\mathrm{x}} \\ \hat{\mathrm{x}} - \bar{\mathrm{x}} \end{bmatrix} = \begin{bmatrix} \delta\mathrm{x} \\ \delta\hat{\mathrm{x}} \end{bmatrix} = \begin{bmatrix} \delta\mathrm{r} \\ \delta\mathrm{v} \\ \delta\hat{\mathrm{r}} \\ \delta\hat{\mathrm{v}} \end{bmatrix}_{12\times 1}^{\mathrm{LVLH}} \tag{4.58}$$

Since the true and estimated states consist of the same variables, the mapping function m_x is equal to the identity matrix and can be ignored.

The mean and covariance of the augmented covariance are then given as:

$$E\left[\mathrm{X}\right] = 0_{12\times 1} \tag{4.59}$$

$$E\left[\mathrm{X}\mathrm{X}^T\right] = C_{12\times 12} \tag{4.60}$$

4.8 Linear Covariance Dynamic Model

The augmented state is propagated as:

$$\mathrm{X}(t_{i+1}) = \mathcal{F}_i \mathrm{X}(t_i) + \mathcal{W} w_{d,i} \tag{4.61}$$

where:

$$\mathcal{F}_i = \begin{bmatrix} \Phi(t_{i+1}, t_i) & 0_{6\times 6} \\ 0_{6\times 6} & \hat{\Phi}(t_{i+1}, t_i) \end{bmatrix} \tag{4.62}$$

$$\mathcal{W} = \begin{bmatrix} I_{6\times6} & 0_{6\times6} \\ 0_{6\times6} & 0_{6\times6} \end{bmatrix} \tag{4.63}$$

Using these equations, the augmented state covariance can be propagated as:

$$C(t_{i+1}) = \mathcal{F}_i C(t_i) \mathcal{F}_i^T + \mathcal{W} Q_d \mathcal{W}^T \tag{4.64}$$

4.9 Linear Covariance Measurement Update Model

When either a lidar or camera measurement is available, the augmented state is updated as:

$$\mathrm{X}(t_j^+) = \mathcal{A}_j \mathrm{X}(t_j^-) + \mathcal{B}_j \nu_j \tag{4.65}$$

where:

$$\mathcal{A}_j = \begin{bmatrix} I_{6\times6} & 0_{6\times6} \\ \hat{K}_j H_j & I - \hat{K}_j \hat{H}_j \end{bmatrix} \tag{4.66}$$

$$\mathcal{B}_j = \begin{bmatrix} 0_{6\times6} \\ \hat{K}_j \end{bmatrix} \tag{4.67}$$

We note here that $H_j = \hat{H}_j$ since the true sensor models match the navigation sensor models.

The augmented state covariance is updated as:

$$C(t_j^+) = \mathcal{A}_j C(t_j^-) \mathcal{A}_j^T + \mathcal{B}_j R_\nu \mathcal{B}_j^T \tag{4.68}$$

4.10 Linear Covariance Maneuver Correction Model

The augmented state is corrected whenever an impulsive control maneuver is commanded according to the following equation:

$$\mathrm{X}(t_k^{+c}) = \mathcal{D}_k \mathrm{X}(t_k^{-c}) + \mathcal{N} \eta_k \tag{4.69}$$

where:

$$\mathcal{D}_k = \begin{bmatrix} I_{6\times 6} & \hat{B}\hat{G}_k \\ 0_{6\times 6} & I + \hat{B}\hat{G}_k \end{bmatrix} \tag{4.70}$$

$$\mathcal{N} = \begin{bmatrix} \hat{B} \\ 0_{6\times 6} \end{bmatrix} \tag{4.71}$$

Finally, the augmented state covariance is corrected for each maneuver as:

$$C(t_k^{+c}) = \mathcal{D}_k C(t_k^{-c}) \mathcal{D}_k^T + \mathcal{N} S_{\Delta V} \mathcal{N}^T \tag{4.72}$$

4.11 RPO Performance Statistics

To assess the performance of the closed-loop RPO system, we will evaluate three key performance metric: the true position dispersion, the true range dispersion, and the ΔV dispersions.

4.11.1 True Position Dispersion

The true position dispersion indicates the dispersion (or spread) of the possible true trajectories away from the nominal trajectory. This dispersion is given as:

$$D(t_i) = E\left[(\delta \mathrm{r})(\delta \mathrm{r})^T\right] \tag{4.73}$$

$$D(t_i) = \begin{bmatrix} I_{3\times 3} & 0_{3\times 9} \end{bmatrix} C(t_i) \begin{bmatrix} I_{3\times 3} & 0_{3\times 9} \end{bmatrix}^T \tag{4.74}$$

The true position dispersion at any instant describes an error ellipsoid centered on the nominal trajectory as depicted in Figure 4.6. The volume defined by the ellipsoid represent the one sigma (1σ) boundary of the three-dimensional Gaussian distribution. The three-dimensional ellipsoid can also be projected into each plane as an two-dimensional ellipse as depicted in Figure 4.7. The values can be scaled to determine the 1, 2, and 3σ boundaries of the two-dimensional Gaussian distribution.

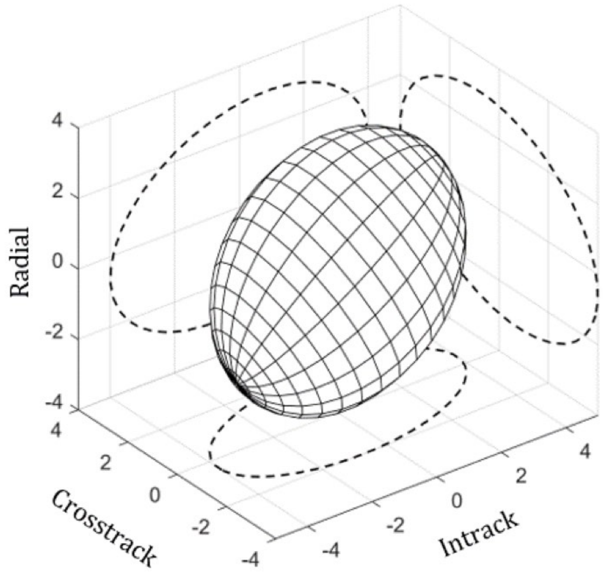

Fig. 4.6: True position dispersion ellipsoid with orthogonal ellipse projections.

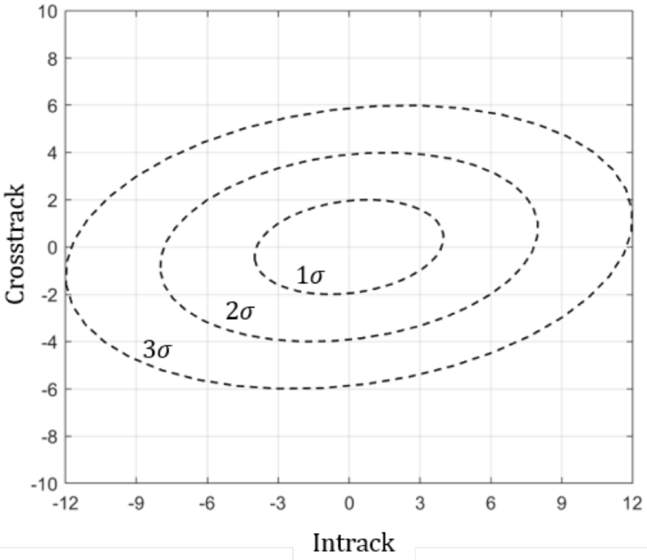

Fig. 4.7: True position dispersion ellipses in the Intrack-Crosstrack plane.

While the true position dispersion of the trajectory at each time step will be important when we look at safety constraints, for now we will be primarily concerned with the true position dispersion at the final time:

$$D(t_f) = \begin{bmatrix} I_{3\times3} & 0_{3\times9} \end{bmatrix} C(t_f) \begin{bmatrix} I_{3\times3} & 0_{3\times9} \end{bmatrix}^T \tag{4.75}$$

The final true position covariance, a 3 × 3 matrix, can be reduced to a single value for constraint verification. The matrix, in the LVLH frame, is given as:

$$D(t_f) = \begin{bmatrix} D_{xx} & D_{xy} & D_{xz} \\ D_{xy} & D_{yy} & D_{yz} \\ D_{xz} & D_{yz} & D_{zz} \end{bmatrix} \tag{4.76}$$

The square root of the largest diagonal element represents the largest uncertainty of the Cartesian LVLH components. This single values can be used to compare against a cubic constraint (i.e. the final true dispersion must lie within a cube centered on the final position).

$$S_{D_f} = \sqrt{\max(D_{xx},\ D_{yy},\ D_{zz})} \tag{4.77}$$

To evaluate the true final dispersion against a spherical constraint, the covariance matrix can be converted into its principle axes:

$$D'(t_f) = T_{eig} D(t_f) T_{eig}^T = \begin{bmatrix} D'_{xx} & 0 & 0 \\ 0 & D'_{yy} & 0 \\ 0 & 0 & D'_{zz} \end{bmatrix} \tag{4.78}$$

where T_{eig} is a rotation matrix defined by the eigenvectors of the matrix D, rotating the original matrix from the LVLH frame to the ellipsoid's principal coordinate frame. The single value metric then represents the maximum 1σ distance from the final position.

$$S_{D'_f} = \sqrt{\max(D'_{xx},\ D'_{yy},\ D'_{zz})} \tag{4.79}$$

4.11.2 True Range Dispersion

The second key metric is the range true range dispersion and is closely related to the true position dispersion. The true range dispersion is calculated by projecting the true position dispersion into the line-of-sight direction between the vehicle and the RSO, yielding a single dispersion value. Mapped into this line-of-sight direction, this dispersion provides a metric for determining how close the vehicle might be to the RSO as opposed to just knowing the nominal range, $\|\bar{\mathrm{r}}\|$.

The true range dispersion at any given time t_i is given as:

$$D_R(t_i) = \hat{\imath}_R^T D(t_i) \hat{\imath}_R = \sigma_R^2 \tag{4.80}$$

$$D_R(t_i) = \hat{\imath}_R^T \begin{bmatrix} I_{3\times3} & 0_{3\times9} \end{bmatrix} C(t_i) \begin{bmatrix} I_{3\times3} & 0_{3\times9} \end{bmatrix}^T \hat{\imath}_R \tag{4.81}$$

where σ_R^2 is the standard deviation of the range uncertainty and $\hat{\imath}_R$ is a unit vector along the line-of-sight vector between the vehicle and the RSO:

$$\hat{\imath}_R = \frac{\bar{\mathrm{r}}(t_i)}{\|\bar{\mathrm{r}}(t_i)\|} \tag{4.82}$$

4.11.3 Navigation Error Covariance

The true navigation covariance represents the true error in the onboard navigation, an indication of how accurately the system can estimate the true states. The true navigation covariance is calculated as:

$$P_{\text{true}}(t) = \begin{bmatrix} -I_{3\times3} & I_{3\times3} \end{bmatrix} C(t) \begin{bmatrix} -I_{3\times3} & I_{3\times3} \end{bmatrix}^T \tag{4.83}$$

4.11.4 True ΔV Dispersion

The final key metric used to evaluate the closed-loop RPO system performance is the amount of ΔV that may be consumed. The ΔV dispersion for each maneuver is based on the range of estimated positions in the navigation filter at the time the maneuver is calculated and is given as:

$$D_{\Delta V_k} = E\left[(\delta \Delta V_k)(\delta \Delta V_k)^T\right] \tag{4.84}$$

$$D_{\Delta V_k} = \begin{bmatrix} 0_{3\times 6} & \hat{G} \end{bmatrix} C_k^{-c} \begin{bmatrix} 0_{3\times 6} & \hat{G} \end{bmatrix}^T + S_{\Delta V_k} \tag{4.85}$$

The 3σ fuel consumption is determined by adding the nominal ΔV to the calculated dispersion:

$$\Delta V_k^{3\sigma} = \left\|\Delta \bar{V}_k\right\| + 3\sqrt{Tr[D_{\Delta V_k}]} \tag{4.86}$$

where $\left\|\Delta \bar{V}\right\|$ is the 2-norm of the nominal ΔV vector and $Tr[D_{\Delta V}]$ is the trace of the ΔV dispersion matrix. The trace of an n-dimensional, square matrix is given as:

$$Tr[A] = \sum_{i=1}^{n} a_{ii} = a_{11} + a_{22} + \cdots + a_{nn} \tag{4.87}$$

The total 3σ ΔV consumption is gained by simply summing the values for each individual maneuver:

$$\Delta V^{3\sigma} = \sum_{k=1}^{n} \Delta V_k^{3\sigma} \tag{4.88}$$

The evaluation of the 3σ fuel consumption is important since maneuvers may require the same nominal ΔV but result in vastly different ΔV dispersions due to position dispersions. When optimizing a trajectory is becomes valuable to select a maneuver with small dispersions even if the nominal value is slightly larger than other options.

4.12 Nominal Trajectory

To initialize the linear covariance simulation we must also define the nominal trajectory about which the performance will be analyzed. This trajectory begins at the initial state, $\bar{\mathrm{x}}_0$, and terminates t_f seconds later at the final state, $\bar{\mathrm{x}}_f$. Between these states, the trajectory may also pass through a number (n_{wpt}) of intermediate waypoints, R^*, with their associated transfer times, t^*. Using the relative motion dynamics derived in Section 2.1.2 and the guidance methods described in Section 4.6, these inputs completely define the nominal relative trajectory to be followed.

A minimum of two maneuvers are required for all transfers (one maneuver to achieve a position on the final trajectory and a second to match the velocity of the final trajectory). Intermediate waypoints are given as:

$$R^* = \begin{bmatrix} R_1^* & R_2^* & \cdots & R_{n_{\text{wpt}}}^* \end{bmatrix} \tag{4.89}$$

where $R^* \in \mathbb{R}^{3 \times n_{\text{wpt}}}$. Each waypoint's transfer time is given as:

$$\Delta t^* = \begin{bmatrix} \Delta t_1^* & \Delta t_2^* & \cdots & \Delta t_{n_{\text{wpt}}}^* \end{bmatrix} \tag{4.90}$$

where $\Delta t^* \in \mathbb{R}^{1 \times n_{\text{wpt}}}$. The trajectory includes two additional transfer times, Δt_0 and $\Delta t_{n_{\text{wpt}+1}}$. Δt_0 is the initial coast time before executing the first maneuver, and $\Delta t_{n_{\text{wpt}+1}}$ is the transfer time to reach the final derived waypoint R_c^* as defined by Equation (4.52). The full set of transfer times are given as:

$$\Delta t = \begin{bmatrix} \Delta t_0 & \Delta t_1^* & \Delta t_2^* & \cdots & \Delta t_{n_{\text{wpt}}}^* & \Delta t_{n_{\text{wpt}+1}} \end{bmatrix} \tag{4.91}$$

The sum of the transfer times must be less than or equal to the defined final time:

$$\sum_{i=1}^{n_{\text{wpt}}+1} \Delta t_i \leq t_f \tag{4.92}$$

If the sum of these times is less than t_f then the final coast time to reach $\bar{\mathrm{x}}_f$ is calculated as:

$$\Delta t_c = t_f - \sum_{i=1}^{n_{\text{wpt}}+1} \Delta t_i \tag{4.93}$$

An example nominal trajectory for a single waypoint ($n_{\mathbf{wpt}} = 1$) is shown in Figure 4.8 and defined in Tables 4.1 and 4.2. The example trajectory's initial state begins at relative perigee of a 500x1000 km NMC and transfers to a 250x500 km NMC, finishing again at relative perigee. The spacecraft begins at the initial state, $\bar{x}(t_0)$ and coasts for $\Delta t_0 = 800$ seconds before the first maneuver, ΔV_1, is executed. This first maneuver is calculated using

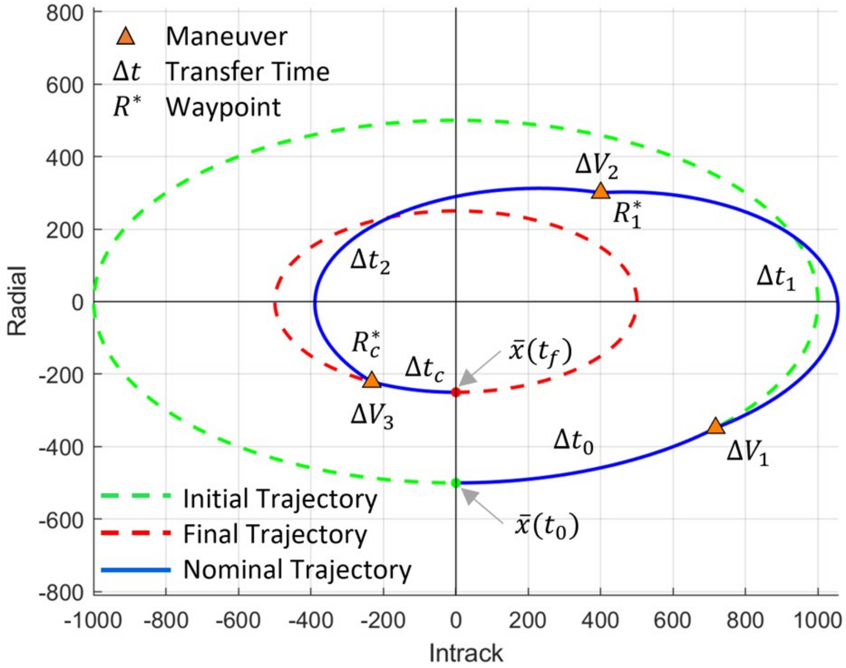

Fig. 4.8: Example nominal trajectory for a single waypoint transfer in the Intrack-Crosstrack plane.

the waypoint guidance algorithm to reach the first commanded waypoint, R_1^*, located at $(400, 300, 0)$ m with a transfer time of $\Delta t_1 = 2500$ seconds. The spacecraft then coasts for Δt_1 seconds to reach this waypoint (and location of the second maneuver). The final waypoint, R_c^*, is located on the final trajectory and is determined by back propagating the final state, $\bar{x}(t_f)$, by Δt_c seconds. This coast time is calculated as:

$$\Delta t_c = t_f - \Delta t_0 - \Delta t_1 - \Delta t_2$$

This derived waypoint is targeted using the waypoint guidance algorithm to generate ΔV_2 with a transfer time of $\Delta t_2 = 2500$ seconds. The final maneuver, ΔV_3, is executed upon arrival at R_c^* to match the final trajectory velocity. The spacecraft then coasts for the final $\Delta t_c = 483$ seconds to reach the final state, $\bar{x}(t_f)$, at the final time $t_f = 6283$ seconds.

It is important to note for later analysis that the spacecraft does not necessarily target the desired final state, $\bar{\mathrm{x}}(t_f)$, as a waypoint unless $\Delta t_1 + \Delta t_2 + \Delta t_3 = t_f$.

4.13 Linear Covariance Summary

With the nominal trajectory, the linear covariance simulation is initialized with the initial navigation covariance, $\hat{P}_0$:

$$\hat{P}_0 = \begin{bmatrix} \hat{P}_{\text{rr}0} & \hat{P}_{\text{rv}0}^T \\ \hat{P}_{\text{rv}0} & \hat{P}_{\text{vv}0} \end{bmatrix}_{6\times 6} \tag{4.94}$$

The initial augmented covariance matrix is defined as:

$$C_0 = \begin{bmatrix} \hat{P}_0 & 0_{6\times 6} \\ 0_{6\times 6} & 0_{6\times 6} \end{bmatrix}_{12\times 12} \tag{4.95}$$

This initialization indicates that the true position dispersions match those of the navigation uncertainty and that the true navigation dispersion is zero. While this may not always be the case, it is a valid assumption since the navigation covariance should be set to the best estimate of the true uncertainty.

The estimated state covariance and the augmented state covariance are propagated forward using Equations (4.33) and (4.64):

$$\hat{P}(t_{i+1}) = \left[\hat{\Phi}(t_{i+1}, t_i)\right] \hat{P}(t_i) \left[\hat{\Phi}(t_{i+1}, t_i)\right]^T + \hat{Q}_d$$

$$C(t_{i+1}) = \mathcal{F} C(t_i) \mathcal{F}^T + \mathcal{G} Q_d \mathcal{G}^T$$

When navigation measurement becomes available, the estimated state covariance is updated according to Equations (4.36) or (4.40):

$$\hat{P}_j^+ = \left[I - \hat{K}_j \hat{H}\right] \hat{P}_j^- \left[I - \hat{K}_j \hat{H}\right]^T + \hat{K}_j R_\nu \hat{K}_j^T$$

and the augmented state covariance is updated according to Equation (4.68):

$$C(t_j^+) = \mathcal{A} C(t_j^-) \mathcal{A}^T + \mathcal{B} R_\nu \mathcal{B}^T$$

When maneuvers are executed, the estimated state covariance and augmented state covariance are corrected according to Equations (4.44) and (4.72):

$$\hat{P}_k^{+c} = \hat{P}_k^{-c} + \hat{B} S_{\Delta V} \hat{B}^T$$

$$C(t_k^{+c}) = \mathcal{D} C(t_k^{-c}) \mathcal{D}^T + \mathcal{N} S_{\Delta V} \mathcal{N}^T$$

These equations, along with the nominal trajectory and relevant noise parameters, are all that is necessary to run the closed-loop RPO, linear covariance simulation. In the following chapters we will use this simulation to analyze trajectory planning techniques (the selection of appropriate waypoints and transfer times) to achieve mission objectives.

4.14 Example Scenario

In order to demonstrate the performance of the RPO linear covariance model developed in this chapter, we will include the results for a single run based on the nominal trajectory shown in Section 4.12. The run will be for an angles-only system that does not include Lidar.

The inputs to define the nominal trajectory are given in Table 4.1. The waypoints and transfer times are given in Table 4.2. The noise properties for are given in Table 4.3. Finally, the camera constraints are given in Table 4.4. Results for the scenario run are given in Figures 4.9-4.14.

Table 4.1: Nominal trajectory inputs.

Parameter	Value	Units
$\hat{\mathrm{r}}(t_0)$	$[-500,\ 0,\ 0]^T$	m
$\hat{\mathrm{v}}(t_0)$	$[0,\ 1,\ 0]^T$	m/s
$\hat{\mathrm{r}}(t_f)$	$[-250,\ 0,\ 0]^T$	m
$\hat{\mathrm{v}}(t_f)$	$[0,\ 0.5,\ 0]^T$	m/s
ω	0.001	rad/s
t_f	$6283\ (= T)$	s
$\hat{s}(t_0)$	$[0,1,0]^T$	-

Table 4.2: Waypoint and transfer time inputs.

Parameter	Value	Units
R_1^*	$[300,\ 400,\ 0]^T$	m
Δt_0	800	s
Δt_1	2500	s
Δt_2	2500	s
R_c^*	$[-221.4, -232.2, 0]^T$	m
Δt_c	483	s

Table 4.3: Scenario noise parameters.

Parameter	Value	Units
$\hat{P}_{\mathrm{rr}0}$	$(5)^2 I_{3\times3}$	m^2
$\hat{P}_{\mathrm{vv}0}$	$(0.01)^2 I_{3\times3}$	m^2/s^2
$\hat{P}_{\mathrm{rv}0}$	$0_{3\times3}$	m^2/s
Q_w	$1.11 \times 10^{-19} I_{3\times3}$	m^2/s^3
$S_{\Delta V}$	$0.02\Delta\bar{V} I_{3\times3}$	m^2/s^2
R_ν^{AON}	$1.11 \times 10^{-7} I_{3\times3}$	rad^2
R_ν^{lidar}	n/a	n/a

Table 4.4: Camera constraint parameters.

Parameter	Value	Units
$\theta_{\oplus}^{lim}$	10	deg
$\theta_{\odot}^{lim}$	30	deg

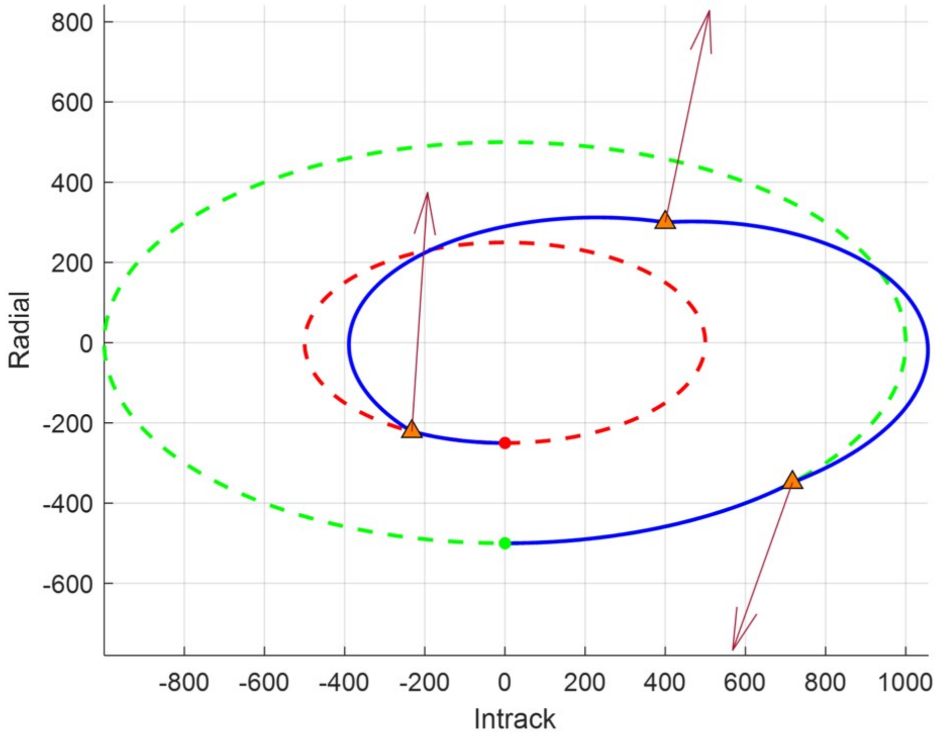

Fig. 4.9: Nominal trajectory with maneuver directions. Initial trajectory is green. Final trajectory is red.

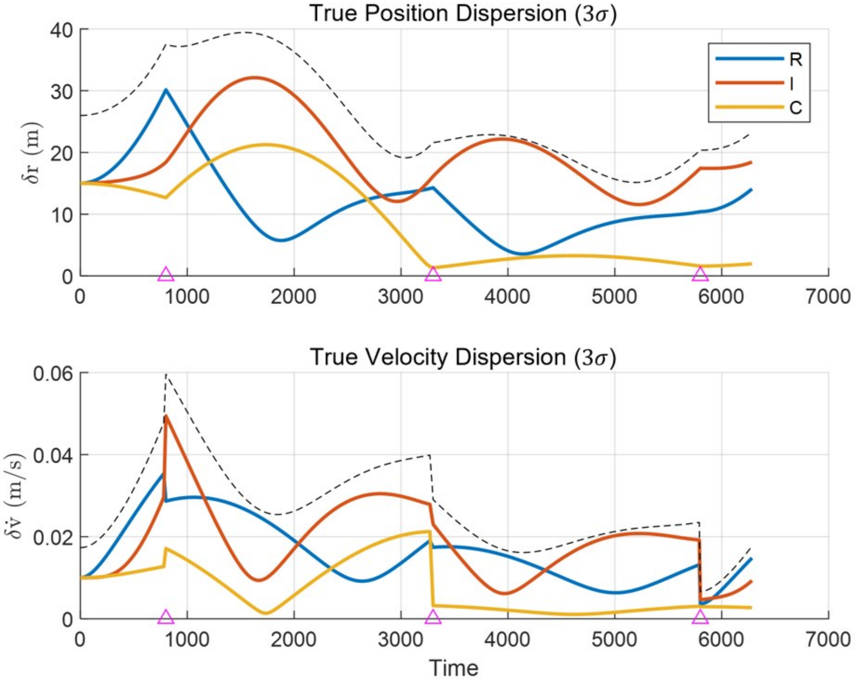

Fig. 4.10: True position and velocity dispersions. Maneuvers times are marked by the triangles. Dashed lines represent the RSS of the three values.

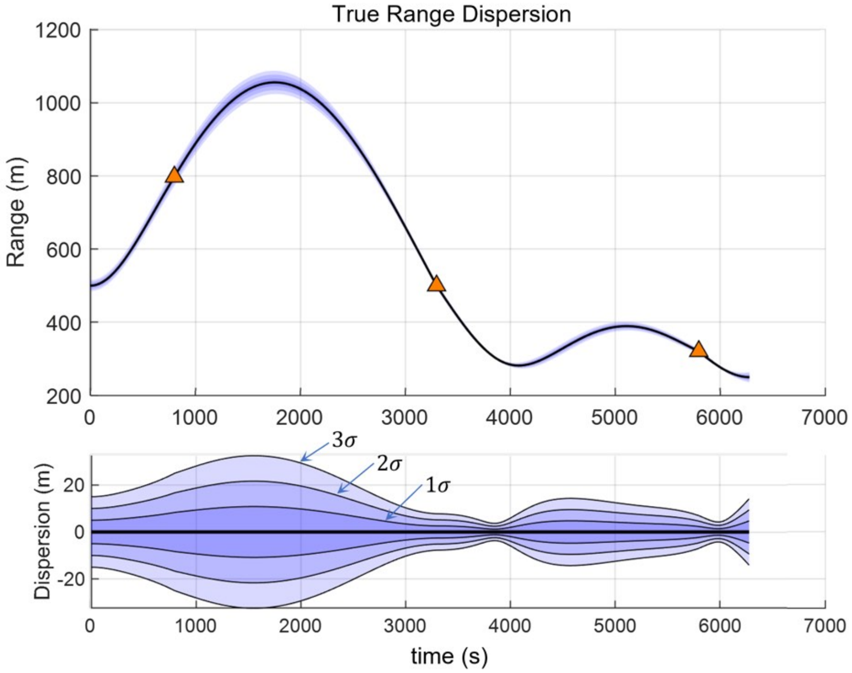

Fig. 4.11: True range dispersions. Top plot shows the nominal range with the 3σ range dispersion overlay. The bottom plot shows the range dispersion.

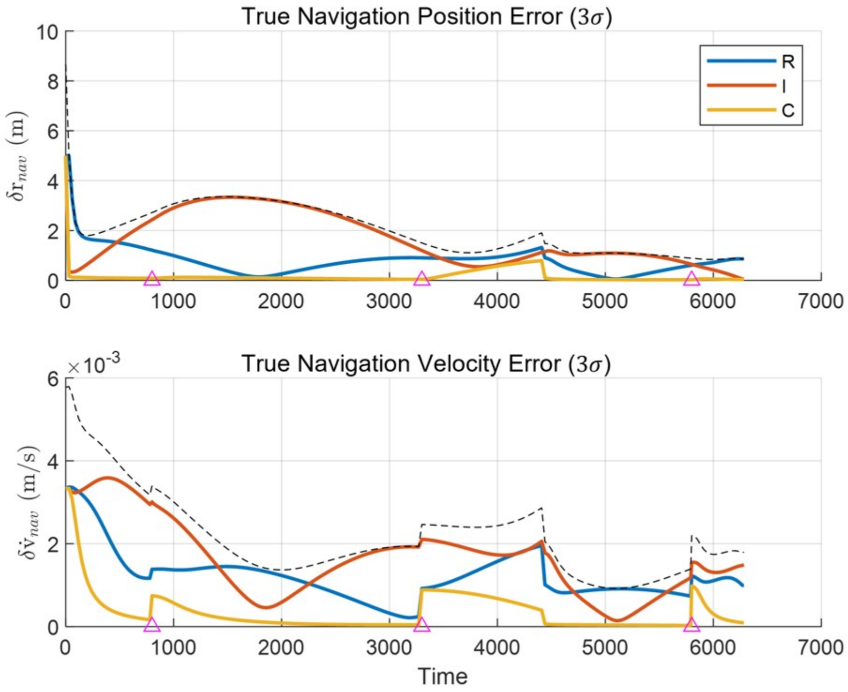

Fig. 4.12: True navigation position and velocity error covariance. Maneuvers times are marked by the triangles. Dashed lines represent the RSS of the three values.

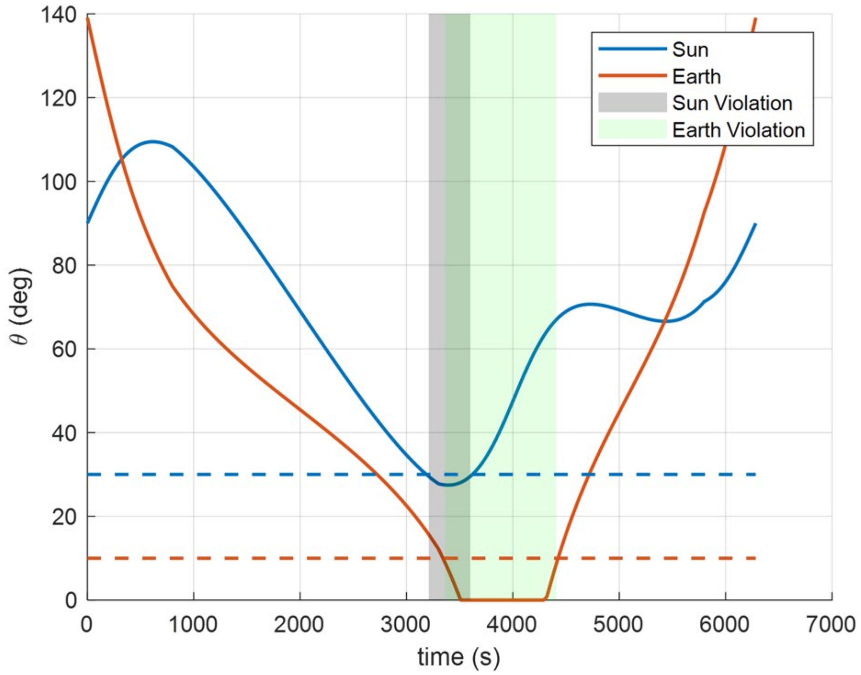

Fig. 4.13: Sun and earth angles. The system system is unable to take images due to both sun and earth angular violations. Dashed lines are the exclusion angle limits $\theta_{\oplus}^{lim}$ and $\theta_{\odot}^{lim}$.

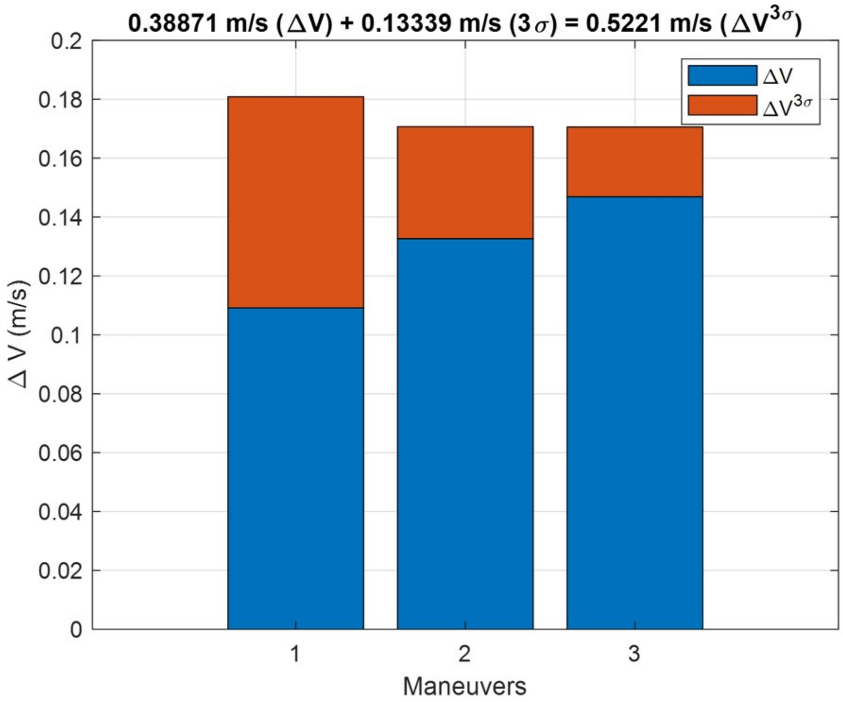

Fig. 4.14: Nominal ΔV (blue) and 3σ dispersions (orange) for each maneuver. Total $\Delta\bar{V}^{3\sigma} = 0.5221$ m/s.

CHAPTER 5

DETERMINISTIC VS CONSTRAINED TWO-IMPULSE RELATIVE PATH PLANNING

In and effort to better understand the trajectory planning problem, this chapter addresses the differences between a deterministic approach and the constrained, stochastic path planning approach developed in Chapter 4 when evaluating two-impulse maneuver sequences for RPO path planning. For the purpose of this analysis, the deterministic approach is only interested in minimizing the ΔV required to achieve the transfer. In contrast, the constrained, stochastic approach will include dynamics, navigation, and maneuver errors to determine a minimum 3σ fuel dispersion ($\Delta V^{3\sigma}$) when constrained by a final true position dispersion requirement. The results for all possible transfer options will be analyzed together to provide the desired insight and to inform future optimization efforts.

Two scenarios are evaluated. Scenario 1 is an NMC resizing problem and scenario 2 is a V-bar hop (where the spacecraft moves between two stationary position along the intrack axis). V-bar is another term for the intrack axis since it is aligned with the velocity vector. It is important to note here that two-impulse scenarios were selected specifically because the problem reduces to just two variables, the times of the two maneuvers. This simplicity allows us to easily visualize the otherwise complex results in two-dimensional plots to gain the desired insight. This approach should not be mistaken for a preferred or optimal transfer scenario.

5.1 NMC Resizing Scenario

The first transfer scenario used for this analysis is a simple NMC resizing problem between two planar closed-path relative orbits as originally described in Section 4.12 and analyzed in Section 4.14. The transfer has fixed initial and final states and a fixed transfer time to move between them. The transfer is defined by two time intervals, 1) the coast time in the initial NMC trajectory before executing the first maneuver of the transfer, Δt_0, and 2) the transfer time, Δt_1, to reach a waypoint located on the final NMC trajectory, R_c^*, where the second maneuver is executed. The scenario finishes with a second coast period, Δt_c, in the final NMC to reach the desired final state at the specified final time ($t_f = \Delta t_0 + \Delta t_1 + \Delta t_c$). Given Δt_0 and Δt_1, the waypoint location of the second maneuver can be determined by back propagating the final state by $-\Delta t_c$ to the corresponding position and velocity. This position is the targeted waypoint at maneuver #1. Maneuver #2 is applied to match the desired velocity. The fixed inputs for the scenario are listed in Table 5.1 and a representative transfer is shown in Figure 5.1. The total transfer time is equal to two times the orbital period, T.

The nominal ΔV for all feasible combinations of Δt_0 and Δt_1 is shown in Figure 5.2 in the form of a surface map. This data and subsequent data sets for scenarios in this chapter were generated by discretizing the transfer time values into 30 second intervals for a total of 419 steps for both Δt_0 and Δt_1. The upper right half of the surface map, colored dark gray, represents the areas outside the scenario where $\Delta t_0 + \Delta t_1 > t_f$.

Of particular interest in this scenario is that there are repeating, non-unique global minima representing four unique transfers between the two trajectories. These minima are identified with numbered markers and their locations and ΔV are given in Table 5.2. Numbers indicated with an asterisk are the same as the original label but delayed by one

Table 5.1: Two-impulse NMC resizing parameters.

Parameter	Value	Units
$\bar{r}(t_0)$	$\begin{bmatrix}-500 & 0 & 0\end{bmatrix}^T$	m
$\bar{v}(t_0)$	$\begin{bmatrix}0 & 1 & 0\end{bmatrix}^T$	m/s
$\bar{r}(t_f)$	$\begin{bmatrix}-250 & 0 & 0\end{bmatrix}^T$	m
$\bar{v}(t_f)$	$\begin{bmatrix}0 & 0.5 & 0\end{bmatrix}^T$	m/s
t_f	12,566 (=2T)	s
ω	0.001	rad/s

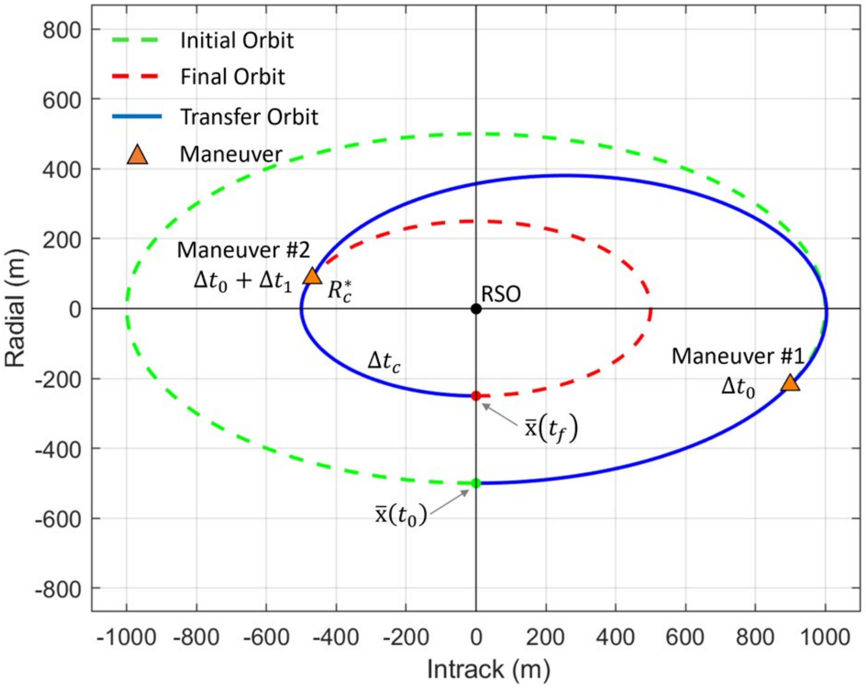

Fig. 5.1: Representative two-impulse transfer between planar NMC orbits.

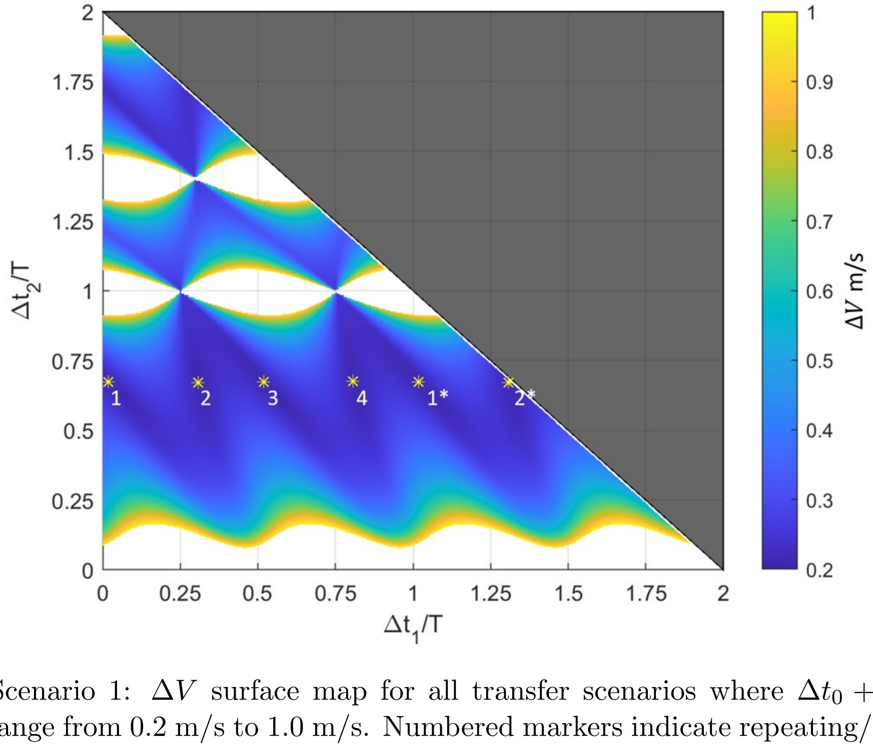

Fig. 5.2: Scenario 1: ΔV surface map for all transfer scenarios where $\Delta t_0 + \Delta t_1 \leq t_f$. Contours range from 0.2 m/s to 1.0 m/s. Numbered markers indicate repeating/non-unique global minima.

orbit. The reported minima are approximate since the values were found from the discretized time steps with no effort to refine the solutions. The four transfers are shown in Figure 5.3.

The first optimal maneuver (marker 1) occurs shortly after relative perigee ($\Delta t_0 \approx 0.02T$) with the subsequent transfers occurring at alternating increments of $0.29T$ and $0.21T$. Each waypoint transfer (Δt_1) is approximately $0.67T$. It is interesting to note that the four transfers group into two pairs that represent very similar paths. Also of note are the long zigzagging troughs that connect the four minima. These continuous troughs deviate from the minima (0.215 m/s) by no more than 15% indicating a wide range of possible transfers with near optimal ΔV solutions.

Table 5.2: Optimal transfer data for the four global minima.

Transfer	$\Delta t_0/T$	$\Delta t_1/T$	ΔV m/s
1	0.01893	0.67335	0.21502
2	0.30727	0.67126	0.21502
3	0.51929	0.67390	0.21502
4	0.80665	0.67474	0.21502
1*	1.01893	0.67444	0.21502
2*	1.30727	0.67126	0.21502

While these four optimal transfer options may be identical for a deterministic ΔV analysis, the same scenario may show significant differences between the otherwise identical transfers when evaluated for a closed-loop stochastic system. In addition to nominal ΔV, analysis of such a system would also provide statistical fuel usage and final true position dispersions for each combination of transfer times that. This data, in turn, could be used to determine optimal trajectories under any prescribed constraint. The time required to perform this analysis using the Monte Carlo analysis technique developed in Chapter 4, however, would be prohibitive. A total of 87,781 function calls were required to generate the ΔV transfer contour map in Figure 5.2. Running 850 Monte Carlo runs for each transfer case (yielding a 10% confidence interval) would require a total of 74,613,850 evaluations. At even 0.01 seconds per run (the approximate time currently required to run the simulation),

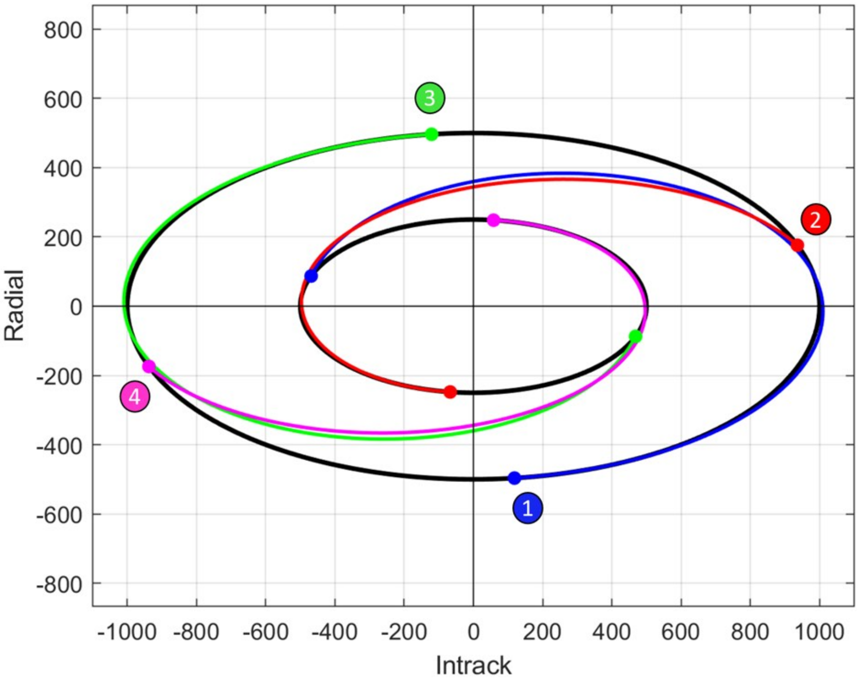

Fig. 5.3: Transfer orbits corresponding to the four global minima. Numbered markers indicate the initial burn locations.

it would take more than eight-and-a-half days to complete the full analysis! Any analysis taking even a tenth of this time to complete would not be conducive to an operational assessment.

In order to efficiently analyze the statistical performance of the system for each transfer case, we will implement the closed-loop linear covariance analysis technique derived in Chapter 4 for three unique cases. Each cases will include a high-accuracy lidar-based navigation system to determine the estimated states, but employ unique maneuver execution error valuations. Cases A and B include high and low-accuracy propulsion systems, respectively, with maneuver execution errors fixed at 1% and 5% (3σ) of the nominal maneuver. Case C has a fixed maneuver execution error independent of the nominal maneuver. The simulation parameters are given in Table 5.3 and the maneuver error parameters for each case are given in Table 5.4.

Table 5.3: Scenario parameters for the two-impulse linear covariance analysis.

Parameter	Value	Units
Measurement Update Interval: Δt_{lidar}	30	s
Initial Position Uncertainty: $C_0(rr)$ (3σ)	$(10)^2 I_{3\times3}$	m^2
Initial Velocity Uncertainty: $C_0(vv)$ (3σ)	$(0.02)^2 I_{3\times3}$	$\mathrm{m}^2/\mathrm{s}^2$
Dynamics Process Noise: Q_w (3σ)	$(10^{-7})^2 I_{3\times3}$	$\mathrm{m}^2/\mathrm{s}^3$
Lidar Measurement Covariance: R (3σ)	$(0.1)^2 I_{3\times3}$	m^2

Table 5.4: Maneuver error uncertainty for three scenarios.

Parameter	Case A	Case B	Case C
Maneuver Error: $S_{\Delta V}$ (3σ)	$(0.01\|\|\Delta\bar{V}\|\|)^2 I_{3\times3}$	$(0.05\|\|\Delta\bar{V}\|\|)^2 I_{3\times3}$	$(0.001 m/s)^2 I_{3\times3}$

5.1.1 LinCov Evaluation of Deterministic Results

The LinCov analysis was first used to evaluate the four optimal solutions identified by the deterministic analysis. The calculated three-sigma ΔV dispersion, $\Delta V^{3\sigma}$, and final true position dispersion, D_f, for each case are shown in Table 5.5. Accounting for the range of potential maneuvers and their associated errors, the ΔV dispersion shows an increase of fuel consumption between 10% and 40% (above the 0.21502 m/s nominal) for all three cases . The final true position dispersions, however, show a much larger variation with dispersions ranging between 4.4 and 296 meters. In general, the $\Delta V^{3\sigma}$ values increase as we move sequentially through the options (1 to 4 to 2*), while the final true position dispersions decrease.

Table 5.5: LinCov dispersion results for the non-constrained global optimal transfers for each scenario.

Transfer	Case A		Case B		Case C	
#	$\Delta V^{3\sigma}$ (m/s)	D_f (m)	$\Delta V^{3\sigma}$ (m/s)	D_f (m)	$\Delta V^{3\sigma}$ (m/s)	D_f (m)
1	0.23480	114.62	0.23714	149.48	0.24104	296.50
2	0.26088	81.08	0.26695	405.39	0.26315	241.52
3	0.27072	17.00	0.27119	84.98	0.27241	231.30
4	0.26533	49.29	0.27006	246.45	0.26704	146.85
1*	0.27150	4.42	0.27218	22.11	0.27371	67.71
2*	0.30620	10.32	0.30955	51.64	0.30740	30.46

5.1.2 LinCov Global Evaluation

The second step of the analysis was to evaluate the entire solution space using the LinCov model in an approach similar to that used in the deterministic evaluation. The final position dispersion and ΔV dispersion surface maps for Case A are shown in Figures 5.4 and 5.5. For Case B, the final position and ΔV dispersion surface maps are shown in Figures 5.6 and 5.7, and then results for Case C are shown in Figures 5.8 and 5.9. The final position dispersion contour values have been limited to a maximum of 100 m for all cases to better show the areas of interest. Full surface maps for the three cases are located at the end of this section. Additionally, the 25 m and 50 m contours from the true position dispersions are overlayed on the ΔV dispersions to allow for easier comparisons between the two metrics.

With the full statistical data for each case it is possible to determine the minimum $\Delta V^{3\sigma}$ given a maximum final true dispersion constraint. A global search of the data was performed for final true position dispersion constraints of 25 m and 50 m. The optimal solutions are indicated on their respective surface maps by the $\triangle$ symbol (in the color corresponding to the evaluated position dispersion contour). In the event that an optimal solution corresponds to a multi-revolution transfer ($\Delta t_1/T > 1$) the single revolution transfer ($\Delta t_1/T \leq 1$) was also determined and is indicated by the $\bigtriangledown$ symbol. This single revolution evaluation was performed since the previously determined deterministic optimal solutions all correspond to single revolution transfers and are of more practical interest. The results for each case are shown in Table 5.6. In addition to the $\Delta V^{3\sigma}$ values, the table includes the nominal ΔV at that location, the final position dispersion, and the calculated times as a fraction of the orbital period.

For this scenario it can be observed that the constrained optimal solutions were generally located near the non-constrained optimal solutions (assuming single revolution transfer options). This trend was even present in Case C when the final position dispersion constraints did not encompass the non-constrained optimal solutions. This would indicated that non-constrained solutions may serve as a decent initial guess when searching for con-

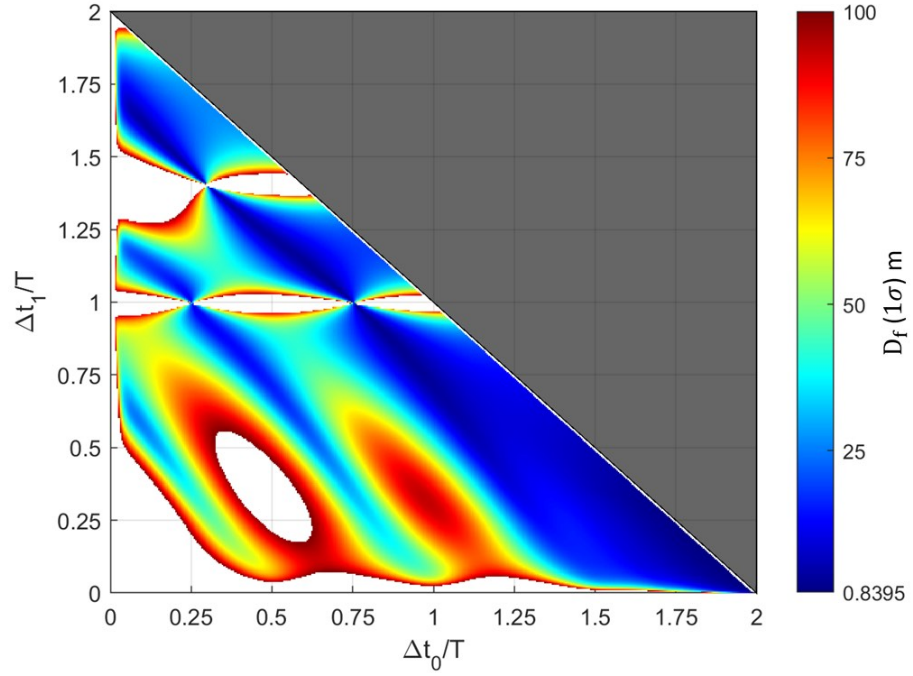

Fig. 5.4: Case A: Final position dispersion for all combinations of Δt_0 and Δt_1.

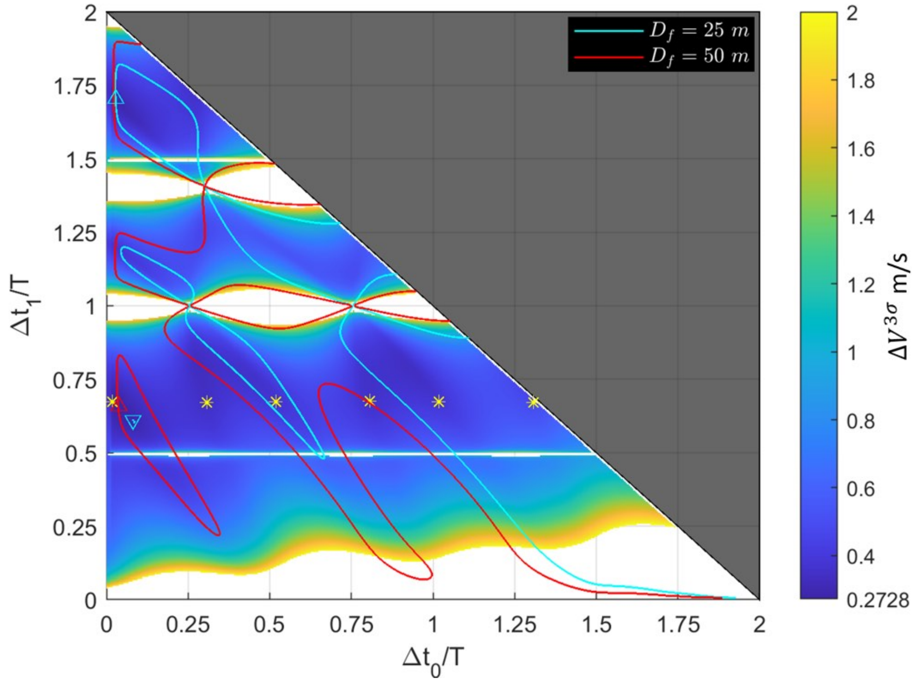

Fig. 5.5: Case A: ΔV dispersion for all combinations of Δt_0 and Δt_1 with overlayed final position dispersion contours.

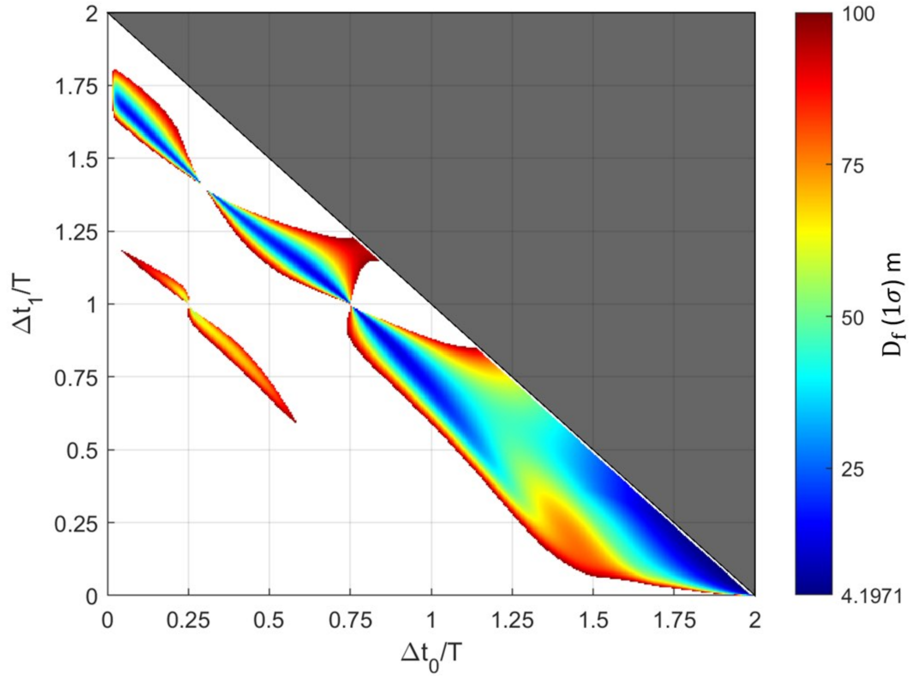

Fig. 5.6: Case B: Final position dispersion for all combinations of Δt_0 and Δt_1.

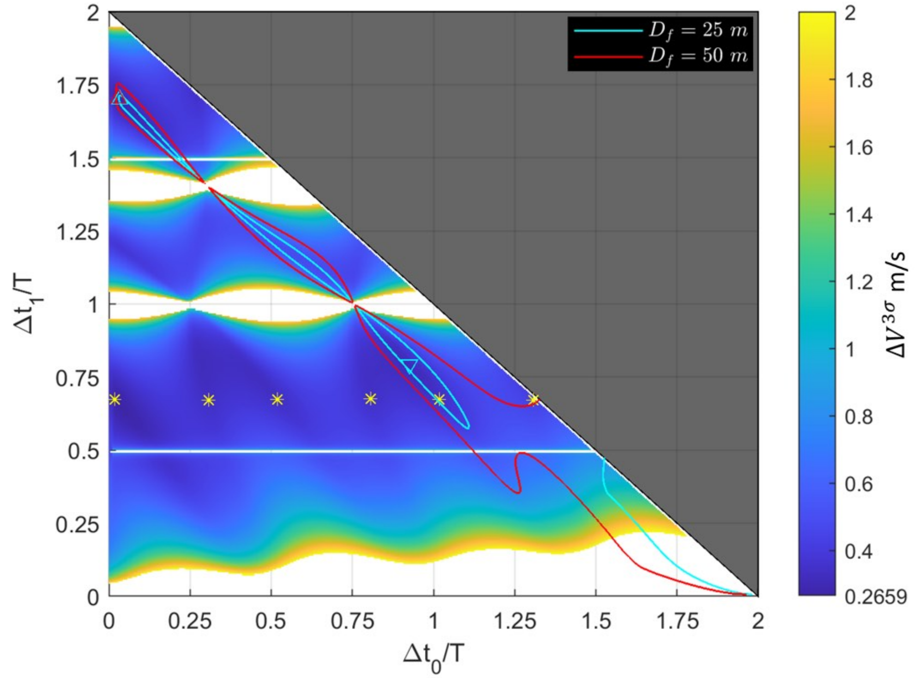

Fig. 5.7: Case B: ΔV dispersion for all combinations of Δt_0 and Δt_1 with overlayed final position dispersion contours.

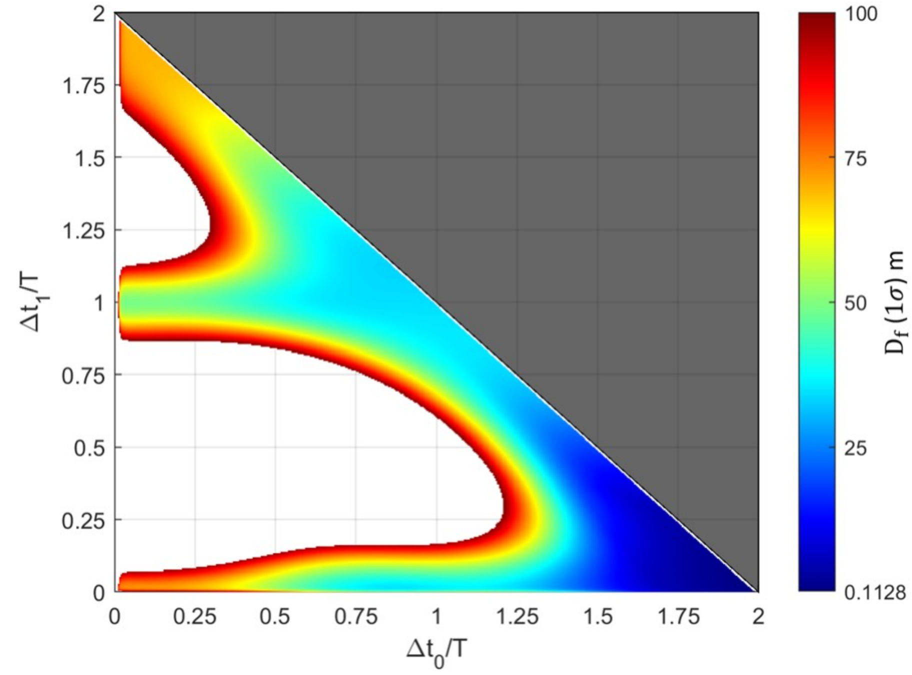

Fig. 5.8: Case C: Final position dispersion for all combinations of Δt_0 and Δt_1.

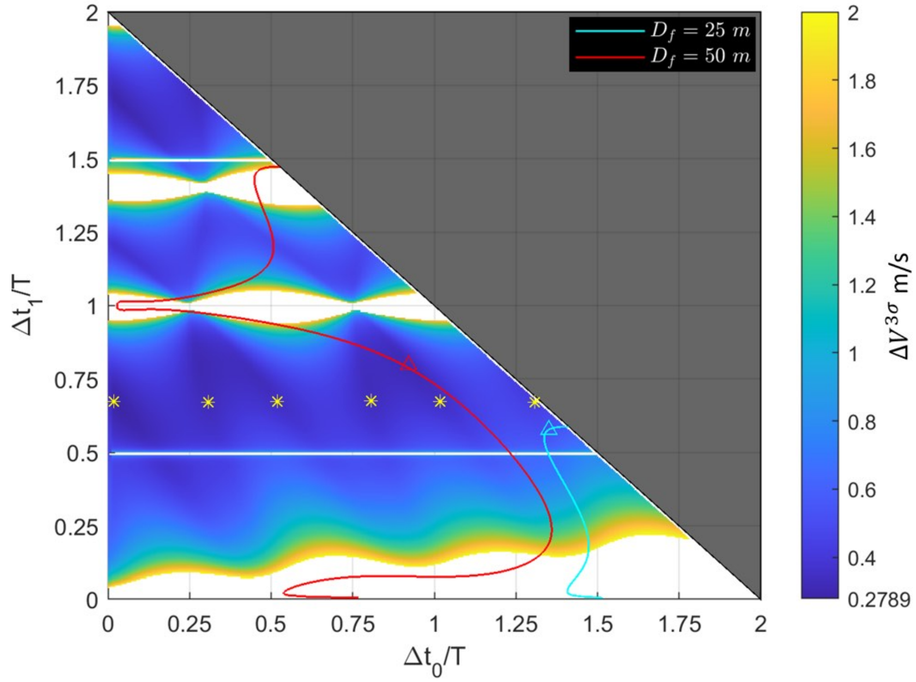

Fig. 5.9: Case C: ΔV dispersion for all combinations of Δt_0 and Δt_1 with overlayed final position dispersion contours.

Table 5.6: Optimal results for the constrained global optimal transfers for the NMC resizing scenario. $\Delta\bar{V} = 0.21502$ m/s.

Case	Constraint	$\Delta V^{3\sigma}$ (m/s)	ΔV (m/s)	$\Delta V/\Delta\bar{V}$	D_f (m)	$\Delta t_0/T$	$\Delta t_1/T$
A (△)	$D_f \leq 50$ m	0.27355	0.21519	1.0009	43.99	0.03820	0.65892
A (△)	$D_f \leq 25$ m	0.29254	0.23525	1.0942	23.32	0.02865	1.70460
A (▽)	$D_f \leq 25$ m	0.29264	0.21762	1.0122	24.97	0.08117	0.61117
B (△)	$D_f \leq 50$ m	0.28704	0.23525	1.0941	27.04	0.02865	1.70460
B (△)	$D_f \leq 25$ m	0.28717	0.23531	1.0944	21.76	0.03342	1.69982
B (▽)	$D_f \leq 50$ m	0.31695	0.22060	1.0260	16.60	0.92631	0.79261
B (▽)	$D_f \leq 25$ m	0.31695	0.22060	1.0260	16.60	0.92631	0.79261
C (△)	$D_f \leq 50$ m	0.31913	0.22102	1.0279	47.81	0.92153	0.79739
C (△)	$D_f \leq 25$ m	0.52177	0.22543	1.0484	24.99	1.35127	0.57775

strained solutions. This observation was aided in this scenario by the large troughs identified in the deterministic solution, indicating large areas with generally equivalent ΔV solutions.

Figure 5.10 shows a comparison between the Case A single revolution constrained optimal solution and the deterministic optimal solution #1. The constrained solution performs the first maneuver just prior to relative perigee but arrives at nearly the same location in the final orbit as the deterministic solution.

This scenario also reveals that maneuver execution errors play a significant role in determining the final position dispersions but do not necessarily have a large impact on the optimal solution. The optimal solution for Case B is near the 1^* deterministic solution, a single revolution delay to the Case A solution near the 1 deterministic solution. The 50 m constrained solution for Case C is again, nearly identical despite the final position dispersion map being drastically different in structure from the percentage-based maneuver execution errors in Cases A and B.

A significant (but intuitive) result from this analysis is that optimal transfers typically have short coasting times from the final maneuver to the terminal time. This approach, relying on the accurate relative navigation, allows the calculated final maneuver to correct for any dispersions that may have resulted from the initial state uncertainty and maneuver

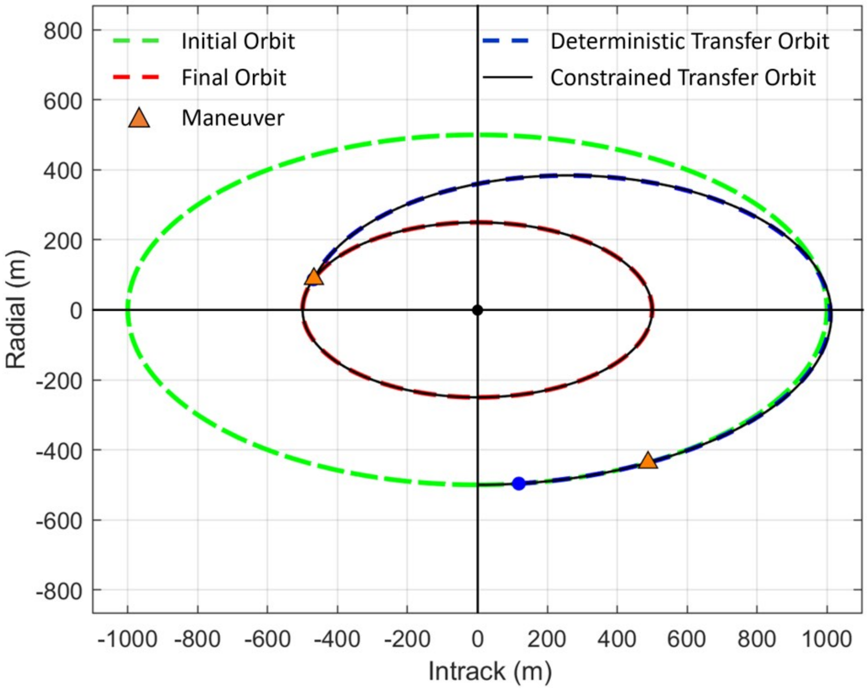

Fig. 5.10: Comparison between deterministic optimal transfer #1 and the constrained, single revolution optimal transfer for Case A ($\Delta t_0 = 0.08117T$ and $\Delta t_1 = 0.61117T$).

execution errors in the first burn. The short final transfer time prevents maneuver execution errors in the second burn and the dynamics process noise from growing beyond the stated constraint.

Finally, overly restrictive final position constraints can significantly alter the constrained optimal solution pushing them away from the unconstrained deterministic solutions. Meeting restrictive constraints relies on the same principles as the previous conclusion (short final transfer times) but pushes the maneuvers towards extremely short final transfer times which may drive significantly larger ΔV usage and dispersions and potentially infeasible maneuver execution timelines.

Full Contour Maps

Since the true position dispersion contour maps above were limited for clarity, expanded versions of the these contour maps using logarithmic scaling are shown in Figures 5.11, 5.12, and 5.13.

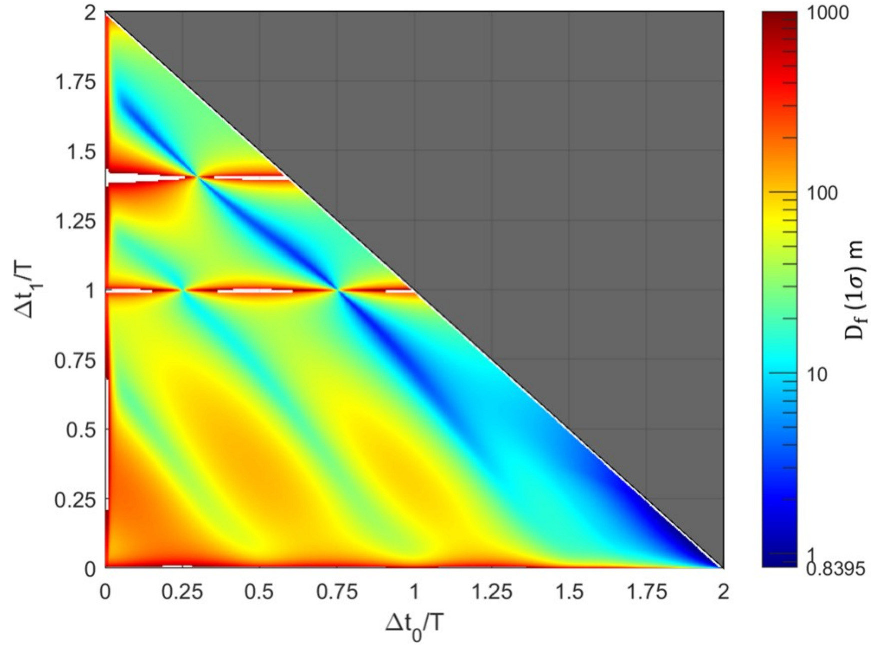

Fig. 5.11: Case A: Final position dispersion for all combinations of Δt_0 and Δt_1 using logarithmic scaling.

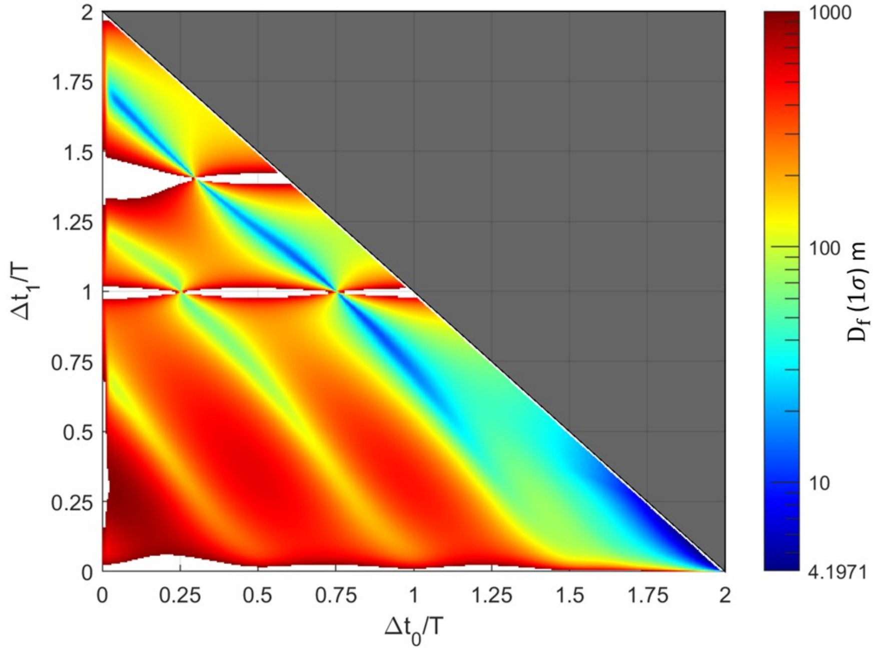

Fig. 5.12: Case B: Final position dispersion for all combinations of Δt_0 and Δt_1 using logarithmic scaling.

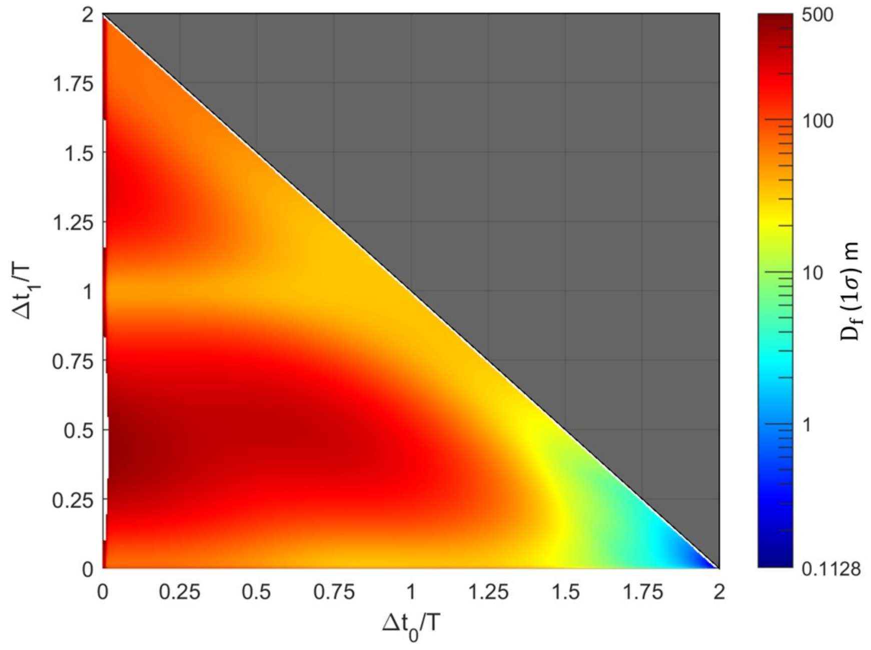

Fig. 5.13: Case C: Final position dispersion for all combinations of Δt_0 and Δt_1 using logarithmic scaling.

5.2 V-bar Hop Scenario

The second scenario is a simple transfer where the maneuvering satellite moves from a stationary position 250 m behind the RSO to a second stationary position 150 m behind the RSO. This is generally referred to as a V-bar hop since the satellite appears to hop from one location on the V-bar to another location on the V-bar. Station-keeping on the V-bar can be useful in RPO operations since the location requires minimal ΔV to maintain for long durations and offers a fixed vantage point for observing an RSO. The first evaluation parameter for the scenario is the initial wait time, Δt_0, before executing the first maneuver targeting the final offset. The second parameter is the transfer time between the two offsets, Δt_1. Finally, the vehicle remains at the new intrack offset until the end of the evaluation period. A representative transfer is shown in Figure 5.14.

As with scenario 1, the full range of transfer options were evaluated to determine the unconstrained nominal ΔV according to the parameters in Table 5.7. The deterministic ΔV results are shown in Figure 5.15.

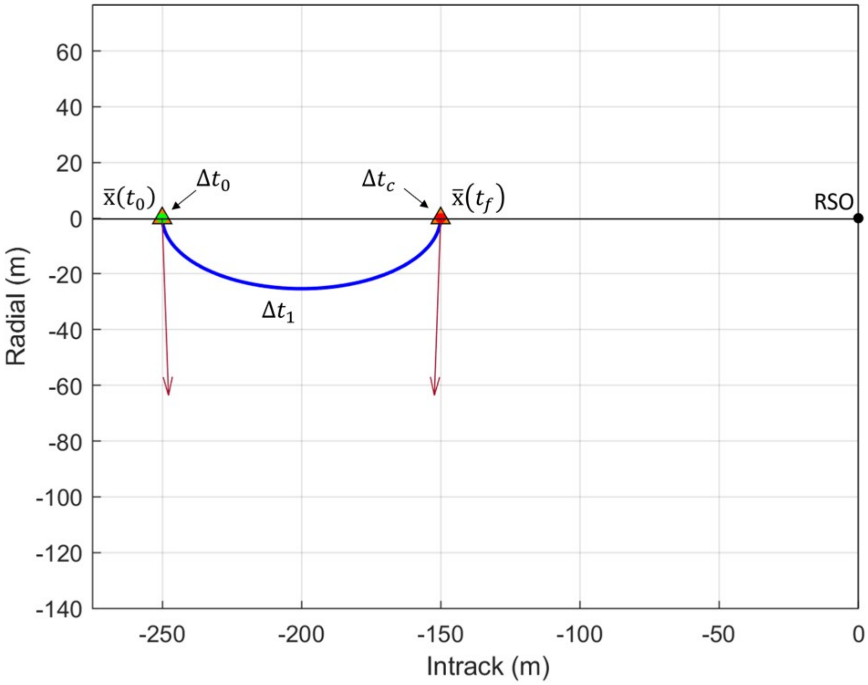

Fig. 5.14: Representative two-impulse V-bar hop transfer between two stationary intrack offsets.

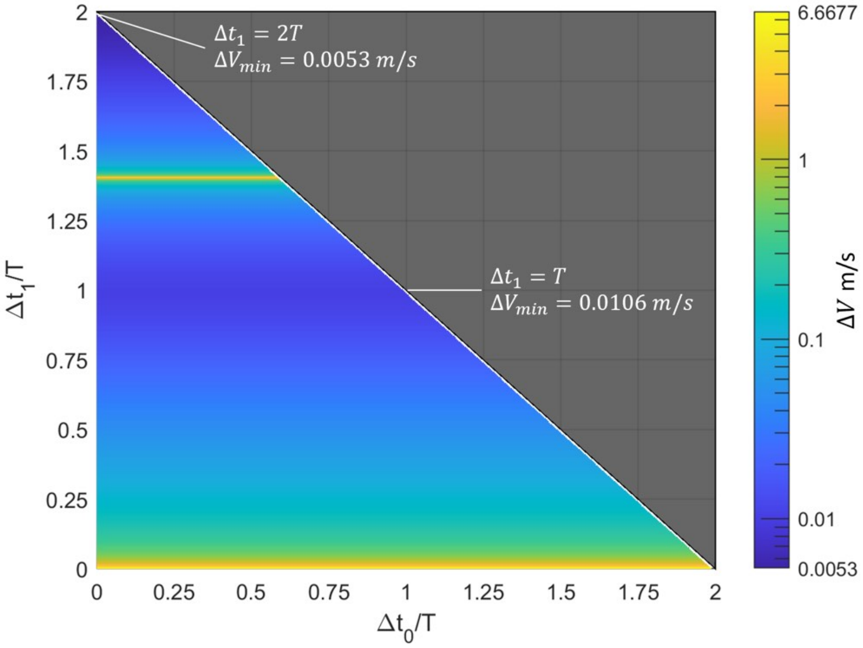

Fig. 5.15: Scenario 2: ΔV surface map for all transfer scenarios where $\Delta t_0 + \Delta t_1 \leq t_f$. Contours range between 0.0053 m/s and 1.0 m/s.

Table 5.7: Two-impulse V-bar hop scenario parameters.

Parameter	Value	Units
$\bar{\mathrm{r}}(t_0)$	$\begin{bmatrix}0 & -250 & 0\end{bmatrix}^T$	m
$\bar{\mathrm{v}}(t_0)$	$\begin{bmatrix}0 & 0 & 0\end{bmatrix}^T$	m/s
$\bar{\mathrm{r}}(t_f)$	$\begin{bmatrix}0 & -150 & 0\end{bmatrix}^T$	m
$\bar{\mathrm{v}}(t_f)$	$\begin{bmatrix}0 & 0 & 0\end{bmatrix}^T$	m/s
t_f	12,566 $(= 2T)$	s
ω	0.001	rad/s

The most obvious observation from an inspection of the deterministic results is that ΔV is only dependent on the transfer time, Δt_1. This result is expected due to stationary initial and final states and the fixed transfer distance between them. A closer inspection reveals that the minimum ΔV occurs with the longest transfer time ($\Delta t_1 = 2T$). More generally, a local minimum occurs whenever the transfer time is equal to an integer interval of the orbital period. The larger the integer the lower the ΔV. For this scenario, the minimum ΔV (0.0053 m/s) occurs when $\Delta t_0 = 0$ and $\Delta t_1 = 2T$. The single revolution transfer ($\Delta t_1 = T$) yields a local minimum ΔV of 0.0106 m/s, twice the value of the two revolution solution.

A second observation is the high ΔV cost for transfers near $1.4T$. This transfer time corresponds to a transition point in the solution where the maneuver vector rapidly switches directions. As a result, the transfer trajectory near $1.4T$ is a large looping trajectory rather than a simple hop. A small variation in the transfer time can result in a large change to the calculated ΔV.

5.2.1 LinCov Global Results

To provide the desired statistical data, the V-bar hop scenario was evaluated using the linear covariance model with the parameters defined in Table 5.8. The final true position dispersions are shown in Figure 5.16 and the $\Delta V^{3\sigma}$ values are shown in Figure 5.17. Similar to before, final true position dispersion contours for 5, 10, 15, and 20 m are overlayed on the ΔV dispersion surface map for easy comparison.

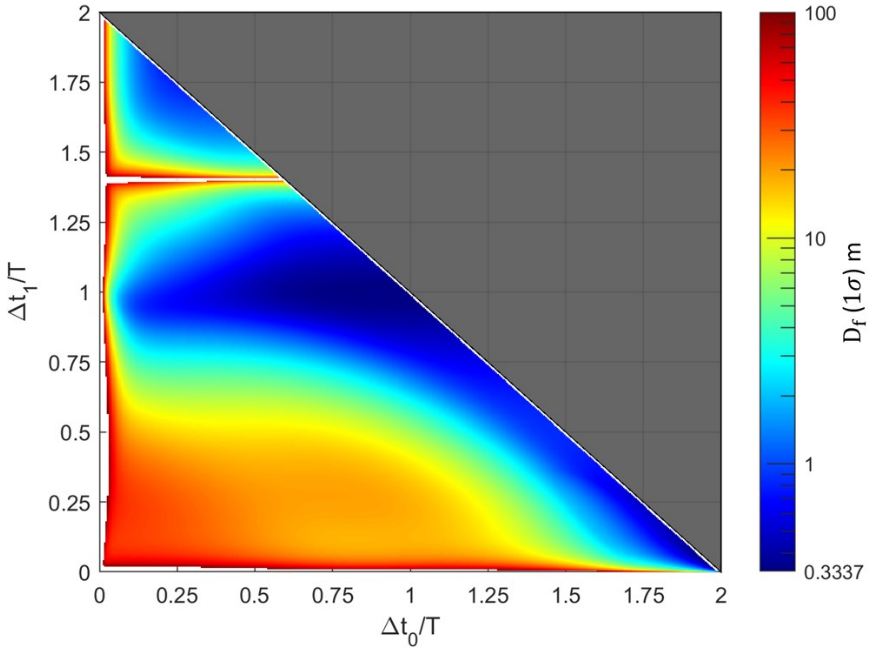

Fig. 5.16: Scenario 2: Final position dispersion for all combinations of Δt_0 and Δt_1.

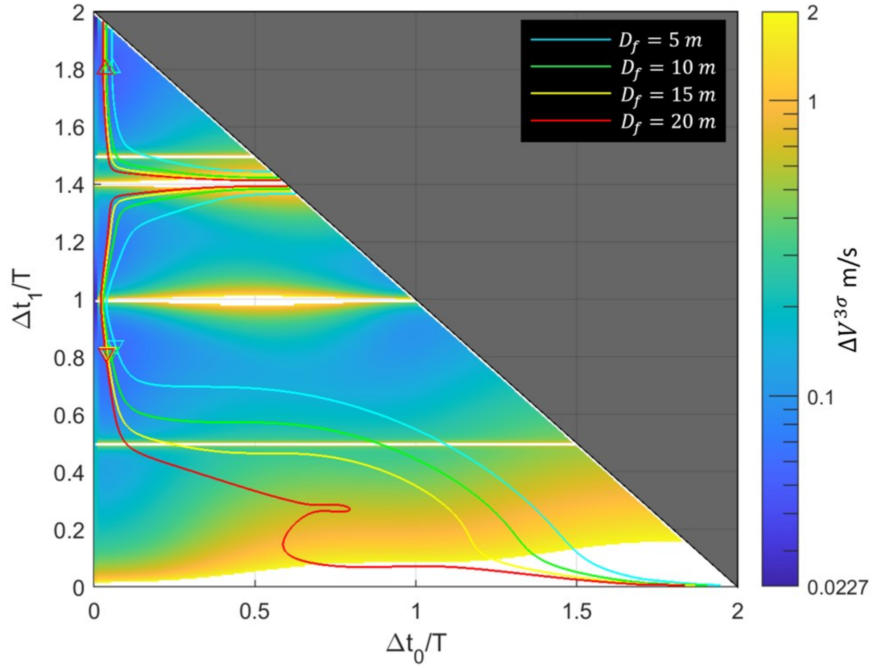

Fig. 5.17: Scenario 2: Final true ΔV dispersion for all combinations of Δt_0 and Δt_1 with overlayed final position dispersion contours.

Table 5.8: LinCov parameters for the V-bar hop scenario.

Parameter	Value	Units
Measurement Update Interval: Δt_{lidar}	30	s
Initial Position Uncertainty: $C_0(rr)$ (3σ)	$(10)^2 I_{3\times3}$	m^2
Initial Velocity Uncertainty: $C_0(vv)$ (3σ)	$(0.02)^2 I_{3\times3}$	m^2/s^2
Dynamics Process Noise: Q_w (3σ)	$(10^{-9})^2 I_{3\times3}$	m^2/s^3
Lidar Measurement Covariance: R (3σ)	$(0.1)^2 I_{3\times3}$	m^2
Maneuver Error: $S_{\Delta V}$ (3σ)	$(0.01\|\|\Delta\bar{V}\|\|)^2 I_{3\times3}$	m^2/s^2

An inspection of the final true position dispersion reveals two observations. The first is that the smallest dispersions occur near the optimal, single revolution transfer ($\Delta t_1 = T$). This would indicate that there are large natural regions with small true position dispersions. The second observation (which was also true for scenario 1) is that small final true position dispersions are commonly associated with short final coast times after the second maneuver. The combination of these two effects yields a particularly small final position dispersion on the order of 0.34 m when $\Delta t_0 > 0.75T$ and $\Delta t_1 \approx T$.

A related observation is that the transition point at $1.4T$ yields a large true position dispersion. As with the deterministic observation this result shows that small variations can results in extremely large dispersions. Transition areas such as this are poor options for both ΔV and position dispersion constraints and should be avoided when planning maneuvers.

The statistical data again allows us to find the minimum $\Delta V^{3\sigma}$ corresponding to a desired final position dispersion. A global search for final position constraints of 5, 10, 15, and 20 meters was performed for comparison to the deterministic analysis. Results for the multiple revolution solutions ($\Delta t_1 > T$) are marked with a $\triangle$ symbol and are detailed in Table 5.9. Results for the single revolution solutions ($\Delta t_1 \leq T$) are marked with a $\bigtriangledown$ symbol and are detailed in Table 5.10.

The optimal $\Delta V^{3\sigma}$ values shows a strong preference for quickly executing the first maneuver. The main difference between the optimal solution for the different constraints is just a slight delay (on the order of minutes) in execution of the first maneuver. Waiting a few extra minutes, though relatively short for the overall scenario, allows the navigation system

Table 5.9: Optimal $\Delta V^{3\sigma}$ results for the multi-revolution ($\Delta t_1 > T$) constrained V-bar hop transfers. $\Delta\bar{V} = 0.0053$ m/s.

Constraint	$\Delta V^{3\sigma}$ (m/s)	ΔV (m/s)	$\Delta V/\Delta\bar{V}$	D_f (m)	$\Delta t_0/T$	$\Delta t_1/T$
$D_f = 5$ m	0.05184	0.00871	1.6421	4.79	0.05252	1.8001
$D_f = 10$ m	0.05075	0.00888	1.6736	8.23	0.03820	1.7953
$D_f = 15$ m	0.05042	0.00888	1.6736	13.76	0.02865	1.7953
$D_f = 20$ m	0.05042	0.00888	1.6736	13.76	0.02865	1.7953

Table 5.10: Optimal $\Delta V^{3\sigma}$ results for the single revolution ($\Delta t_1 < T$) constrained V-bar hop transfers. $\Delta\bar{V} = 0.0106$ m/s.

Constraint	$\Delta V^{3\sigma}$ (m/s)	ΔV (m/s)	$\Delta V/\Delta\bar{V}$	D_f (m)	$\Delta t_0/T$	$\Delta t_1/T$
$D_f = 5$ m	0.06006	0.01468	1.3817	5.00	0.06207	0.83559
$D_f = 10$ m	0.05844	0.01593	1.5002	9.29	0.04775	0.81171
$D_f = 15$ m	0.05790	0.01593	1.5002	13.83	0.03820	0.81171
$D_f = 20$ m	0.05773	0.01620	1.5255	18.23	0.03342	0.80694

to further refine the orbit solution thus yielding more accurate maneuvers. Longer delays, however, allow the state dispersions to grow having a negative impact on the range of potential maneuvers that must be executed. Even with short delays before the first maneuver the ΔV dispersions are significantly larger than deterministic minimum. Additionally, the optimal $\Delta V^{3\sigma}$ values are 6x or 12x the nominal values for the single and multi-revolution transfers, respectively. These values may seem excessively large to trajectory planners who view the process as a deterministic system. These values, however, are quite reasonable compared to the 100x (or more) increases over the nominal value that are revealed through the stochastic analysis.

5.3 Conclusions

Two different two-impulse orbital transfers were evaluated in order to assess the differences between deterministic and constrained stochastic optimal transfers planning based on the RPO models developed in Chapter 4. While the two scenarios represented different aspects of RPO trajectory planning, there are a number of general observations that can be made:

- Constrained stochastic optimal solutions are frequently located near the deterministic optimal solutions.
- There generally exists a broad range of near-optimal trajectories near the optimal deterministic solution that provide a useful solution space for finding the constrained stochastic optimal solution.
- Short coast times between the final maneuver and the final transfer time aid is reducing the final true position dispersion size.
- Solution spaces near transition or singularity points in the deterministic solution typically result in both large ΔV dispersions and position dispersions.
- Thruster errors affect the range of final true position dispersions, but do not necessarily have a large affect the optimal $\Delta V^{3\sigma}$ solution.
- Constrained stochastic optimal solutions do not necessarily lie on the boundary of the position constraint unless the constraint is overly restrictive.
- Overly restrictive constraints on the final position dispersion may lead to infeasible solutions or drive the solution away from the optimal regions and can result in significantly large ΔV dispersions.

These conclusions will help drive the optimization techniques implemented in Chapter 6 as we seek to solve more complex scenarios.

CHAPTER 6

OPTIMAL RELATIVE PATH PLANNING WITH POSITION DISPERSION CONSTRAINTS

Mission planning for rendezvous and proximity operations missions is a complicated task that often requires operators to balance competing constraints while still satisfying mission objectives. As missions have become more complex, the planning process has migrated from a manually focused task to one involving more sophisticated tools capable of optimally managing the constraints and objectives. The inclusion of a closed-loop RPO linear covariance model into the planning process provides a significant new planning capability and requires modern optimization tools for solving in a timely manner.

In this chapter we will develop a hybrid optimization approach used to determine optimal trajectories for our stochastic, closed-loop RPO system when there are imposed constraints on the final position dispersion and performance limitations in the navigation system. The tool is aptly named Stochastic Trajectory Optimization for RPO Missions (STORM). To demonstrate STORM's ability to find global optimal solutions, it will be used to analyze the two-impulse NMC resizing and V-bar hop scenarios from Chapter 5 that were previously solved using a global search method.

6.1 Stochastic Trajectory Optimization for RPO Missions (STORM)

This section describes the implementation of the STORM analysis tool that pairs a hybrid optimization approach with the closed-loop RPO linear covariance model for determining optimal relative trajectories when constrained by final position dispersion.

6.1.1 Optimization Approach

Based on the lessons learned in Chapter 5, a two-step hybrid optimization process was implemented to find the global optimal transfer path between the initial state and the final state. The first step applies a Genetic Algorithm (GA) to find a solution near the global optimum. A stochastic solver was selected due to the complex solution space, as seen in the solutions to the relatively simple two-impulse scenarios (as depicted in Figures 5.5, 5.7, and 5.9). The GA solution is then fed to a non-linear solver (MATLAB's *fmincon* routine) to refine the solution and to find the global optimum. This hybrid approach was chosen based on the observation that the solution space near the global optimum is potentially convex and conducive to direct optimization approaches. While every effort is made to find the global optimum (or the adjacent solution space), there is no guarantee that the GA will be able to find the desired solution space and may represent a local minimum. As noted in Chapter 5, the solution space for even a simple two-impulse problem can have multiple similar local minima. Fortunately for the mission planner (and their fuel budget), there may be relatively small differences between the local and global minima.

This optimization approach is shown in Figure 6.1. The overall approach is managed by the STORM tool that controls the scenario inputs and optimization settings and reports the optimal trajectory data. Both the GA and *fmincon* solvers feed the decision variable, x, to the RPO LinCov analysis tool and optimize based on the returned cost function, J. The final solution from the GA is used as the initial guess when initializing *fmincon*.

6.1.2 Genetic Algorithm

The Genetic Algorithm, a member of the larger class of evolutionary algorithms, uses principles of biological evolution to solve global optimization problems. By mimicking the concept of natural selection, a population of potential solutions are evaluated as they evolve across multiple generations through the mechanisms of selection, crossover, mutation, and elitism. Each potential solution in the population, x_i^{pop}, is referred to as an individual with its own unique solution set, referred to as chromosomes, and fitness function, the equivalent of a cost function. Chromosomes can be combined or mutated over time to generate new

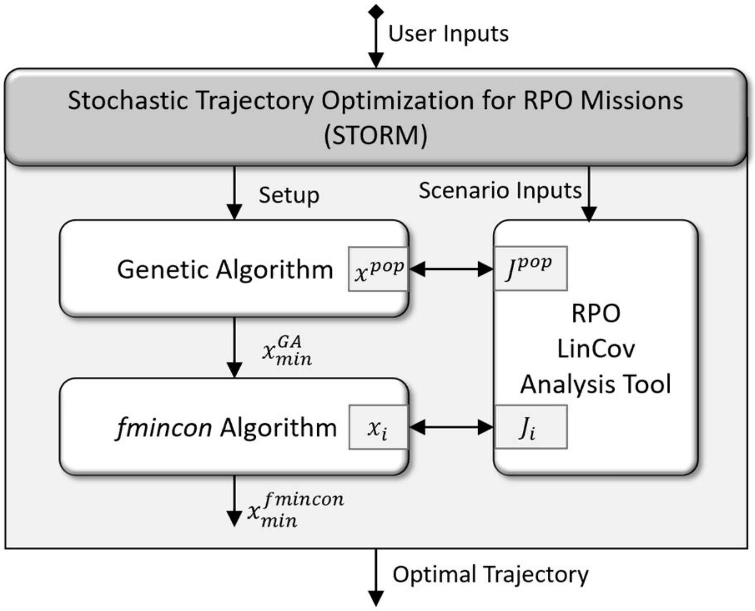

Fig. 6.1: STORM analysis tool implementing a hybrid optimization approach.

individuals in search of an optimal fitness function. A general flow diagram of the GA is shown in Figure 6.2. The following details are a mixture of general GA methods and specifics provided by the MATLAB Global Optimization Toolbox documentation [112]. For additional details, the reader is directed to the introductory work by Spall [113] or the application focused work by Goldberg [114].

The chromosomes of the initial population are randomly selected to provide a diverse population across the feasible solution space while also meeting all constraints. A typical population contains between 100 and 300 individuals depending on the complexity of the problem. Increasing the population size results in a more effective search of the solution space at the expense of computational time.

After calculating the fitness function for each member of the population, the GA works to evolve the population by first determining which individuals will be used as parents to generate offspring for the next generation. This process is referred to as *selection*. MATLAB's default selection algorithm, stochastic uniform sampling, creates a line of finite length where each individual of the population represents a segment of the line proportional to their scaled fitness function. The algorithm then steps along the line at uniform steps to

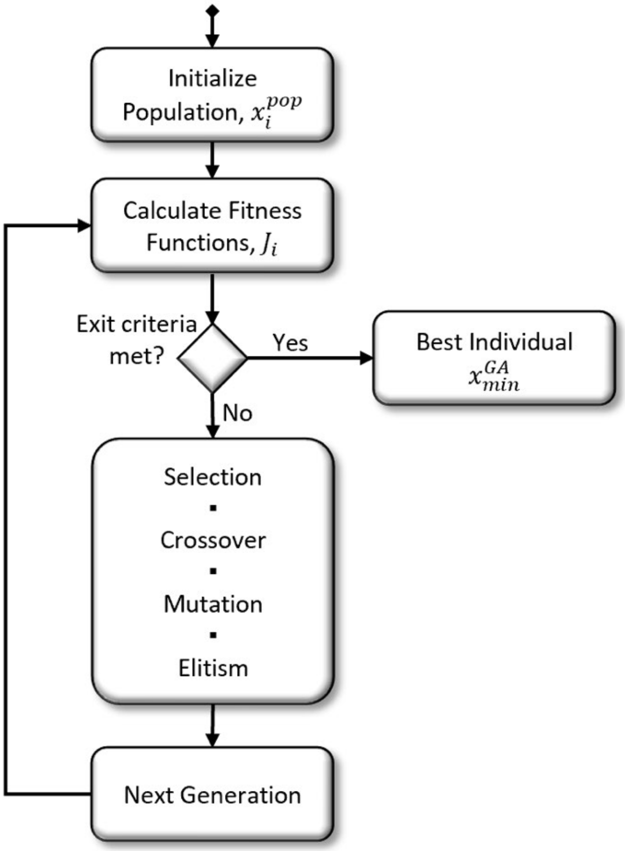
Initialize Population, x_i^{pop}
Calculate Fitness Functions, J_i
Exit criteria met?
Yes
Best Individual x_{min}^{GA}
No
Selection
Crossover
Mutation
Elitism
Next Generation

Fig. 6.2: Genetic Algorithm flow diagram.

select parent pairs. Individuals with lower fitness function values (i.e. more optimal) are more likely to be selected as parents but all individuals have to potential to pass along their information to the next generation.

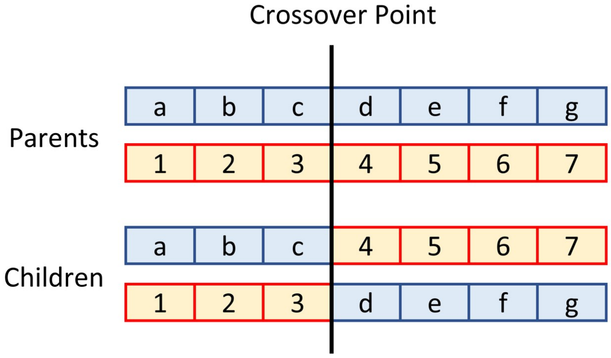

Fig. 6.3: Single-point crossover function where the chromosomes from parents are used to generate children for the next generation.

Children for the next generation are created through a process called *crossover* where the chromosomes from two parents are crossed at a specific locations to create two new chromosomes. Crossover can occur at a single location or at multiple locations. Figure 6.3 depicts a single point crossover function.

In order to introduce and/or maintain genetic diversity in the population, one or more values in a child's chromosome may be randomly altered by a process called *mutation*. Mutation allows for the GA to efficiently explore the solution space by creating and evaluating small (or large) variations in the population. For a binary solution space a mutation can be as simple as a single bit flip in an individual's chromosome. For real value representations the mutation may be a random offset applied to each value. The mutation processes are depicted in Figure 6.4. For a more complex solutions space, the mutation process may be

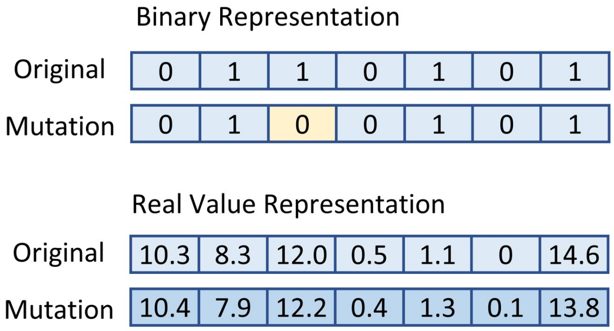
Binary Representation
Original 0 1 1 0 1 0 1
Mutation 0 1 0 0 1 0 1
Real Value Representation
Original 10.3 8.3 12.0 0.5 1.1 0 14.6
Mutation 10.4 7.9 12.2 0.4 1.3 0.1 13.8

Fig. 6.4: Sample mutation processes for binary and real valued solution space chromosomes.

finite steps along a direction determined by results from the previous generation towards an optimal solution. Mutation is only applied to a subset of generated children to prevent the process from turning the GA into a random search algorithm. Mutation in the MATLAB GA is performed by adding a random vector, sampled from a Gaussian distribution, to a child's chromosome. The standard deviation of the mutation typically decreases with each generation to allow for finer adjustments over time.

The final step in the GA is to select a number of individuals with the best fitness functions to transition unchanged into the next generation though the process called *elitism*. This process ensures that the best individuals are not lost between generations and continue to serve as parents if more suitable individuals are not identified through subsequent evaluations. Approximately 5% of the population are carried over through elitism, though this can be adjusted as necessary for a given problem. Figure 6.5 depicts the combined mechanics of selection, crossover, mutation, and elitism.

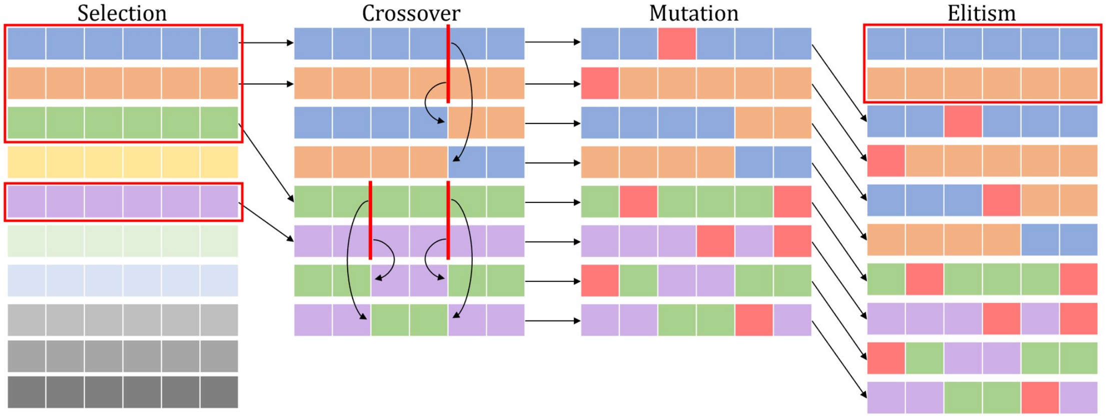

Fig. 6.5: Selection, crossover, mutation, and elitism mechanics of a generic Genetic Algorithm.

6.1.3 MATLAB *fmincon* Algorithm

The best individual from the GA is passed to MATLAB's constrained, non-linear multivariable optimization routine, *fmincon*, for generating a final optimal solution. This function call uses the same linear constraints and bounds as the GA function call. Whereas the

GA is used to perform the bulk of the optimization process to place the solution in the vicinity of the global minima, this *fmincon* function call is used to quickly home in on the optimal solution. Published details of the *fmincon* algorithm are available in the MATLAB Optimization Toolbox documentation [115].

6.1.4 Fitness Function

In line with the evolutionary theme, the cost function of the GA is referred to as the *fitness function*, invoking the Darwinian concept of survival of the fittest. For simplicity, this nomenclature is maintained when discussing the cost function of either the GA or *fmincon* algorithms.

For the constrained scenario, the fitness function to be minimized is:

$$\min_{x} \quad J = \Delta V^{3\sigma} + P_{\text{disp}} \tag{6.1}$$

where $\Delta V^{3\sigma}$ is the 3σ ΔV consumption and dispersion defined in Equations (4.86) and (4.88):

$$\Delta V^{3\sigma} = \sum_{k=1}^{n} \left(\left\| \Delta \bar{V}_k \right\| + 3\sqrt{Tr[D_{\Delta V_k}]} \right)$$

and P_{disp} is a penalty function for not achieving the required final true position dispersion. It should be noted that the fitness function in Equation (6.1) is a function of x through the linear covariance model but is not indicated in the equation for simplicity. The penalty is calculated as:

$$P_{\text{disp}} = \begin{cases} 100\ S_{D_f}/S_{\text{lim}}, & \text{if } S_{D_f} > S_{\text{lim}} \\ 0, & \text{otherwise} \end{cases} \tag{6.2}$$

where S_{D_f} is the final true position dispersion defined in Equation (4.77) and S_{lim} is the user specified final true position dispersion requirement. It is important to mention that this penalty function is designed to guide the solution towards values that meet the final dispersion constraint. More severe penalties risk becoming a guess and shoot method simply attempting to find the proverbial needle in a haystack when the desired region is relatively small compared to the search space. The penalty function is depicted in Figure 6.6.

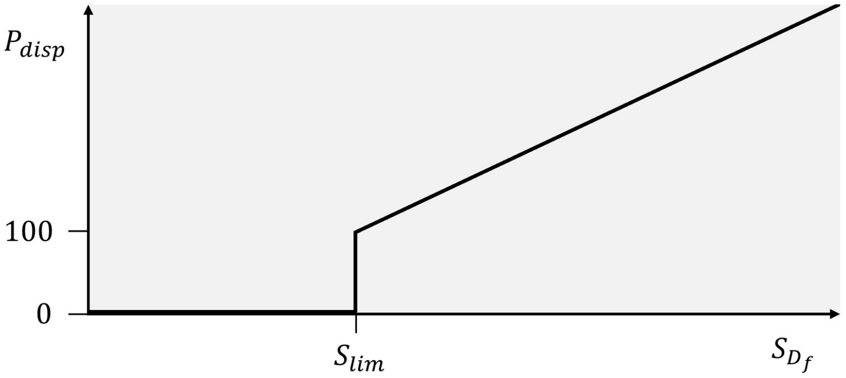

Fig. 6.6: Penalty shape as a function of S_{D_f} used to aid convergence.

The scale factor of 100 in Equation (6.2) was selected to be larger than any nominal ΔV values expected to be encountered in the represented scenarios. Care should be taken when selected this scale factor to ensure the penalty aids in convergence and does not allow for non-compliant solutions to pass as the top individual. Such a scenario may occur where the GA minimizes the penalty and ΔV but does not meet the final position dispersion constraint. The scale factor should be selected so that, once the constraint has been met, the GA moves to quickly disregards non-compliant trajectories through the selection process and elitism processes.

As a baseline for comparison against the constrained optimal solutions we are also interested in calculating the deterministic optimal solution for each scenario. Rather than employing a global search, as was done in Chapter 5, we will again use the STORM analysis tool with a simplified fitness, including only the total ΔV for the scenario, to generate the baseline data without consideration of the stochastic linear covariance model. The RPO linear covariance model can then be used to generate the stochastic characteristics of the baselined, deterministic solution such as the final true position and ΔV dispersions for comparison. The deterministic fitness function to be minimized is:

$$\min_{x} \quad J = \Delta\bar{V} = \sum_{k=1}^{n} \left\|\Delta\bar{V}_k\right\| \tag{6.3}$$

6.1.5 Decision Variable Size and Constraints

In order to generate an initial population, the size of the decision variable (i.e. the chromosome of individuals in the population) and any linear constraints must first be specified. The decision variable size, n_{vars}, is dependent on the number of intermediate waypoints, n_{wpts}, in the solved-for solution and is evaluated as:

$$n_{\text{vars}} = \begin{cases} 2, & n_{\text{wpts}} = 0 \\ 2 + 4 \cdot n_{\text{wpts}}, & n_{\text{wpts}} \geq 1 \end{cases} \tag{6.4}$$

For the two-impulse scenario, ($n_{\text{wpts}} = 0$), the decision variable is arranged as:

$$x^{(0)} = \begin{bmatrix} \Delta t_0 & \Delta t_1 \end{bmatrix}^T \tag{6.5}$$

where Δt_0 and Δt_1 are the transfer times as defined in Section 5.1. For a scenario with one or two intermediate waypoints ($n_{\text{wpts}} = 1$ or 2) the decision variables are arranged as:

$$x^{(1)} = \begin{bmatrix} \Delta t_0 & \Delta t_1 & \Delta t_2 & R_x^{(1)} & R_y^{(1)} & R_z^{(1)} \end{bmatrix}^T \tag{6.6}$$

$$x^{(2)} = \begin{bmatrix} \Delta t_0 & \Delta t_1 & \Delta t_2 & \Delta t_3 & R_x^{(1)} & R_y^{(1)} & R_z^{(1)} & R_x^{(2)} & R_y^{(2)} & R_z^{(2)} \end{bmatrix}^T \tag{6.7}$$

where Δt_2 and Δt_3 are the additional transfer times and $R_x^{(n)}$, $R_y^{(n)}$, and $R_z^{(n)}$ are the components of the waypoint position vectors in the LVLH coordinate frame. Each additional waypoint adds one extra transfer time and three extra waypoint position values.

In order to ensure the scenario time matches the fixed final time, the decision variable is subject to the following linear inequality constraint:

$$Ax \leq b \tag{6.8}$$

For $n_{\text{wpts}} = 0$ the values for A and b are:

$$A^{(0)} = \begin{bmatrix} 1 & 1 \end{bmatrix}, \qquad b = t_f \tag{6.9}$$

so that $\Delta t_1 + \Delta t_2 \leq t_f$. For $n_{\text{wpts}} = 1$, the values for A and b are:

$$A^{(1)} = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \end{bmatrix}, \qquad b = t_f \tag{6.10}$$

so that $\Delta t_0 + \Delta t_1 + \Delta t_2 \leq t_f$. Linear inequalities for additional waypoints are similarly constructed. Time values are forced to be positive (e.g. $\Delta t_1 \geq 0$) by enforcing lower bounds in the GA function. This approach was implemented, rather than additional linear inequalities, to due to internal limitations in the MATLAB GA that allows for individuals that violate the linear inequality constraints. Individuals that violate the constraints are applied additional penalties to ensure they're not carried forward, but these individuals can slow down the optimization process and are better limited with the upper and lower bounds when applicable.

It is important to note here that the waypoints are not subject to equality or inequality constraints but are constrained by setting upper and lower bounds on their values. These bounds are determined by assuming all waypoints are sufficiently close to the RSO and by defining an appropriately sized box around the RSO. For most scenarios this box was defined as a cube whose side is three times the largest component magnitude of the initial or final position states. This box size is somewhat arbitrary and may need to be adjusted for each scenario but its application sufficiently constrains the GA populations and helps reduce overall run time.

Finally, all values in the GA decision variable are defined to be integer values. This configuration prevents the GA evolution process from making minor inconsequential adjustments to the population and instead helps focus on the GA's coarse optimization before the top individual solution is passed to the non-linear solver. The *fmincon* solver does not implement this same integer constraint. Because the values are defined as integers, the

MATLAB GA may assign values to an individual's chromosome that violate the inequality constraints. In these cases the fitness functions are assigned an additional penalty (internal to the algorithm) that reduces to zero when the constraints are met.

6.1.6 Free Final Time

While the majority of this work focuses on scenarios with a fixed final time, t_f, it is also possible to construct the optimization problem for a free final time scenario. This is accomplished by adding the final coast time, Δt_c, to the decision variable as a means of fully defining t_f. For a two-impulse free final time scenario, the decision variable becomes:

$$x_{\texttt{fft}}^{(0)} = \begin{bmatrix} \Delta t_0 & \Delta t_1 & \Delta t_c \end{bmatrix}^T \tag{6.11}$$

where $\Delta t_0 + \Delta t_1 + \Delta t_c = t_f$. Equivalently, the three-impulse decision variable becomes:

$$x_{\texttt{fft}}^{(1)} = \begin{bmatrix} \Delta t_0 & \Delta t_1 & \Delta t_2 & \Delta t_c & R_x^{(1)} & R_y^{(1)} & R_z^{(1)} \end{bmatrix}^T \tag{6.12}$$

This approach also requires modification to the linear inequality constraints by selecting an arbitrarily large limit on b. STORM sets an upper limit on the transfer time of three times the orbital period ($t_f \leq 3T$). The selection of $3T$ is arbitrary and can be modified depending on the problem and operational constraints.

6.2 Optimal Two-Impulse Path Planning

In this section we will apply the hybrid optimization approach described above to the two-impulse scenarios developed and analyzed in Chapter 5. Comparing the results of the global search and the hybrid optimization approaches will demonstrate the optimization approach's ability to effectively determine the global optimum. To evaluate this performance we will revisit the NMC resizing in scenario 1, Case A (Section 5.1) and the V-bar hop in scenario 2 (Section 5.2). Details of the inputs used for the MATLAB GA are provided in Appendix E. Included are the stopping conditions. While function tolerances are defined,

Table 6.1: LinCov global search and optimization results for the single revolution NMC resizing scenario with final position requirement, $S_{lim} = 25$ m. $\Delta\bar{V} = 0.21502$ m/s.

Method	$\Delta V^{3\sigma}$ (m/s)	ΔV (m/s)	$\Delta V/\Delta\bar{V}$	D_f (m)	$\Delta t_0/T$	$\Delta t_1/T$
Global Search (A▽)	0.29264	0.21762	1.0122	24.97	0.08117	0.61117
Optimization	0.28959	0.21717	1.0100	24.62	0.07162	0.61615

the maximum number of generations or the maximum number of stall generations (i.e. generation without an improvement in the fitness function) are the most common reason for the GA to stop.

6.2.1 Optimal Two-Impulse NMC Resizing Results

The hybrid optimization approach was applied to the single revolution NMC resizing scenario as defined in Section 5.1. In order to determine the single revolution transfer solution, the linear inequality constraint for the GA and *fmincon* solvers defined in Equation (6.9) was modified as:

$$A^{(0)} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}, \qquad b = \begin{bmatrix} t_f \\ T \end{bmatrix} \tag{6.13}$$

so that $\Delta t_1 \leq T$.

The optimal results for both the global search and hybrid optimization methods, given a 25 m final position dispersion requirement, are given in Table 6.1. A comparison of the two methods show that the hybrid optimization was successful at locating the global optimum and resulted in a lower fitness function than the global search. The optimal trajectory and maneuver vectors determined by the hybrid approach are shown in Figure 6.7. The final true position dispersion is shown against the 25 m requirement in Figure 6.8 demonstrating that the constraint was successfully met.

To show the GA's ability to converge for this scenario, trajectories for the top 100 individuals of the final populations are shown in 6.9. Maneuvers for the final population are grouped together in regions that correspond to low values of the fitness function.

Values from the initial and final populations are also overlayed onto the ΔV dispersion surface maps in Figure 6.10 (as originally depicted in Figure 5.5). The optimal solution, determined by the *fmincon* algorithm, is located very near that optimal solution for the scenario as was shown in Table 6.1. It is also apparent the many of the individuals in the final population are grouped near the deterministic, non-unique global optimum values as was expected from a visual assessment of the final population trajectories.

Finally, the fitness functions for all individuals in the initial and final populations are shown in Figure 6.11. Given the simplicity of the solution space, the fittest individual of the initial population is near the final solutions. This condition was the result of the random initialization and may not always be applicable. The largest values are the result of the GA's internal penalty function.

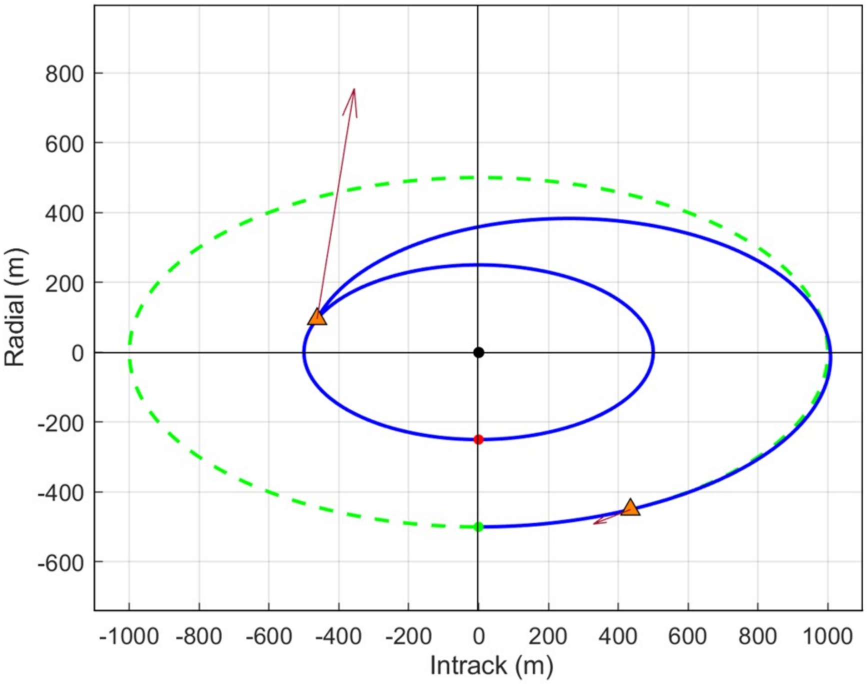

Fig. 6.7: Optimal NMC resizing trajectory and maneuvers.

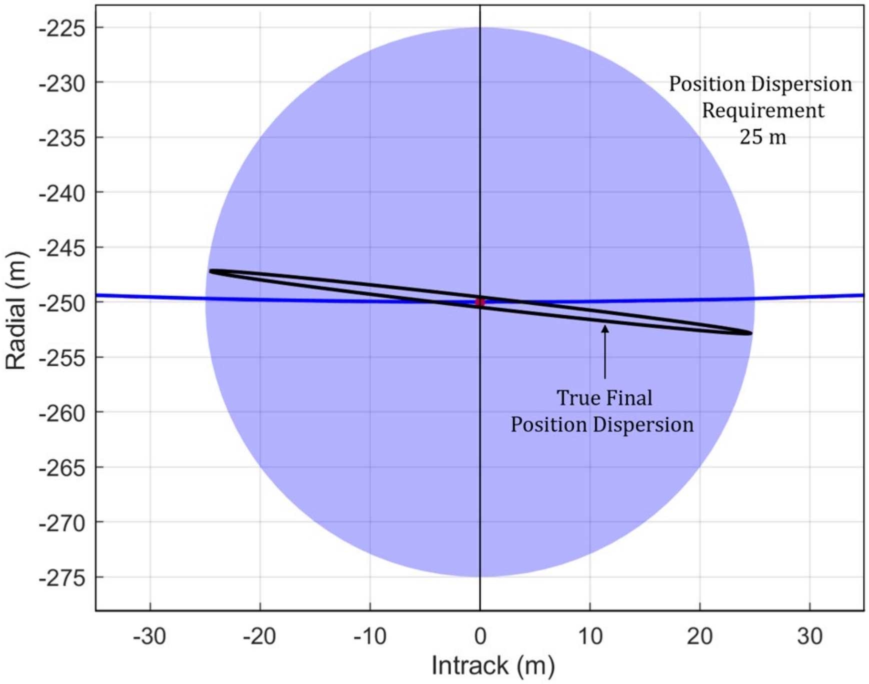

Fig. 6.8: Final true position dispersion for the optimal NMC resizing trajectory.

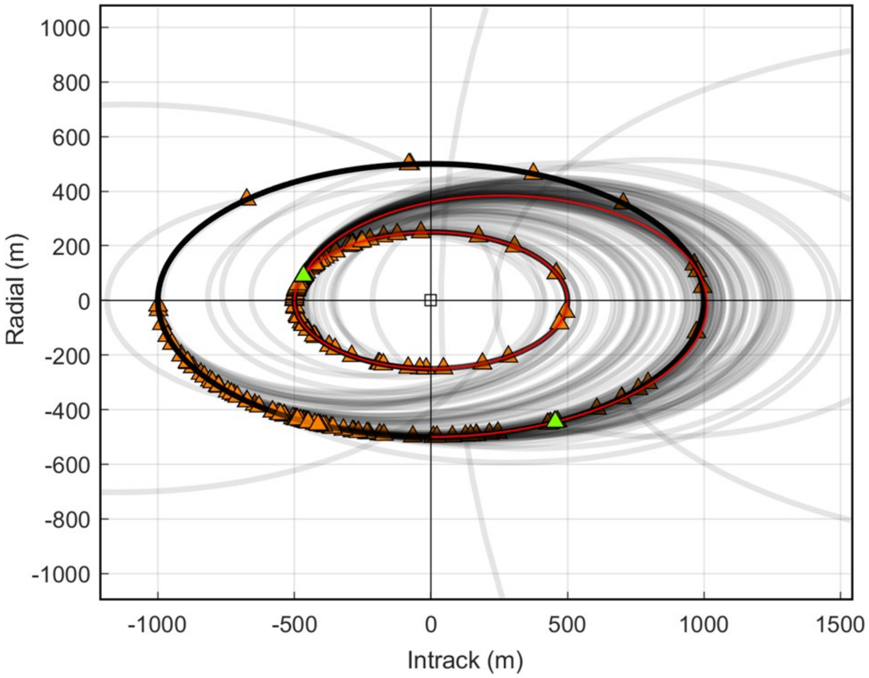

Fig. 6.9: Trajectory solutions for the GA final population. Optimal maneuvers are green. Optimal trajectory is red.

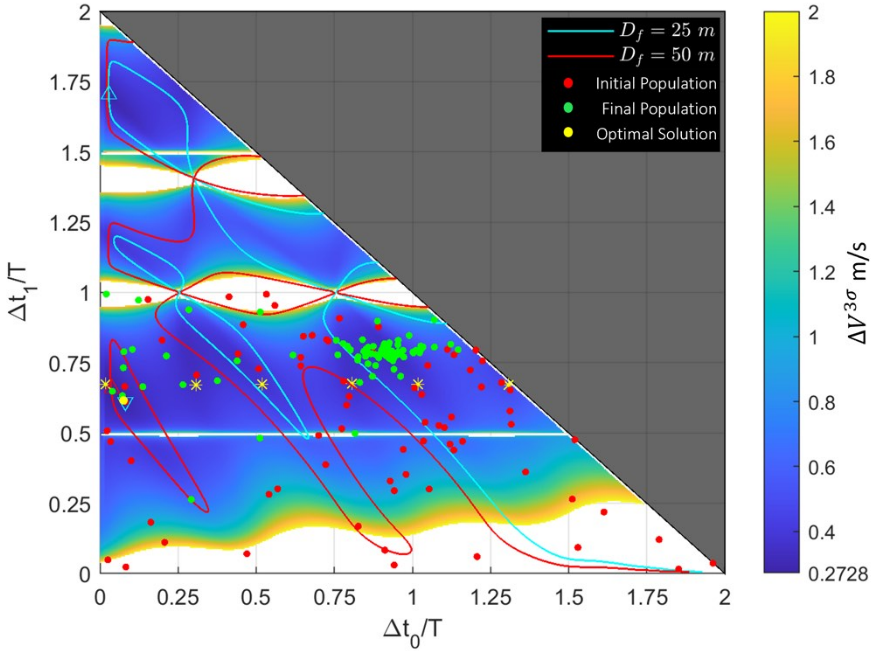

Fig. 6.10: Locations of the GA initial and final populations and the *fmincon* optimal solution overlayed on the NMC resizing ΔV dispersion surface map.

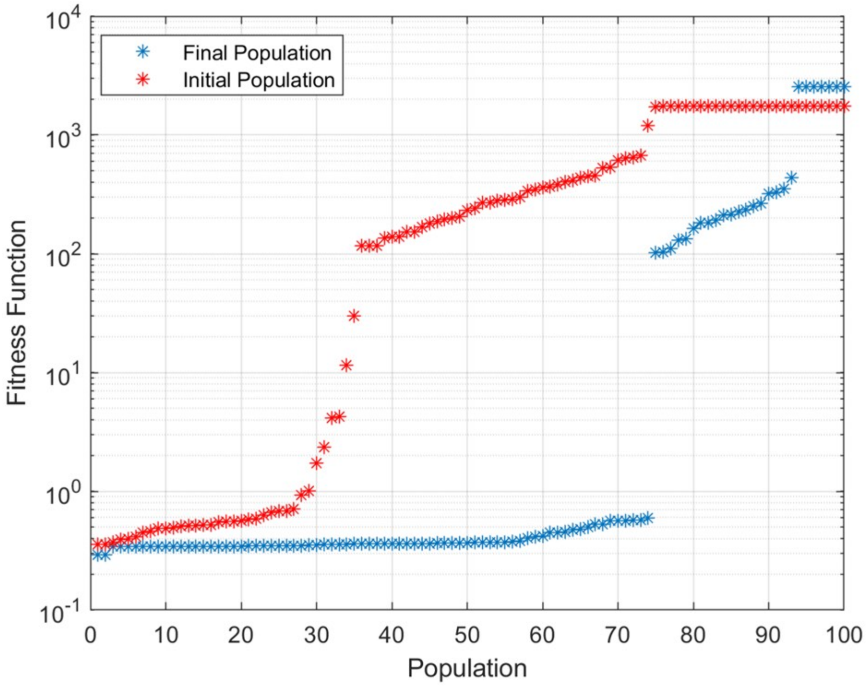

Fig. 6.11: Fitness function values for the initial and final GA populations of the NMC resizing scenario.

6.2.2 Optimal Two-Impulse V-bar Hop Results

Next, STORM was applied to the V-bar hop scenario as described in Section 5.2. The original linear equality constraint, as defined in Equation (6.9), was used to explore the full range of single and multi-revolution solutions.

The optimal results for both the global search and hybrid optimization methods, given a 5 m final position dispersion requirement, are given in Table 6.2. A comparison of the two methods show that the hybrid optimization was again successful at locating the global optimum and resulted in a lower fitness function than the global search. The results from these two scenarios shows the hybrid approach's ability to find constrained global optimal values using the linear covariance model.

The optimal trajectory and maneuver vectors as determined by the hybrid approach are shown in Figure 6.12. The final true position dispersion is shown against the 2 m requirement in Figure 6.13 demonstrating that the constraint was successfully met.

Table 6.2: LinCov global search and optimization results for the V-bar hop scenario with a final position requirement, $D_f \leq 5$ m. $\Delta\bar{V} = 0.0053$ m/s.

Method	$\Delta V^{3\sigma}$ (m/s)	ΔV (m/s)	$\Delta V/\Delta\bar{V}$	D_f (m)	$\Delta t_0/T$	$\Delta t_1/T$
Global Search	0.05184	0.00871	1.6421	4.79	0.05252	1.8001
Optimization	0.05141	0.00871	1.6433	4.80	0.04775	1.80485

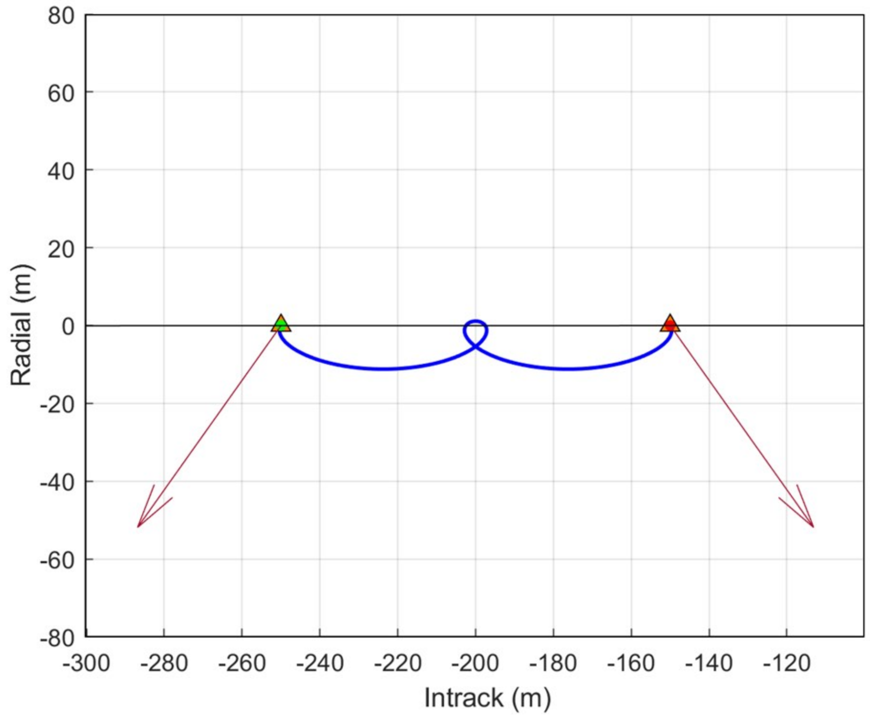

Fig. 6.12: Optimal V-bar hop trajectory and maneuvers determined by the hybrid optimization method.

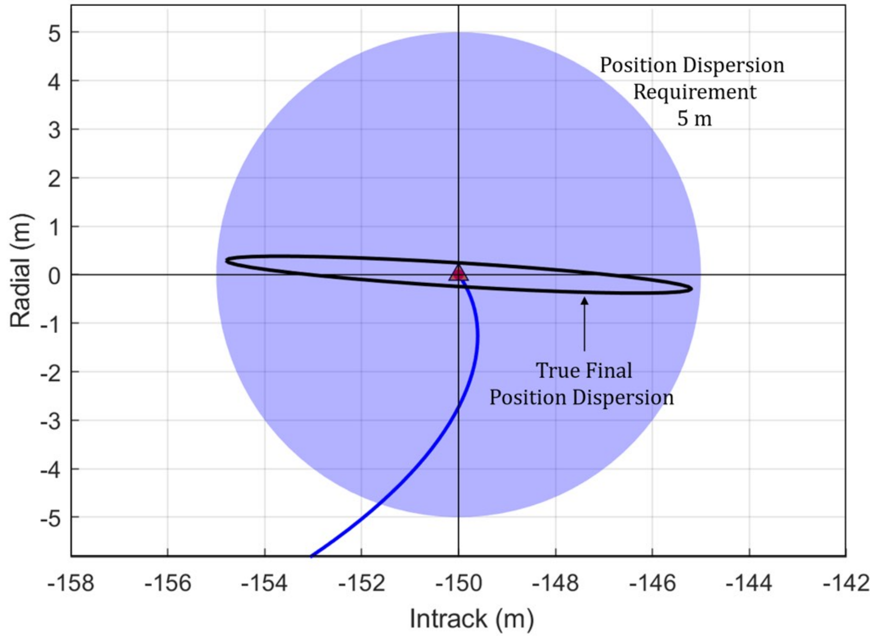

Fig. 6.13: Final true position dispersion for the optimal V-bar hop trajectory.

Values from the initial and final populations are overlayed onto the ΔV dispersion surface maps in Figure 6.14 (as originally depicted in Figure 5.7) to show the evolution of the GA population. The optimal solution, determined by the *fmincon* algorithm, is located very near the global search optimal solution as was shown in Table 6.2. As with the NMC resizing scenario, many of the individuals in the final population are grouped near the deterministic optimum solution.

Finally, the fitness functions for individuals in the initial and final populations are shown in Figure 6.15. Given the simplicity of the solution space, the fittest individual of the initial population is near the final solutions. This condition was the result of the random initialization and may not always be applicable.

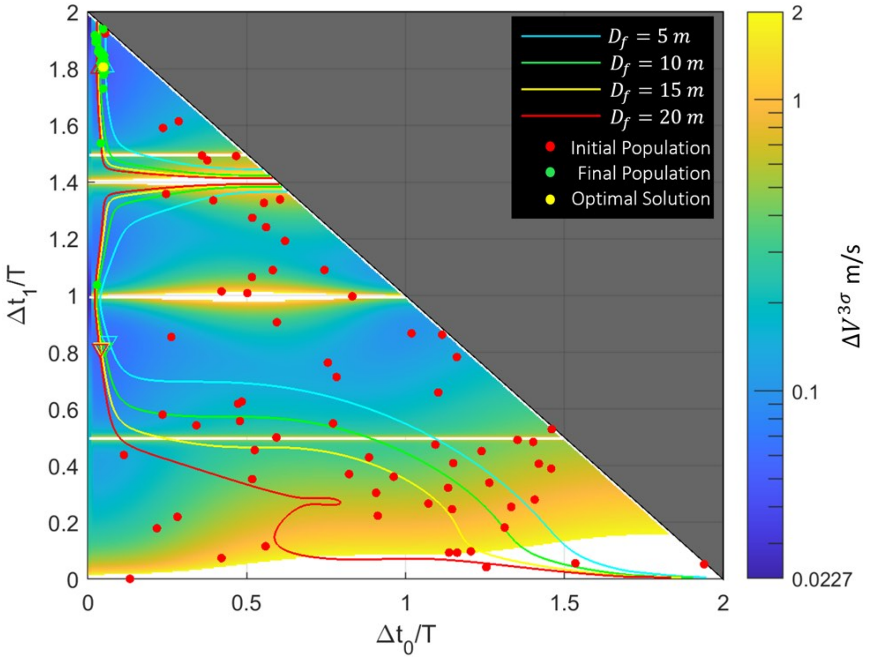

Fig. 6.14: Locations of the GA initial and final populations and the optimal solution overlayed on the V-bar hop ΔV dispersion surface map.

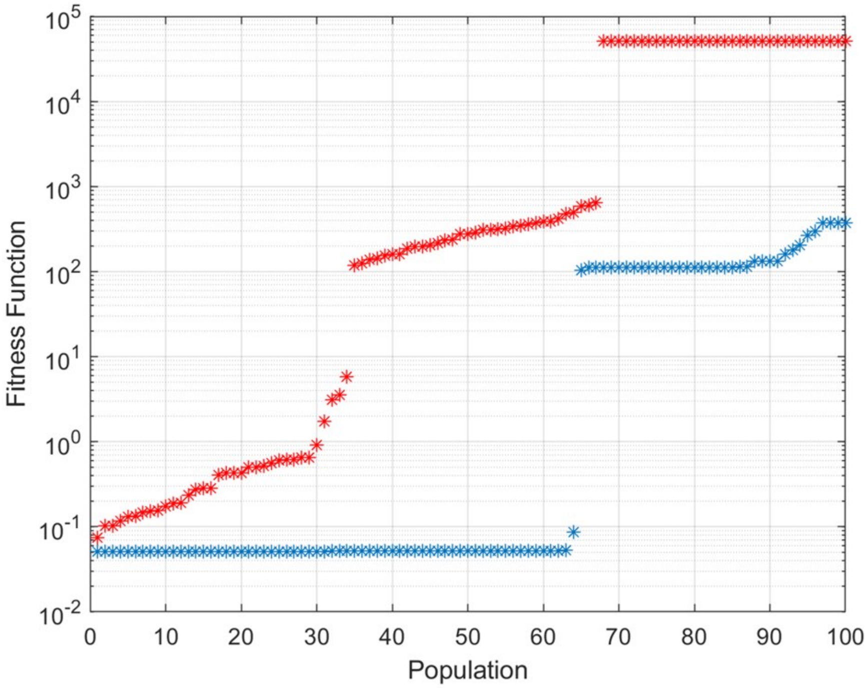

Fig. 6.15: Fitness function values for the initial and final GA populations of the V-bar hop scenario.

6.3 Conclusions

In this chapter we developed the STORM hybrid optimization tool to solve the stochastic constrained trajectory planning problem using a closed-loop linear covariance model. The approach focuses on determining the optimal trajectory by minimizing the ΔV dispersion while subject to a penalty for not achieving a specified true final position dispersion. Dynamic noise, sensor noise and constraints, navigation errors, and maneuver execution errors are all accounted for in determining the optimal solution.

To demonstrate this new approach's ability to find the global optimum, STORM solutions were compared to the global search method applied to the two-impulse scenarios in Chapter 5. For both the NMC resizing and V-bar hop scenarios, STORM successfully located the global optimum through it's hybrid approach of pairing a genetic algorithm with a nonlinear solver.

CHAPTER 7

MULTI-WAYPOINT RELATIVE PATH PLANNING WITH POSITION DISPERSION CONSTRAINTS

In search of more operationally relevant scenarios, we will now expand the STORM optimal path planning analysis to include transfers composed of multiple waypoints. These new scenarios will also employ the passive, angle-only relative navigation systems common to many modern RPO systems. Unlike the lidar-based navigation used to this point, the passive AON approach imposes additional complexity to the planning process due to difficulties in determining range [17] and the potential for optics being blinded by the sun or earth as described in Section 4.2.4.

Up to this point all of the scenarios analyzed have been two-impulse sequences consisting of a single waypoint located on the final orbit and whose position is uniquely determined by the transfer times. To expand the scope of this analysis and explore more complex scenarios, this section will focus on scenarios that include additional intermediate waypoints as defined in Section 4.12. In addition to the transfer times, the STORM analysis tool must determine the location of these intermediate waypoints to determine the optimal trajectory. The addition of these values to the decision variable complicates the optimization process but provides increased flexibility in determining optimal solutions and reducing final position dispersions.

Two three-impulse scenarios are evaluated in this chapter including the now familiar NMC resizing problem and a more complex scenario where the active vehicle transitions from a co-elliptic drift into an NMC around the RSO. To test STORM's performance, the scenarios will be evaluated for varying lighting conditions, final position dispersion constraints, and final times.

7.1 Three-Impulse NMC Resizing

In this section we will revisit the co-planar NMC resizing scenario described in Sections 4.12 and 5.1 but now for a three-impulse sequence. The updated scenario parameters are provided in Table 7.1 and the linear covariance simulation parameters are provided in Table 7.2. Since this scenario employs a passive, angles-only relative navigation scheme, the camera constraints are also defined in Table 7.2 where $\theta_{\oplus}^{lim}$ and $\theta_{\odot}^{lim}$ are the angle below which measurements are inhibited due to the earth and sun, respectively.

To evaluate the efficacy of the new STORM approach, we will look at three different cases with varying lighting conditions and fitness functions. The variations between the three cases are defined in Table 7.3. The first case will serve as the baseline by solving the scenario with the deterministic fitness function defined in Equation (6.3). As a reminder, the deterministic solution solves for the minimum ΔV solution without consideration for the stochastic properties generated by the linear covariance model. Stochastic properties are then generated using the good lighting conditions where the sun begins in the positive intrack direction, or 90 degrees off of the camera boresight and 45 degrees from the camera imaging limit. The STORM algorithm will also implement a maximum final position dispersion of 5 m for comparison purposes only. The second case will use this good lighting condition, final dispersion requirement, and utilize the constrained fitness function defined in Equation (6.1). Finally, the third case will start from a poor lighting condition where the sun is 43 degrees above the positive intrack vector, a mere 2 degrees from the camera

Table 7.1: Nominal trajectory inputs for NMC resizing scenario.

Parameter	Value	Units
$\hat{r}(t_0)$	$[-500,\ 0,\ 0]^T$	m
$\hat{v}(t_0)$	$[0,\ 1,\ 0]^T$	m/s
$\hat{r}(t_f)$	$[-250,\ 0,\ 0]^T$	m
$\hat{v}(t_f)$	$[0,\ 0.5,\ 0]^T$	m/s
ω	0.001	rad/s
t_f	$6283\ (= T)$	s

Table 7.2: NMC resizing scenario noise parameters and camera constraints.

Parameter	Value	Units
$C_0(rr)$ (3σ)	$(10)^2 I_{3\times3}$	m^2
$C_0(vv)$ (3σ)	$(0.01)^2 I_{3\times3}$	$\mathrm{m}^2/\mathrm{s}^2$
$C_0(rv)$ (3σ)	$0_{3\times3}$	m^2/s
Q_w (3σ)	$(10^{-9})^2 I_{3\times3}$	$\mathrm{m}^2/\mathrm{s}^3$
$S_{\Delta V}$ (3σ)	$(0.03\Delta\bar{V})^2 I_{3\times3}$	$\mathrm{m}^2/\mathrm{s}^2$
R_ν^{AON} (3σ)	$(10^{-3})^2 I_{3\times3}$	rad^2
R_ν^{lidar} (3σ)	n/a	n/a
Earth Exclusion, $\theta_\oplus^{lim}$	10	deg
Sun Exclusion, $\theta_\odot^{lim}$	45	deg

Table 7.3: Three-impulse NMC resizing scenario test cases.

Case	Initial Sun Vector $\hat{s}(t_0)$	Fitness Function J	Dispersion Requirement S_{lim}
Deterministic	$[0, 1, 0]^T$	ΔV	n/a
Good Lighting	$[0, 1, 0]^T$	$\Delta V^{3\sigma} + P_{\text{disp}}$	5 m
Poor Lighting	$[0.6820, 0.7314, 0]^T$	$\Delta V^{3\sigma} + P_{\text{disp}}$	5 m

imaging limit. These results will help to articulate one of the key advantages of this new optimization approach by being adaptive to real world limitations.

Detailed results for each case are presented in the following sections but are summarized here. The linear covariance results for the deterministic, good lighting, and poor lighting cases are given in Table 7.4. The optimal times and position of the intermediate waypoint in the LVLH frame for each case are given in Table 7.5. The three optimal trajectories, and the locations of their respective maneuvers, are depicted in Figure 7.1.

Table 7.4: LinCov results for the three-impulse NMC resizing scenario with final position requirement. $\Delta\bar{V} = 0.1250$ m/s.

Case	$\Delta V^{3\sigma}$ (m/s)	ΔV (m/s)	$\Delta V/\Delta\bar{V}$	D_f (m)
Deterministic	4.79235	0.12500	1.0000	21.98
Good Lighting	0.41167	0.21198	1.6958	5.00
Poor Lighting	1.14185	0.67547	5.4038	4.98

Table 7.5: Optimal time and waypoint results for the three-impulse NMC resizing scenario with final position requirement.

Case	$\Delta t_0/T$	$\Delta t_1/T$	$\Delta t_2/T$	R_x (m)	R_y (m)	R_z (m)
Deterministic	0	0.5	0.49936	375.0	293.7	-2.6
Good Lighting	0.26862	0.40440	0.25068	189.8	-409.1	0
Poor Lighting	0.08290	0.73943	0.17715	160.7	-177.1	22.9

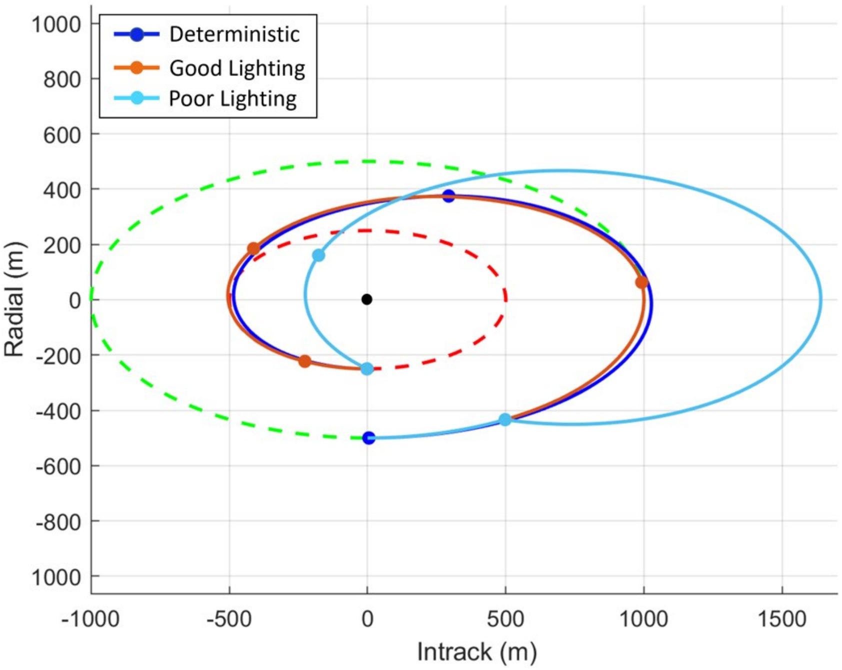

Fig. 7.1: Optimal trajectory and maneuver locations for the three-impulse NMC resizing scenarios.

7.1.1 Deterministic Optimal Solution

The optimal deterministic trajectory and maneuvers are shown in Figure 7.2. Overlaid on the trajectory are the regions where the angle measurements are available and lighting constraints are not violated. Angle measurement availability and violation of the Earth/Sun lighting constraints are shown in Figure 7.3. The sun angle never drops below the 45° limit.

The final true dispersion ellipse in the Intrack-Radial plane is shown in Figure 7.4. The long axis of the dispersion ellipse is more than fourfold larger than than the 5 m requirement.

The nominal ΔV and dispersions for the three maneuvers are depicted in Figure 7.5. It is interesting to note that there is almost no dispersion on the initial maneuver since it occurs at $t = 0$ and the navigation filter has yet to process any measurements. For this maneuver, $\Delta V_1 = 0.0329$ m/s and $\Delta V_1^{3\sigma} = 0.0016$ m/s. This would indicate that, regardless of the true state, the guidance algorithm only has the single *a priori* estimate for calculating the initial maneuver. Maneuvers 2 and 3, however, show significant dispersions with values orders of magnitude larger than the nominal ΔV's.

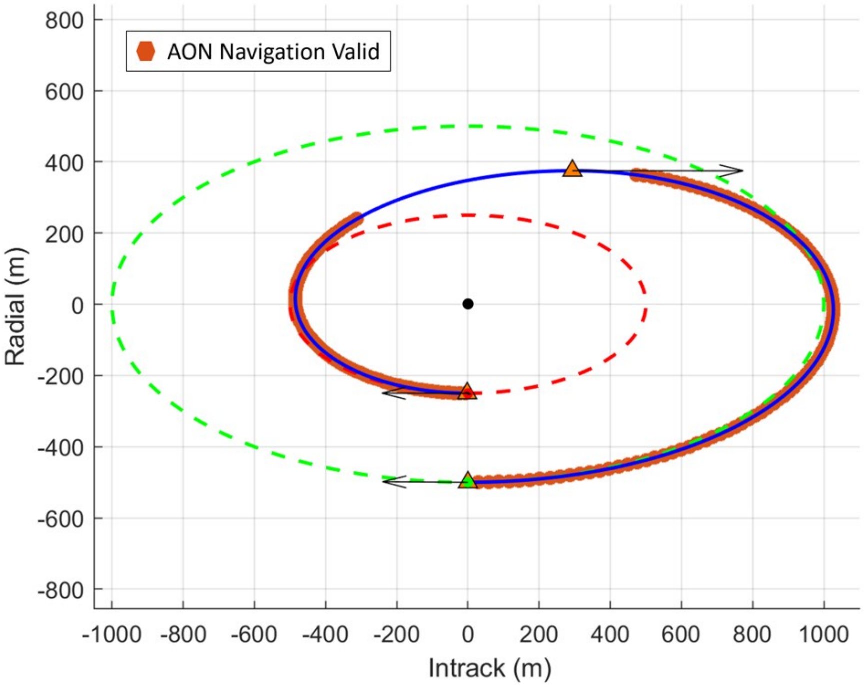

Fig. 7.2: Three-impulse NMC resizing scenario for the deterministic optimal solution.

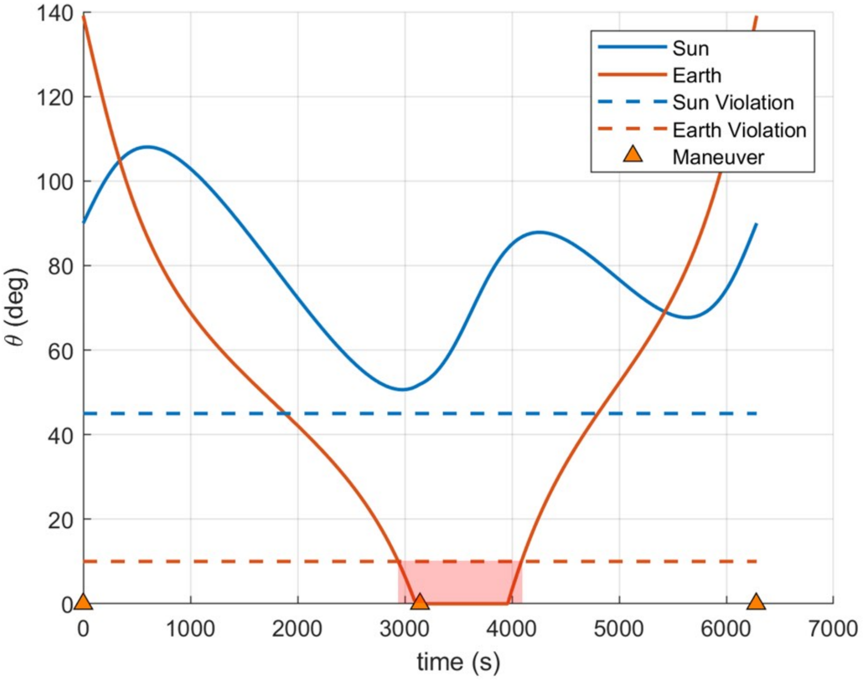

Fig. 7.3: Sun and earth angles for deterministic optimal solution of the three-impulse NMC resizing scenario.

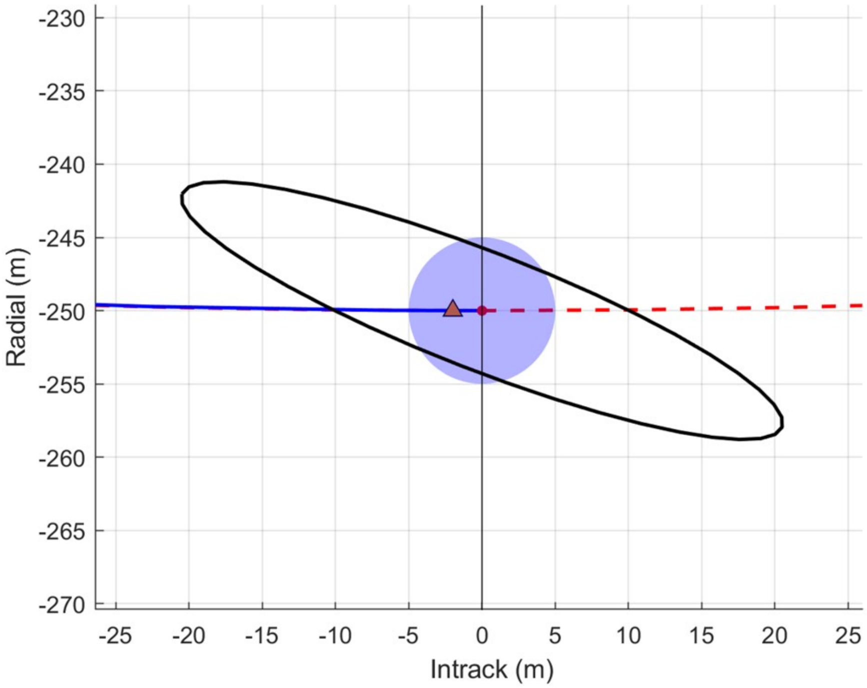

Fig. 7.4: Final true position dispersion for the deterministic optimal solution of the three-impulse NMC resizing scenario. Blue circle represents a desired final position dispersion constraint.

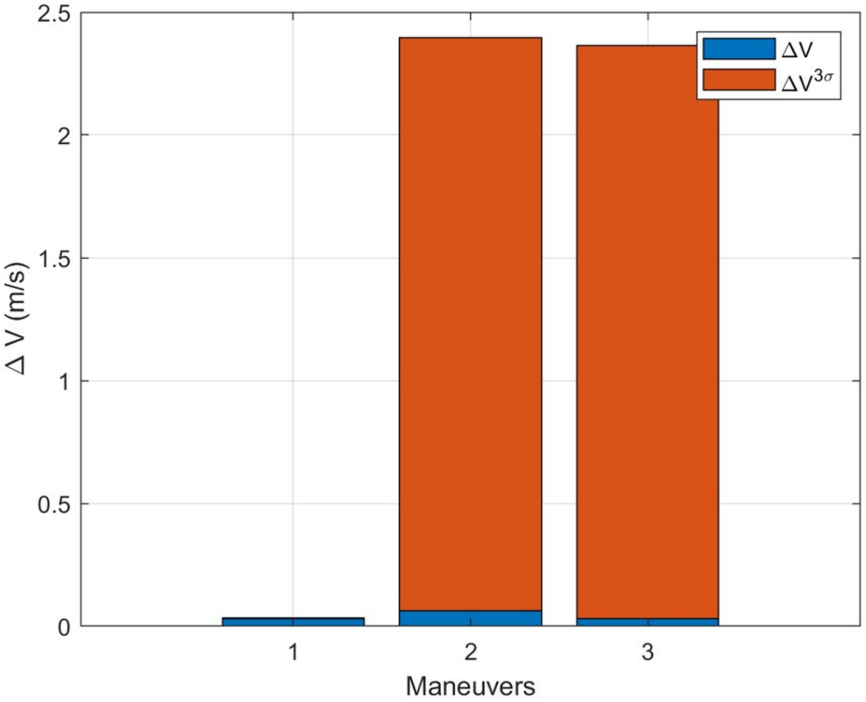

Fig. 7.5: ΔV and 3σ ΔV dispersion values for the deterministic optimal solution.

7.1.2 Constrained Optimal Solution - Good Lighting

The constrained optimal trajectory and maneuvers for the good lighting case are shown in Figure 7.6. Overlaid on the trajectory are the regions where the angle measurements are available and lighting constraints are not violated. Angle measurement availability and violation of the Earth/Sun lighting constraints are shown in Figure 7.7. The sun angle for this case never drops below the 45° sun exclusion limit.

The final true dispersion ellipse in the Intrack-Radial plane is shown in Figure 7.8. The long axis of the ellipse exactly meets the 5 m requirement. The nominal ΔV and dispersions for the three maneuvers are depicted in Figure 7.9.

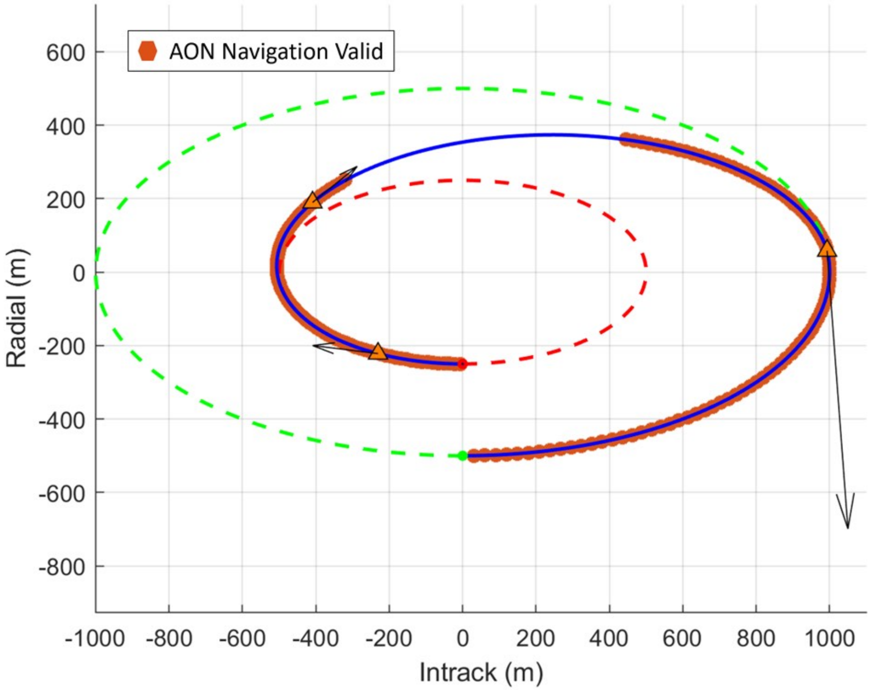

Fig. 7.6: Three-impulse NMC resizing scenario for the constrained optimal solution with good lighting. Navigation outage is due to earth angle violation.

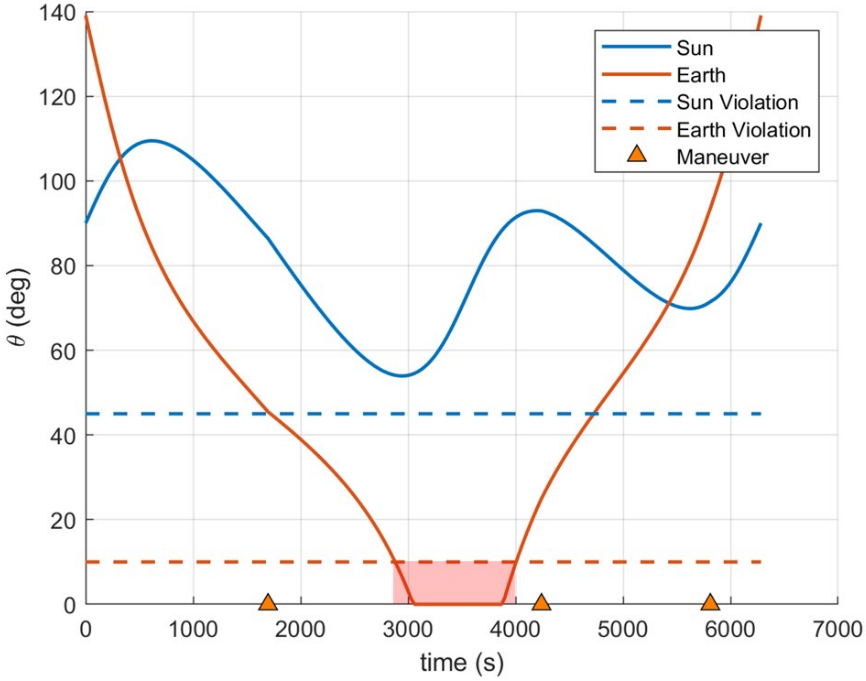

Fig. 7.7: Sun and earth angles for constrained optimal solution of the three-impulse NMC resizing scenario with good lighting.

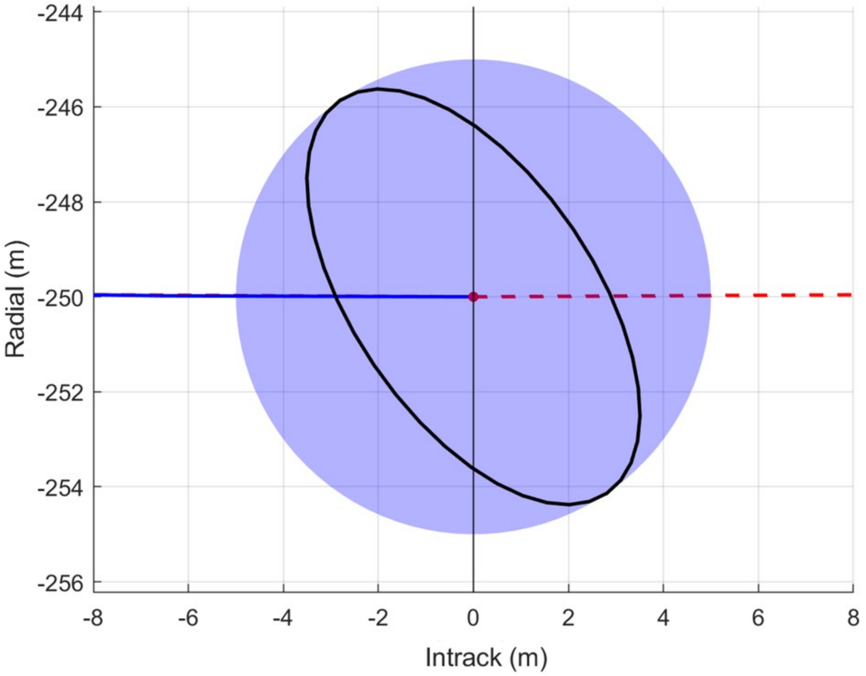

Fig. 7.8: Final true position dispersion for the deterministic optimal solution of the three-impulse NMC resizing scenario with good lighting. Blue circle represents a final position dispersion constraint.

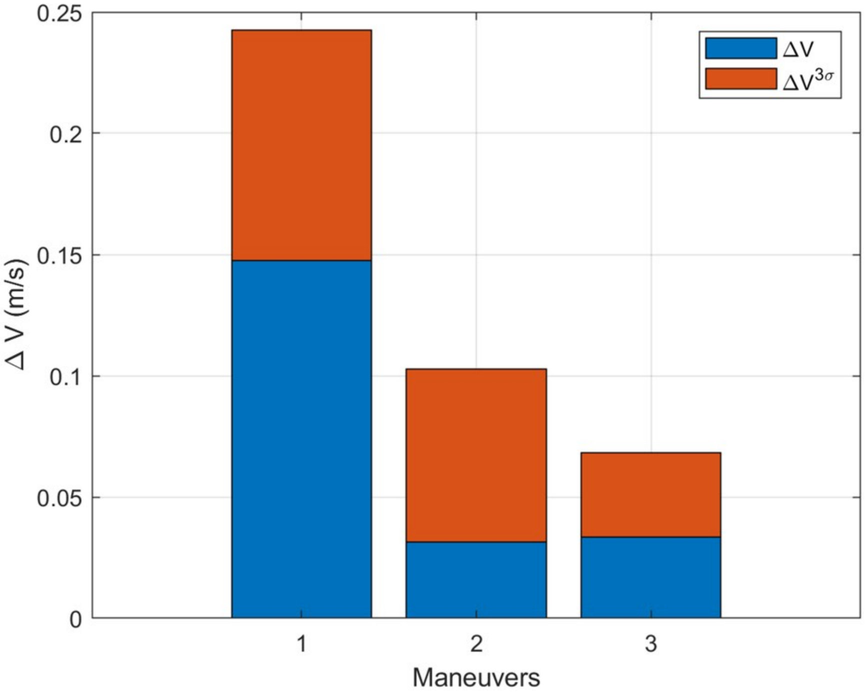

Fig. 7.9: ΔV and 3σ ΔV dispersion values for the constrained optimal solution with good lighting.

7.1.3 Constrained Optimal Solution - Poor Lighting

The constrained optimal trajectory and maneuvers for the poor lighting case are shown in Figure 7.10. Overlaid on the trajectory are the regions where the angle measurements are available and lighting constraints are not violated. Though not easily visible, there is a single valid navigation measurement just prior to the execution of the final maneuver, aiding the the maneuver accuracy in order to achieve the dispersion requirement. Angle measurement availability and violation of the Earth/Sun lighting constraints are shown in Figure 7.11.

The final true dispersion ellipse in the Intrack-Radial plane is shown in Figure 7.12. The long axis of the ellipse is just under the 5 m requirement. The nominal ΔV and dispersions for the three maneuvers are depicted in Figure 7.13.

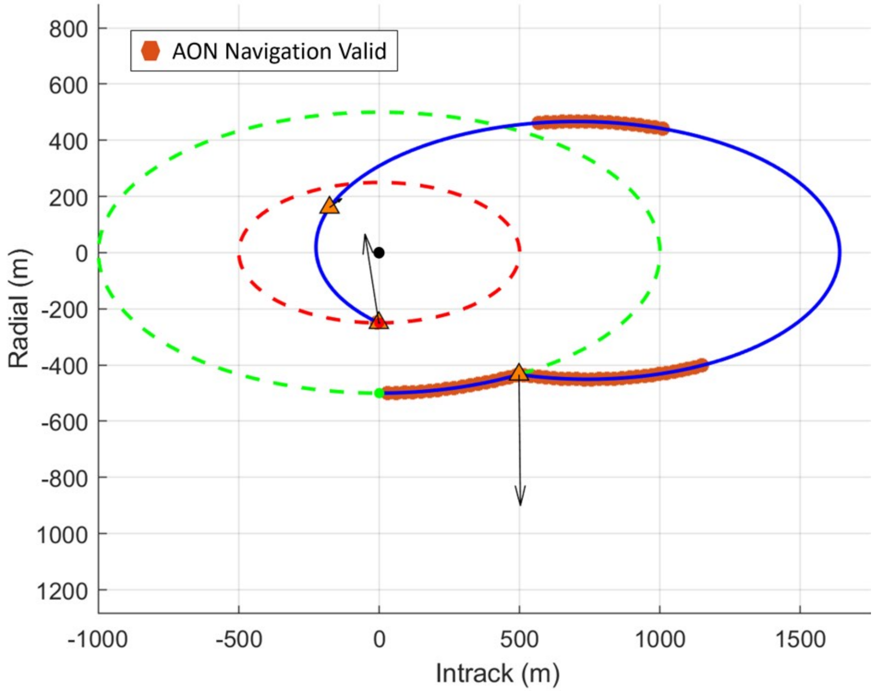

Fig. 7.10: Three-impulse NMC resizing scenario for the constrained optimal solution with poor lighting. Navigation outage due to earth and sun angle violations.

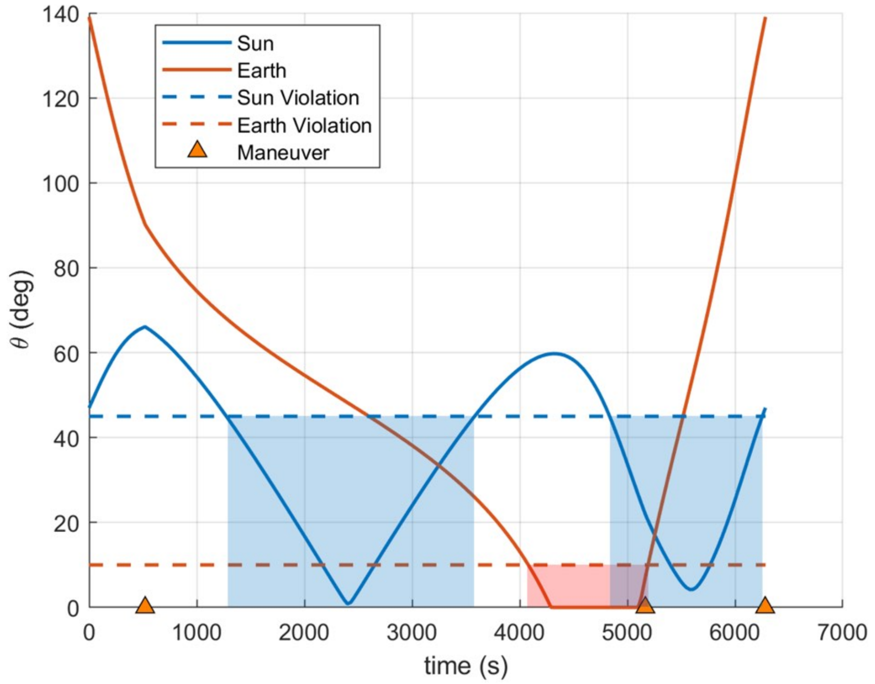

Fig. 7.11: Sun and earth angles for constrained optimal solution of the three-impulse NMC resizing scenario with poor lighting.

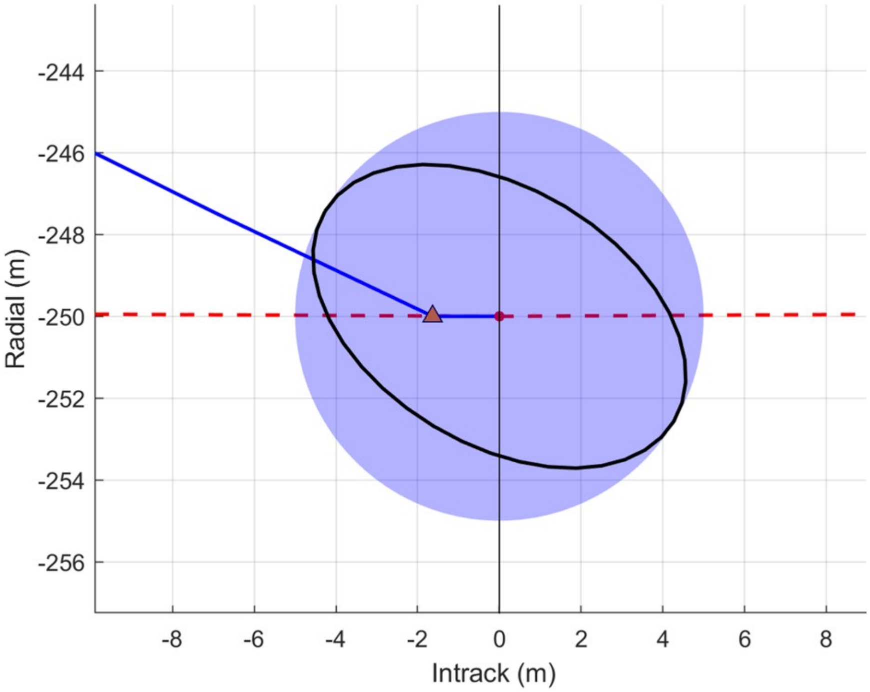

Fig. 7.12: Final true position dispersion for the deterministic optimal solution of the three-impulse NMC resizing scenario with poor lighting. Blue circle represents a final position dispersion constraint.

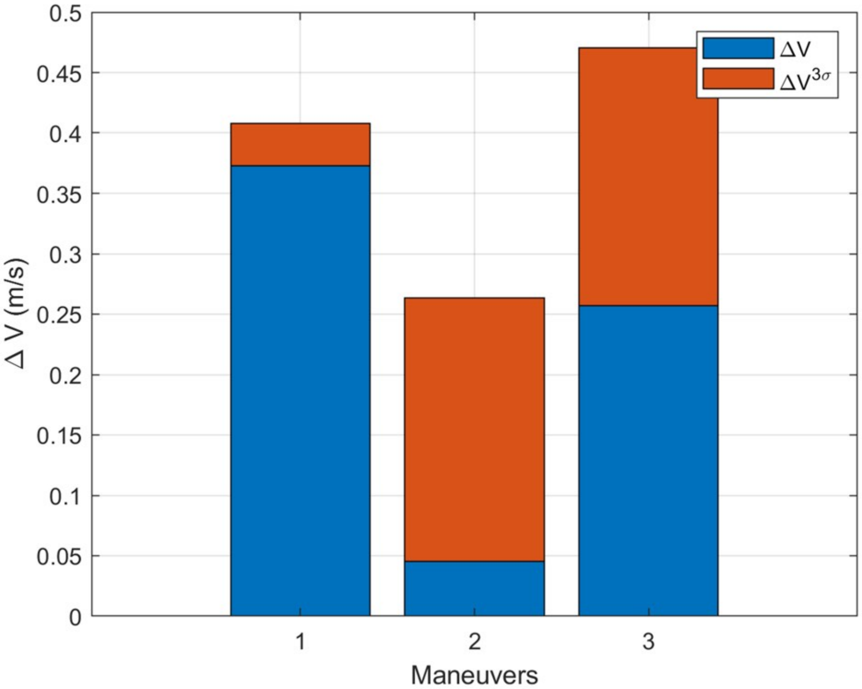

Fig. 7.13: ΔV and 3σ ΔV dispersion values for the constrained optimal solution with poor lighting.

7.2 Three-impulse Co-elliptic Drift to NMC

In this section we will introduce a new relative motion trajectory common in RPO operations, the co-elliptic drift. Sometime referred to as a straight-line drift, this trajectory appears to travel in a straight line along the Intrack direction with a constant Radial offset relative to the RSO when viewed in the LVLH frame. This trajectory is achieved when the active vehicle is in a circular orbit (similar to the RSO) but with a small difference in the orbit radius. Since there is no radial motion ($\ddot{x} = \dot{x} = 0$), the vehicle's motion in the LVLH frame is defined by Equation (2.26a):

$$\dot{y} = -\frac{3}{2}\omega x_0 \tag{7.1}$$

where $\dot{y}$ is the Intrack velocity and x_0 is the constant Radial offset. With a positive radial offset the vehicle will travel in the negative Intrack direction corresponding to a slower orbit. Conversely, with a negative radial offset, where the vehicle is below the RSO, a positive Intrack velocity achieved.

For this scenario the vehicle begins in a co-elliptic drift that is 750 m below and 7065 m behind the RSO. Assuming $\omega = 0.001$, the vehicle begins with an Intrack velocity of 1.1250 m/s and will travel 7068 m each orbital period placing it directly beneath the RSO after 6,283 seconds. The vehicle must transfer from this co-elliptic drift into an NMC as depicted in Figure 7.14. Like the co-elliptic drift, the 150×300 m NMC does not have any Crosstrack motion for a planer scenario. The final vehicle position is located at local perigee (i.e. bottom) of the NMC. This scenario represents a common rendezvous approach where a vehicle arrives at an RSO and must transfer to a non-drifting trajectory to begin proximity operations such as inspection or docking.

The scenario parameters, including initial and final states, are given in Table 7.6. The transfer time for the scenario is set to twice the orbital period. The linear covariance parameters, initial sun vector, and camera constraints used in the STORM algorithm are detailed in Table 7.7.

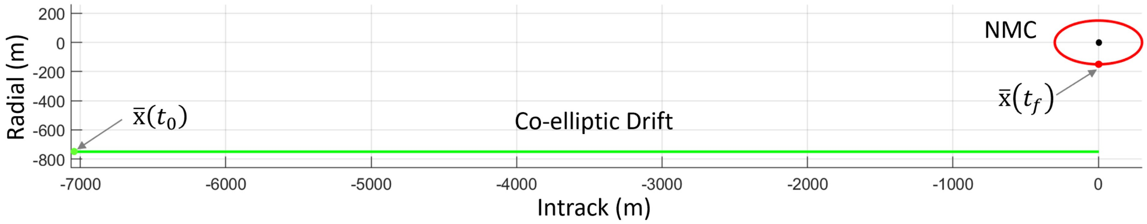

Fig. 7.14: Co-elliptic and NMC trajectories.

Table 7.6: Nominal trajectory inputs for co-elliptic to NMC scenario.

Parameter	Value	Units
$\hat{r}(t_0)$	$[-750,\ -7065,\ 0]^T$	m
$\hat{v}(t_0)$	$[0,\ 1.1250,\ 0]^T$	m/s
$\hat{r}(t_f)$	$[-150,\ 0,\ 0]^T$	m
$\hat{v}(t_f)$	$[0,\ 0.3,\ 0]^T$	m/s
ω	0.001	rad/s
t_f	12566 $(= 2T)$	s

Table 7.7: Co-elliptic to NMC scenario noise parameters and camera constraints.

Parameter	Value	Units
$C_0(rr)\ (3\sigma)$	$(10)^2 I_{3\times3}$	m^2
$C_0(vv)\ (3\sigma)$	$(0.01)^2 I_{3\times3}$	$\mathrm{m}^2/\mathrm{s}^2$
$C_0(rv)\ (3\sigma)$	$0_{3\times3}$	m^2/s
$Q_w\ (3\sigma)$	$(10^{-9})^2 I_{3\times3}$	$\mathrm{m}^2/\mathrm{s}^3$
$S_{\Delta V}\ (3\sigma)$	$(0.03\Delta\bar{V})^2 I_{3\times3}$	$\mathrm{m}^2/\mathrm{s}^2$
$R_\nu^{AON}\ (3\sigma)$	$(10^{-3})^2 I_{3\times3}$	rad^2
$R_\nu^{lidar}\ (3\sigma)$	n/a	n/a
Earth Exclusion, $\theta_\oplus^{lim}$	10	deg
Sun Exclusion, $\theta_\odot^{lim}$	45	deg
Initial Sun Vector, $\hat{s}(t_0)$	$[-1,0,0]^T$	n/a

Table 7.8: Three-impulse co-elliptic to NMC scenario test cases.

Case	Fitness Function J	Dispersion Requirement S_{lim}	Final Time t_f
Deterministic	ΔV	n/a	2T
Constrained	$\Delta V^{3\sigma} + P_{\text{disp}}$	15 m	2T
Constrained: FFT	$\Delta V^{3\sigma} + P_{\text{disp}}$	15 m	Free
Constrained: Tight	$\Delta V^{3\sigma} + P_{\text{disp}}$	3 m	2T

In this scenario we will evaluate four unique cases, defined in Table 7.8, to demonstrate different aspects of the STORM algorithm. The first case serves as the scenario baseline, optimizing on the nominal, deterministic ΔV for comparison to later constrained cases. For this case, and all other cases in the scenario, the sun begins in the negative radial direction so that the vehicle enters the NMC with ideal lighting conditions.

The second case, identified as *constrained*, will add a 15 m dispersion requirement on the final position dispersion and optimize on the stochastic mean ΔV plus 3σ dispersion, $\Delta V^{3\sigma}$. The third case will again evaluate the scenario with these constraints but with a free final time, allowing the STORM algorithm to include this parameter in its optimization space to determine an improvement to the constrained solution. The final case, identified as *tight constraint*, returns to the fixed final time but reduces the final position dispersion requirement to 3 m. This case helps to demonstrate STORM's ability to meet tight requirements that may be part of an operational mission.

As before, the results for the scenario's four cases are summarized in this section with individual details and figures provided in subsequent sections. Tables 7.9 provides the ΔV and 3σ ΔV dispersions, the final true position dispersion, and the solved-for transfer times of each case. Table 7.10 provides the solved-for waypoint location of each case. The trajectories and maneuver locations for each case for both the Radial-Intrack and Crosstrack-Intrack planes are depicted in Figures 7.15 and 7.16, respectively.

It is interesting to observe the rather large position dispersion for the deterministic case. An dispersion of 153 m is an unacceptable error when the vehicle is only supposed to

Table 7.9: LinCov results for the three-impulse co-elliptic to NMC scenario with final position requirement. $\Delta\bar{V} = 0.3750$ m/s.

Case	$\Delta V^{3\sigma}$ (m/s)	ΔV (m/s)	$\Delta V/\Delta\bar{V}$	D_f (m)
Deterministic	0.50895	0.37500	1.0000	153.1
Constrained	0.56363	0.37539	1.0010	15.0
FFT	0.54446	0.37583	1.0022	3.9
Tight	3.20037	0.91690	2.4451	3.0

Table 7.10: Optimal time and waypoint results for the three-impulse co-elliptic to NMC scenario with final position requirement. T_c is an optimization parameter for the free final time case only.

Case	$\Delta t_0/T$	$\Delta t_1/T$	$\Delta t_2/T$	$\Delta t_c/T$	R_x (m)	R_y (m)	R_z (m)
Deterministic	0.36022	0.64881	0.71272	0.27825	-390.5	-1675.0	0.0
Constrained	0.20123	1.2358	0.33588	0.22701	-410.6	-423.4	0.8
Free Final Time	0.06207	1.21787	0.32819	0.04551	-514.9	-739.2	-0.02
Tight Constraint	0.14695	1.76345	0.07155	0.01805	-16.4	-75.1	27.0

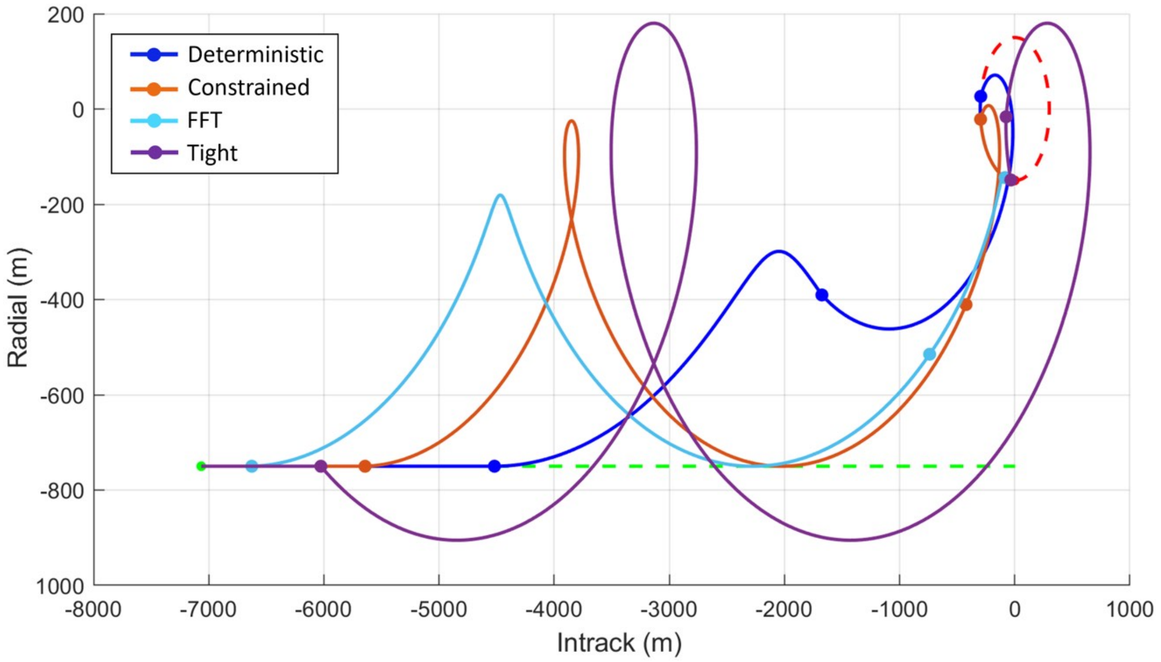

Fig. 7.15: Optimal trajectory and maneuver locations for the three-impulse co-elliptic to NMC scenarios in the Radial-Intrack plane. Axes are not equal to help clarify the trajectories.

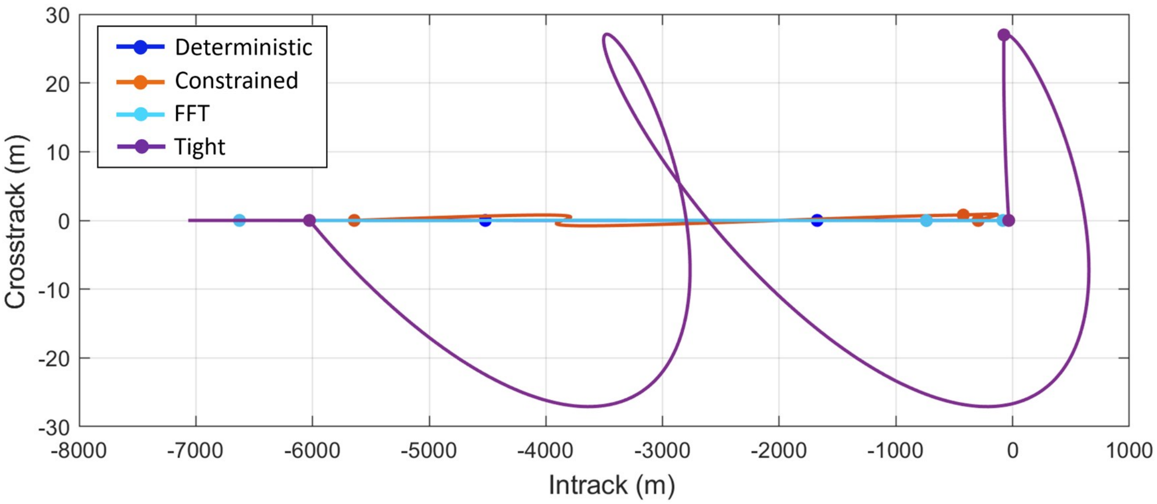

Fig. 7.16: Optimal trajectory and maneuver locations for the three-impulse co-elliptic to NMC scenarios in the Crosstrack-Intrack plane. Axes are not equal to help clarify the trajectories.

be 150 m away from the RSO. For small increases in mean ΔV and $\Delta V^{3\sigma}$, the constrained solution provides a 10x reduction in the final position dispersion while maintaining the final state and specified final sun angle.

Another key observation is that the free final time case has similar mean ΔV and $\Delta V^{3\sigma}$ values to the constrained case but with a significantly smaller final dispersion. By modifying the final time ($1.65T$ vs $2T$) the final dispersion is nearly as small as for the tight constraint. The tight constraint, however, requires a 2.4x increase in mean ΔV and a 5.7x increase in $\Delta V^{3\sigma}$ over those determined for the constrained case.

While all scenarios analyzed to this point have been in the full three dimensions of the LVLH frame, the tight constraint case for this scenario is the first time a significant Crosstrack component was applied to the planned trajectory. The first maneuver generates about 27 m of Crosstrack motion when targeting the intermediate waypoint. This motion helps to generate range observability for the angles-only navigation critical to meeting the final position dispersion requirement. By comparison, the first maneuver in the constrained cases generates 1 m of Crosstrack. In many RPO scenarios, a small amount of Crosstrack motion can be useful for adjusting the lighting angles to more favorable values. Additionally, the changes in Crosstrack motion can serve as effective observability maneuvers to reduce navigation errors [17].

7.2.1 Deterministic Optimal Solution

The optimal deterministic trajectory and maneuvers are shown in Figure 7.17. Overlaid on the trajectory are the regions where the angle measurements are available and lighting constraints are not violated. The angles-only solutions are unavailable across certain time frames due to two violations of the sun constraint as shown in Figure 7.18. The earth angle for this case never drops below the 10° threshold.

The final true dispersion ellipse in the Intrack-Radial plane is shown in Figure 7.19. While the long axis of the dispersion ellipse, at 153 m, is significantly larger than the desired constraint, the short axis of the dispersion also fails to meet the 15 m value used for the constrained case.

The nominal ΔV and dispersions for the three maneuvers are depicted in Figure 7.20. The maneuvers, as depicted in Figure 7.17, are all in the Intrack direction. This was also the case for the deterministic optimal solution to the three-impulse NMC resizing scenario.

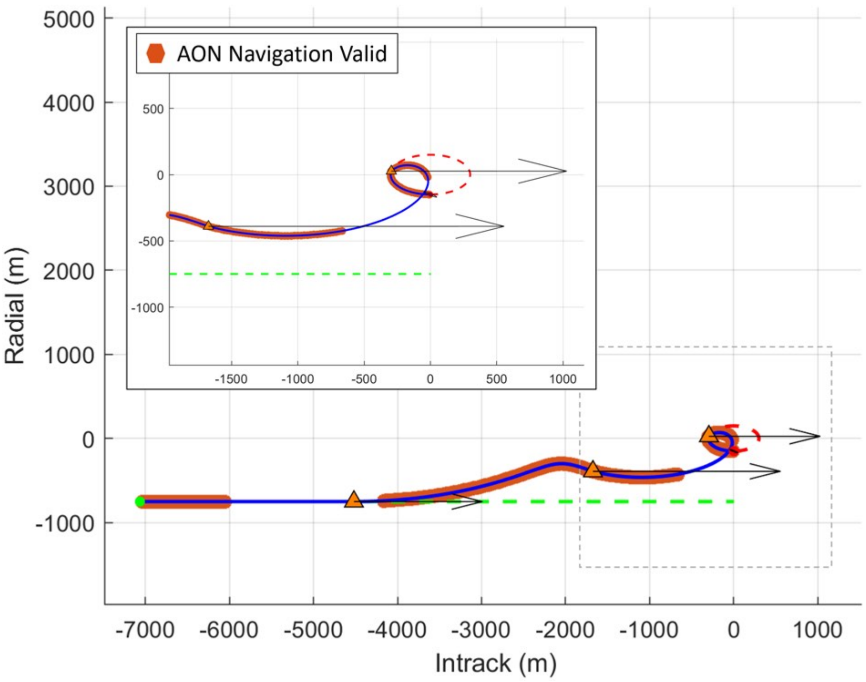

Fig. 7.17: Three-impulse co-elliptic to NMC scenario for the deterministic optimal solution.

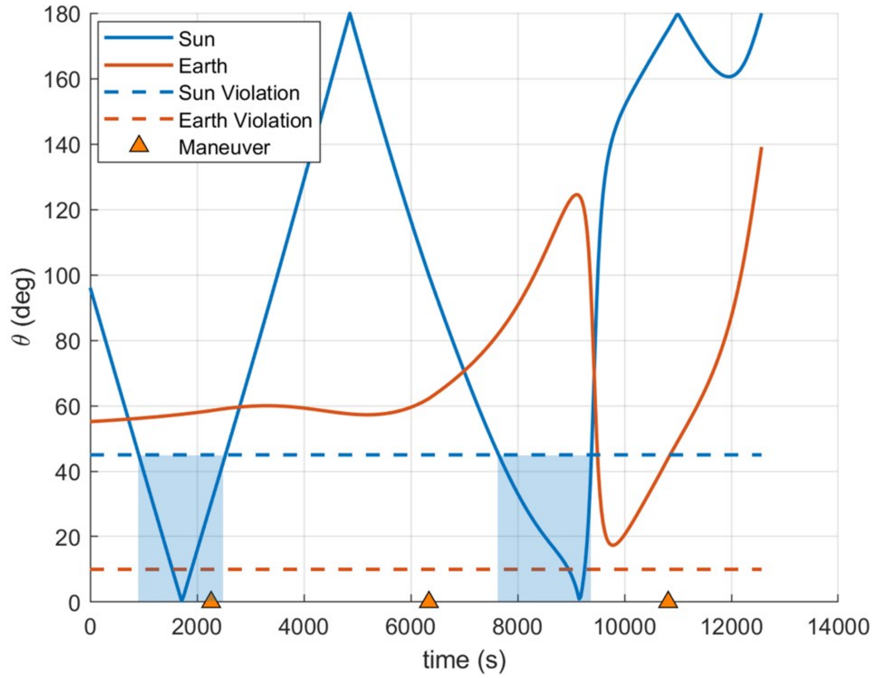

Fig. 7.18: Sun and earth angles for deterministic optimal solution of the three-impulse co-elliptic to NMC scenario.

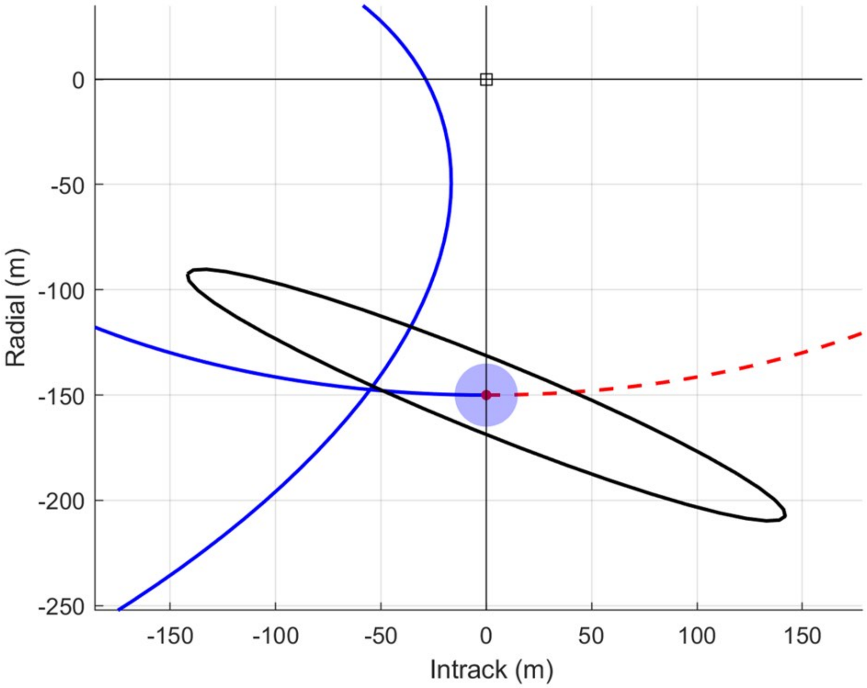

Fig. 7.19: Final true position dispersion for the deterministic optimal solution of the three-impulse co-elliptic to NMC scenario. Blue circle represents a 15 m dispersion.

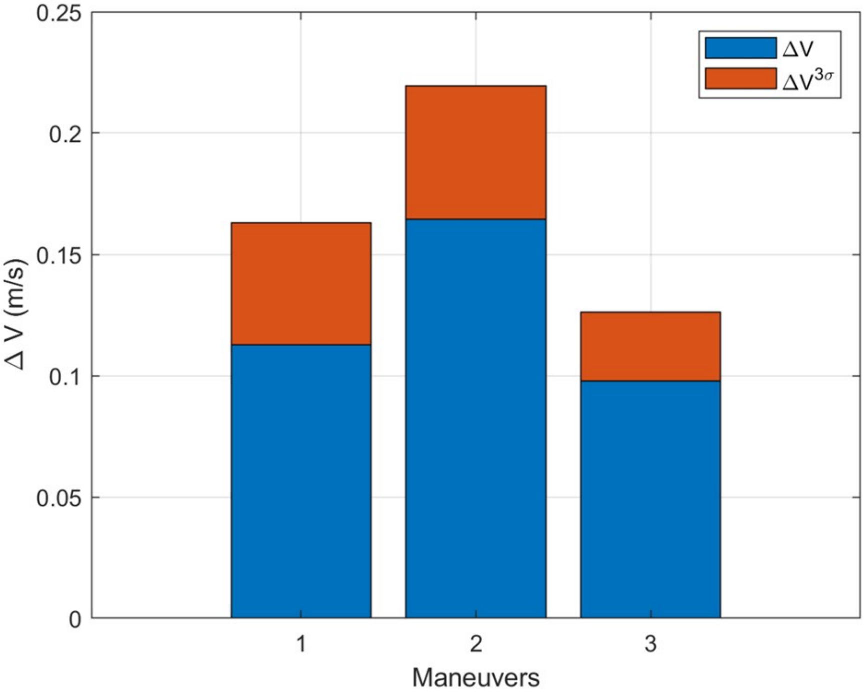

Fig. 7.20: ΔV and 3σ ΔV dispersion values for the deterministic optimal solution.

7.2.2 Constrained Optimal Solution

The constrained optimal trajectory and maneuvers are shown in Figure 7.21. Overlaid on the trajectory are the regions where the angle measurements are available and lighting constraints are not violated. The angles-only solutions are unavailable across certain time frames due to violations of the sun constraint as shown in Figure 7.22. The earth angle for this case never drops below the 10° threshold.

The final true dispersion ellipse in the Intrack-Radial plane is shown in Figure 7.23. The dispersion just meets the 15 m requirement.

The nominal ΔV and dispersions for the three maneuvers are depicted in Figure 7.24. We will note that the second maneuver has a small nominal value compared to the 3σ dispersion. This combination is typical of correction maneuvers used to reduce trajectory dispersions.

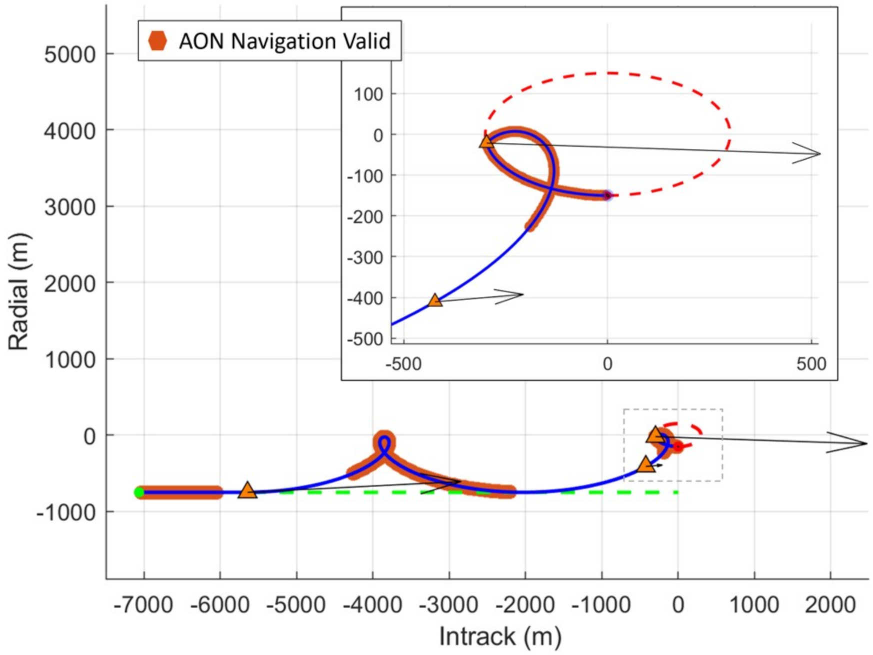

Fig. 7.21: Three-impulse co-elliptic to NMC scenario path for the constrained optimal solution.

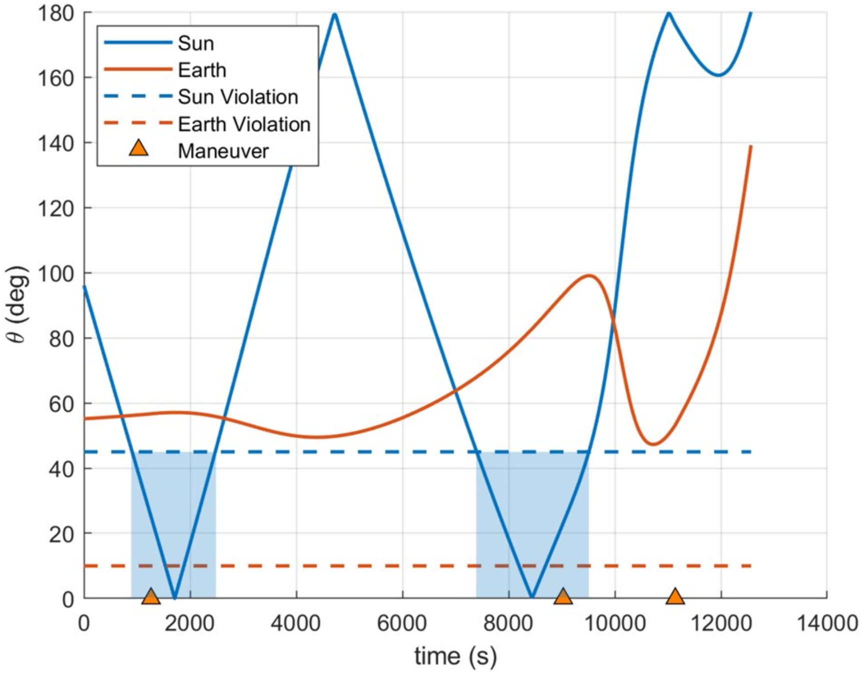

Fig. 7.22: Sun and earth angles for deterministic optimal solution of the three-impulse co-elliptic to NMC scenario.

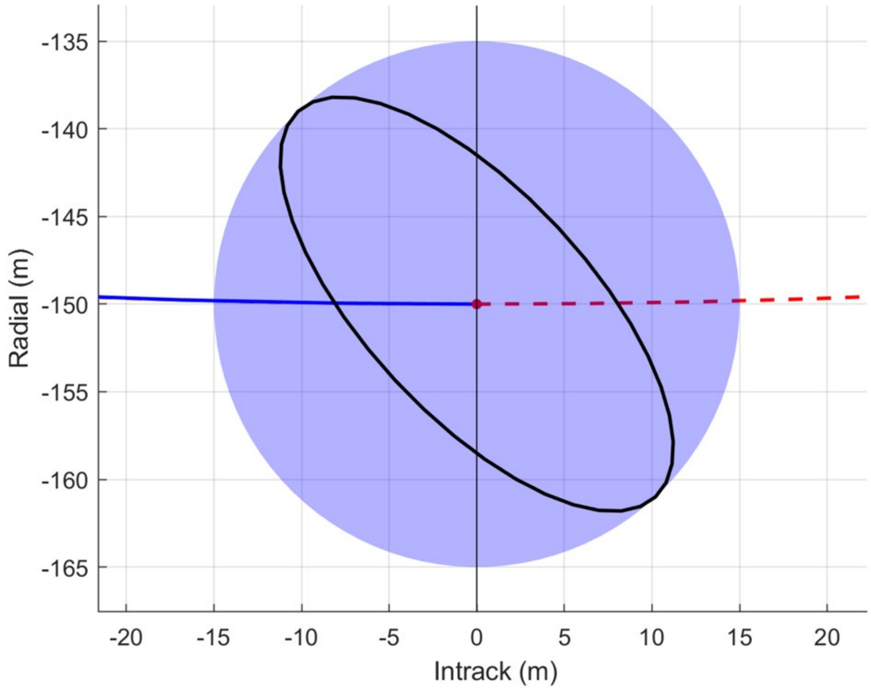

Fig. 7.23: Final true position dispersion for the constrained optimal solution of the three-impulse co-elliptic to NMC scenario. Blue circle represents a 15 m dispersion.

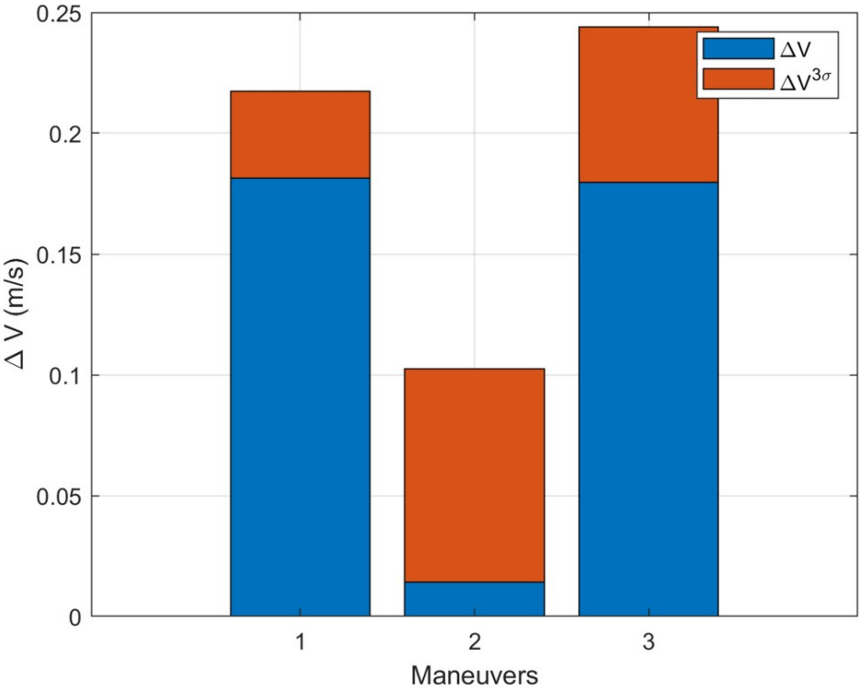

Fig. 7.24: ΔV and 3σ ΔV dispersion values for the constrained optimal solution.

7.2.3 Constrained Optimal Solution - Free Final Time

The free final time optimal trajectory and maneuvers are shown in Figure 7.25. Overlaid on the trajectory are the regions where the angle measurements are available and lighting constraints are not violated. The angles-only solutions are unavailable across certain time frames due to violations of the sun constraint as shown in Figure 7.26. The earth angle for this case never drops below the 10° threshold.

Given flexibility in the final arrival time, the free final time case reached the final NMC state after 10390 seconds or $1.654T$, as seen in Table 7.9. One significant effect of this arrival time is the change in the NMC lighting angle compared to the other cases. Whereas the other trajectories allowed for the vehicle to arrive at the final state with the sun at its back, the this case arrives with the sun with the sun just 55° off of the sensor boresight.

The final true dispersion ellipse in the Intrack-Radial plane is shown in Figure 7.27. The dispersion is well under the 15 m requirement. Since the solution was not limited by the final position dispersion constraint, the scenario was rerun without the constraint to assess the solution. The unconstrained solution resulted in $\Delta V^{3\sigma} = 0.43816$ m/s, $\Delta V = 0.37574$ m/s, and $D_f = 30.9$ m. One would typically expect to find the same solution when removing the constraint. However, this result would indicate that all of the local solutions near the unconstrained global optimum violate the constraint and finding the constrained solution requires searching in the vicinity of a different minima. These results were verified by achieving consistent results across multiple reruns of these two conditions.

The nominal ΔV and dispersions for the three maneuvers are depicted in Figure 7.28. The second maneuver again shows the characteristics of a mid-course correction maneuver as seen in the constrained case.

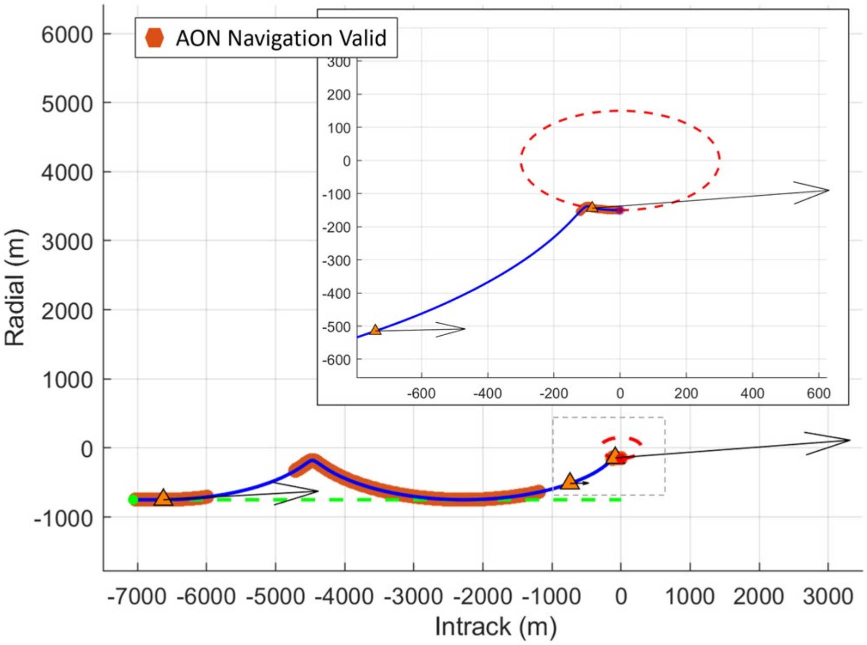

Fig. 7.25: Three-impulse co-elliptic to NMC scenario for the free final time optimal solution.

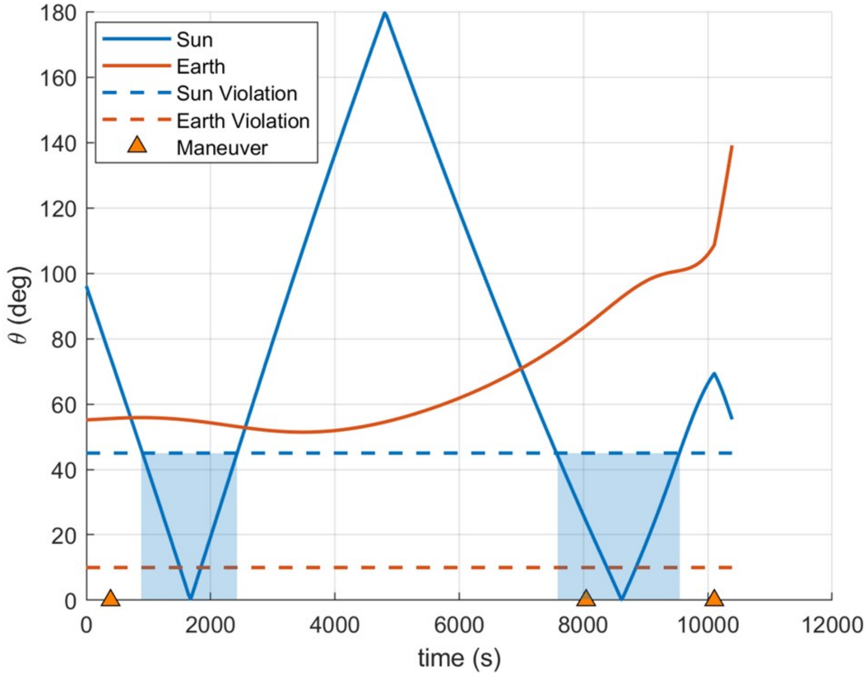

Fig. 7.26: Sun and earth angles for the free final time optimal solution of the three-impulse co-elliptic to NMC scenario.

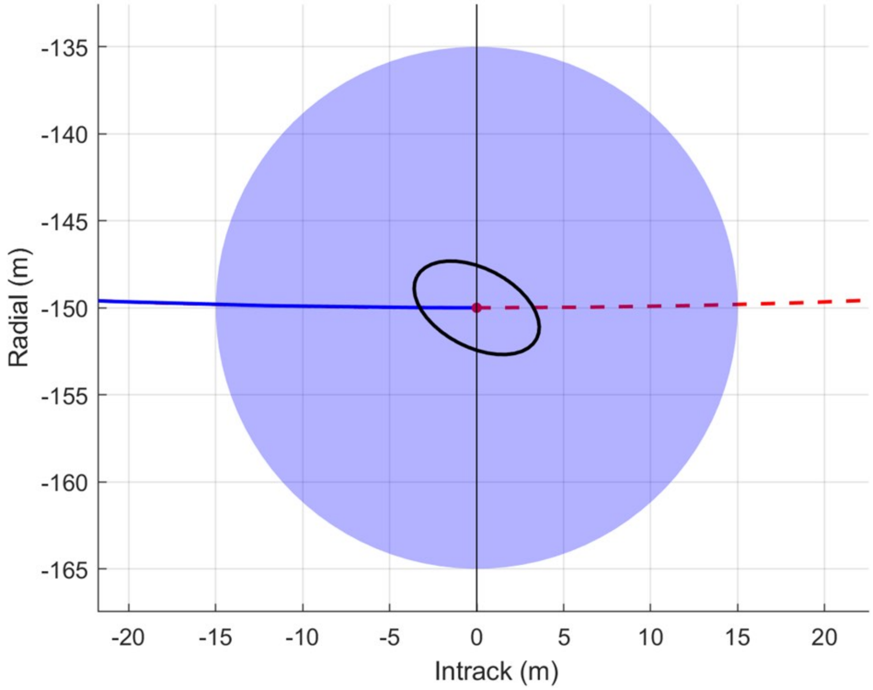

Fig. 7.27: Final true position dispersion for the free final time optimal solution of the three-impulse co-elliptic to NMC scenario. Blue circle represents a 15 m dispersion.

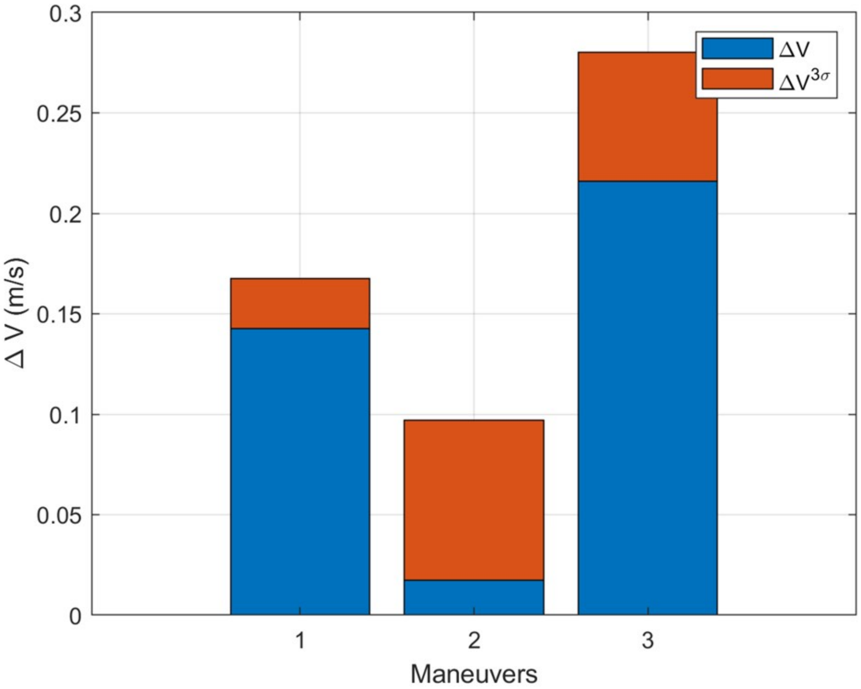

Fig. 7.28: ΔV and 3σ ΔV dispersion values for the free final time optimal solution.

7.2.4 Constrained Optimal Solution - Tight Constraint

The tightly constrained optimal trajectory and maneuvers are shown in Figure 7.29. Overlaid on the trajectory are the regions where the angle measurements are available and lighting constraints are not violated. The angles-only solutions are unavailable across certain time frames due to two violations of the sun constraint and a violation of the earth constraint as shown in Figure 7.30.

The final true dispersion ellipse in the Intrack-Radial plane is shown in Figure 7.31. The dispersion just meets the 3 m requirement and nearly fills the allowed space. To achieve this tight constraint, the final two maneuvers occur closely together near the final time and after all lighting violations have passed. This allows the angles-only navigation to accurately estimate the vehicle state that, in turn, provides the requisite knowledge required for accurate maneuver guidance calculations.

The nominal ΔV and dispersions for the three maneuvers are depicted in Figure 7.32. Both the second and third maneuvers show the characteristics of mid-course correction maneuvers.

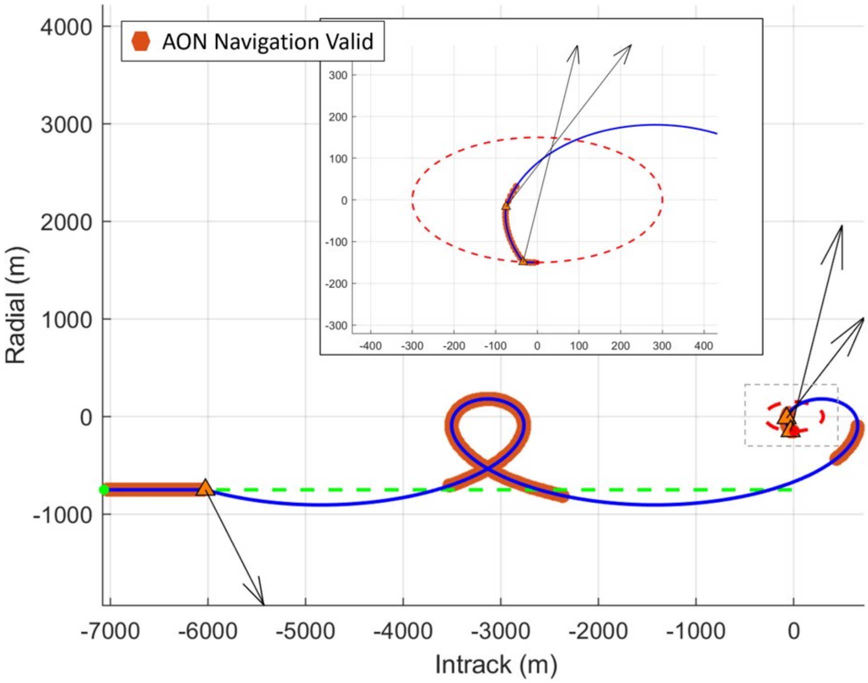

Fig. 7.29: Three-impulse co-elliptic to NMC scenario for the tightly constrained optimal solution.

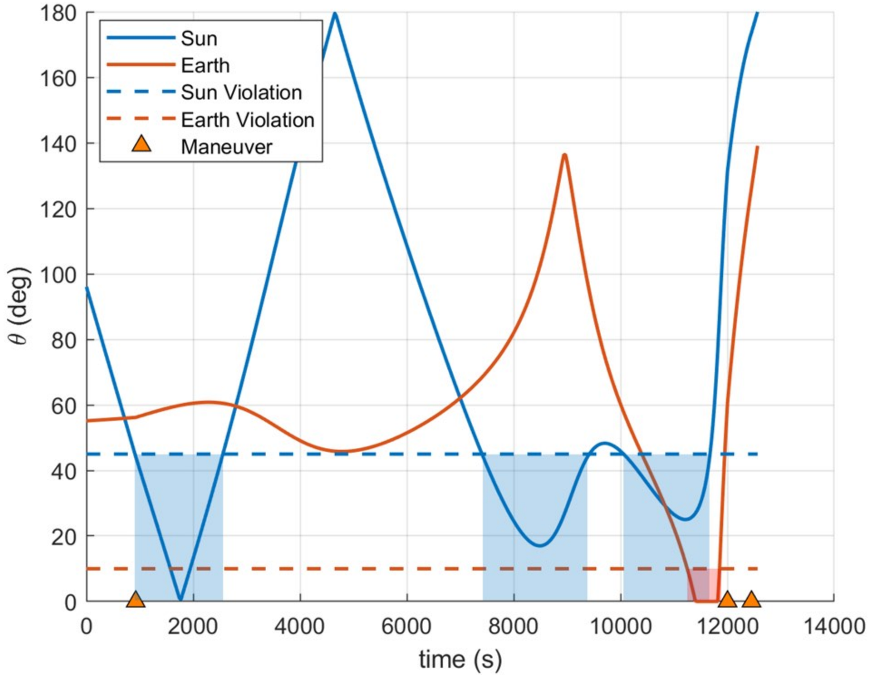

Fig. 7.30: Sun and earth angles for tightly constrained optimal solution of the three-impulse co-elliptic to NMC scenario.

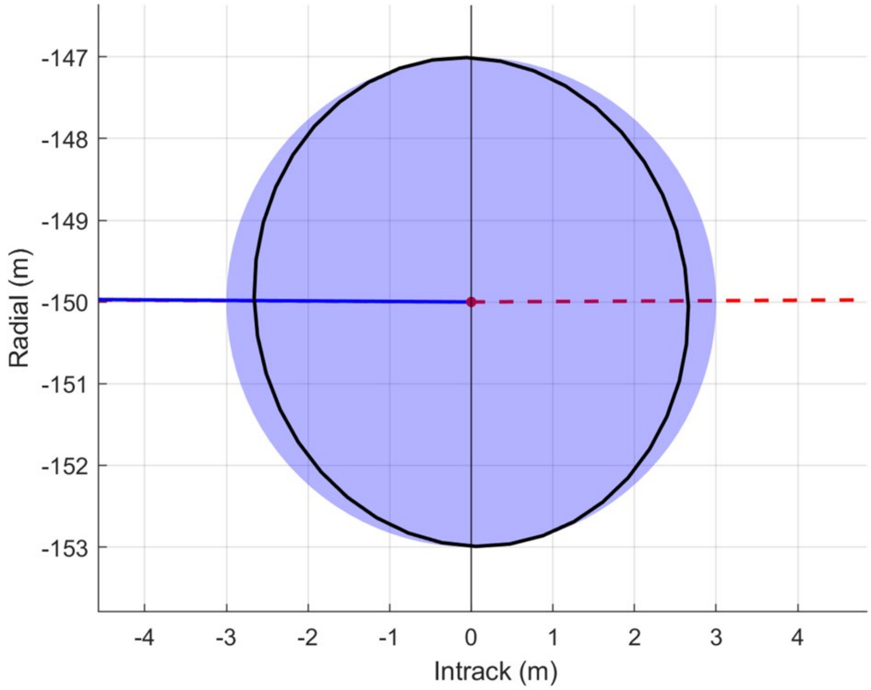

Fig. 7.31: Final true position dispersion for the tightly constrained optimal solution of the three-impulse co-elliptic to NMC scenario. Blue circle represents a 3 m dispersion.

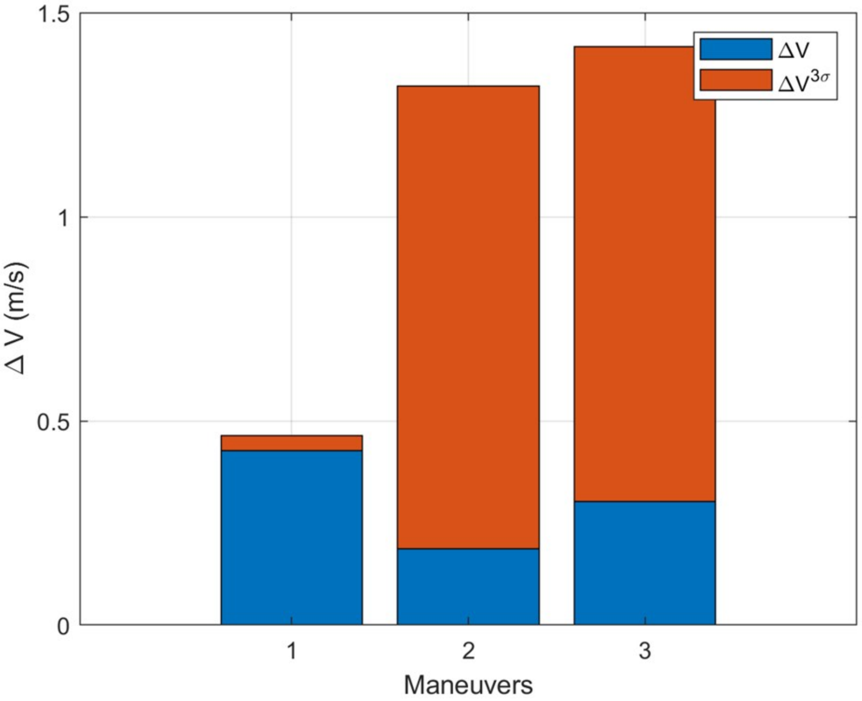

Fig. 7.32: ΔV and 3σ ΔV dispersions for the tightly constrained optimal solution.

7.3 Multi-Waypoint Solutions

For simplicity, only three-impulse scenarios were considered for analysis in this Chapter to this point. For comparisons of 3, 4, and multi-impulse rendezvous options using linear covariance techniques, the reader is directed to the initial work by the author [108] and Jin [107]. The deterministic approach to these multi-impulse scenarios is addressed in the expansive work by Prussing [116–118].

To verify performance, and the ability of the STORM algorithm to converge to a solution, the constrained NMC resizing scenario with good lighting was rerun for up to a 9-impulse solution. The key results of mean ΔV, total $\Delta V^{3\sigma}$, and final position dispersion, D_f, are depicted in Figure 7.33. While the 3, 4, 5, and 6-impulse solutions all meet the position dispersion requirement and have low total $\Delta V^{3\sigma}$ dispersion values, the three-impulse solution has the smallest fitness function value. It is interesting to note that increasing the

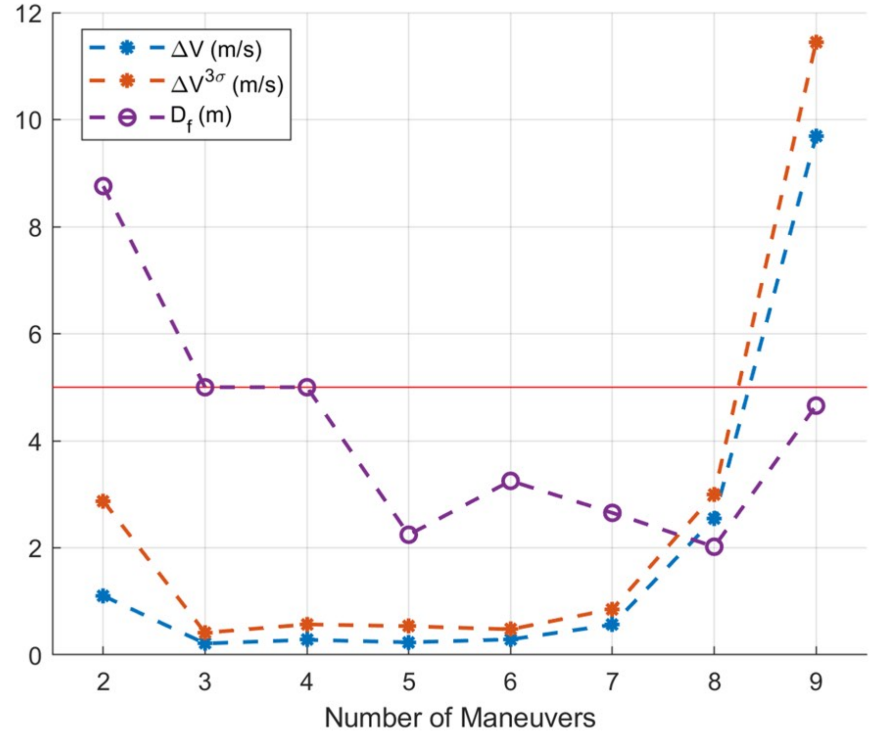

Fig. 7.33: Comparison of mean ΔV, total $\Delta V^{3\sigma}$, and D_f solutions of the NMC resizing scenario using multiple waypoints. Red line represents the 5 m position dispersion requirement.

number of maneuvers can help to reduce the final true position dispersion. The two-impulse solution is the only solution that does not meet the 5 m position dispersion requirement.

7.4 Conclusions

The STORM analysis tool was used to evaluate three-impulse scenarios where the algorithm must solve for the location of an intermediate waypoint in addition to the related transfer time parameters. For the three-impulse NMC resizing, different lighting conditions were applied to demonstrate the effects of navigation performance on the optimal trajectory while still meeting mission constraints. In the most extreme case, the STORM algorithm was able to achieve the mission objectives with the sun starting (and finishing) just 2° off of the navigation sensor's sun exclusion angle. The ability to meeting difficult mission objectives, such as a tough lighting scenario, provides mission planners the ability to construct optimal mission plans by trading mission objectives against fuel budgets without guesswork.

The more complex co-elliptic drift to NMC scenario was also analyzed to demonstrate additional aspects of STORM for solving complex, operationally relevant mission planning tasks. Results showed that small changes in maneuver locations or the final time can significantly improve the mission performance with minimal increases to fuel consumption when compared to the deterministic solution. The ability to solve the free final time planning problem was also demonstrated.

While all of the STORM solutions showed significant improvements in mission performance over the deterministic solutions, no consideration was made for the critical areas of spacecraft safety and collision avoidance. In several cases the trajectory brings the vehicle unnecessarily close to the RSO. For a truly robust planner, these considerations must be accounted for when determining optimal trajectories. In the next chapter we will evaluate different methods for assessing safety and develop the necessary modifications to the STORM algorithm for their implementation.

CHAPTER 8

PROXIMITY OPERATIONS SAFETY ANALYSIS

On April 15, 2005, NASA launched the Demonstration for Autonomous Rendezvous Technology (DART) satellite seeking to advance its rendezvous and proximity operations technology [16]. As the spacecraft performed a series of close-range maneuvers around its RSO (the previously launched MUBLCOM satellite), the spacecraft's Guidance, Navigation and Control (GNC) software experienced navigation difficulties and overshot an important waypoint that was designed to trigger the transition to a more advanced navigation method. At longer ranges, DART relied on relative bearing data similar to the angles-only navigation method develop in Section 4.2.3. At closer ranges, DART was to transition to the Advanced Video Guidance System (AVGS) that would use laser returns to generate accurate relative range and bearing data (similar to the lidar-based methods developed in Section 4.2.1.

The transition to AVGS was to be triggered when the spacecraft arrived at a waypoint 200 meters behind MUBLCOM (with a tolerance of just 6.3 meters). Unfortunately, DART failed to achieve the waypoint, missing the allowable tolerance by less than 2 meters, and continued advancing towards MUBLCOM until the two collided. At the time of the collision DART believed it was retreating from the RSO at 0.3 meters per second, but instead bumped into MUBLCOM at 1.5 meters per second. Despite hosting an onboard collision avoidance system, the system had not anticipated such large inaccuracies and failed to prevent the collision.

Because the mission failed to achieve its main mission objectives, NASA declared a "Type A" mishap resulting in an exhaustive review of the mishap's root causes. While the proximate cause of the mishap included erroneous data from a GPS receiver, the report faulted larger systemic root causes for the mishap. Some of those causes include:

- High risk, low budget nature of the procurement
- Inadequate training and experience
- Inadequate incorporation of previous "lessons learned"
- Inadequate GNC software development processes
- International Traffic in Arms Regulations (ITAR) restrictions

Analysis showed that the UK-built GPS receiver produced a velocity measurement with a bias of 0.6 meters per second. This bias was a known issue with the receiver, but the issue was never discovered by the design team due to ITAR restrictions and complications dealing with foreign vendors.

The Air Force Research Laboratory's XSS-11 spacecraft, also designed for proximity operations, had launched just days prior to DART. Though safety had been a major consideration in the design and planned operations of XSS-11, the DART collision brought increased scrutiny. Many of the author's mentors at AFRL had previously worked XSS-11 operations and brought that safety mindset to follow-on programs where he learned the ropes on rendezvous and proximity operations.

As critical as the technical considerations may be for RPO safety, sometimes it comes down to simple trust and confidence in the team. Prior to executing a particular operation on the ANGELS mission, the Mission Director walked over to the author's console and asked how he was feeling. As the Lead Mission Planner, the author had designed and analyzed the planned trajectory and had developed many of the safety checks and protocols that would be implemented. When he responded that he felt confident, the Mission Director turned to the rest of the team and declared "Alright. Let's go."

Several of the concepts presented in this chapter are offshoots of those experiences in designing, planning, and flying RPO missions. What is new however is merging these concepts with a closed-loop linear covariance analysis. This research element is a major contribution to the field of rendezvous and proximity operations.

8.1 RPO Safety

In this section we will address some of the basic methods for addressing spacecraft trajectory safety during proximity operations. These methods are commonly used in deterministic trajectory analysis. It will be noted here that safety is not unique definition and can be achieved through numerous methods. Key to evaluating safety is having a consistent mathematical approach with well under criteria.

8.1.1 Keep-out Zone Analysis

The first approach to preventing a collision is to define a simple keep-out zone (KOZ) around the RSO that the active vehicle should not enter. Analysis can then be performed to ensure that the spacecraft's nominally planned trajectory does not intersect or cut through that zone. For a spherical keep-out zone, that analysis is a simple range check that flags mission planners to the violation and initiates a trajectory replan without the keep-out zone violation. Similarly, a speed-limit zone can be defined where the vehicle's velocity (or directional range-rate) are limited to a maximum value.

A mission's keep-out zone may vary by mission phase or a spacecraft's performance capabilities. The International Space Station (ISS) has a two tiered system with an initial Approach Ellipsoid (AE) and a smaller spherical keep-out zone depicted in Figure 8.1. The AE is a 4×2×2 km ellipsoid defined in the LVLH frame such that the 4 km semi-major axis is in the intrack direction. The KOZ is a 200 m radius sphere centered on the ISS [119].

8.1.2 Tree Analysis

A natural extension of the keep-out zone analysis is to consider the possibility of missing a planned maneuver and evaluating all possible free-drifting trajectories for violation of the keep-out zone. A free-drifting trajectory would result should any planned maneuver fail to execute and the spacecraft continues on its natural path.

As part of a spacecraft's fault detection and correction system, major faults are likely to disable or inhibit the propulsion system in order to prevent the vehicle from inadvertently maneuvering towards the RSO and increasing the risk of collision. Due to this common

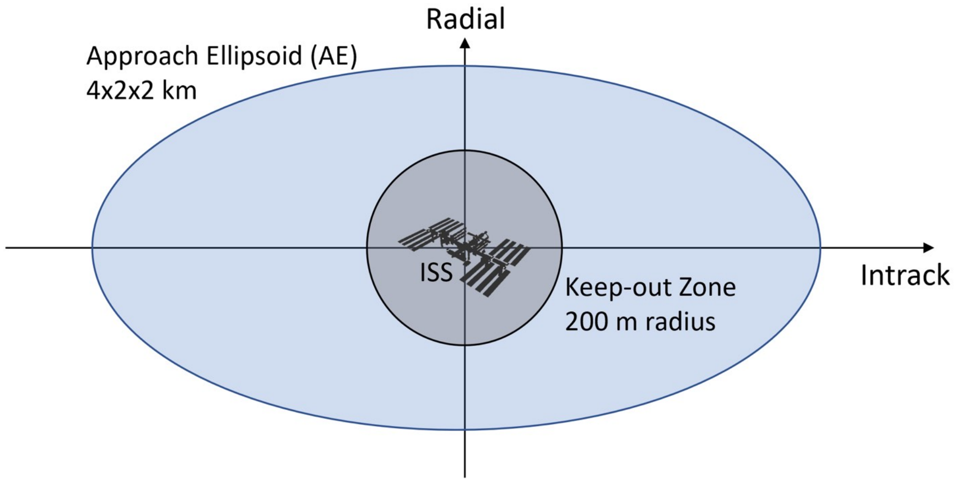

Fig. 8.1: Approach Ellipsoid and Keep-out Zones for the International Space Station represent method for keep-out zone safety during proximity operations. Figure is not to scale.

fault correction approach, each failed maneuver can be treated as a singular event where all future maneuvers are also inhibited until ground operators are able to intervene and a new trajectory is generated and uploaded for execution.

We will refer to this evaluation as a *tree analysis* where the nominal trajectory (executing all planned maneuvers) is considered the main tree trunk and each missed burn trajectory is a branch of the tree. Violation of the keep-out zone by any branch of the trajectory (over a specified time window) would trigger a replan to increase safety. An example of a simple tree analysis for the three-impulse NMC resizing scenario is shown in Figures 8.2 and 8.3, presenting the relative trajectories and relative range, respectively.

8.1.3 Trajectory Assessment

The obvious issue with a simplified keep-out zone approach (with our without an accompanying tree analysis) is that it relies on a nominal trajectory and ignores the inevitable trajectory dispersions that arise in a closed-loop GNC system. A true safety assessment should include dispersions due to off-nominal trajectories.

Fortunately, the trajectory designer has a few options at their disposal to address this deficiency. The first option would be to perform a full Monte Carlo analysis of the planned

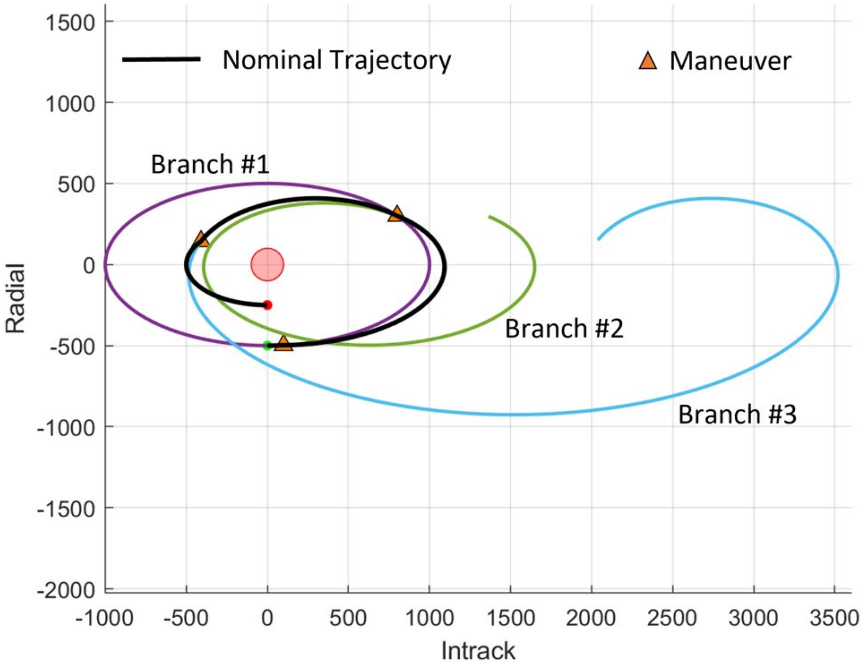

Fig. 8.2: Tree analysis of the three-impulse NMC resizing scenario showing the free-drifting trajectory branches. All units are in meters.

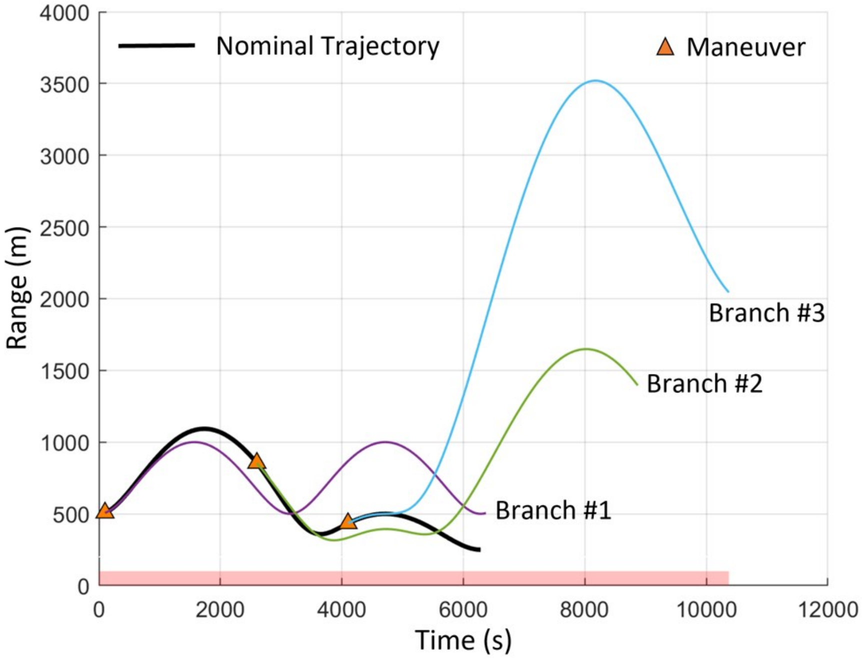

Fig. 8.3: Tree analysis of the three-impulse NMC resizing scenario showing the range to the RSO for each free-drifting trajectory branch.

trajectory and assess each sample trajectory against the keep-out zone. Alternatively, the ensemble statistics from the Monte Carlo analysis could be used to perform a statistical analysis against the KOZ. As discussed earlier, a Monte Carlo analysis is a time consuming process that does not meet operational timelines of an RPO mission making this an undesirable option.

A second and much more desirable option is to perform a closed-loop linear covariance analysis which can assess the planned trajectory for safety violations in a single run. Since individual sample trajectories don't exist for the linear covariance approach, we will need to develop an alternative constraint check based on the statistical data provided by a linear covariance model. In the following sections we will discuss multiple probability-based options for evaluating safety and identify their strengths and weaknesses for accurately assessing safety.

A third option is sometimes considered that performs a standard linear covariance analysis to evaluate a trajectory. Unlike the more advanced closed-loop variant, a standard covariance analysis only accounts for navigation errors along the nominal trajectory and not the true dispersions resulting from a full closed-loop system. While attractive for its simplicity, this approach does not meet the needs of complex closed-loop RPO missions as seen in the two-impulse scenarios of Chapter 5.

8.2 Probabilistic Trajectory Safety

This section will develop several approaches for assessing trajectory safety based on the statistical data (i.e true position dispersions) generated by the closed-loop linear covariance models. The first approach is to directly calculate the probability of collision between the two vehicles at each time step across the scenario. The equations for this instantaneous probability of collision are derived as well as an approximation for simplifying the calculation. Given limitations in this method we will next look at bounding methods that provide statistical safety information without directly calculating the probability of collision. These approaches include the range uncertainty and the Mahalanobis distance. Finally, a preferred approach is derived that bounds the probability of collision by comparing a scaled value of the true position dispersion against a keep-out zone around the RSO.

8.2.1 Probability of Collision

The probability density function (*pdf*) of a multivariate, Gaussian (normal) random vector [94] is defined as:

$$f_n(\bar{x}) = \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} \exp\left\{-\frac{1}{2}(x-\bar{\mu})^T\Sigma^{-1}(x-\bar{\mu})\right\} \tag{8.1}$$

where Σ is the positive semi-definite covariance matrix and $|\cdot|$ denotes the matrix determinant. $\bar{\mu}$ and Σ are the random variable's mean and covariance, respectively, and n is the dimension of the random variable, x. For simplicity in this analysis we will assume, with no loss of generality, that the random variable is zero-mean ($\bar{\mu} = 0$).

$$f_n(x) = \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} \exp\left\{-\frac{1}{2}x^T\Sigma^{-1}x\right\} \tag{8.2}$$

For a univariate case ($n = 1$) the equation simplifies to:

$$f_1(x) = \frac{1}{\sqrt{2\pi}\,\sigma_x} \exp\left\{-\frac{1}{2}\frac{x^2}{\sigma_x^2}\right\} \tag{8.3}$$

where σ_x is the standard deviation. The 2-dimensional ($n = 2$) equation is given as:

$$f_2(x) = \frac{1}{2\pi|\Sigma|^{1/2}} \exp\left\{-\frac{1}{2}x^T\Sigma^{-1}x\right\} \tag{8.4}$$

And the 3-dimensional ($n = 3$) equation is given as:

$$f_3(\bar{x}) = \frac{1}{(2\pi)^{3/2}|\Sigma|^{1/2}} \exp\left\{-\frac{1}{2}x^T\Sigma^{-1}x\right\} \tag{8.5}$$

The 3-dimensional covariance matrix is given as:

$$\Sigma = \begin{bmatrix} \sigma_x^2 & \rho_{xy}\sigma_x\sigma_y & \rho_{xz}\sigma_x\sigma_z \\ \rho_{xy}\sigma_x\sigma_y & \sigma_y^2 & \rho_{yz}\sigma_y\sigma_z \\ \rho_{xz}\sigma_x\sigma_z & \rho_{yz}\sigma_y\sigma_z & \sigma_z^2 \end{bmatrix} \tag{8.6}$$

where σ_i is the standard deviation in the x, y, and z axes and ρ_{ij} provides the cross-correlation between the different axes. The correlation factor, ρ_{ij}, can take on values in the range $-1 \leq \rho_{ij} \leq 1$ where $\rho_{ij} = \pm 1$ represents a perfect correlation between axes and $\rho_{ij} = 0$ represents zero correlation between axes.

The covariance matrix can be rotated into its principal axes, generating a diagonal matrix, through a singular value decomposition [120]:

$$\Sigma = U \cdot S \cdot V^T \tag{8.7}$$

$$\Sigma' = S = \begin{bmatrix} \sigma_{x'}^2 & 0 & 0 \\ 0 & \sigma_{y'}^2 & 0 \\ 0 & 0 & \sigma_{z'}^2 \end{bmatrix} \tag{8.8}$$

where Σ' is the covariance matrix in the principle frame and $\sigma_{x'}$, $\sigma_{y'}$, and $\sigma_{z'}$ are the standard deviations in this principal frame. A 2-dimensional example depicting the relationship between the two frames is depicted in Figure 8.4. In this principal frame Equation (8.5) simplifies to the following form:

$$f_3(\bar{x}') = \frac{1}{(2\pi)^{3/2}\,\sigma_{x'}\sigma_{y'}\sigma_{z'}} \exp\left\{-\frac{1}{2}\left[\left(\frac{x'}{\sigma_{x'}}\right)^2 + \left(\frac{y'}{\sigma_{x'}}\right)^2 + \left(\frac{z'}{\sigma_{x'}}\right)^2\right]\right\} \tag{8.9}$$

The probability of collision, P_c, for a given relative position vector, $\bar{r}$, and true position dispersion matrix, $D_{\bar{r}}$, is calculated by integrating the *pdf* over the hard-body keep-out volume, V.

$$P_c = \iiint\limits_V f_3(\bar{r})dV \tag{8.10}$$

$$P_c = \frac{1}{(2\pi)^{3/2}|\Sigma_{\bar{r}}|} \iiint\limits_{V} \exp\left\{-\frac{1}{2}\bar{r}^T \Sigma_{\bar{r}}^{-1} \bar{r}\right\} dV \tag{8.11}$$

Equivalently, P_c can be calculated in the principal frame by transforming the state position vector into the covariance's principal frame:

$$P_c = \iiint\limits_{V'} f_3(\bar{r}')dV' \tag{8.12}$$

$$P_c = \frac{1}{(2\pi)^{3/2}\sigma_{x'}\sigma_{y'}\sigma_{z'}} \iiint\limits_{V'} \exp\left\{-\frac{1}{2}\left[\left(\frac{x'}{\sigma_{x'}}\right)^2 + \left(\frac{y'}{\sigma_{x'}}\right)^2 + \left(\frac{z'}{\sigma_{x'}}\right)^2\right]\right\} dV' \tag{8.13}$$

While an accurate volume integral would require knowledge of both spacecraft's shape and attitude, the volume, V, can be simplified to a sphere with radius equivalent to the sum of the two spacecrafts' largest dimension. This radius is referred to as the hard-body radius, R_{hb}. Under this simplification, the volume used for integration is given as:

$$V = \frac{4}{3}\pi R_{hb}^3 \tag{8.14}$$

where V and V' are equivalent.

The instantaneous probability of collision between the spacecraft and RSO, can be calculated by numerically integrating Equation (8.11) or (8.13) over the hard-body volume

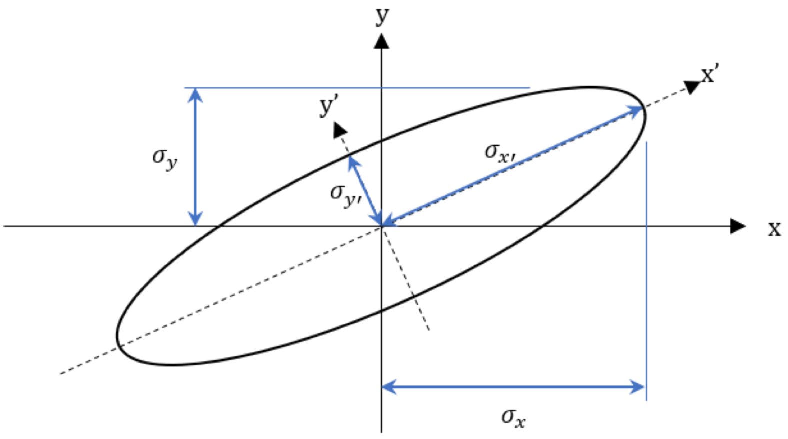

Fig. 8.4: 1σ covariance boundary in the local and principal axes for a 2-dimensional system.

in Equation (8.14). Chan provides multiple methods for numerically calculating this triple integral [57]. These methods, however, are computationally intensive and not conducive to our overall objectives for trajectory planning. When implemented into the STORM analysis tool, numerical integration of Equation (8.13) at each time step increases the software run-time by a full order of magnitude. In an effort to reduce computational time we will look to simplify the P_c calculation.

8.2.2 Simplified Probability of Collision

To reduce computational time we will look to linearize the calculation of P_c. This is done by assuming the hard-body radius is small compared to the standard deviations and that the *pdf* is constant over the hard-body volume. Under this assumption the resulting approximation for P_c is:

$$P_c \approx \frac{2}{3} \frac{R_{hb}^3}{\sqrt{2\pi}\sigma_{x'}\sigma_{y'}\sigma_{z'}} \exp\left\{-\frac{1}{2}\left[\left(\frac{x_0'}{\sigma_{x'}}\right)^2 + \left(\frac{y_0'}{\sigma_{x'}}\right)^2 + \left(\frac{z_0'}{\sigma_{x'}}\right)^2\right]\right\} \tag{8.15}$$

The probability of collision for the nominal trajectory and branches of the NMC resizing scenario shown in Section 8.1.2 was analyzed against a hard-body radius of 10 meters. The results are shown in Figure 8.5. A zoomed in view is shown in Figure 8.6 to clarify the maximum calculated probabilities of collision, $P_c^{\max}$. Note that the numerically calculated value of P_c for the nominal trajectory is zero across the scenario and therefore does not appear on the logarithmic plot. The true value of P_c is not zero but limited by the numerical precision of MATLAB's floating-point doubles. The smallest positive normalized floating-point number in IEEE double precision is equal to 2^{-1022} or approximately 2.2×10^{-308} [121].

The calculated P_c values are overlayed on the range values for the nominal trajectory and each branch in Figure 8.7. Note that increases in P_c are not necessarily tied to reduced ranges to the RSO.

Before proceeding, it is important to point out some of the issues with using P_c as our safety metric. The first, as was identified above, is that the time required to calculate an accurate P_c value, Equation (8.13), in the STORM analysis tool is prohibitive. To address

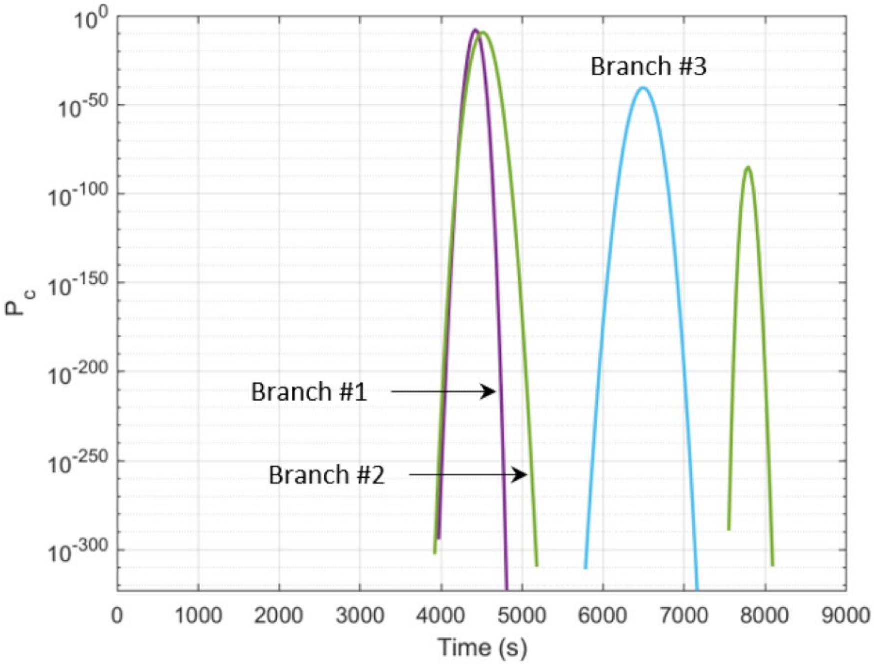

Fig. 8.5: Probability of collision values for tree analysis of the NMC resizing scenario.

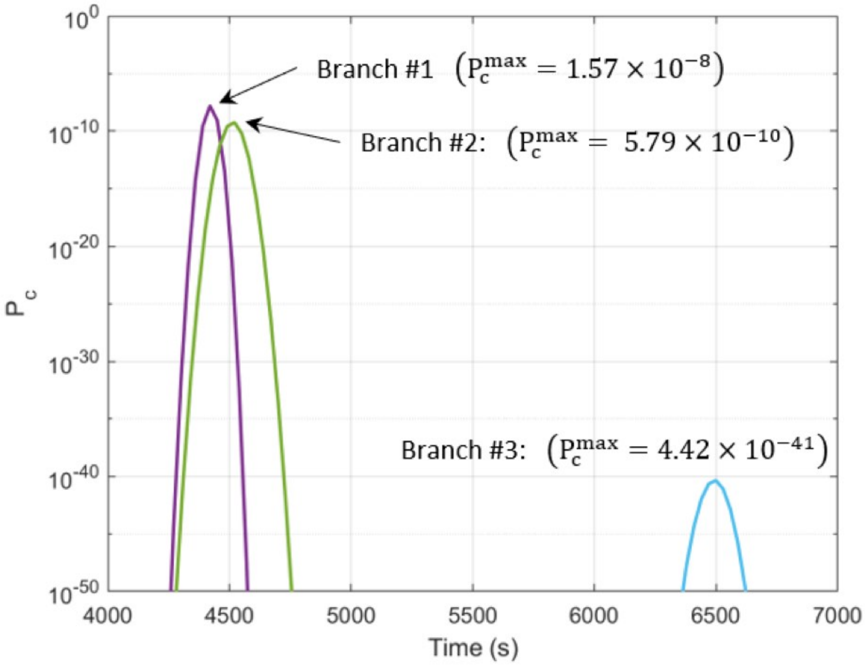

Fig. 8.6: Maximum probability of collision values for tree analysis of the NMC resizing scenario.

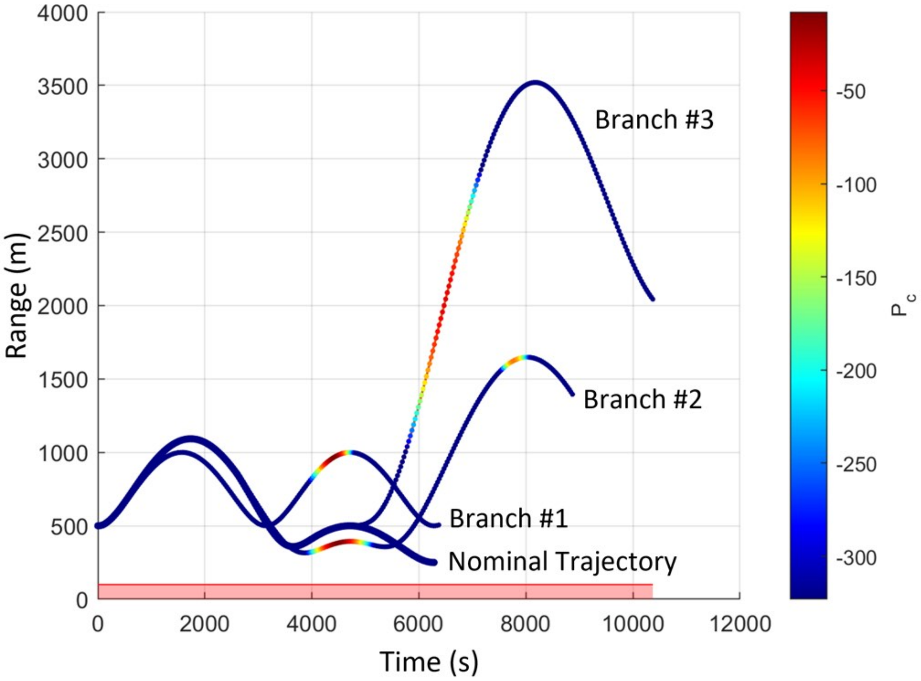

Fig. 8.7: Relative range for tree analysis of the NMC resizing scenario with overlayed P_c values.

the timing issue we can simplify the P_c calculation as was done in Equation (8.15) assuming we accept the assumptions used in its derivation.

A less obvious issue with both of these approaches is, at what value of P_c is the risk too large and the trajectory in need of replanning. To minimize space debris, NASA directs maneuverable spacecraft to initiate avoidance maneuvers when P_c exceeds 1×10^{-4} (1 in 10,000) [122, 123]. This requirement, however, is based on an event probability and not on the instantaneous probability derived above. Additionally, this requirement does not apply to intentional operations such as RPO. While efforts have been made to bring uniformity to the different mission types, no standardized value is accepted across the space community for proximity operations.

Finally, both P_c calculations were based on a hard-body radius and not an arbitrary keep-out zone as may be required for operational systems. Vehicle performance limitations or government policy may require a spacecraft to remain outside of a specified KOZ that may be 100 m, 1 km, or even 10 km. Increasing the volume of integration beyond the hard-body volume disconnects the calculated P_c value from any actual collision risk further complicating the selection of a P_c threshold.

Given these issues, it is the author's recommendation that a direct calculation of P_c is not an advisable option for trajectory planning. As an alternative, we will look at several methods for bounding P_c that may be used for in conjunction with an arbitrary KOZ.

8.2.3 Range Uncertainty

As an extension, the relative range at closest approach combined with the relative range uncertainty is another method that can be used to assess safety. This one-dimensional value provides a simple and intuitive assessment indicating whether or not the spacecraft's current range with its related uncertainty intersects the RSO keep-out zone. Given the true position dispersion, $D_{\bar{r}}$, the variance along the relative position vector is calculated as:

$$\sigma_r^2 = \hat{i}_r^T D_{\bar{r}} \hat{i}_r \tag{8.16}$$

where $\hat{i}_r$ is the unit vector along the relative position vector:

$$\hat{i}_r = \frac{\mathbf{r}}{\|\mathbf{r}\|} \tag{8.17}$$

This method is typically adequate when the spacecraft is at large ranges, compared to the size of the position dispersion ellipsoid. As the range is reduced, or equivalently the position dispersion increases, this method can produce inaccurate results, giving false-positive indications of a range error. This failing is particularly apparent when the uncertainty dispersion is long and skinny along the line-of-sight vector as is common with angles-only navigation schemes.

To demonstrate this condition we will look at a simple two-dimensional scenario with the following values:

$$D_{\bar{r}} = \begin{bmatrix} 4.0 & 5.4 & 0 \\ 5.4 & 9.0 & 0 \\ 0 & 0 & 0 \end{bmatrix} m^2$$

$$\mathbf{r}_0 = \begin{bmatrix} 2.0 & 8.0 & 0 \end{bmatrix}^T \; m$$

$$R_{\text{koz}} = 1.0 \; m$$

In this case the mean range between the vehicle and RSO is 8.25 m with a standard deviation of 3.35 m. According to the range uncertainty method, the KOZ lies on the interval between 2.16σ and 2.76σ. The position dispersion ellipses for the 1, 2, and 3σ bounds are shown along with the equivalent range uncertainty bounds in Figure 8.8. It is apparent that the RSO KOZ lies between the 2σ and 3σ range uncertainty bounds as calculated. This assessment is, however, misleading since the KOZ is entirely outside of the 3σ ellipse's bounds.

The tree analysis of the NMC resizing scenario was repeated using the range uncertainty method and is shown in Figure 8.9. Range uncertainty bounds for the nominal range and each branch are depicted as shaded contours. Note that the 4σ uncertainty on branch 1 violates the KOZ threshold just prior to $t = 6000$. This interval, however, does not coincide with a high P_c as depicted in Figure 8.7. While intuitive, this metric is not always accurate for assessing true dispersion errors and is not well suited for our purpose.

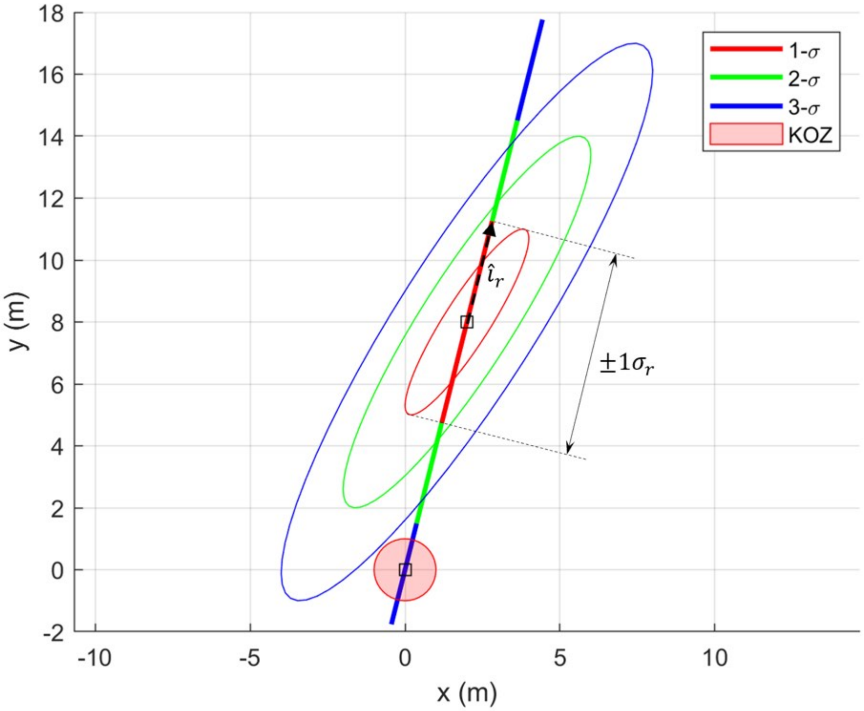

Fig. 8.8: Position dispersion ellipse and the resulting range uncertainties. The 3σ range uncertainty intersects the keep-out zone while the 3σ ellipse does not.

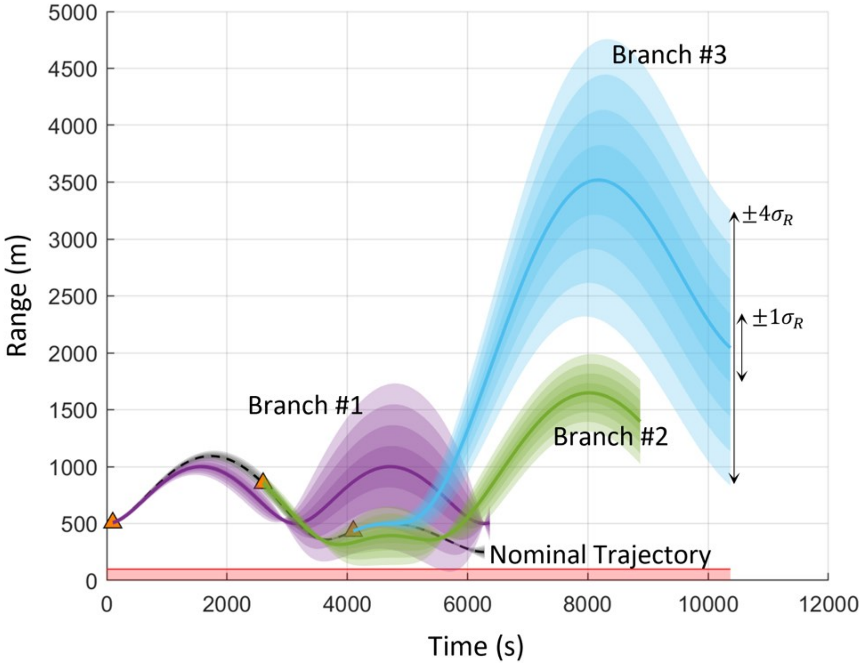

Fig. 8.9: Relative range for tree analysis of the NMC resizing scenario with overlayed range uncertainty bounds.

8.2.4 Mahalanobis Distance

One method that can be considered to avoid the issues observed with range uncertainty is to determine the Mahalanobis distance between the position dispersion and the RSO. The Mahalanobis distance represents the statistical distance between a point and a multi-dimensional distribution, equivalent to determining the number of standard deviations a point is away from the mean for a univariate distribution [124].

For a Gaussian distribution, the true position dispersion matrix, D, represents an ellipsoid whose surface, x, defines the 1σ boundary.

$$(x-\bar{x})^T D^{-1}(x-\bar{x})-1=0 \tag{8.18}$$

This equation can be rewritten in a more general form as:

$$(x-\bar{x})^T D^{-1}(x-\bar{x})-l^2=0 \tag{8.19}$$

where l is a constant. The Mahalanobis distance for any point, x^*, is then calculated as:

$$l=\sqrt{(x^*-\bar{x})^T D^{-1}(x^*-\bar{x})} \tag{8.20}$$

Revisiting the previous simplified example ($x^* = [0,\ 0,\ 0]^T$), the Mahalanobis distance to the RSO is calculated to be 4.17, meaning the RSO is at the 4.17σ boundary of the true position dispersion. This is depicted in Figure 8.10.

This method is also simple and intuitive but does not account for the RSO's keep-out zone. One way around this would be to select the point on the keep-out zone that corresponds to the line-of-sight vector rather than the origin. With this approximation ($x = [0.2425,\ 0.9701,\ 0]^T$) the resulting Mahalanobis distance is 3.67. While smaller than the previously calculated Mahalanobis distance, this boundary intersects the keep-out zone at other locations potentially leading to false-negative safety checks. This false-negative is most pronounced when the ellipsoid is long and skinny, similar to the range uncertainty method.

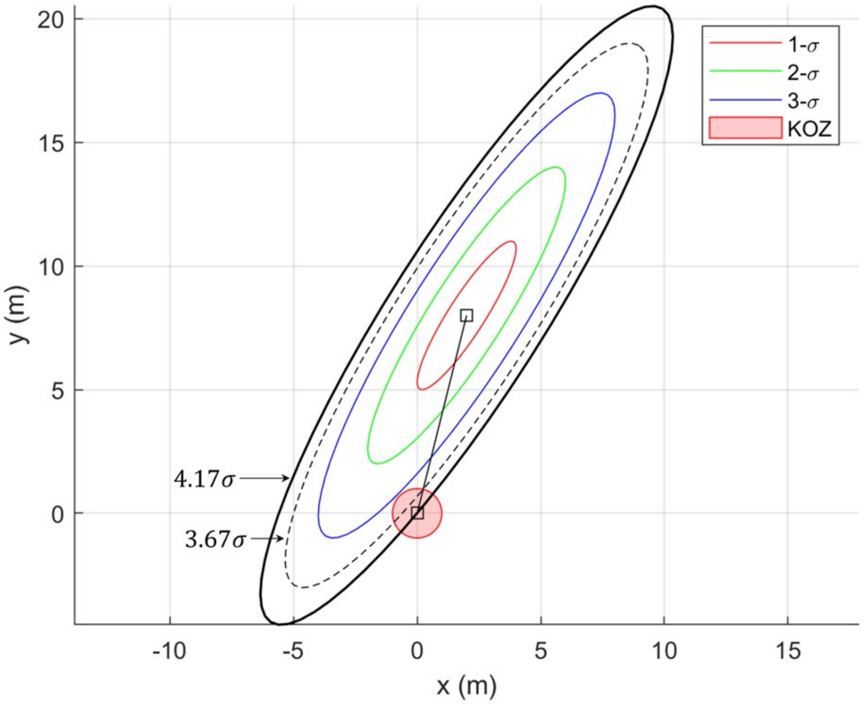

Fig. 8.10: Position dispersion ellipse and resulting Mahalanobis distance to the RSO (4.17σ) and KOZ along the line-of-sight vector (3.67σ).

8.2.5 Bounded Probability of Collision

To address the issues with the previous methods, we will develop an alternate methods that properly accounts for the RSO keep-out zone and, rather than calculating P_c, accurately bounds the probability of collision. This method will also prevent false-negatives and false-positives.

We will begin by looking at how we can use the true position dispersion matrix to bound the probability of collision calculation. A multi-dimensional covariance matrix, D, represents a hyperellipsoid:

$$(x - \bar{x})^T D^{-1} (x - \bar{x}) = l^2 \tag{8.21}$$

where l is a constant. For a one-dimensional system this represents a single line between $-l$ and $+l$ (similar to our range uncertainty calculations). For two and three-dimensional systems this equation represents an ellipse or ellipsoid, respectively. Figure 8.4 depicts the 1σ boundary ($l = 1$) for a two-dimensional system. As show previously in Figure 4.6, the

Table 8.1: Probability values that x lies within the hyperellipsoid boundary defined by the equation $x^T D^{-1} x = l^2$ for 1, 2, and 3-dimensional distributions.

$n \setminus l$	1	2	3	4	5	6
1	0.6827	0.9545	0.9973	0.999937	0.9999994	0.999999998
2	0.3935	0.8647	0.9889	0.999665	0.9999963	0.999999985
3	0.1987	0.7385	0.9707	0.998866	0.9999846	0.999999925

3×3 position dispersion (mathematically equivalent to the covariance matrix) represents a three-dimensional ellipsoid that statistically bounds the 1σ range of possible values that can be assumed by the true position.

It can be shown that the probability of finding x inside of a hyperellipsoid is [125]:

$$P^n(l) = \left[\frac{1}{(2\pi)^{(n/2)}}\right] \int_0^l \exp\left\{-\frac{1}{2}r^2\right\} f(r)\, dr \tag{8.22}$$

where r is the radius of the hypersphere and $f(r)dr$ is the spherically symmetric volume in the n-dimensional space. Evaluating this integral for $n = 1,\ 2,$ and 3 yields:

$$P^1(l) = \sqrt{\frac{2}{\pi}} \int_0^l \exp\left\{-\frac{1}{2}r^2\right\} dr = \operatorname{erf}\left(\frac{l}{\sqrt{2}}\right) \tag{8.23}$$

$$P^2(l) = \int_0^l \exp\left\{-\frac{1}{2}r^2\right\} r\ dr = 1 - \exp\left(-\frac{l^2}{2}\right) \tag{8.24}$$

$$P^3(l) = \sqrt{\frac{2}{\pi}} \int_0^l \exp\left\{-\frac{1}{2}r^2\right\} r^2\ dr = \operatorname{erf}\left(\frac{l}{\sqrt{2}}\right) - \sqrt{\frac{2}{\pi}}\, l \exp\left(-\frac{l^2}{2}\right) \tag{8.25}$$

The probability for each case, as a function of l, is shown in Figure 8.11. The integer values for l between 1 and 6 are shown in Table 8.1. For a one-dimensional system, the 3σ boundary is the well known 99.73%, indicating that any randomly sampled point has a 99.73% chance of lying within the 3σ bounds. It must be noted that for a three-dimensional system (such as our position dispersion) that the probability within the 3σ boundary is only 97.07%.

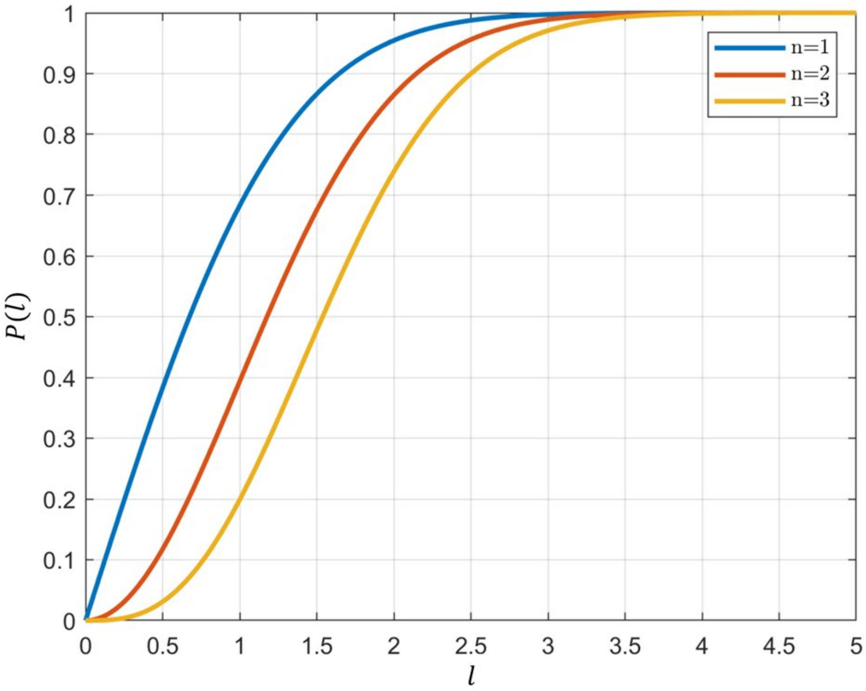

Fig. 8.11: Probability that x lies within the hyperellipsoid boundary defined by the equation $x^T D^{-1} x = l^2$.

Table 8.2: Probability of collision bound for select 3-dimensional covariance bounds.

l	3	4	5	6	7
$P_c <$	2.93×10^{-2}	1.13×10^{-3}	1.54×10^{-5}	7.48×10^{-8}	1.30×10^{-10}

Given a desired probability for the true position dispersion ($n = 3$) we can determine the corresponding σ-boundary that encompasses that probability. For example, a bounding probability of 99.99% would equate to a value of 4.5942σ. (For one and two-dimensional dispersions, 99.99% equates to values of 3.8906σ and 4.2919σ, respectively). A calculated P_c for a hard-body volume entirely outside this boundary is likely much smaller than 1×10^{-4}, but we can guarantee is is not larger.

In terms of probability of collision, using a 6σ boundary for a three-dimensional system indicates that $P_c < 7.5 \times 10^{-8}$, or less than a 1 in 13 million chance of collision. Achieving similar results from a Monte Carlo analysis would be a time consuming process. Bounded P_c values for integer values of sigma between 3 and 7 are shown in Table 8.2.

8.3 Keep-out Zone Intersection

The next step in evaluating trajectory safety is to determine if the selected bounded true position dispersion ellipsoid intersects the RSO keep-out zone. To evaluate this condition we must determine the minimum distance from the RSO keep-out zone and our specified $n\sigma$-boundary.

For this calculation we will use the derivation proposed by Hart for determining the minimum distance between any point, x_0, and an ellipsoid [126]. We will assume, without loss of generality, that the ellipsoid is located at the origin of the frame with principal axes parallel to the coordinate axes. The line segment corresponding to this minimum distance will be normal to the ellipsoid at x and will satisfy the following equation:

$$\mathbf{x}_0 - \mathbf{x} = \alpha \boldsymbol{n}(\mathbf{x}) \tag{8.26}$$

where $\boldsymbol{n}(\mathbf{x})$ is the ellipsoid surface normal at the point $\mathbf{x}$ on the ellipsoid surface. The ellipsoid surface is implicitly defined as:

$$f(\mathbf{x}) = \frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} - 1 \tag{8.27}$$

where a, b, and c are the principal semi-axes of the ellipsoid. The surface normal at point $\mathbf{x}$ is defined by the gradient of the surface:

$$\boldsymbol{n}(\mathbf{x}) = \nabla f(\mathbf{x}) = \begin{bmatrix} \frac{\partial f}{\delta x} & \frac{\partial f}{\delta y} & \frac{\partial f}{\delta z} \end{bmatrix} = 2 \begin{bmatrix} \frac{x}{a^2} & \frac{y}{b^2} & \frac{z}{c^2} \end{bmatrix} \tag{8.28}$$

We will drop the 2 from the gradient since we are only interested in the direction of the surface normal. Substituting Equation (8.28) into Equation (8.26) and solving for $\mathbf{x}$ yields:

$$x = \frac{a^2 x_0}{\alpha + a^2} \tag{8.29a}$$

$$y = \frac{b^2 y_0}{\alpha + b^2} \tag{8.29b}$$

$$z = \frac{c^2 z_0}{\alpha + c^2} \tag{8.29c}$$

Substituting Equation (8.29) into Equation (8.27) yields:

$$\frac{a^2 x_0^2}{(\alpha + a^2)^2} + \frac{b^2 y_0^2}{(\alpha + b^2)^2} + \frac{c^2 z_0^2}{(\alpha + c^2)^2} = 1 \tag{8.30}$$

This equation can be rearranged to provide a sixth-order polynomial for α:

$$p_6\alpha^6 + p_5\alpha^5 + p_4\alpha^4 + p_3\alpha^3 + p_2\alpha^2 + p_1\alpha + p_0 = 0 \tag{8.31}$$

where:

$$\begin{aligned}
p_6 =& - 1 \\
p_5 =& - 2\left(a^2 + b^2 + c^2\right) \\
p_4 =& - 4\left(a^2b^2 + a^2c^2 + b^2c^2\right) + a^2x^2 + b^2y^2 + c^2z^2 - a^4 - b^4 - c^4 \\
p_3 =& - 2\left(a^2c^4 + b^2c^4 + a^2b^4 + a^4b^2 + a^4c^2 + b^4c^2\right) - 8a^2b^2c^2 \\
& + 2a^2b^2\left(x^2 + y^2\right) + 2a^2c^2\left(x^2 + z^2\right) + 2b^2c^2\left(y^2 + z^2\right) \\
p_2 =& - 4c^2\left(a^4b^2 + a^2b^4\right) - a^4b^4 - c^4\left(a^4 + 4a^2b^2 + b^4\right) \\
& + \left(a^2b^4 + a^2c^4\right)x^2 + \left(a^4b^2 + b^2c^4\right)y^2 + \left(a^4c^2 + b^4c^2\right)z^2 \\
& + 4a^2b^2c^2\left(x^2 + y^2 + z^2\right) \\
p_1 =& - 2\left(a^4b^2c^4 + a^2b^4c^4 + a^4b^4c^2\right) + 2\left(a^2b^2c^4 + a^2b^4c^2\right)x^2 \\
& + 2\left(a^2b^2c^4 + a^4b^2c^2\right)y^2 + 2\left(a^2b^4c^2 + a^4b^2c^2\right)z^2 \\
p_0 =& - \left(a^4b^4c^4 + a^2b^4c^4x^2 + a^4b^2c^4y^2 + a^4b^4c^2z^2\right)
\end{aligned} \tag{8.32}$$

The distance between the point $\mathbf{x}_0$ and the ellipsoid corresponds to the largest (and only positive) root of Equation (8.31).

$$d = \alpha_{\max} \tag{8.33}$$

This distance can be substituted back into Equation (8.29) to determine the point on the ellipsoid corresponding to the minimum distance.

If the largest root for α is negative then the point lies within the ellipsoid.

Using a spherical keep-out zone, the true position dispersion intersects the keep-out zone if the distance from the RSO to the true position dispersion is less than the keep-out zone radius:

$$I_{\text{koz}} = \begin{cases} 1, & \text{if } d < R_{\text{koz}} \\ 0, & \text{otherwise} \end{cases} \tag{8.34}$$

where I_{koz} is a flag indicating intersection and R_{koz} is the radius of the keep-out zone.

It is possible to extend this approach to non-spherical keep-out zones by iterating between the two ellipsoids. Using the previous calculated point on the ellipsoid as a new starting point, an estimate for the distance to the second ellipsoid is calculated. Iterations continue until the change in the estimated distance falls below a specified tolerance. In this case, an intersection would be declared if the calculated distance at any iterations goes negative. Alternatively, an intersection between two ellipsoids can be calculated directly according to the method put forth by Alfano [127]. The benefit of using the approach derived above is that it provides the calculated minimum distance at each time step for visual assessment rather than just an indication of intersection.

To demonstrate this method, we will reprise our previous example defined by the following covariance matrix, offset, and keep-out zone:

$$D_{\bar{r}} = \begin{bmatrix} 4.0 & 5.4 & 0 \\ 5.4 & 9.0 & 0 \\ 0 & 0 & 0 \end{bmatrix} m^2$$

$$\mathbf{r}_0 = \begin{bmatrix} 2.0 & 8.0 & 0 \end{bmatrix}^T \; m$$

$$R_{\text{koz}} = 1.0 \; m$$

The covariance and position vector in the principal coordinate frame can be obtained by performing a singular value decomposition:

$$D_{\bar{r}} = U \cdot S \cdot V^T \tag{8.35}$$

such that:

$$S = \begin{bmatrix} 12.4506 & 0 & 0 \\ 0 & 0.5494 & 0 \\ 0 & 0 & 0 \end{bmatrix} m^2$$

$$\mathbf{x}_0 = U\mathbf{r}_0 = \begin{bmatrix} -7.8181 & 2.6224 & 0 \end{bmatrix}^T \; m$$

This gives the following inputs for the polynomial: $a = 3.53$, $b = 0.74$, $c = 0$, $x = 7.8467$, $y = -2.5355$, and $z = 0$.

The calculated distance (i.e. the minimum distance between the dispersion ellipse and RSO at the origin) as a function of l is given in Figure 8.12. The ellipse just touches the keep-out zone at $l = 3.09$. The ellipse intersects the RSO at the origin at $l = 4.17$; the same value found when evaluating the Mahalanobis distance in Section 8.2.4. If the bounded safety parameter had been set at 3σ then an intersection would not have been flagged ($I_{\text{koz}} = 0$). A bounded safety parameter set at 4σ would have been flagged for intersection ($I_{\text{koz}} = 1$)

The minimum distance locations for the 1, 2, and 3σ ellipses are shown in Figure 8.13. The location of the minimum distance point as the ellipse scaling is increased is shown in Figure 8.14. Note the sharp turn in the location around the 2σ boundary due to the long skinny ellipse that corresponds to the areas of concern when using the range uncertainty and Mahalanobis methods.

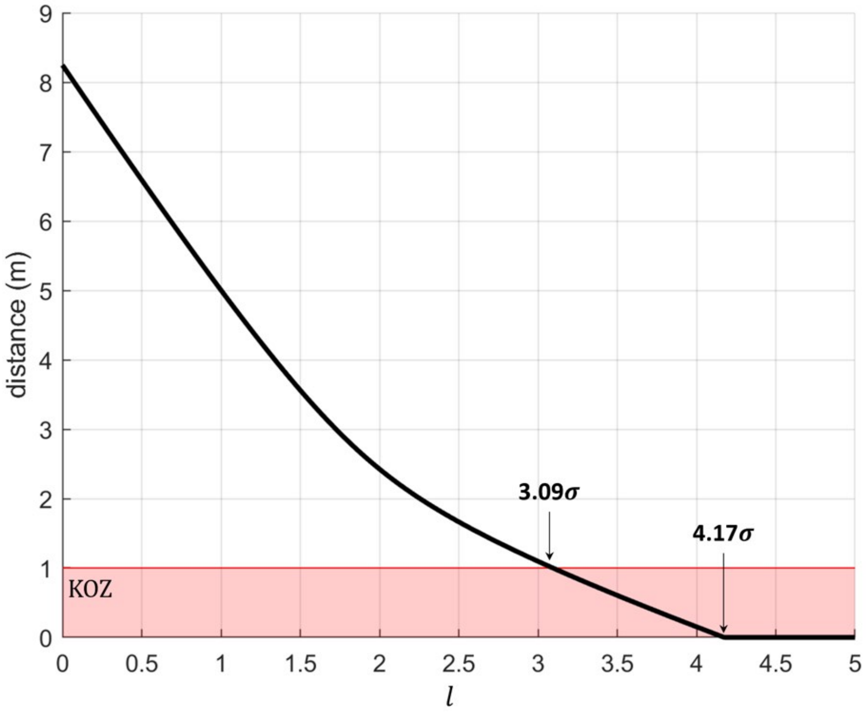

Fig. 8.12: Calculated distance between scaled ellipse and the origin.

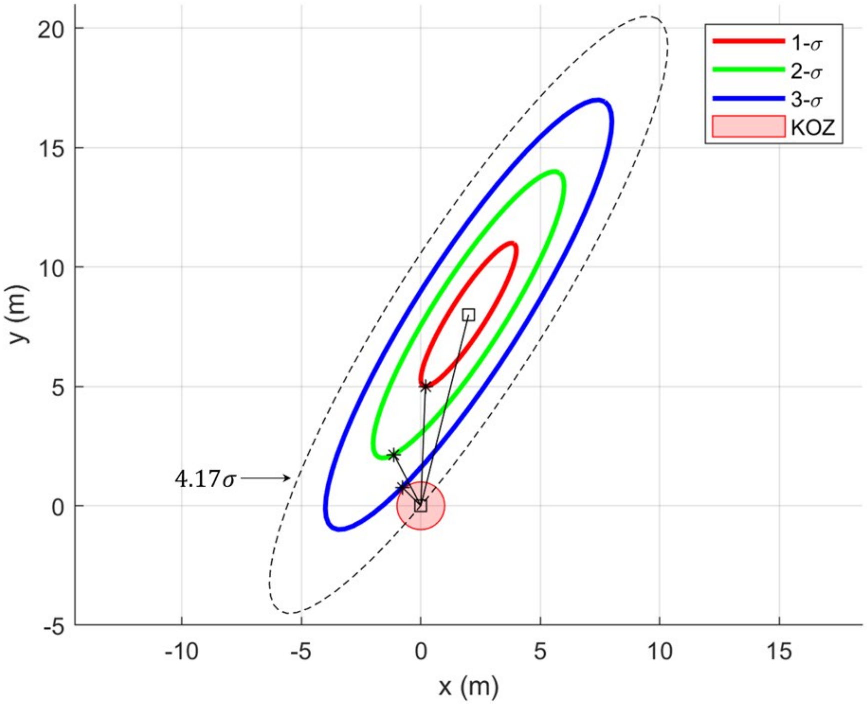

Fig. 8.13: Minimum distance between the 1, 2, and 3σ ellipses and the origin.

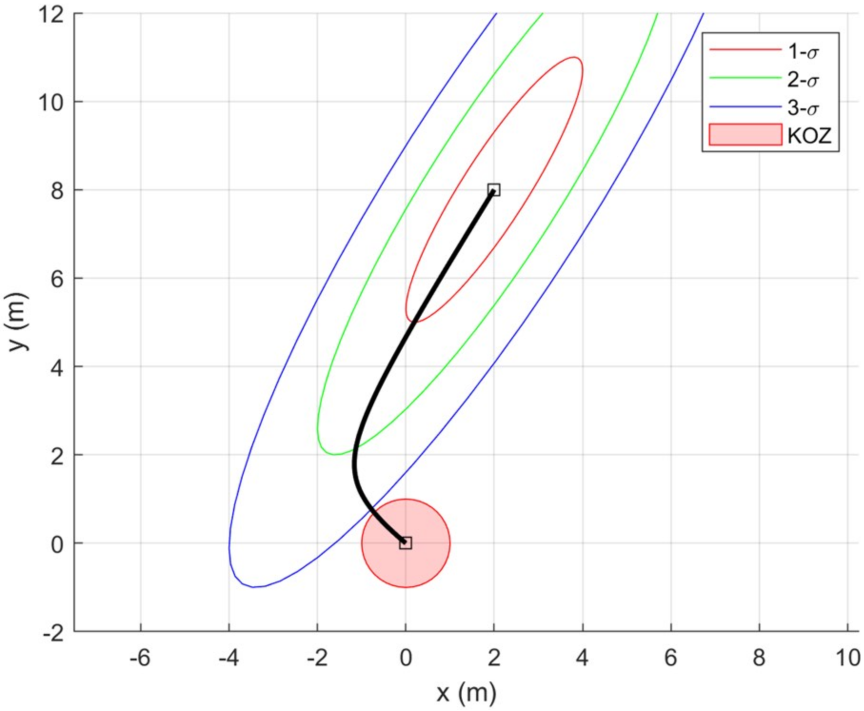

Fig. 8.14: Location of the point on the ellipse corresponding to the minimum distance to the origin as l increases from 0. The 1, 2, and 3σ ellipses are shown for reference.

With this approach we can now easily assess the safety of a relative trajectory by evaluating the desired $n\sigma$ true position dispersion at any given time step against the spherical keep-out zone. This method does not have the same assumptions and limitations at calculating the probability of collision, and accurately bounds the probability to a desired threshold.

Bounding the probability is more intuitive than specifying a threshold P_c. With this method it can be stated that a trajectory does not intersect the KOZ (either arbitrary or defined by the hard-body radius) to a given standard deviation. To maintain consistency with the NASA directive (though not direct correlation), trajectories should be evaluated at a minimum of the 4σ bound ($P_c < 1.13 \times 10^{-3}$) or at the 5σ bound ($P_c < 1.54 \times 10^{-5}$) for more conservatism.

Before demonstrating this bounded probability method for the NMC resizing scenario, we will expand the RPO closed-loop linear covariance models to include the tree analysis.

8.4 Linear Covariance Tree Analysis

Another step in determining spacecraft safety is the tree analysis that evaluates free-drift trajectories should maneuvers be halted. In this section we will derive the linear covariance equations for these branches of the analysis so that they can also be evaluated using the bounded probability technique developed in Sections 8.2.5 and 8.3.

The tree analysis is simplified by two assumptions: there are no additional maneuvers to be executed, and navigation has stopped. These assumptions are consistent with fault conditions and represent a conservative approach to the dispersions.

Each branch of the trajectory begins just prior to a planned maneuver, t_k^{-c}. True state dispersions be in the time between the previous maneuver (or initialization) and the start of each new branch are accounted for in the nominal trajectory analysis since the true state dispsersions are unaffected by navigation. This allows us to pull the augmented state covariance just prior to a maneuver to perform the tree analysis without any loss of accuracy should a fault occur immediately following a maneuver or at any point up to the next maneuver. The augmented state covariance matrix prior to a maneuver is $C(t_k^{-c})$ and corresponds to the initial augmented state covariance for each branch:

$$C^k(t_0) = C(t_k^{-c}) \tag{8.36}$$

where the superscript k indicates the branch index and corresponds to matching maneuver index. The linear covariance propagation model for each branch is based on Equation (4.64) and is given as:

$$C^k(t_{i+1}) = \mathcal{F} C^k(t_i) \mathcal{F}^T + \mathcal{G} Q_d \mathcal{G}^T \tag{8.37}$$

The analysis is performed by propagating each branch forward by a specified amount of time according to mission safety needs.

The true position dispersion at each time step is similar to Equation (4.74) and is given as:

$$D^k(t_i) = \begin{bmatrix} I_{3\times3} & 0_{3\times9} \end{bmatrix} C^k(t_i) \begin{bmatrix} I_{3\times3} & 0_{3\times9} \end{bmatrix}^T \tag{8.38}$$

The tree analysis of the NMC resizing scenario from Section 8.1.2 is repeated in Figure 8.15 using the bound probability approach to determine the minimum distance between the 4σ position dispersion and the RSO. Figure 8.16 shows a zoomed in view of the tree analysis indicating that branch #2 violates the 100 m keep-out zone and that branch #1 comes close to the keep-out zone but does not violate.

It is interesting to note that the nominal trajectory bounds are significantly smaller than the bounds on each of the free-drifting branches of the analysis. These uncertainties, associated with anomalous spacecraft behavior, are a significantly higher source of risk than nominally planned trajectories.

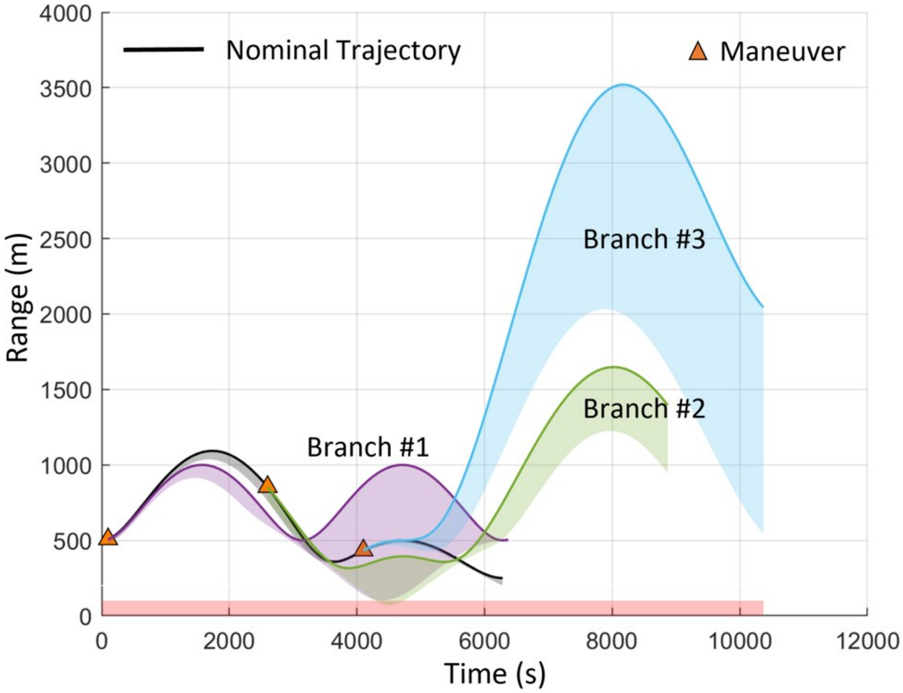

Fig. 8.15: Relative range for tree analysis of the NMC resizing scenario with overlayed minimum distance for the 4σ bounded probability covariance.

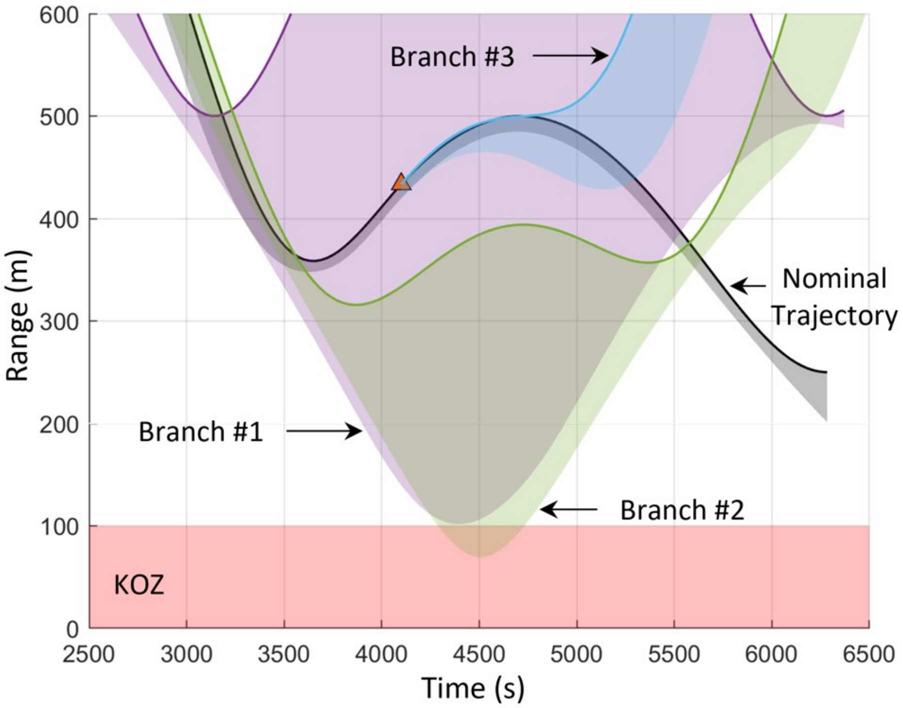

Fig. 8.16: Relative range for tree analysis of the NMC resizing scenario with overlayed minimum distance for the 4σ bounded probability covariance. Results are zoomed in to show the keep-out zone violation.

8.5 Conclusions

This chapter addressed the issues of safety and collision risk when planning and executing rendezvous and proximity operations missions. The key concepts of keep-out zones and tree analysis were discussed for assessing relative trajectories. A keep-out zone is used to ensure safety by maintaining a minimum distance between the active vehicle and the RSO. Keep-out zones are based on vehicle performance, mission requirements, or government policy. A tree analysis is used to ensure that keep-out zone restrictions are followed even in the event of missed maneuvers, as may occur during a spacecraft anomaly.

These concepts were then evaluated using various probabilistic methods that helped to inform planners when safety conditions may be violated. Methods including probability of collision, range uncertainty, and the Mahalanobis distance were explored but were found to be insufficient as reliable and interpretable metrics.

A new approach was developed that pairs the true position dispersions generated in a closed-loop linear covariance model with the familiar concept of bounded probability (i.e. the probability bounded by a specified number of standard deviations). This approach allows trajectory planners to determine if the range of possible trajectories violate a specified keep-out zone to a specified probability.

Finally, the RPO linear covariance model developed in Chapter 4 was expanded to include a tree analysis where true dispersions are generated for free-drift trajectories when planned maneuvers are not executed. This expansion will allow the STORM analysis tool, developed in Chapter 6, to account for trajectory safety when determining constrained optimal trajectories. This approach will be explored in the penultimate chapter of this work.

CHAPTER 9

OPTIMAL RELATIVE PATH PLANNING WITH POSITION DISPERSION AND SAFETY CONSTRAINTS

With the development of the bounded probability safety metric, we have all of the necessary elements to plan and evaluate safe, constrained relative trajectories for RPO spacecraft missions. To provide this full mission planning capability, this chapter will expand the STORM analysis tool (developed in Chapter 6) to include the bounded probability safety metric (developed in Chapter 8). A new optimization fitness function is developed to include path safety for both the nominal trajectory and branches of a tree analysis.

To understand the impacts of the new safety constraint on trajectory planning, a risk assessment of the previously evaluated three-impulse scenarios from Chapter 9 will be performed using the updated analysis tool. This assessment includes the NMC resizing and co-elliptic to NMC scenarios. These two scenarios are then optimized with the new safety constrained fitness function in order to evaluate the impacts of the safety constraint on the final trajectory. The V-bar hop scenario is also evaluated and compared to determine the impacts of the safety constraint. Since this scenario was only analyzed for the two-impulse transfer in Chapter 6, full trajectory assessments will be generated with and without the safety constraint.

9.1 Fitness Function

The optimization fitness function used by both the genetic algorithm and *fmincon* for the the safe trajectory planning problem is given as:

$$\min_{x} \quad J = \Delta V^{3\sigma} + P_{\text{disp}} + P_{\text{safety}} \tag{9.1}$$

where $\Delta V^{3\sigma}$ is the total 3σ ΔV consumption, P_{disp} is a penalty for violating the final position dispersion, and P_{safety} is a penalty term for violating the keep-out zone at any instant in time. As before, the total 3σ ΔV consumption, Equation (4.88), is defined as:

$$\Delta V^{3\sigma} = \sum_{j=1}^{n} \Delta V_j^{3\sigma}$$

and the dispersion penalty, Equation (6.2), is defined as:

$$P_{\text{disp}} = \begin{cases} 100\, S_{D_f}/S_{\text{lim}}, & \text{if } S_{D_f} > S_{\text{lim}} \\ 0, & \text{otherwise} \end{cases}$$

The safety penalty, P_{safety}, is calculated as:

$$P_{\text{safety}} = 100 \sum_{k=0}^{m} \sum_{i=1}^{n} I_{\text{koz}}(t_{\Delta V(k)} + t_i^k) \tag{9.2}$$

where I_{koz} is the flag indicating a safety violation as defined in Equation (8.34). Safety violations are evaluated at each time step (i) along the nominal trajectory and each branch (k) of the trajectory, t_i^k. The nominal trajectory starts at $t_0 = t_{\Delta V(0)}$ and is identified by time steps t_i^0. The m branch trajectories start at the time of their associated maneuver, $t_{\Delta V(k)}$, and occur over the time steps t_i^k. The total number time steps depends on both the duration of the scenario and the length of the safety look ahead window. Violations are assessed using the bounded probability method developed in Section 8.2.5 where violations are indicated by an intersection between the $l\sigma$ true position dispersion and the RSO keep-out zone. The safety penalty is designed to reduce to zero as the analyzed trajectories are pushed away from the keep-out zone.

As with the final position dispersion constraint in Section 6.1.4, the scale factor of 100 in Equation (9.2) was selected to create a penalty larger than any expected value for ΔV. Care should be taken when selected a scale factor to ensure the final solution meets the safety constraint.

9.2 NMC Resizing

This section will revisit the constrained, three-impulse NMC resizing scenario defined and analyzed in Sections 5.1 and 7.1. Analysis will specifically focus on the good lighting case of this scenario where the sun begins 90 deg off the sensor boresignt. For clarity, this previously analyzed trajectory will be referred to as *simply constrained* to distinguish if from analysis that includes the safety constraint, referred to as *safety constrained*. The safety constrained case contains both the final position dispersion constraint associated with the nominal trajectory and the bounded probability safety constraint as shown in the updated fitness function, Equation (9.1). Inputs for both scenarios are defined in Tables 9.1 and 9.2

Table 9.1: Nominal trajectory inputs for NMC resizing scenario.

Parameter	Value	Units
$\hat{\mathrm{r}}(t_0)$	$[-500,\ 0,\ 0]^T$	m
$\hat{\mathrm{v}}(t_0)$	$[0,\ 1,\ 0]^T$	m/s
$\hat{\mathrm{r}}(t_f)$	$[-250,\ 0,\ 0]^T$	m
$\hat{\mathrm{v}}(t_f)$	$[0,\ 0.5,\ 0]^T$	m/s
ω	0.001	rad/s
t_f	$6283\ (= T)$	s

Table 9.2: NMC resizing scenario noise parameters, camera constraints, and safety parameters.

Parameter	Value	Units
$C_0(rr)$ (3σ)	$(10)^2 I_{3\times 3}$	m^2
$C_0(vv)$ (3σ)	$(0.01)^2 I_{3\times 3}$	m^2/s^2
$C_0(rv)$ (3σ)	$0_{3\times 3}$	m^2/s
Q_w (3σ)	$(10^{-9})^2 I_{3\times 3}$	m^2/s^3
$S_{\Delta V}$ (3σ)	$(0.03\Delta\bar{V})^2 I_{3\times 3}$	m^2/s^2
R_ν^{AON} (3σ)	$(10^{-3})^2 I_{3\times 3}$	rad^2
R_ν^{lidar} (3σ)	n/a	n/a
Earth Exclusion, $\theta_\oplus^{lim}$	10	deg
Sun Exclusion, $\theta_\odot^{lim}$	45	deg
Initial Sun Vector, $\hat{s}(t_0)$	$[0, 1, 0]^T$	n/a
Bounded Sigma, l	4	n/a
KOZ Radius, R_{koz}	200	m
Hard-body Radius, R_{hb}	10	m
Branch Horizon, Δt	6283 $(= T)$	s
Dispersion Requirement, S_{lim}	5	m

9.2.1 Simply Constrained Results

The simply constrained optimal trajectory was evaluated to determine whether or not it met the proposed safety considerations. As defined in Table 9.2, safety is assessed against the 4σ bounded probability and 200 m RSO keep-out zone. The nominal range for the trajectory and tree analysis, as well as the minimum range, are shown in Figure 9.1. The second branch of the tree analysis (green) violates the keep-out zone with a minimum range of 178 m.

Though not used explicitly for the trajectory planning, the probability of collision was also calculated for comparison purposes and shown in Figure 9.2. The maximum probability of collision (also occurring along the second branch) is $P_c = 1.54 \times 10^{-8}$, assuming a combined hard-body radius of 10 m. The P_c value for the nominal trajectory never exceeds 10^{-50}. It should be noted that these P_c values were calculated using a numerical integration of Equation (8.13) rather than the simplified version in Equation (8.15) in order to generate accurate results. Inclusion of the numerical integration calculation in the RPO LinCov analysis tool increases the function run-time by a factor of 31 (from approximately 0.048 seconds per run up to 1.5 seconds). Slowing down the STORM algorithm to this degree would render the analysis tool operationally irrelevant.

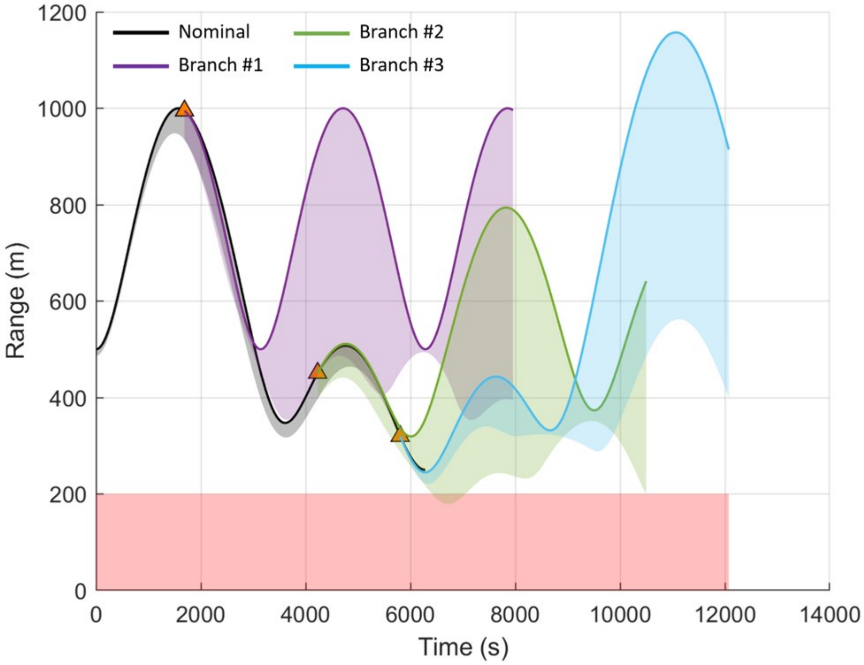

Fig. 9.1: Range and 4σ minimum range projection for simply constrained NMC resizing scenario.

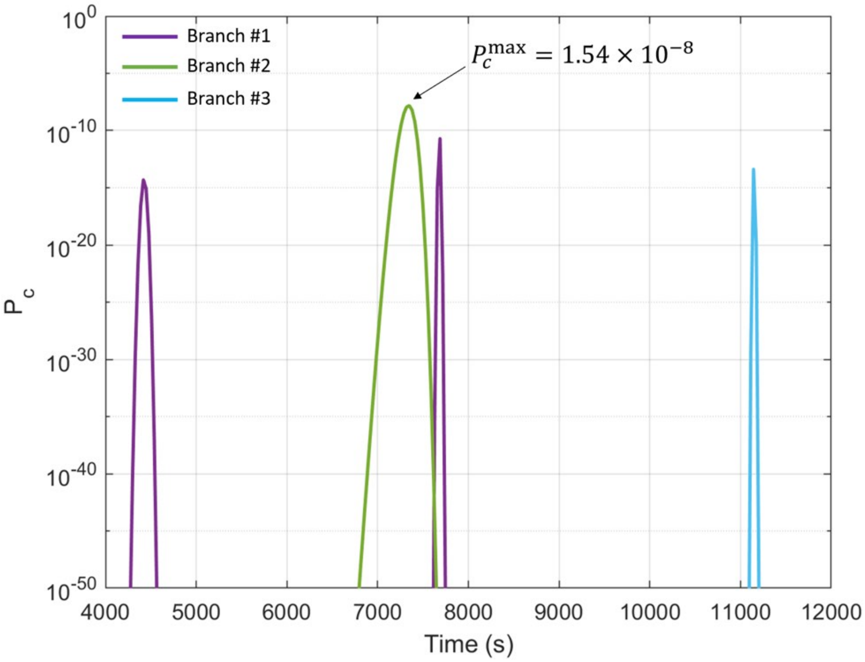

Fig. 9.2: Probability of collision for simply constrained NMC resizing scenario.

9.2.2 Safety Constrained Results

The STORM analysis tool was used to solve the safety constrained optimal NMC resizing scenario with the revised fitness function in Equation (9.1). The optimization results for ΔV and final position dispersion are given in Table 9.3, and the results for timing and waypoint location are provided in Table 9.4. Values are shown next to the previously computed values for the simply constrained case.

The optimal trajectory and maneuvers for the safety constrained case are shown in Figure 9.3. Overlaid on the trajectory are the regions where the angle measurements are available and lighting constraints are not violated. The nominal trajectory and three branches are depicted in Figure 9.4. Neither the nominal trajectory nor the branches violate the RSO keep-out zone. The nominal range between the vehicle and the RSO, as well as the 4σ minimum range are shown in Figure 9.5. The addition of the safety constraint successfully modified the trajectory such that the minimum range no longer violates the keep-out zone. The minimum range along the third branch (blue), however, does brush up against the keep-out zone. It should also be noted that the addition of the safety constraint led to a smaller final position dispersion of 3.59 m against the 5 m constraint. Rerunning the scenario without the 5 m final position dispersion requirement yielded the same results. The deterministic minimum fuel usage, $\Delta\bar{V}$, was previously determined to be 0.1250 m/s.

Table 9.3: LinCov results for the three-impulse NMC resizing scenario. $\Delta\bar{V} = 0.1250$ m/s.

Case	$\Delta V^{3\sigma}$ (m/s)	ΔV (m/s)	$\Delta V/\Delta\bar{V}$	D_f (m)
Simply Constrained	0.41167	0.21198	1.6958	5.00
Safety Constrained	0.87016	0.52279	4.1823	3.59

Table 9.4: Optimal time and waypoint results for the three-impulse NMC resizing scenario.

Case	$\Delta t_0/T$	$\Delta t_1/T$	$\Delta t_2/T$	R_x (m)	R_y (m)	R_z (m)
Simply Constrained	0.26862	0.40440	0.25068	189.8	-409.1	0.0
Safety Constrained	0.35396	0.57090	0.07481	-291.0	-327.8	0.0

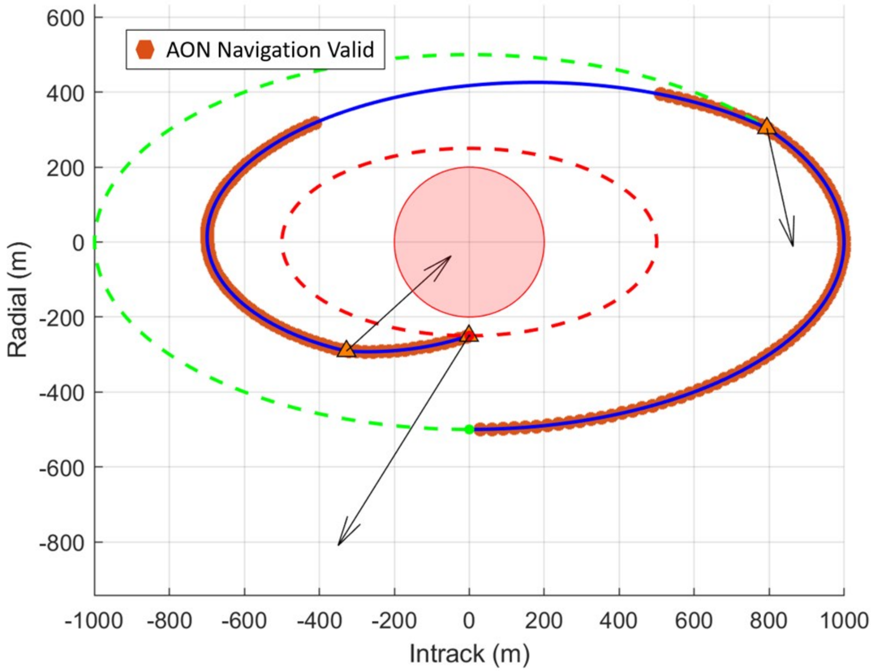

Fig. 9.3: NMC resizing scenario trajectory path for the safety constrained optimal solution.

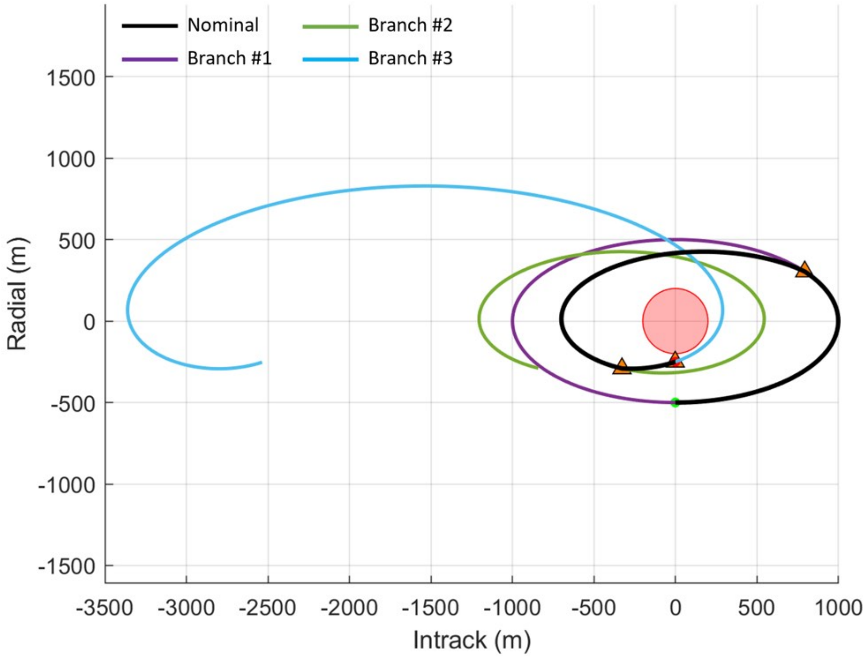

Fig. 9.4: Tree analysis of safety constrained NMC resizing scenario.

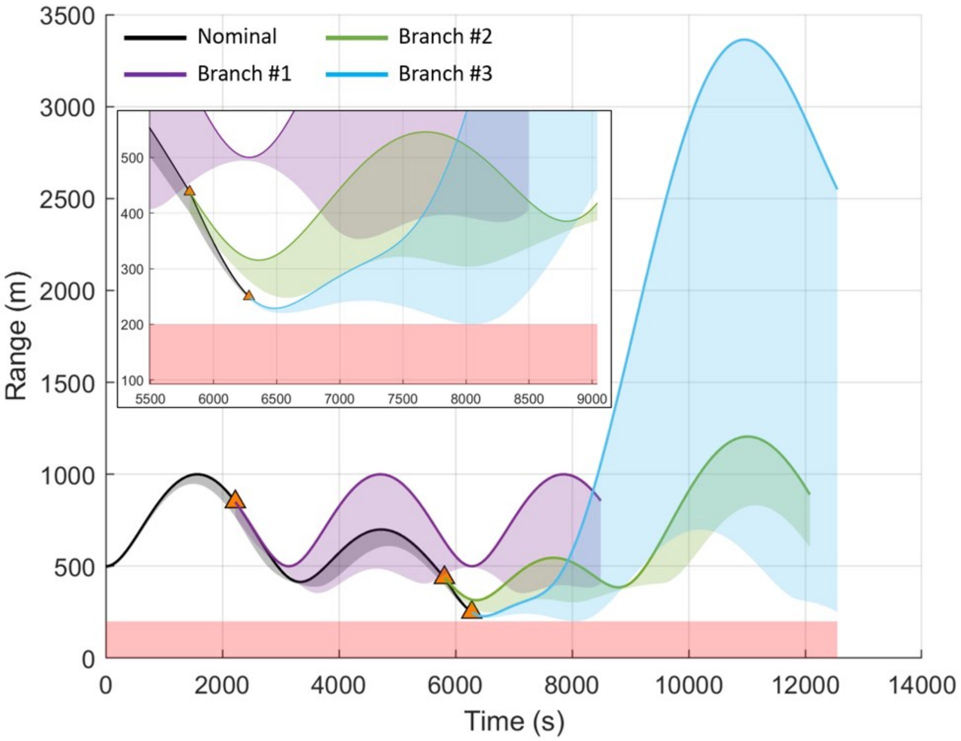

Fig. 9.5: Range and 4σ minimum range for safety constrained NMC resizing scenario.

The probability of collision is shown in Figure 9.6. The maximum calculated probability, $P_c^{\max} = 2.44 \times 10^{-11}$, now corresponds to branch 1, rather than branch 2 as was the case in the simply constrained trajectory. The maximum probability along branch 2 has dropped significantly to 10^{-40} and branch 3 does not appear at this scale. Again, P_c for the nominal trajectory does not appear on the scale plotted.

The nominal ΔV and dispersions for the three maneuvers are depicted in Figure 9.7. The addition of the safety constraint increases the optimal $\Delta V^{3\sigma}$ by 110% when compared to the simply constrained trajectory. While the keep-out zone size was selected to demonstrate the benefit of accounting for safety, doubling the fuel consumption can be significant for mission planning. If the constraint is not absolutely necessary, changing the keep-out zone to 150 m would reduce the $\Delta V^{3\sigma}$ back to the original simply constrained case since at 150 m no safety constraints are violated.

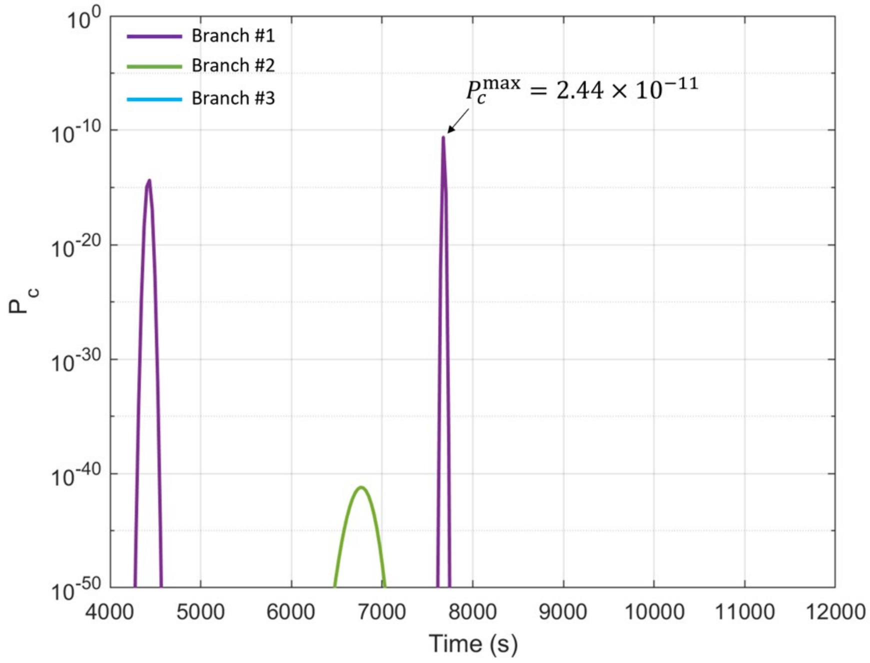

Fig. 9.6: Probability of collision for safety constrained NMC resizing scenario.

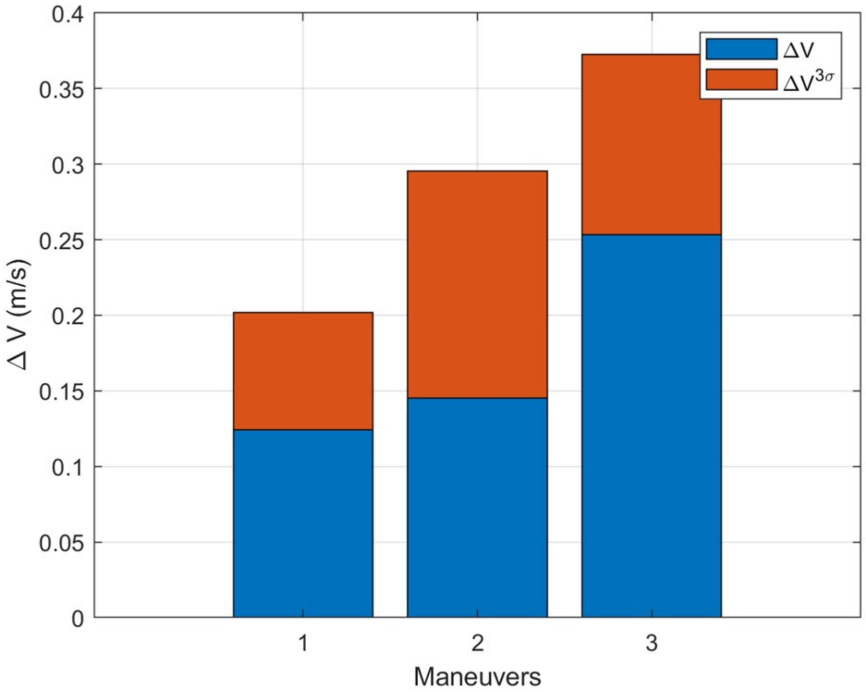

Fig. 9.7: ΔV and 3σ ΔV dispersion values for the safety constrained NMC resizing scenario.

9.3 Co-elliptic to NMC

This section will revisit the constrained, three-impulse co-elliptic to NMC scenario defined and analyzed in Section 7.2. Analysis will specifically focus on the constrained case of this scenario with a fixed final time and a final position dispersion requirement of 15 m. For clarity, this previously analyzed trajectory will again be referred to as *simply constrained* to distinguish if from analysis that includes the safety constraint, referred to as *safety constrained*. The inputs for both scenarios are defined in Tables 9.5 and 9.6

Table 9.5: Nominal trajectory inputs for the co-elliptic to NMC scenario.

Parameter	Value	Units
$\hat{\mathrm{r}}(t_0)$	$[-750,\ -7065,\ 0]^T$	m
$\hat{\mathrm{v}}(t_0)$	$[0,\ 1.1250,\ 0]^T$	m/s
$\hat{\mathrm{r}}(t_f)$	$[-150,\ 0,\ 0]^T$	m
$\hat{\mathrm{v}}(t_f)$	$[0,\ 0.3,\ 0]^T$	m/s
ω	0.001	rad/s
t_f	12566 $(= 2T)$	s

Table 9.6: Co-elliptic to NMC scenario noise parameters, camera constraints, and safety parameters.

Parameter	Value	Units
$C_0(rr)$ (3σ)	$(10)^2 I_{3\times3}$	m^2
$C_0(vv)$ (3σ)	$(0.01)^2 I_{3\times3}$	m^2/s^2
$C_0(rv)$ (3σ)	$0_{3\times3}$	m^2/s
Q_w (3σ)	$(10^{-9})^2 I_{3\times3}$	m^2/s^3
$S_{\Delta V}$ (3σ)	$(0.03\Delta\bar{V})^2 I_{3\times3}$	m^2/s^2
R_ν^{AON} (3σ)	$(10^{-3})^2 I_{3\times3}$	rad^2
R_ν^{lidar} (3σ)	n/a	n/a
Earth Exclusion, $\theta_\oplus^{lim}$	10	deg
Sun Exclusion, $\theta_\odot^{lim}$	45	deg
Initial Sun Vector, $\hat{s}(t_0)$	$[-1, 0, 0]^T$	n/a
Bounded Sigma, l	4	n/a
KOZ Radius, R_{koz}	100	m
Hard-body Radius, R_{hb}	10	m
Branch Horizon, Δt	12566 $(= 2T)$	s
Dispersion Requirement, S_{lim}	15	m

9.3.1 Simply Constrained Results

The simply constrained optimal trajectory was evaluated to determine whether or not it meets the proposed safety considerations. As defined in Table 9.6, safety is assessed against the 4σ true position dispersion and 100 m RSO keep-out zone. The nominal and minimum ranges for the trajectory and tree analysis are shown in Figure 9.8. The keep-out zone is violated by both the nominal trajectory and the second branch of the tree analysis with minimum ranges of 29 m and 19 m, respectively.

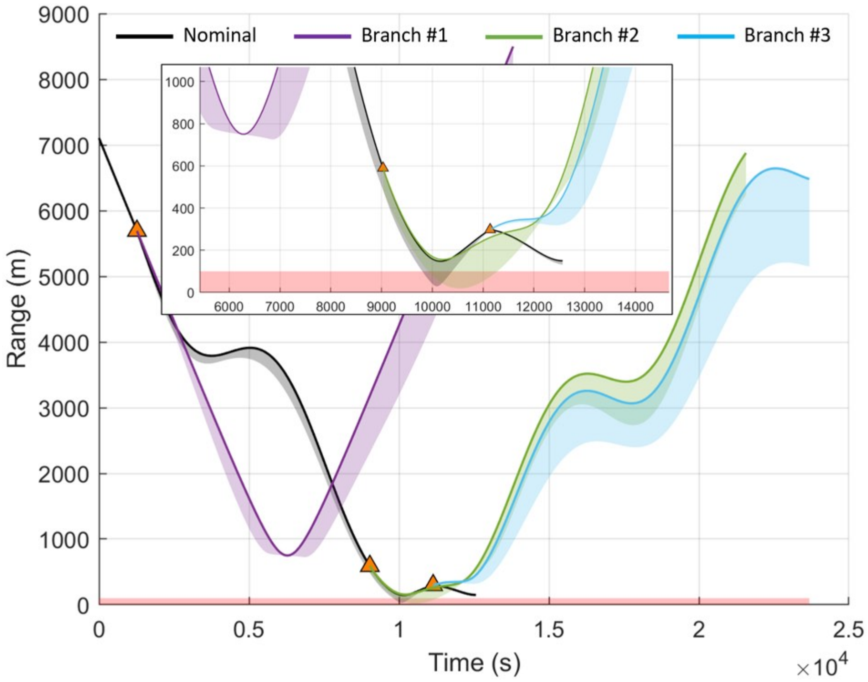

Fig. 9.8: Range and 4σ minimum range for co-elliptic to NMC scenario without safety constraint. Minimum range violates RSO keep-out zone.

9.3.2 Safety Constrained Results

The STORM analysis tool was used to solve the safety constrained optimal co-elliptic to NMC transfer scenario with the revised fitness function in Equation (9.1). The optimization results for ΔV and final position dispersion are given in Table 9.7, and the results for timing and waypoint location are provided in Table 9.8. The optimal trajectory and maneuvers for the safety constrained scenario are shown in Figure 9.8. Overlaid on the trajectory are the regions where the angle measurements are available and lighting constraints are not violated. The nominal trajectory and three branches are depicted in Figure 9.10. A significant amount of safety is achieved through Crosstrack motion as seen in Figure 9.11. Neither the nominal trajectory nor the branches violate the RSO keep-out zone.

The nominal range between the vehicle and the RSO, as well as the 4σ minimum range are shown in Figure 9.12. The addition of the safety constraint successfully modified the trajectory such that the minimum range no longer violates the 100 m keep-out zone. The minimum range along the third branch, however, does brush up against the keep-out zone.

Table 9.7: LinCov results for the three-impulse coelliptic to NMC scenario. $\Delta\bar{V} = 0.3750$ m/s.

Case	$\Delta V^{3\sigma}$ (m/s)	ΔV (m/s)	$\Delta V/\Delta\bar{V}$	D_f (m)
Simply Constrained	0.56363	0.37539	1.0010	15.0
Safety Constrained	1.09473	0.76334	2.03558	15.0

Table 9.8: Optimal time and waypoint results for the three-impulse co-elliptic to NMC scenario.

Case	$\Delta t_0/T$	$\Delta t_1/T$	$\Delta t_2/T$	R_x (m)	R_y (m)	R_z (m)
Simply Constrained	0.26862	0.40440	0.25068	189.8	-409.1	0.0
Safety Constrained	1.07263	0.63409	0.28243	415.5	-107.5	-91.0

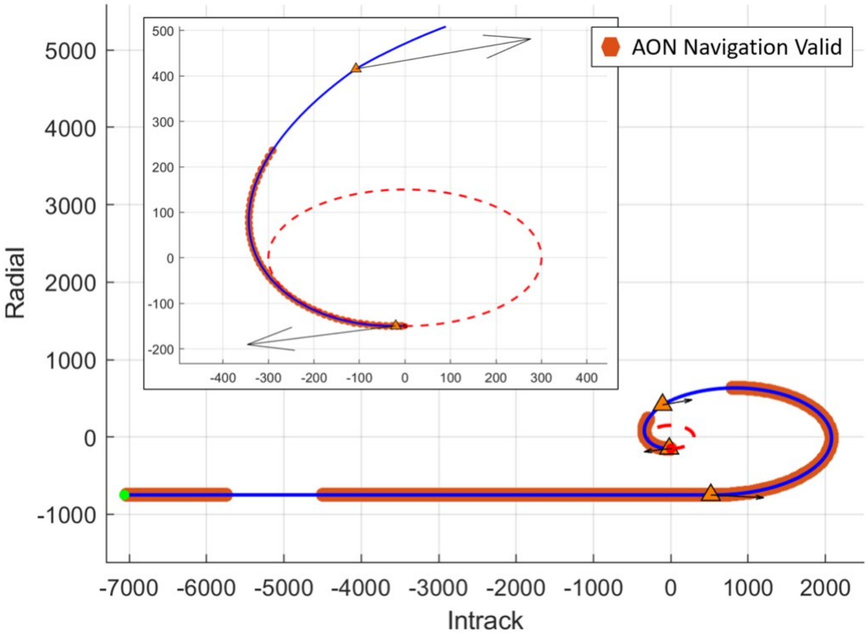

Fig. 9.9: Co-elliptic to NMC scenario trajectory for the safety constrained optimal solution.

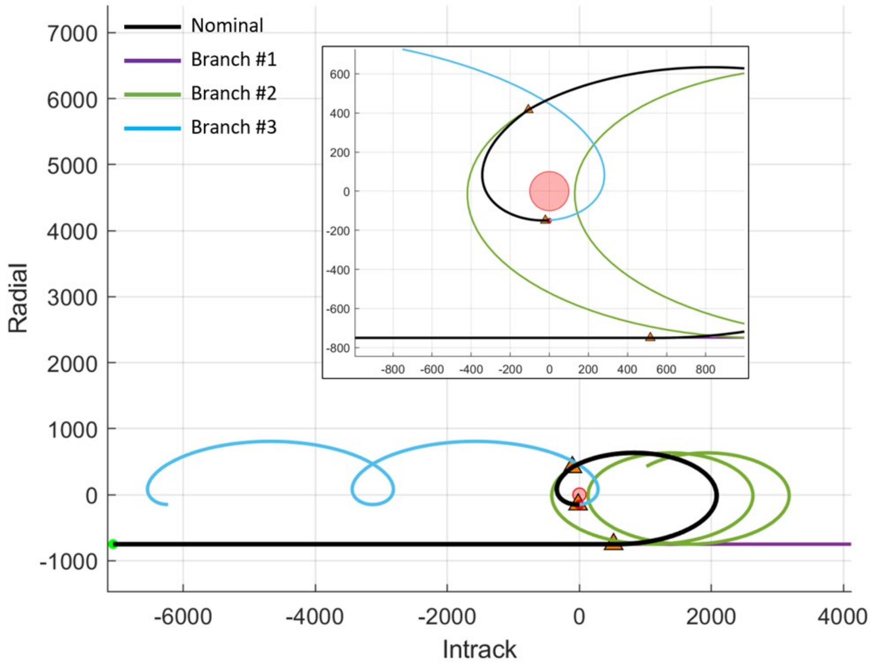

Fig. 9.10: Tree analysis of the safety constrained co-elliptic to NMC scenario. Radial-Intrack plane. Branch 1 not shown in full for clarity.

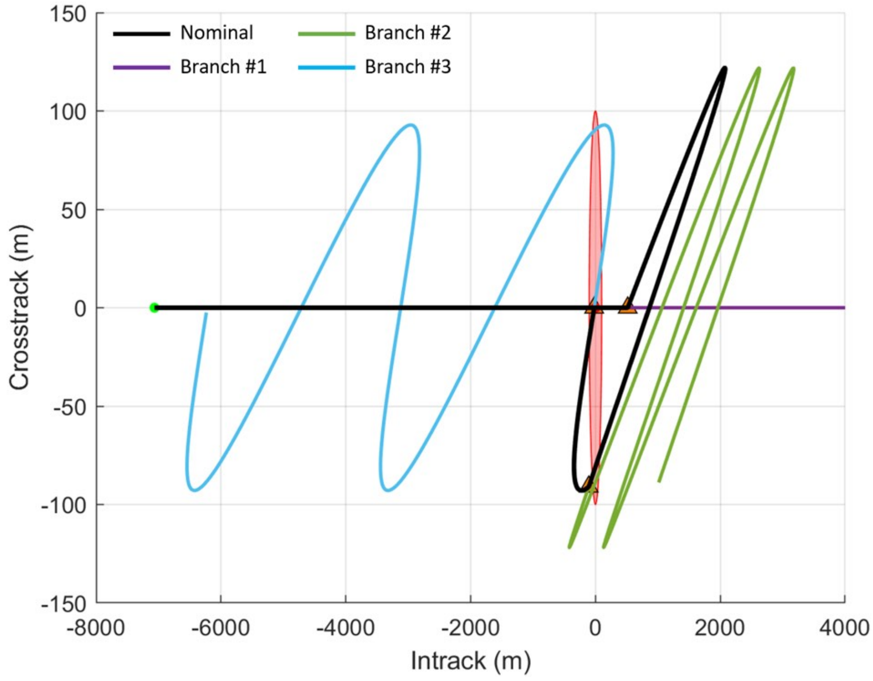

Fig. 9.11: Tree analysis of safety constrained co-elliptic to NMC scenario. Crosstrack-Intrack plane. Branch 1 not shown in full for clarity.

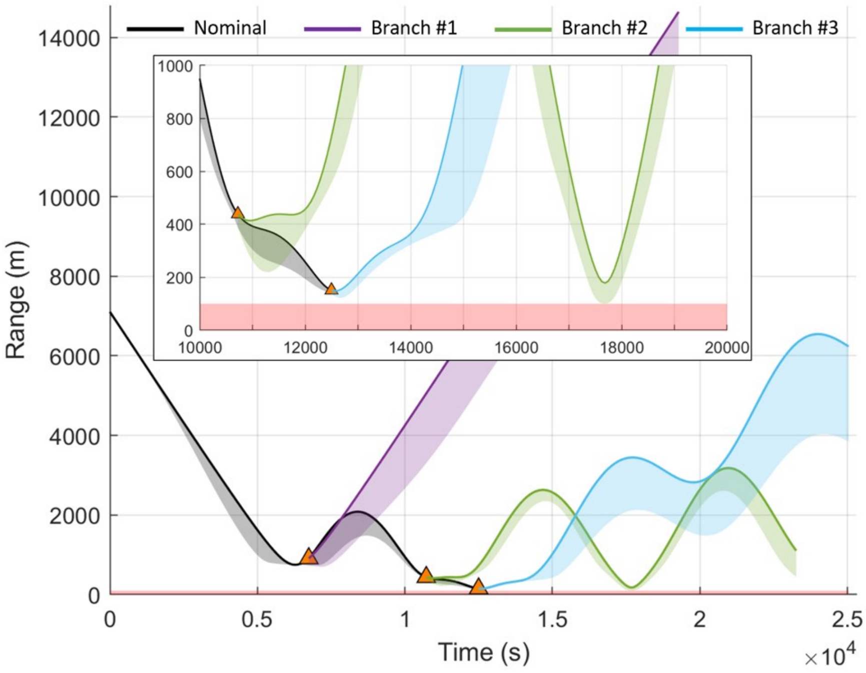

Fig. 9.12: Range and 4σ minimum range for safety constrained co-elliptic to NMC scenario.

To better depict the spatial relation between the RSO keep-out zone and the true position dispersion, three-dimensional views of the minimum 4σ distance for branches 2 and 3 and shown in Figures 9.13 and 9.14. Note that the true position ellipsoid for branch 2 and the spherical RSO keep-out zone just barely touch as shown in Figure 9.12.

The nominal ΔV and dispersions for the three maneuvers are depicted in Figure 9.15. The addition of the safety constraint increases the optimal $\Delta V^{3\sigma}$ by 94% compared to the simply constrained case.

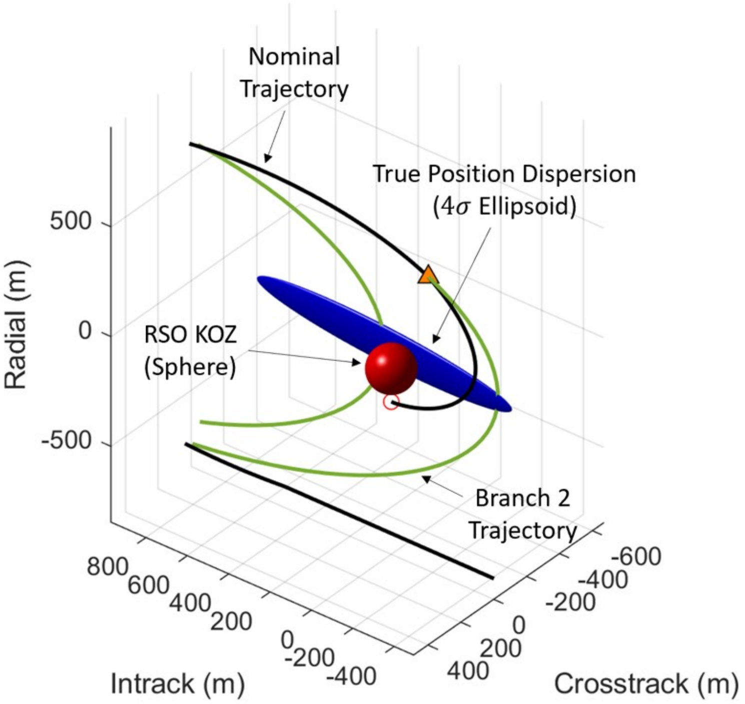

Fig. 9.13: True position dispersion ellipsoid for tree analysis branch 2 at the time of minimum distance relative to the RSO keep-out zone.

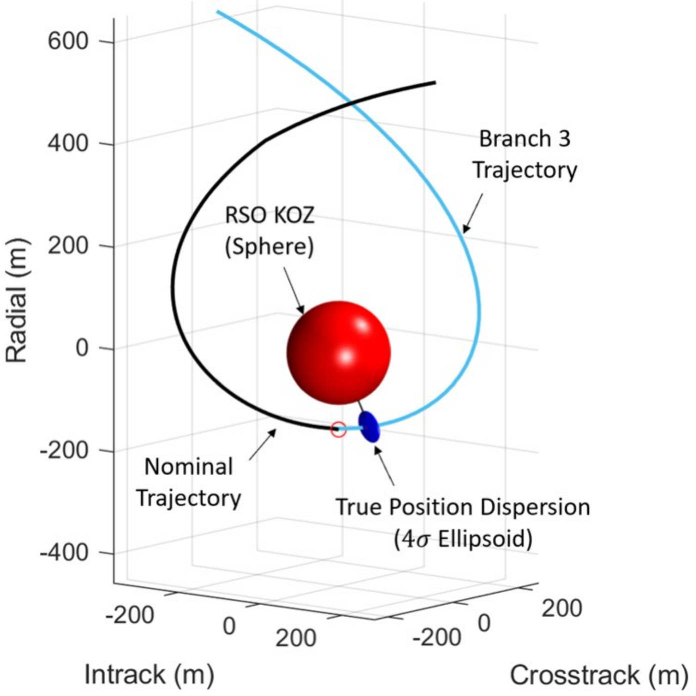

Fig. 9.14: True position dispersion ellipsoid for tree analysis branch 3 at the time of minimum distance relative to the RSO keep-out zone.

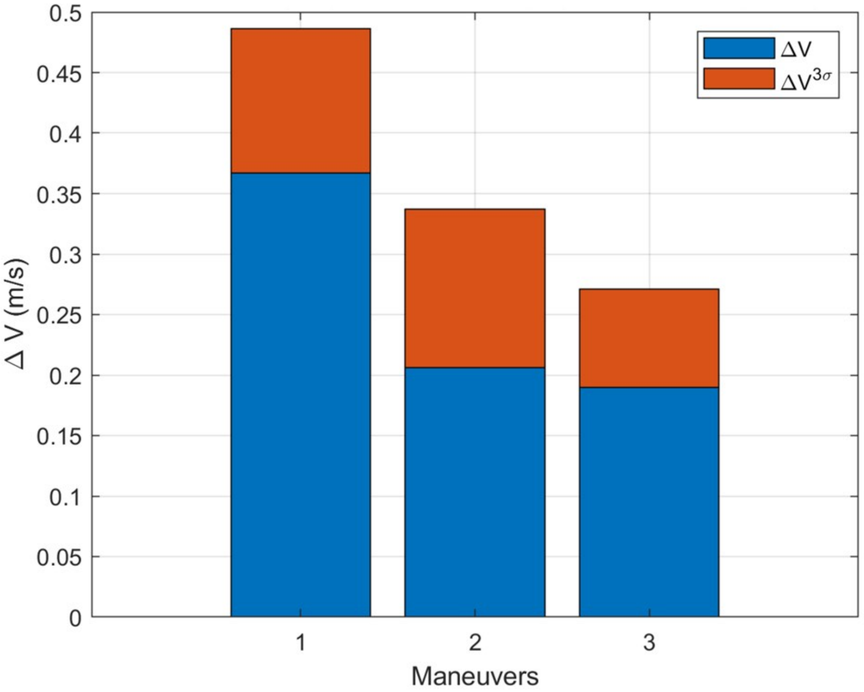

Fig. 9.15: ΔV and 3σ ΔV dispersion values for the safety constrained co-elliptic to NMC scenario.

9.4 V-bar Hop

This section will revisit the V-bar hop scenario from Section 5.2. Since the scenario was previously analyzed only for the constrained, two-impulse transfer, a complete three-impulse analysis will be performed using STORM. The navigation scheme will also be switched to incorporate passive, angles-only measurements. Two cases will be analyzed. The first, referred to as *simply constrained*, will minimize the fitness function in Equation (6.1) in order to minimize $\Delta V^{3\sigma}$ while meeting the final position dispersion constraint. The second analysis, referred to as *safety constrained*, will minimize the fitness function in Equation (9.1) that includes the safety constraint. The scenario and simulation parameters are given in Tables 9.9 and 9.10.

Keeping a spacecraft on the V-bar is an inherently difficult and risky operation given the unstable equilibrium of the relative dynamics. Being slightly above or below the V-bar can cause the spacecraft to slowly drift towards (or away from) the RSO, leading to a potential collision. The risk is increased when using passive, angles-only measurements given their difficulty in measuring range. Many missions avoid operating on the V-bar due to these risks. The initial position and velocity dispersions in Table 9.10 are lower than in previous cases to reflect tighter control requirements for V-bar operations. The results for the simply and safety constrained analysis are shown in Tables 9.11 and 9.12. ΔV values are compared against the deterministic single revolution transfer minimum of 0.0106 m/s previously calculated in Section 5.2.

Table 9.9: Nominal trajectory inputs for the V-bar hop scenario.

Parameter	Value	Units
$\hat{\mathrm{r}}(t_0)$	$[-250,\ 0,\ 0]^T$	m
$\hat{\mathrm{v}}(t_0)$	$[0,\ 0,\ 0]^T$	m/s
$\hat{\mathrm{r}}(t_f)$	$[-150,\ 0,\ 0]^T$	m
$\hat{\mathrm{v}}(t_f)$	$[0,\ 0,\ 0]^T$	m/s
ω	0.001	rad/s
t_f	$12566\ (= 2T)$	s

Table 9.10: V-bar hop scenario noise parameters, camera constraints, and safety parameters.

Parameter	Value	Units
$C_0(rr)$ (3σ)	$(1)^2 I_{3\times3}$	m^2
$C_0(vv)$ (3σ)	$(0.002)^2 I_{3\times3}$	$\mathrm{m}^2/\mathrm{s}^2$
$C_0(rv)$ (3σ)	$0_{3\times3}$	m^2/s
Q_w (3σ)	$(10^{-9})^2 I_{3\times3}$	$\mathrm{m}^2/\mathrm{s}^3$
$S_{\Delta V}$ (3σ)	$(0.03\Delta\bar{V})^2 I_{3\times3}$	$\mathrm{m}^2/\mathrm{s}^2$
R_ν^{AON} (3σ)	$(10^{-3})^2 I_{3\times3}$	rad^2
R_ν^{lidar} (3σ)	n/a	n/a
Earth Exclusion, $\theta_\oplus^{lim}$	10	deg
Sun Exclusion, $\theta_\odot^{lim}$	45	deg
Initial Sun Vector, $\hat{s}(t_0)$	$[-1,0,0]^T$	n/a
Bounded Sigma, l	4	n/a
KOZ Radius, R_{koz}	100	m
Hard-body Radius, R_{hb}	10	m
Branch Horizon, Δt	12566 $(= 2T)$	s
Dispersion Requirement, S_{lim}	2	m

Table 9.11: LinCov results for the three-impulse V-bar hop scenario. $\Delta\bar{V} = 0.0106$ m/s.

Case	$\Delta V^{3\sigma}$ (m/s)	ΔV (m/s)	$\Delta V/\Delta\bar{V}$	D_f (m)
Simply Constrained	0.02236	0.01070	1.0094	0.6
Safety Constrained	0.12142	0.07543	7.1160	1.4

Table 9.12: Time and waypoint results for the three-impulse V-bar hop scenario.

Case	$\Delta t_0/T$	$\Delta t_1/T$	$\Delta t_2/T$	R_x (m)	R_y (m)	R_z (m)
Simply Constrained	0.92824	0.76177	0.25236	-10.2	-152.9	0.0
Safety Constrained	1.27296	0.54624	0.14468	6.0	-158.0	0.0

9.4.1 Simply Constrained Results

The simply constrained optimal trajectory and maneuvers are shown in Figure 9.16. Overlaid on the trajectory are the regions where the angle measurements are available and lighting constraints are not violated. The angles-only navigation measurements are unavailable across certain time frames due to violations of the sun constraint as shown in Figure 9.17. The final true position dispersion, at 0.6 m, is far under the dispersion requirement as shown in Figure 9.18. Rerunning the scenario without the final position dispersion requirement yields the same optimal trajectory.

Figure 9.19 plots the tree analysis for the transfer where both the second and third branches drift into the keep-out zone. Figure 9.20 shows the range and 4σ minimum range for the nominal trajectory and branches, further depicting the KOZ violation. Branch 2 has a minimum range of 6.8 m and branch 3 a minimum range of 0.1 m. While the transfer is near an absolute minimum for fuel consumption, the trajectory is clearly a high risk for collision.

The ΔV results are shown in Figure 9.21. It is important to note that the results indicate two distinct maneuvers equal in magnitude at the beginning and end of the transfer, while the middle maneuver serves as a mid-course correction. A mid-course correction is characterized by the nominal ΔV value of zero and is performed to help reduce the final true position dispersion. Because of this corrective maneuver, the nominal branches 2 and 3 in Figure 9.19 lie on top of each other. Maneuver execution errors, however, cause the minimum range for the two branches to diverge slightly as seen in Figure 9.20.

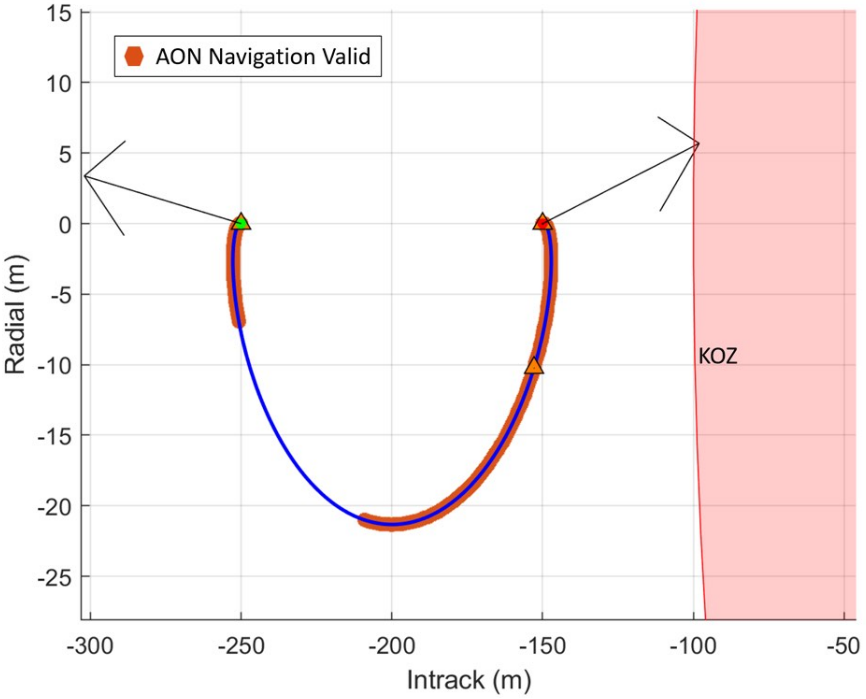

Fig. 9.16: ΔV and 3σ ΔV dispersions for the simply constrained V-bar hop scenario.

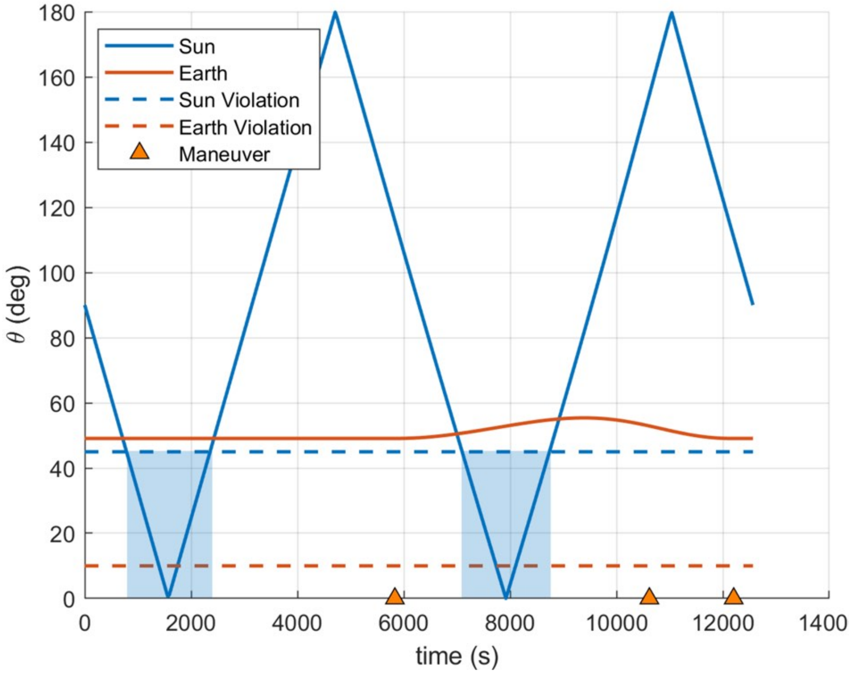

Fig. 9.17: Sun and earth angles for the simply constrained V-bar hop scenario.

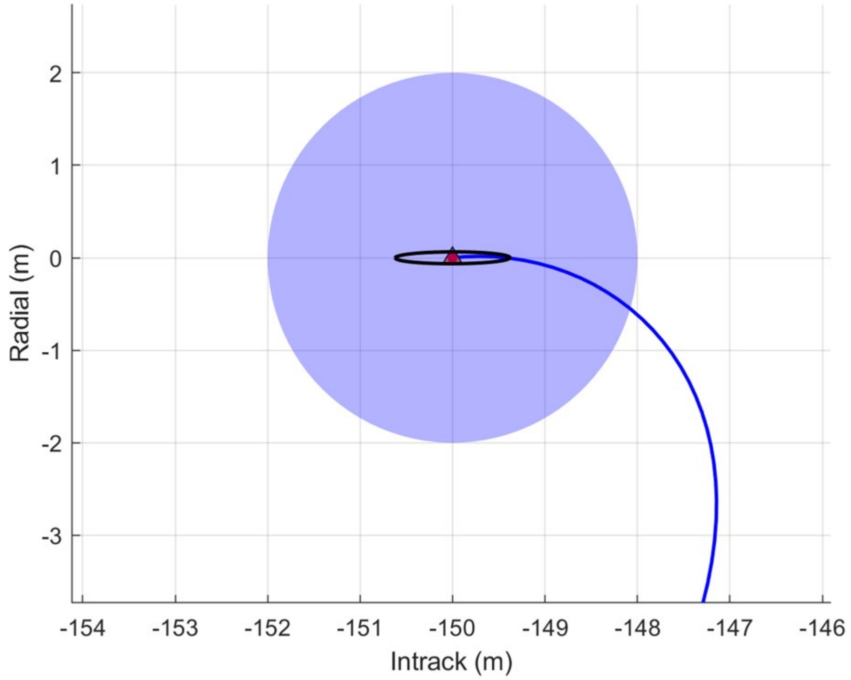

Fig. 9.18: Final true position dispersion for the simply constrained V-bar hop scenario. Blue circle represents the 2 m dispersion.

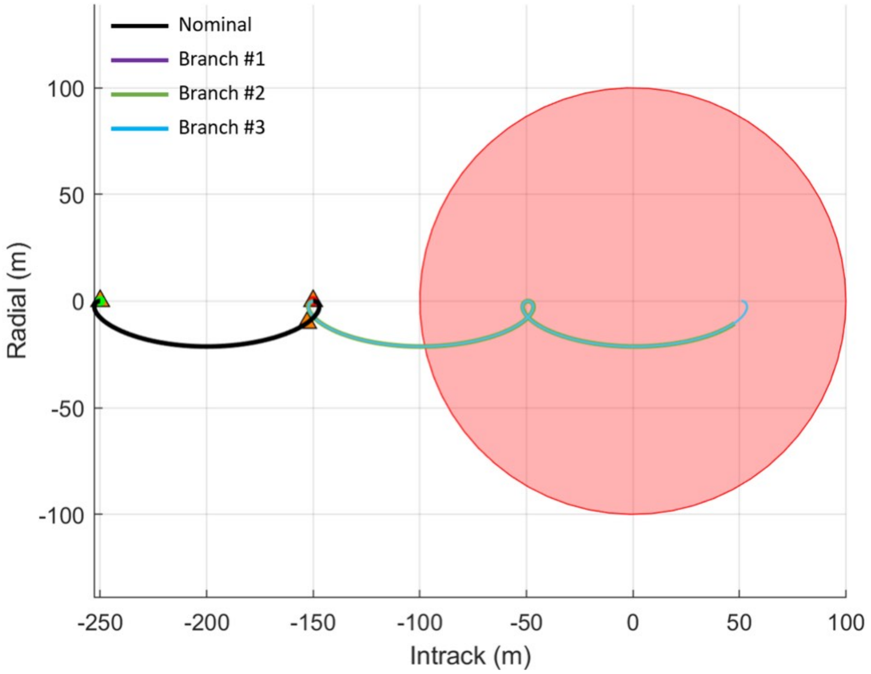

Fig. 9.19: Tree analysis of simply constrained V-bar hop scenario.

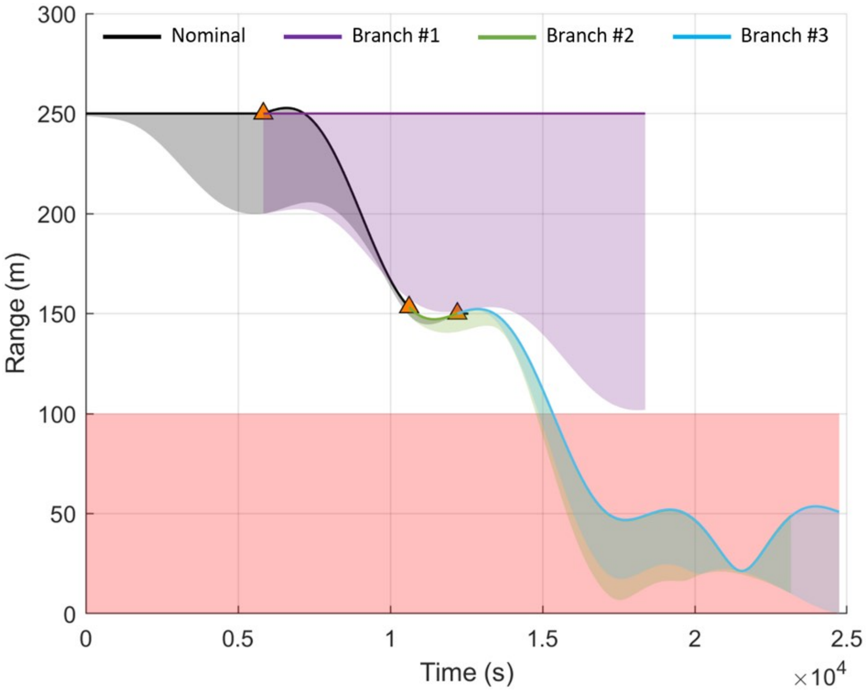

Fig. 9.20: Range and 4σ minimum range projection for simply constrained V-bar hop scenario.

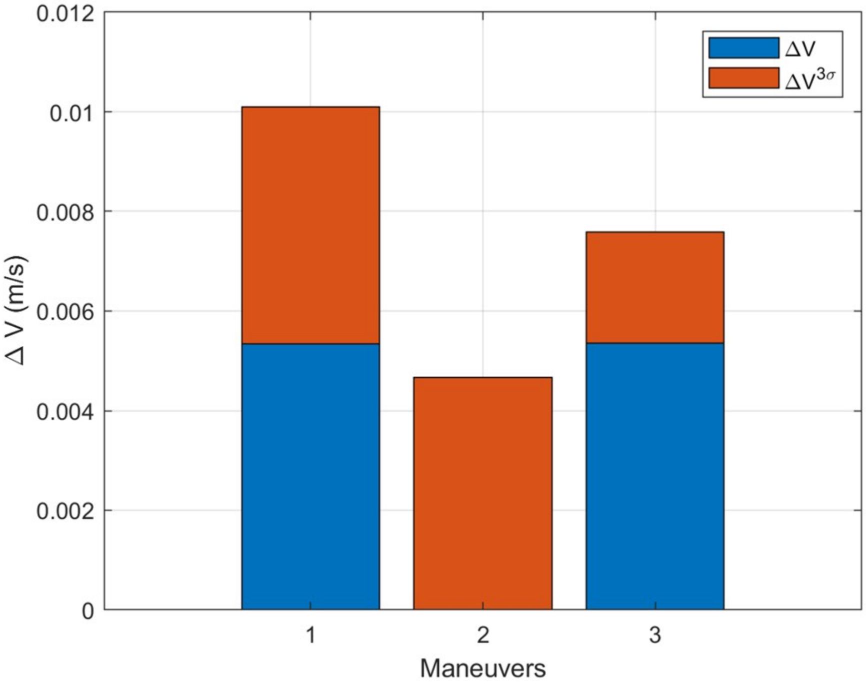

Fig. 9.21: ΔV and 3σ ΔV dispersions for the simply constrained V-bar hop scenario.

9.4.2 Safety Constrained Results

The optimal, safety constrained trajectory and maneuvers generated by STORM are shown in Figure 9.22. Overlaid on the trajectory are the regions where the angle measurements are available and lighting constraints are not violated. Figure 9.23 plots the tree analysis for the transfer. Figure 9.24 shows the range and 4σ minimum range for the nominal trajectory and branches. To improve safety and meet the 4σ constraint, the first maneuver (creating branch 2) places the spacecraft onto a small offset NMC trajectory. This helps get the vehicle get near the final position without inducing any significant drift towards the RSO. The second maneuver achieves the final position but creates a trajectory that drifts away from the RSO as seen in branch 3. These maneuvers indicate a significant departure from the minimum ΔV solutions typically used to perform a V-bar hop sequence and demonstrate that such transfers can be achieved with adequate safety margins.

The ΔV results are shown in Figure 9.25. Unlike the simply constrained solution, the second maneuver is the largest and there are no mid-course corrections applied. Achieving the safety constraint increases the fuel consumption by small amount.

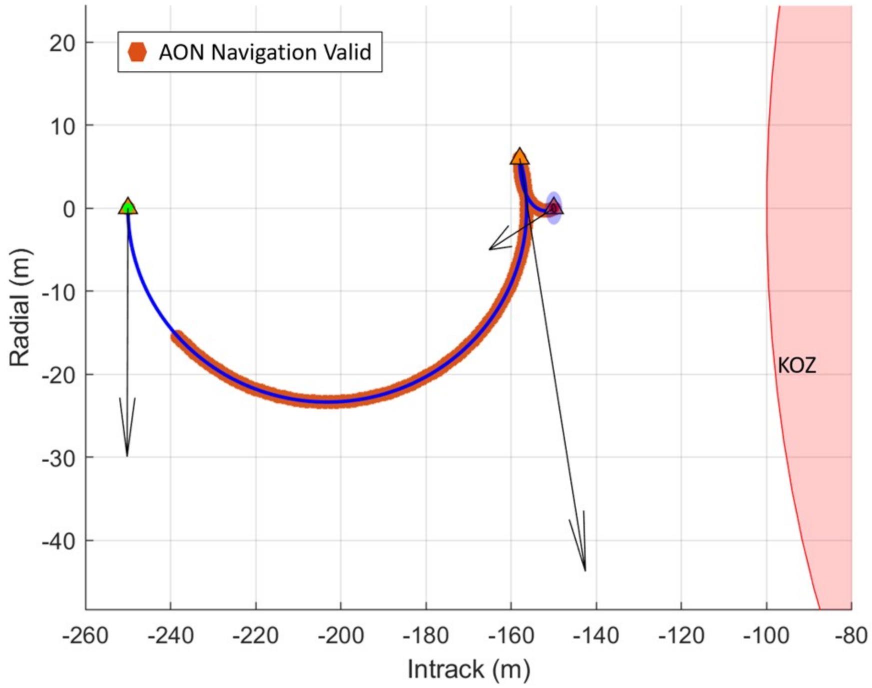

Fig. 9.22: V-bar hop scenario trajectory for the safety constrained optimal solution.

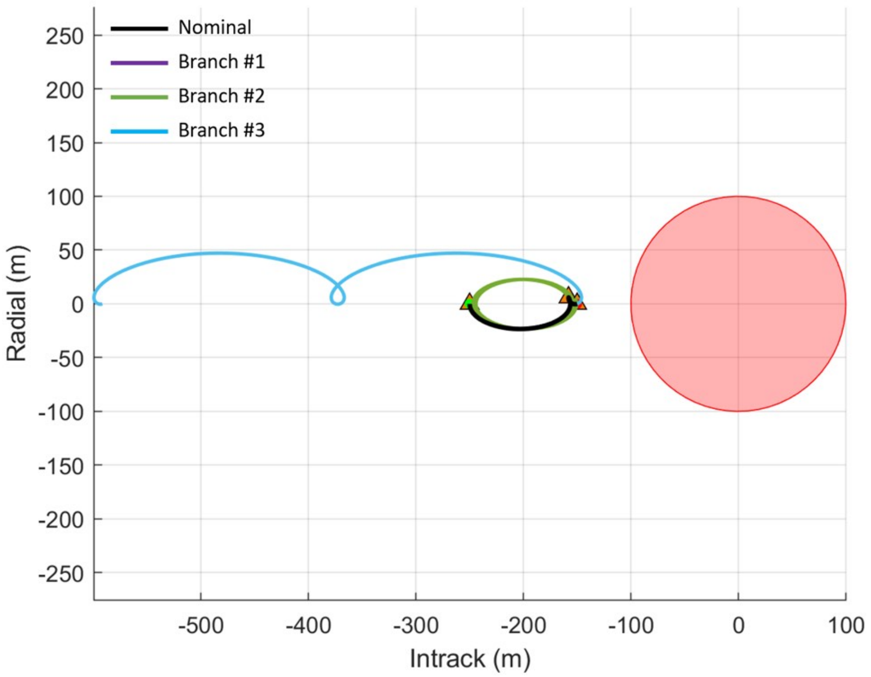

Fig. 9.23: Tree analysis of the V-bar hop scenario with safety constraint.

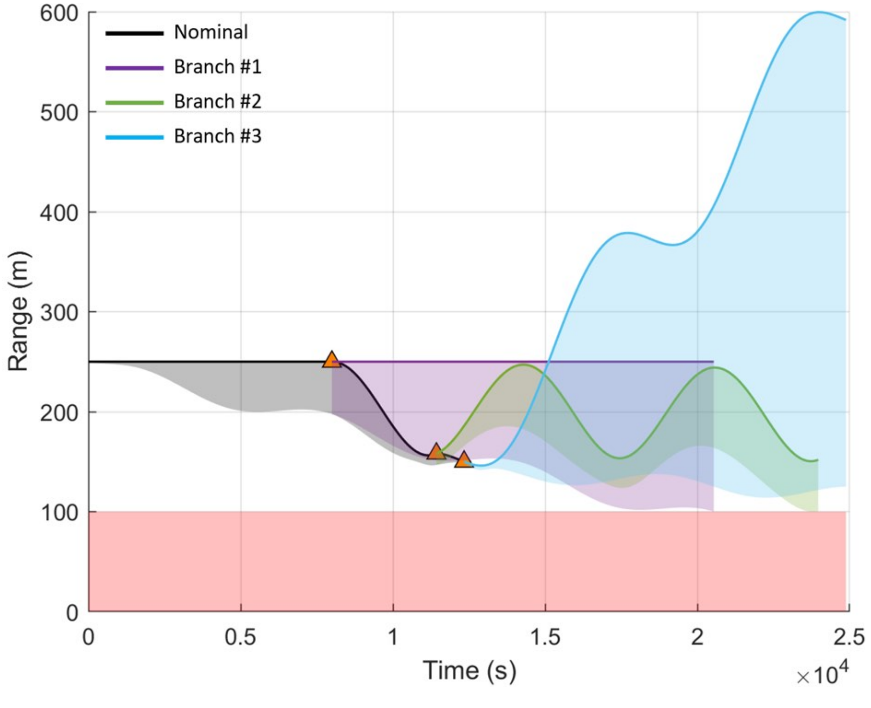

Fig. 9.24: Range and 4σ minimum range projection for the V-bar hop scenario with safety constraint.

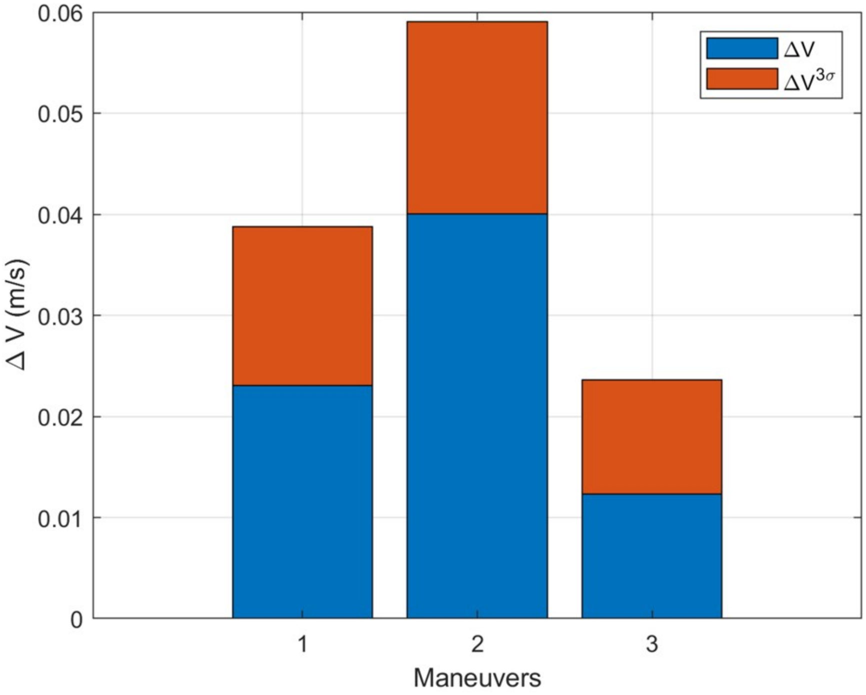

Fig. 9.25: ΔV and 3σ ΔV dispersion values for the V-bar hop scenario with safety constraint.

9.5 Conclusions

In this chapter the STORM analysis tool was updated to include a safety constraint based on the bounded probability method developed in Chapter 8. The safety constraint is applied to both the nominal and branch trajectories as part of a tree analysis. The impact of the safety constraint for three scenarios was evaluated by comparing the results for the simply and safety constrained trajectories. Key comparisons included the relative trajectory, the minimum range uncertainty, and ΔV consumption. Detailed conclusions from this chapter are provided in Chapter 10.

CHAPTER 10
CONCLUSIONS

Execution of on-orbit rendezvous and proximity operations is a difficult and complicated endeavor. As RPO vehicles and the missions they execute become more complicated, mission operators have required more advanced tools to plan and execute those missions. Space is a congested, contested, and competitive domain [128], and it is the responsibility of leadership and operators to conduct RPO operations so as to minimize the risk of collisions and the potential for creating space debris. RPO missions planners require the ability to plan complex maneuvers, achieve exceptional precision, and minimize fuel consumption while maintaining complete safety.

It has been the objective of this research to assist this effort through the development of advanced, ground-based mission planning techniques that can reduce the risk of collision while accounting for many of the limitations and constrained that impact RPO missions. To this end, Chapters 1 and 2 introduced the key concepts of relative motion dynamics, maneuvers, and spacecraft safety. Research relevant to this objective and used to inform recent RPO missions was also identified. Spacecraft safety was identified as a key requirement for RPO missions where accidental collisions can negatively impact the entire space environment and individual flight operations. Since flight operations are inherently stochastic in nature, Chapters 3 and 4 provided the basis for a statistical understanding of RPO operations including trajectory dispersions and ΔV dispersions. A tailor-made version of the LinCov tool was developed for RPO missions that provides options for either lidar or optical-based navigation and accounts for maneuver execution error.

With these foundational concepts in place we were able to pose the central thesis of this research. Is it possible to create a mission planning tool that can assure safe operations?

10.1 Constrained Path Planning Thesis

Closed-loop linear covariance techniques can be coupled with stochastic optimization techniques to develop optimal relative spacecraft trajectories that meet safety constraints while simultaneously meeting performance requirements in the presence of sensor noise, actuator errors, and dynamics modeling errors.

To validate this thesis, an initial study implementing two-impulse transfers was performed in Chapter 5 to visualize and understand the solutions space. From these results it was observed that, when accounting for stochastic processes, constrained optimal solutions typically exist near the deterministic optimal solution and that there generally exists a broad range of near-optimal trajectories adjacent to the optimal deterministic trajectory. From these observations we see that finding constrained optimal solutions can be a matter of making small tweaks to deterministic solutions. Constrained optimal solutions should generally look and feel like the solutions an experience trajectory planner would expect to find when generating deterministic solutions. These intuitions, however, do not always hold and strongly depend on the mission constraints.

In order to significantly reduce the true final dispersion, maneuvers must occur later in the transfer timeline and shortly before reaching the desired final state. This allows for the maneuver to accurately target the desired final state without giving uncertainty in the dynamics or maneuver execution errors time to dominate the solution. Achieving an overly restrictive true final dispersion requirement, however, may also drive the solution away from the optimal region and result in a significant increase in the total ΔV dispersion. Additionally, transfers occurring near transitions or singularities (e.g. half-period transfers) can result in both large position and ΔV dispersions and should be avoided. Detailed conclusions from this two-impulse analysis are provided in Section 5.3.

To build on these observations, a novel trajectory planning tool (STORM) was developed in Chapter 6 that uses a genetic algorithm followed by the non-linear *fmincon* solver to optimize relative trajectory transfers. The genetic algorithm is used to find feasible solutions near the global optimum and *fmincon* to refine the solution. This approach was selected as a direct result of observation from the previous two-impulse analysis. The hybrid approach helps to reduce the overall run time and to improve the optimal solution when compared to running just a genetic algorithm.

It will also be noted here that the optimization fitness function seeks to minimize the 3σ ΔV dispersion, $\Delta V^{3\sigma}$, rather than just the nominal ΔV. This is an important distinction when considering the stochastic nature of RPO spacecraft. It's easy for mission planners and operators to believe that the system always performs nominally, operating exactly as commanded. The cautionary tale of the DART collision is a reminder that sensor noise and maneuver execution errors must be considered when planning precision activities. Optimizing on $\Delta V^{3\sigma}$ helps to bound the potential trajectory dispersions by accounting for and planning around system limitations. Under this approach, individual maneuvers may seem large (when considering the stochastic nature of on-orbit operations) but total overall fuel consumption will be reduced when viewed in aggregate.

The STORM analysis tool was then used solve the previously analyzed two-impulse scenarios, demonstrating its ability to successfully locate the global minimum transfer. When properly configured, the genetic algorithm efficiently searches the entire solution space with a large fraction of its final population typically located in the regions containing the both local and global minimum. STORM was also able to differentiate between the two in successfully determining a final optimal solution.

While this analysis would seem to partially prove the thesis, there were still two major questions to be addressed related to operationally relevant scenarios (i.e. complex multi-impulse transfers) and vehicle safety. In Chapter 7, STORM was used to solve variations of the three-impulse transfer in order to demonstrate the ability to generate complex trajectories where the location of intermediate waypoints must also be determined. While solving the NMC resizing scenario, STORM showed how a transfer orbit can be adjusted to compensate for variations in lighting condition and still meet final position dispersion constraints. This result showed that it is possible to generate feasible results for increasingly complex mission constraints. Obtaining these constrained results often come at the expense increase fuel usage. It is important to note, however, that no feasible results would be possible from a deterministic solution since no inputs exist for constraints like lighting.

The co-elliptic to NMC transfer scenario was solved for a simply constrained solution, a tightly constrained solution, and a free final time solution. As observed previously, it is possible to achieve a smaller final position dispersion by altering the trajectory and maneuver timeline. Achieving the smaller dispersion, however, comes at the cost of increased fuel consumption. Allowing the final time to vary, however, can achieve the same results with no impact to fuel consumption. This is another key observation that indicates mission planners can achieve tight constraints by allowing for flexibility in time (or equivalently lighting angles) when feasible. Though not explored in this work, it would be possible to account for this flexibility in the optimization fitness function by bounding the final time to a permissible window.

Results for the NMC resizing scenario also show that this approach can successfully implement multi-impulse transfers and is not limited to just two or three-impulse transfers. Additional maneuvers do not add significantly to the total fuel consumption but they can help to reduce the final position dispersion. This is achieved in part through mid-course correction maneuvers that can be placed near the end of a transfer to reduce dispersions. This result is consistent with the simplified two-impulse observations. Too many maneuvers, however, will increase fuel consumption. Transfers should be limited between 3 and 6 total maneuvers with the upper end only pursued when attempting to achieve tight final position dispersion requirements.

To address spacecraft safety, a number of approaches were evaluated for their accuracy, speed, and ability to account for operational constraints (such as an arbitrary keep-out zone). Approaches evaluated in Chapter 8 included 1) probability of collision, 2) range uncertainty, and 3) Mahalanobis distance. While each approach has it advantages, none were found to be sufficient for efficiently and accurately assessing risk. A novel approach, called bounded probability, was developed that allows planners to bound the probability of collision against any arbitrary keep-out zone. This approach also allows mission planners and operators to discuss safety in terms of standard deviations, terminology related to the common probability of collision metric used for assessing non-RPO risks. It is recommended that RPO safety be analyzed against at least a 4σ bounded probability ($P_c < 1.54 \times 10^{-5}$).

Finally, optimization results for all three scenarios were evaluated in Chapter 9 for safety considerations when including or not including the new safety metric. In all three scenarios, STORM was successful in identifying transfer orbits that met the final position dispersion requirement, maintained 4σ safety, and minimized $\Delta V^{3\sigma}$.

In evaluating the fully constrained results, a number of observations can be made with respect to the thesis. The first is that the size of a keep-out zone (or the proximity in which the spacecraft operates to a keep-out zone) should be carefully selected. For the NMC resizing scenario, results show that a minimal increases in the keep-out zone size (from 150 m up to 200 m) can drive up ΔV by 650%. At 150 m, the safety constrained trajectory is nearly identical to the global, deterministic minimum for the transfer. A second observation is that Crosstrack motion can be an effective mitigation to the risk of collision. This observation is not novel and is commonly addressed through the concept of passive safety. However, this new approach provides a framework for determining the requisite amount of Crosstrack motion needed to maintain safety given the fuel expense of adding and removing Crosstrack motion. Finally, results demonstrate that it is possible to operate safely on the V-bar. To ensure this safety, planners should verify that drift towards the RSO should be minimized or eliminated. Proper selection of waypoints and maneuver timelines in a V-bar hop scenario will yield a different approach strategy than typically considered by planners but can create transfers that maintain passive safety even when accounting for the potential of spacecraft anomalies.

Though not directly related to RPO, it was also observed that a linear covariance model can also be used to determine the optimal location for mid-course correction maneuvers (as seen in the simply constrained V-bar hop scenario). Ongoing research is looking at this approach to determine the optimal location for mid-course corrections on trans-lunar injection orbits where minimizing final position dispersion is critical to achieving an accurate final orbit.

Finally, a new concept of RPO safety was also developed and integrated into STORM; the tree analysis. It is critical to RPO safety assessments to consider the possibility that the spacecraft may encounter an issue or anomaly and fail to execute future maneuvers. While this concept is not inherently unique, results from this research demonstrated the importance of including trajectory dispersion in that analysis. In many cases, it is the minimum distance between the position dispersion and keep-out zone of a branch, not the nominal trajectory, that spikes the risk of collision (or violates the keep-out zone) and must be addressed to adequately address safety.

Results from this research have successfully proven the thesis that a closed-loop linear-covariance model can be used to determine optimal and safe RPO trajectories. This approach provides a significant tool for mission planners and operators to efficiently and safety perform complex rendezvous and proximity operations missions.

10.2 Future Work

As research into this field progressed, the author observed a number of limitations that could be addressed as well as opportunities to be explored. The key limitation in the STORM analysis tool is the dependence on a genetic algorithm to generate the initial estimate of an optimal trajectory. Genetic algorithms can be highly effective but, in this case, frequently showed itself to be inefficient it its search. Solution runs may require upwards of 30 minutes to converge to the desired solution, often with little to no improvements between generations. Typical runs often require 10-15 minutes for convergence, with some requiring as few as 5 minutes. Several variations of the hybrid optimization scheme were tested but all took equally long amounts of time to converge. Other global optimization algorithms, such as grid search and particle swarms, were also evaluated within STORM. While these methods often generated quicker results, they did not always converge to known optimal solutions when analyzing simple two-impulse scenarios. Research into improved global optimization techniques can greatly improve the optimization process by reducing run time and improving convergence to the global optimum.

The STORM framework allows for a multitude of variations that can be included in the optimization space. As mentioned above, work could be performed to expand the free final time analysis to allow planners to select desired lighting windows at the final location, opening the possibility to further reduce fuel consumption or to minimize dispersions.

All analyses were performed with either the lidar or optical-based navigation methods. Many RPO systems have multiple navigation packages and are capable of performing both methods. Additional analysis should be performed to determine the optimal method or locations for transferring between sensor packages. More accurate navigation packages may also be used to assist less accurate packages when operating near the range boundary between the two.

The ability to perform non-standard operations (e.g. forced-motion circumnavigations, corridor approach vectors, or station-keeping) can also be generated withing the STORM framework and would provide a valuable mission planning capability. Similarly, the bounded probability safety metric can also be used to keep the vehicle out of alternative keep-out zones such as RSO sensor exclusion zones.

Finally, the linear covariance model generated for this analysis was intentionally generic in order to be broadly applicable. When implemented for specific RPO spacecraft, care should be taken to match model parameters to the vehicle's expected performance. It is the desire of the author that these concepts and techniques become widely understood and applied, bringing greater capability and safety to future RPO missions.

REFERENCES

[1] "XSS-10 Micro Satellite," `https://www.aerospace-technology.com/projects/xss-10microsatellite/`, Accessed: 2020-07-12.

[2] AFRL Fact Sheet, "XSS-11 Micro Satellite," `https://www.kirtland.af.mil/Portals/52/documents/AFD-111103-035.pdf`, September 2011, Accessed: 2022-01-18.

[3] AFRL Fact Sheet, "Automated Navigation and Guidance Experiment for Local Space (ANGELS)," `https://www.kirtland.af.mil/Portals/52/documents/AFD-131204-039.pdf`, July 2014, Accessed: 2022-01-18.

[4] "Mycroft – Satellite Space Experiment," `https://afresearchlab.com/technology/space-vehicles/mycroft/`, Accessed: 2020-07-06.

[5] AFRL Fact Sheet, "ESPA Augmented Geosynchronous Laboratory Experiment (EAGLE)," `https://www.kirtland.af.mil/Portals/52/documents/EAGLE-factsheet.pdf`, April 2018, Accessed: 2022-01-18.

[6] NASA Facts, "DART Demonstrator To Test Future Autonomous Rendezvous Technologies in Orbit," `https://www.nasa.gov/centers/marshall/pdf/100402main_dart.pdf`, September 2004, Accessed: 2022-01-18.

[7] DARPA Fact Sheet, "Orbital Express," `https://upload.wikimedia.org/wikipedia/commons/7/78/Orbita_Express_fact_sheet.pdf`, March 2007, Accessed: 2022-01-18.

[8] Persson, S., D'Amico, S., and Harr, J., "Flight results from prisma formation flying and rendezvous demonstration mission," *IAC-10-D9.2.8 , 61st International Astronautical Congress*, 2010.

[9] AFSPC Fact Sheet, "Geosynchronous Space Situational Awareness Program (GSSAP)," `https://www.afspc.af.mil/About-Us/Fact-Sheets/Article/730802/geosynchronous-space-situational-awareness-program-gssap/`, March 2017, Accessed: 2022-01-18.

[10] Clohessy, W. H. and Wiltshire, R., "Terminal Guidance System for Satellite Rendezvous," *Journal of the Aero/Space Sciences*, Vol. 27, No. 3, 1960, pp. 653–658.

[11] Woffinden, D. C. and Geller, K. D., "Relative Angles-Only Navigation and Pose Estimation for Autonomous Orbital Rendezvous," *Journal of Guidance, Control, and Dynamics*, Vol. 30, No. 5, 2007, pp. 1455–1469.

[12] Allen, A. M., Langley, C., Mukherji, R., Taylor, A. B., Umasuthan, M., and D., B. T., "Rendezvous Lidar Sensor System for Terminal Rendezvous, Capture, and Berthing to the International Space Station," *Proceedings of SPIE, Vol. 6958, SPIE Conference on Sensors and Systems for Space Applications II*, SPIE, 2008.

[13] D'Amico, S., Benn, M., and Jørgensen, J. L., "Pose estimation of an uncooperative spacecraft from actual space imagery," *International Journal of Space Science and Engineering 5*, Vol. 2, No. 2, 2014, pp. 171–189.

[14] Curtis, H. D., *Orbital Mechanics for Engineering Students*, Butterworth-Heinemann, Cambridge, MA, 4th ed., 2020.

[15] Bhaskaran, S., Riedel, J., Synnott, S., and Wang, T., "The Deep Space 1 Autonomous Navigation System: A Post-Flight Analysis," *AIAA/AAS Astrodynamics Specialist Conference, Navigation and Flight Mechanics Section, Jet Propulsion Laboratory, Denver*, 2000.

[16] DART Mishap Investigation Board, "Overview of the dart mishap investigation results," Tech. rep., NASA, 2006.

[17] Woffinden, D. and Geller, D., "Observability criteria for angles-only navigation," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 45, No. 3, 2009, pp. 1194–1208.

[18] Woffinden, D. C. and Geller, D. K., "Navigating the road to autonomous orbital rendezvous," *Journal of Spacecraft and Rockets*, Vol. 44, No. 4, 2007, pp. 898–909.

[19] Newton, I., "Naturalis philosophiae principia mathematica," *London (1687)*, 1726.

[20] Bate, R. R., Mueller, D. D., and White, J. E., *Fundamentals of Astrodynamics*, Dover Publications, 2020.

[21] Montenbruck, O. and Gill, E., *Satellite Orbits: Models, Methods, and Applications*, Springer Science & Business Media, 2000.

[22] Bond, V. R. and Allman, M. C., *Modern Astrodynamics: Fundamentals and Perturbation Methods*, Vol. 51, Princeton University Press, 1996.

[23] Prussing, J. E. and Conway, B. A., *Orbital Mechanics*, Oxford University Press, USA, 1993.

[24] Lovell, T. A. and Spencer, D. A., "Relative orbital elements formulation based upon the Clohessy-Wiltshire equations," *The Journal of the Astronautical Sciences*, Vol. 61, No. 4, 2014, pp. 341–366.

[25] Hill, G. W., "Researches in the Lunar Theory," *American Journal of Mathematics*, Vol. 1, No. 1, 1878, pp. 5–26.

[26] Sutton, G. P. and Biblarz, O., *Rocket Propulsion Elements*, John Wiley & Sons, 2016.

[27] Parten, R. P. and Mayer, J. P., "Development of the Gemini Operational Rendezvous Plan," *Journal of Spacecraft and Rockets*, Vol. 5, No. 9, 1968, pp. 1023–1028.

[28] Young, K. A. and Alexander, J. D., "Apollo Lunar Rendezvous," *Journal of Spacecraft and Rockets*, Vol. 7, No. 9, 1970, pp. 1083–1086.

[29] Aldrin, B., *Line-of-sight guidance techniques for manned orbital rendezvous*, Ph.D. thesis, Massachusetts Institute of Technology, 1963.

[30] Pearson Don, J., "Shuttle Rendezvous and Proximity Operations," *Space dynamics, Proceedings of the International Symposium*, Toulouse, France, 1989, pp. 833–851.

[31] Goodman, J. L., "History of space shuttle rendezvous and proximity operations," *Journal of Spacecraft and Rockets*, Vol. 43, No. 5, 2006, pp. 944–959.

[32] Rumford, T. E., "Demonstration of Autonomous Rendezvous Technology (Dart) Project Summary," *Proceedings of SPIE, Vol. 5088, SPIE Space System Technology and Operations*, SPIE, Orlando, FL, 2003.

[33] Dennehy, C. J. and R., C. J., "A Summary of the Rendezvous, Proximity Operations, Docking, and Undocking (RPODU) Lessons Learned from the Defense Advanced Research Project Agency (DARPA) Orbital Express (OE) Demonstration System Mission," *NASA/TM-2011-217088*, 2011.

[34] Weismuller, T. and Leinz, M., "GN&C technology demonstrated by the orbital express autonomous rendezvous and capture sensor system," *29th annual AAS guidance and control conference*, American Astronautical Society, 2006, pp. 06–016.

[35] Cohen, R. S., "AFRL Dispatching Satellite to Examine Unresponsive Smallsat," `https://www.airforcemag.com/AFRL-Dispatching-Satellite-to-Examine-Unresponsive-Smallsat/`, 2019-10-18, Accessed 2022-01-18.

[36] McCall, G. H. and Darrah, J. H., "Space Situational Awareness: Difficult, Expensive-and Necessary," *Air & Space Power Journal*, Vol. 28, No. 6, 2014, pp. 6.

[37] Pinard, D., Reynaud, S., Delpy, P., and Strandmoe, S. E., "Accurate and autonomous navigation for the ATV," *Aerospace Science and Technology*, 2007, pp. 490–498.

[38] Larsson, R., Noteborn, R., Chasset, C., Karlsson, T., Carlsson, A., Persson, S., and Bodin, P., "Flight Results from SSC's GNC Experiments within the PRISMA Formation Flying Mission," *61st International Astronautical Congress*, Vol. 7, 2010, pp. 6032–6041.

[39] Yamanaka, K. and Ankersen, F., "New state transition matrix for relative motion on an arbitrary elliptical orbit," *Journal of Guidance, Control, and Dynamics*, Vol. 25, No. 1, 2002, pp. 60–66.

[40] Schweighart, S. A. and Sedwick, R. J., "High-fidelity linearized J2 model for satellite formation flight," *Journal of Guidance, Control, and Dynamics*, Vol. 25, No. 6, 2002, pp. 1073–1080.

[41] Gaias, G., Ardaens, J.-S., and Montenbruck, O., "Model of J_2 perturbed satellite relative motion with time-varying differential drag," *Celestial Mechanics and Dynamical Astronomy*, Vol. 123, No. 4, 2015, pp. 411–433.

[42] Sullivan, J., Grimberg, S., and D'Amico, S., "Comprehensive survey and assessment of spacecraft relative motion dynamics models," *Journal of Guidance, Control, and Dynamics*, Vol. 40, No. 8, 2017, pp. 1837–1859.

[43] DiGirolamo, L. J., Hacker, K. A., Hoskins, A. H., and Spencer, D. B., "A hybrid motion planning algorithm for safe and efficient, close proximity, autonomous spacecraft missions," *AIAA/AAS Astrodynamics Specialist Conference*, 2014, p. 4130.

[44] Munoz, J., Boyarko, G., and Fitz-Coy, N., "Rapid path-planning options for autonomous proximity operations of spacecraft," *AIAA/AAS Astrodynamics Specialist Conference*, 2010, p. 7667.

[45] Bennett, T., Schaub, H., and Roscoe, C. W., "Faster-than-natural spacecraft circumnavigation via way points," *Acta Astronautica*, Vol. 123, 2016, pp. 376–386.

[46] Lu, P. and Liu, X., "Autonomous trajectory planning for rendezvous and proximity operations by conic optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 36, No. 2, 2013, pp. 375–389.

[47] Ortolano, N. G., *Autonomous Trajectory Planning for Satellite RPO and Safety of Flight Using Convex Optimization*, Ph.D. thesis, Utah State University, 2018.

[48] Weiss, A., Kolmanovsky, I., Baldwin, M., and Erwin, R. S., "Model predictive control of three dimensional spacecraft relative motion," *2012 American Control Conference (ACC)*, IEEE, 2012, pp. 173–178.

[49] Weiss, A., Baldwin, M., Erwin, R. S., and Kolmanovsky, I., "Model Predictive Control for Spacecraft Rendezvous and Docking: Strategies for Handling Constraints and Case Studies," *IEEE Transactions on Control Systems Technology*, Vol. 23, No. 4, 2015, pp. 1638–1647.

[50] Pontani, M. and Conway, B. A., "Optimal finite-thrust rendezvous trajectories found via particle swarm algorithm," *Journal of Spacecraft and Rockets*, Vol. 50, No. 6, 2013, pp. 1222–1234.

[51] Pontani, M., Ghosh, P., and Conway, B. A., "Particle swarm optimization of multiple-burn rendezvous trajectories," *Journal of Guidance, Control, and Dynamics*, Vol. 35, No. 4, 2012, pp. 1192–1207.

[52] Li, S., Mehra, R., Smith, R., and Beard, R., "Multi-spacecraft trajectory optimization and control using genetic algorithm techniques," *2000 IEEE Aerospace Conference. Proceedings (Cat. No. 00TH8484)*, Vol. 7, IEEE, 2000, pp. 99–108.

[53] Patera, R. P., "General method for calculating satellite collision probability," *Journal of Guidance, Control, and Dynamics*, Vol. 24, No. 4, 2001, pp. 716–722.

[54] Chan, K., "Analytical expressions for computing spacecraft collision probabilities," *Proceedings of the 11 th Annual AAS/AIAA Space Flight Mechanics Meeting, Santa Barbara, CA*, 2001, pp. 305–320.

[55] Chan, K., “Spacecraft collision probability for long-term encounters,” *Advances in the Astronautical Sciences*, Vol. 116, No. 1, 2003, pp. 767–784.

[56] Chan, K., “Short-term vs. long-term spacecraft encounters,” *AIAA/AAS Astrodynamics Specialist Conference*, 2004, p. 5460.

[57] Chan, F. K., *Spacecraft Collision Probability*, Aerospace Press El Segundo, CA, 2008.

[58] Patera, R. P., “Satellite collision probability for nonlinear relative motion,” *Journal of Guidance, Control, and Dynamics*, Vol. 26, No. 5, 2003, pp. 728–733.

[59] Coppola, V. T., Woodburn, J., and Hujsak, R., “Effects of cross correlated covariance on spacecraft collision probability,” *AAS/AIAA Spaceflight Mechanics Meeting*, 2004, pp. 04–181.

[60] Alfano, S., “Addressing nonlinear relative motion for spacecraft collision probability,” *AIAA/AAS Astrodynamics Specialist Conference*, 2006, p. 6760.

[61] Carpenter, J. R., “Non-parametric collision probability for low-velocity encounters,” *Advances in the Astronautical Sciences*, Vol. 127, No. 1, 2007, pp. 227–242.

[62] Slater, G., Byram, S., and Williams, T., “Collision avoidance for satellites in formation flight,” *Journal of Guidance, Control, and Dynamics*, Vol. 29, No. 5, 2006, pp. 1140–1146.

[63] Gaylor, D. E. and Barbee, B. W., “Algorithms for safe spacecraft proximity operations,” *AAS/AIAA Spaceflight Mechanics Meeting*, 2007.

[64] Sun, Z., Luo, Y., and Niu, Z., “Spacecraft rendezvous trajectory safety quantitative performance index eliminating probability dilution,” *Science China Technological Sciences*, Vol. 57, No. 6, 2014, pp. 1219–1228.

[65] Frey, G. R., Petersen, C. D., Leve, F. A., Kolmanovsky, I. V., and Girard, A. R., “Constrained spacecraft relative motion planning exploiting periodic natural motion trajectories and invariance,” *Journal of Guidance, Control, and Dynamics*, Vol. 40, No. 12, 2017, pp. 3100–3115.

[66] Phillips, M. R., *Spacecraft collision probability estimation for rendezvous and proximity operations*, Master’s thesis, Utah State University, 2012.

[67] Breger, L. S., *Control of spacecraft in proximity orbits*, Ph.D. thesis, Massachusetts Institute of Technology, 2007.

[68] Prince, E. R., *Optimal Finite Thrust Guidance Methods for Constrained Satellite Proximity Operations Inspection Maneuvers*, Ph.D. thesis, Air Force Institute of Technology, 2018.

[69] Breger, L. and How, J. P., “Safe trajectories for autonomous rendezvous of spacecraft,” *Journal of Guidance, Control, and Dynamics*, Vol. 31, No. 5, 2008, pp. 1478–1489.

[70] Breger, L. and How, J. P., “Powered safe abort for autonomous rendezvous of spacecraft,” *AIAA Guidance, Navigation and Control Conference*, 2007.

[71] Prince, E. R., Carr, R. W., and Cobb, R. G., "Computationally Efficient Methods for Fuel Optimal Proximity Maneuvers with Constraints," *AAS/AIAA Astrodynamics Specialist Conference*, Stevenson, WA, 2018, p. 0867.

[72] Richards, A., Schouwenaars, T., How, J. P., and Feron, E., "Spacecraft trajectory planning with avoidance constraints using mixed-integer linear programming," *Journal of Guidance, Control, and Dynamics*, Vol. 25, No. 4, 2002, pp. 755–764.

[73] Roger, A. B. and McInnes, C. R., "Safety constrained free-flyer path planning at the international space station," *Journal of Guidance, Control, and Dynamics*, Vol. 23, No. 6, 2000, pp. 971–979.

[74] Schlanbusch, R., Kristiansen, R., and Nicklasson, P. J., "Spacecraft formation reconfiguration with collision avoidance," *Automatica*, Vol. 47, No. 7, 2011, pp. 1443–1449.

[75] Sauter, L. and Palmer, P., "Onboard semianalytic approach to collision-free formation reconfiguration," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 48, No. 3, 2012, pp. 2638–2652.

[76] Holzinger, M., DiMatteo, J., Schwartz, J., and Milam, M., "Passively safe receding horizon control for satellite proximity operations," *2008 47th IEEE Conference on Decision and Control*, IEEE, 2008, pp. 3433–3440.

[77] Irvin, D. and Cobb, R., "Feasibility study and fuel optimal control techniques for constrained relative satellite orbit geometries," *AIAA/AAS Astrodynamics Specialist Conference*, 2006, p. 6758.

[78] Irvin Jr, D. J., Cobb, R. G., and Lovell, T. A., "Fuel-optimal maneuvers for constrained relative satellite orbits," *Journal of Guidance, Control, and Dynamics*, Vol. 32, No. 3, 2009, pp. 960–973.

[79] Dannemiller, D. P., "Multi-Maneuver Clohessy-Wiltshire Targeting," *AAS/AIAA Astrodynamics Specialist Conference*, Girdwood, Alaska, 2011.

[80] Franquiz, F. J., Muñoz, J. D., Udrea, B., and Balas, M. J., "Optimal range observability maneuvers of a spacecraft formation using angles-only navigation," *Acta Astronautica*, Vol. 153, 2018, pp. 337–348.

[81] Huang, H., Ma, G., Zhuang, Y., and Lv, Y., "Optimal spacecraft formation reconfiguration with collision avoidance using particle swarm optimization," *Information Technology and Control*, Vol. 41, No. 2, 2012, pp. 143–150.

[82] Chari, R., Geller, D., Norris, H., D'Souza, C., and Brand, T., "Autonomous Orbital Rendezvous Using Angles-Only Navigation," *AAS/AIAA Astrodynamics Specialists Conference*, 2001.

[83] Schmidt, J., Geller, D., and Chavez, F. R., "Viability of Angles-only Navigation for Orbital Rendezvous Operation," *AIAA-2010-7755, AIAA Guidance, Navigation, and Control Conference, Toronto, Ontario*, AIAA, Toronto, Ontario, 2010.

[84] Tombasco, J. and Axelrad, P., "Observability of Relative Hybrid Elements, Given Space-Based Angles-Only Observations," *Journal of Guidance, Control, and Dynamics*, Vol. 35, No. 5, 2012, pp. 1681–1686.

[85] Tweddle, B. E. and Saenz-Otero, A., "Relative computer vision-based navigation for small inspection spacecraft," *Journal of Guidance, Control, and Dynamics*, Vol. 38, No. 5, 2015, pp. 969–978.

[86] Howard, R. T. and Bryan, T. C., "Video Guidance Sensor for Automated Capture," *Space Programs and Technologies Conference*, AIAA, Huntsville, AL, 1992, p. 1389.

[87] Junkins, J. L., Hughes, D. C., Wazni, K. P., , and Pariyapong, V., "Vision-Based Navigation for Rendezvous and Docking and Proximity Operations," *22nd Annual AAS Guidance and Control Conference, Breckenridge, CO*, AAS, Breckenridge, CO, 1999, p. 021.

[88] Petit, A., Marchand, E., and Kanan, i., "Vision-based Space Autonomous Rendezvous: A Case Study," *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, IEEE, San Francisco, USA, 2011, pp. 619–624.

[89] Ruel, S., English, C., Anctil, M., and Church, P., "3DLASSO: Real-time pose estimation from 3D data for autonomous satellite servicing," *Proc. ISAIRAS 2005 Conference, Munich, Germany*, 2005.

[90] Liu, L., Zhao, G., and Bo, Y., "Point cloud based relative pose estimation of a satellite in close range," *Sensors*, Vol. 16, No. 6, 2016, pp. 824.

[91] Aghili, F., Kuryllo, M., Okouneva, G., and English, C., "Robust vision-based pose estimation of moving objects for automated rendezvous & docking," *2010 IEEE International Conference on Mechatronics and Automation*, IEEE, 2010, pp. 305–311.

[92] Montenbruck, O. and D'Amico, S., "GPS based relative navigation," *Distributed Space Missions for Earth System Monitoring*, Springer, 2013, pp. 185–223.

[93] Fehse, W., *Automated Rendezvous and Docking of Spacecraft*, Cambridge University Press, New York, 2003.

[94] Maybeck, P. S., *Stochastic Models, Estimation and Control, Volume 1*, Academic Press, Inc., 1979.

[95] Stengel, R. F., *Optimal Control and Estimation*, Courier Corporation, 1994.

[96] Zarchan, P. and Musoff, H., *Fundamentals of Kalman Filtering: A Practical Approach*, The American Institute of Aeronautics and Astronautics, Inc, 2nd ed, 2005.

[97] Crassidis, J. L. and Junkins, J. L., *Optimal Estimation of Dynamic Systems*, CRC press, 2011.

[98] Simon, D., *Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches*, John Wiley & Sons, 2006.

[99] Ristic, B., Arulampalam, S., and Gordon, N., *Beyond the Kalman Filter: Particle Filters for Tracking Applications*, Artech House, 2003.

[100] Schutz, B., Tapley, B., and Born, G. H., *Statistical Orbit Determination*, Elsevier, 2004.

[101] Geller, D. K., "Linear covariance techniques for orbital rendezvous analysis and autonomous onboard mission planning," *Journal of Guidance, Control, and Dynamics*, Vol. 29, No. 6, 2006, pp. 1404–1414.

[102] Geller, D. K. and Christensen, D. P., "Linear covariance analysis for powered lunar descent and landing," *Journal of Spacecraft and Rockets*, Vol. 46, No. 6, 2009, pp. 1231–1248.

[103] Rose, M. B. and Geller, D., "Linear covariance techniques for powered ascent," *AIAA Guidance, Navigation, and Control Conference*, 2010, p. 8175.

[104] Jin, K., Geller, D., and Luo, J., "Development and Validation of Linear Covariance Analysis Tool for Atmospheric Entry," *Journal of Spacecraft and Rockets*, Vol. 56, No. 3, 2019, pp. 854–864.

[105] Christensen, R. S. and Geller, D., "Linear covariance techniques for closed-loop guidance navigation and control system design and analysis," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, Vol. 228, No. 1, 2014, pp. 44–65.

[106] Christensen, R. S., Droge, G., and Leishman, R. C., "Closed-Loop Linear Covariance Framework for Path Planning in Static Uncertain Obstacle Fields," *Journal of Guidance, Control, and Dynamics*, 2021, pp. 1–15.

[107] Jin, K., Geller, D. K., and Luo, J., "Robust Trajectory Design for Rendezvous and Proximity Operations with Uncertainties," *Journal of Guidance, Control, and Dynamics*, Vol. 43, No. 4, 2020, pp. 741–753.

[108] Stastny, N. B., Geller, D. K., and Shuster, S., "Optimal Relative Trajectory Design with Mission Constraints and Performance Requirements," *43rd Annual AAS Guidance, Navigation, and Control Conference*, AAS, Breckenridge, CO, 2020.

[109] Greenberg, M. D., *Foundations of Applied Mathematics*, Courier Corporation, 2013.

[110] Kreyszig, E., *Advanced Engineering Mathematics, 10th Edition*, Wiley, 2009.

[111] Rice, J. A., *Mathematical Statistics and Data Analysis*, Duxbury Press, 1995.

[112] MATLAB and Global Optimization Toolbox, *Version 9.9.0 and 4.4 (Release 2020b)*, The MathWorks Inc., Natick, Massachusetts, 2020.

[113] Spall, J. C., *Introduction to Stochastic Search and Optimization: Estimation, Simulation, and Control*, John Wiley & Sons, Hoboken, NJ, 2005.

[114] Goldberg, D. E., *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, 1989.

[115] MATLAB and Optimization Toolbox, *Version 9.9.0 and 9.0 (Release 2020b)*, The MathWorks Inc., Natick, Massachusetts, 2020.

[116] Prussing, J. E., "Optimal four-impulse fixed-time rendezvous in the vicinity of a circular orbit." *AIAA Journal*, Vol. 7, No. 5, 1969, pp. 928–935.

[117] Prussing, J. E., "Optimal two-and three-impulse fixed-time rendezvous in the vicinityof a circular orbit," *AIAA Journal*, Vol. 8, No. 7, 1970, pp. 1221–1228.

[118] Prussing, J. E. and Chiu, J.-H., "Optimal multiple-impulse time-fixed rendezvous between circular orbits," *Journal of Guidance, Control, and Dynamics*, Vol. 9, No. 1, 1986, pp. 17–22.

[119] Koons, D. S., Schreiber, C., Acevedo, F., and Sechrist, M., "Risk mitigation approach to commercial resupply to the international space station," *Making Safety Matter*, Vol. 680, 2010, pp. 12.

[120] Chapra, S. C. and Canale, R. P., *Numerical Methods for Engineers*, Boston: McGraw-Hill Higher Education, 2010.

[121] "IEEE Standard for Floating-Point Arithmetic," *IEEE Std 754-2008*, 2008, pp. 1–70.

[122] Krage, F. J., "NASA Spacecraft Conjunction Assessment and Collision Avoidance Best Practices Handbook," Tech. Rep. SP-20205011318, NASA, 2020.

[123] "Collision Avoidance for Space Environment Protection," Tech. Rep. NASA Interim Directive 7120.132, NASA, 2020.

[124] McLachlan, G. J., "Mahalanobis distance," *Resonance*, Vol. 4, No. 6, 1999, pp. 20–26.

[125] Bryson, A. E. and Ho, Y.-C., *Applied Optimal Control: Optimization, Estimation and Control*, CRC Press, 1975.

[126] Hart, J. C., *Graphic Gems IV*, chap. II.1, Distance to an Ellipsoid, Elsevier, 1994, pp. 113–119.

[127] Alfano, S. and Greer, M. L., "Determining if two solid ellipsoids intersect," *Journal of Guidance, Control, and Dynamics*, Vol. 26, No. 1, 2003, pp. 106–110.

[128] Harrison, R. G., "Unpacking the Three C's: Congested, Competitive, and Contested Space," *Astropolitics*, Vol. 11, No. 3, 2013, pp. 123–131.

APPENDICES

APPENDIX A

CLOHESSY-WILTSHIRE STATE TRANSITION MATRIX

The continuous form of the Clohessy-Wiltshire equations are derived in Section 2.1.2 and given as:

$$\ddot{x} - 3\omega^2 x - 2\omega\dot{y} = 0 \tag{A.1a}$$

$$\ddot{y} + 2\omega\dot{x} = 0 \tag{A.1b}$$

$$\ddot{z} + \omega^2 z = 0 \tag{A.1c}$$

The discrete form of these equations is defined as:

$$\boldsymbol{X}(t) = \boldsymbol{\Phi}(t)\boldsymbol{X}(t_0) \tag{A.2}$$

where the state vector, $\boldsymbol{X}(t)$, is comprised of the vehicle's position and velocity relative to the reference orbit, and $\boldsymbol{\Phi}(t)$ is the state transition matrix. The initial conditions at $t = 0$ are defined given as:

$$\boldsymbol{X}(t_0) = \begin{bmatrix} x_0 \\ y_0 \\ z_0 \\ \dot{x}_0 \\ \dot{y}_0 \\ \dot{z}_0 \end{bmatrix} \tag{A.3}$$

To determine the state transition matrix we begin with equation (A.1b):

$$\frac{d}{dt}\left(\dot{y} + 2\omega x\right) = 0 \tag{A.4}$$

such that:

$$\dot{y} + 2\omega x = \text{constant} \tag{A.5}$$

This constant is determined by evaluating equation (A.5) at t_0:

$$\dot{y} + 2\omega x = \dot{y}_0 + 2\omega x_0 \tag{A.6}$$

We can now rearrange variables to solve for $\dot{y}$:

$$\dot{y} = \dot{y}_0 + 2\omega\left(x_0 - x\right) \tag{A.7}$$

Substituting equation (A.7) in equation (A.1a) and separating the initial conditions yields:

$$\ddot{x} + \omega^2 x = 2\omega\dot{y}_0 + 4\omega^2 x_0 \tag{A.8}$$

This second-order non-homogeneous ordinary differential equation has a solution of the following form:

$$x = A\sin\omega t + B\cos\omega t + \frac{1}{\omega^2}\left(2\omega\dot{y}_0 + 4\omega x_0\right) \tag{A.9}$$

where A and B are the unknowns of the complementary solution and the remaining portion is the particular solution. Evaluating equation (A.9) at $t = 0$ yields:

$$x_0 = B + \frac{2}{\omega}\dot{y}_0 + 4x_0 \tag{A.10}$$

such that:

$$B = -3x_0 - \frac{2}{\omega}\dot{y}_0 \tag{A.11}$$

The first derivative of equation (A.9) is:

$$\dot{x} = \omega A\cos\omega t - \omega B\sin\omega t \tag{A.12}$$

When evaluated at $t = 0$:

$$\dot{x} = \omega A \tag{A.13}$$

such that:

$$A = \frac{1}{\omega}\dot{x}_0 \tag{A.14}$$

Substituting equations (A.11) and (A.14) back into equation (A.9) and combining terms yields:

$$x = (4 - 3\cos\omega t)\ \ x_0 + \frac{1}{\omega}(\sin\omega t)\ \ \dot{x}_0 + \frac{2}{\omega}(1 - \cos\omega t)\ \ \dot{y}_0 \tag{A.15}$$

Substituting equations (A.11) and (A.14) into equation (A.12) and combining terms yields:

$$\dot{x} = 3\omega\sin\omega t\ x_0 + \cos\omega t\ \dot{x}_0 + 2\sin\omega t\ \dot{y}_0 \tag{A.16}$$

Equation (A.15) can now be substituted back into equation (A.7):

$$\dot{y} = \dot{y}_0 + 2\omega\left(x_0 - (4 - 3\cos\omega t)\ \ x_0 - \frac{1}{\omega}(\sin\omega t)\ \ \dot{x}_0 - \frac{2}{\omega}(1 - \cos\omega t)\ \ \dot{y}_0\right) \tag{A.17}$$

and simplifies to:

$$\dot{y} = 6\omega(\cos\omega t - 1)\,x_0 - 2\sin\omega t\ \dot{x}_0 + (4\cos\omega t - 3)\,\dot{y}_0 \tag{A.18}$$

Equation (A.18) can be integrated to determine the solution for y:

$$y = 6\omega\left(\frac{1}{\omega}\sin\omega t - t\right)x_0 + \frac{2}{\omega}\cos\omega t\ \dot{x}_0 + \left(\frac{4}{\omega}\sin\omega t - 3t\right)\dot{y}_0 + C \tag{A.19}$$

where C is the yet unknown constant of integration. Evaluating at $t = 0$ the equation becomes:

$$y_0 = \frac{2}{\omega}\dot{x}_0 + C \tag{A.20}$$

such that:

$$C = y_0 - \frac{2}{\omega}\dot{x}_0 \tag{A.21}$$

Substituting (A.21) back into (A.19) yields the final solution for y:

$$y = 6\left(\sin \omega t - \omega t\right) x_0 + \frac{2}{\omega}\left(\cos \omega t - 1\right) \dot{x}_0 + y_0 + \left(\frac{4}{\omega} \sin \omega t - 3t\right) \dot{y}_0 \tag{A.22}$$

The solution for uncoupled crosstrack component in equation (A.1c) is:

$$z = D \cos \omega t + E \sin \omega t \tag{A.23}$$

with the first derivative:

$$\dot{z} = -\omega D \sin \omega t + \omega E \sin \omega t \tag{A.24}$$

Evaluating both of these equation at $t = 0$ yields:

$$D = z_0 \tag{A.25}$$

$$E = \frac{\dot{z}_0}{\omega} \tag{A.26}$$

Substituting these constants back into equations (A.23) and (A.24) yields our final equations for the crosstrack motion:

$$z = \cos \omega t \ z_0 + \frac{1}{\omega} \sin \omega t \ \dot{z}_0 \tag{A.27}$$

$$\dot{z} = -\omega \sin \omega t \ z_0 + \sin \omega t \ \dot{z}_0 \tag{A.28}$$

Equations (A.15), (A.16), (A.18), (A.22), (A.27), and (A.28) define the discrete for of the Clohessy-Wiltshire equations. These equations can be written in state-space form by defining the a state vector of the relative position and velocity:

$$\boldsymbol{X} = \begin{bmatrix} x & y & z & \dot{x} & \dot{y} & \dot{z} \end{bmatrix}^T \tag{A.29}$$

The state-space solution for $\boldsymbol{X}(t)$ is given by:

$$\boldsymbol{X}(t) = \boldsymbol{\Phi}(t) \boldsymbol{X}(t_0) \tag{A.30}$$

where the state transition matrix is:

$$\mathbf{\Phi}(t) = \begin{bmatrix} 4-3\cos\omega t & 0 & 0 & \frac{1}{\omega}(\sin\omega t) & \frac{2}{\omega}(1-\cos\omega t) & 0 \\ 6(\sin\omega t-\omega t) & 1 & 0 & \frac{2}{\omega}(\cos\omega t-1) & \frac{1}{\omega}(4\sin\omega t-3\omega t) & 0 \\ 0 & 0 & \cos\omega t & 0 & 0 & \frac{1}{\omega}\sin\omega t \\ 3\omega\sin\omega t & 0 & 0 & \cos\omega t & 2\sin\omega t & 0 \\ 6\omega(\cos\omega t-1) & 0 & 0 & -2\sin\omega t & 4\cos\omega t-3 & 0 \\ 0 & 0 & -\omega\sin\omega t & 0 & 0 & \cos\omega t \end{bmatrix} \tag{A.31}$$

and:

$$\boldsymbol{X}(t_0) = \begin{bmatrix} x_0 & y_0 & z_0 & \dot{x}_0 & \dot{y}_0 & \dot{z}_0 \end{bmatrix}^T \tag{A.32}$$

APPENDIX B

SUN VECTOR IN LVLH FRAME

The ability to predict the location and motion of the sun is an important part of rendezvous and proximity operations when performing relative navigation. This section develops a simplified sun model in the local-vertical, local-horizontal (LVLH) coordinate frame so that it can be easily integrated with the Clohessy-Wiltshire equations.

The inertial position of the sun relative to a spacecraft in orbit around the earth is given as:

$$r_{s/sc} = r_{s/e} - r_{sc/e} \tag{B.1}$$

where the subscripts s, sc, and e represent the sun, spacecraft, and earth, respectively. $r_{s/e}$ is the position of the sun relative to the earth and $r_{sc/e}$ is the position of the spacecraft relative to the earth. The unit vector from the spacecraft to the sun in the inertial frame is then given as:

$$\hat{s}^I = \frac{r_{s/sc}}{||r_{s/sc}||} = \frac{r_{s/e} - r_{sc/e}}{||r_{s/e} - r_{sc/e}||} \tag{B.2}$$

The rate of change of the unit sun vector is:

$$\dot{\hat{s}}^I = \frac{d}{dt}\left(\hat{s}^I\right) \tag{B.3}$$

$$\frac{d}{dt}\left(\frac{r_{s/sc}}{||r_{s/sc}||}\right) = \frac{1}{||r_{s/sc}||}\left(I - (\hat{s}^I)(\hat{s}^I)^T\right)\dot{r}_{s/sc} \tag{B.4}$$

Since $||r_{sc/e}|| \ll ||r_{s/e}||$, we can make the following approximations:

$$r_{s/sc} \approx r_{s/e} \tag{B.5}$$

$$\hat{s}^I \approx \frac{r_{s/e}}{||r_{s/e}||} \tag{B.6}$$

and modify Equation (B.4):

$$\dot{\hat{s}}^I = \frac{1}{||r_{s/e}||} \left(I - (\hat{s}^I)(\hat{s}^I)^T \right) \dot{r}_{s/sc} \tag{B.7}$$

The rate of change of the sun vector in the inertial frame can be expressed in the LVLH frame using the kinematic transport theorem:

$$\dot{\hat{s}}^I = \dot{\hat{s}}^{LVLH} + \Omega_I^{LVLH} \times \hat{s}^{LVLH} \tag{B.8}$$

where:

$$\Omega^{LVLH} = \begin{bmatrix} 0 \\ 0 \\ \omega \end{bmatrix} \tag{B.9}$$

is the angular velocity of the LVLH frame relative to the inertial frame and ω is the mean motion of the spacecraft's orbit. This equation can be modified by replacing the cross product with the skew-symmetric matrix such that:

$$\dot{\hat{s}}^I = \dot{\hat{s}}^{LVLH} + \left[\Omega_I^{LVLH} \times \right] \hat{s}^{LVLH} \tag{B.10}$$

$$\left[\Omega^{LVLH} \times \right] = \begin{bmatrix} 0 & -\omega & 0 \\ \omega & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{B.11}$$

Rearranging Equation (B.10), the rate of change of the sun vector relative to the spacecraft in the LVLH frame is:

$$\dot{\hat{s}}^{LVLH} = \dot{\hat{s}}^I - \left[\Omega^{LVLH} \times \right] \hat{s}^{LVLH} \tag{B.12}$$

Assuming that our models only need to be accurate for a short duration (e.g. hours or days), we can neglect the rate of change of the sun vector in the inertial frame:

$$\dot{\hat{s}}^I \approx 0 \tag{B.13}$$

and simplify Equation (B.12):

$$\dot{\hat{s}}^{LVLH} = -\left[\Omega^{LVLH}\times\right]\hat{s}^{LVLH} \tag{B.14}$$

The solution to Equation (B.14) is given as:

$$\hat{s}^{LVLH}(t) = \Phi(t,t_0)\hat{s}^{LVLH}(t_0) \tag{B.15}$$

where:

$$\Phi(t,t_0) = e^{-[\Omega\times](t-t_0)} \tag{B.16}$$

$$\Phi(t,t_0) = \begin{bmatrix} \cos\omega(t-t_0) & \sin\omega(t-t_0) & 0 \\ -\sin\omega(t-t_0) & \cos\omega(t-t_0) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{B.17}$$

The initial sun vector in the LVLH frame can be determined from high accuracy models as:

$$\hat{s}^{LVLH}(t_0) = T_I^{LVLH}(t_0)\left(\frac{r_{sc/e}(t_0) - r_{s/e}(t_0)}{||r_{sc/e}(t_0) - r_{s/e}(t_0)||}\right) \tag{B.18}$$

$$\hat{s}^{LVLH}(t_0) \approx -T_I^{LVLH}(t_0)\left(\frac{r_{s/e}(t_0)}{||r_{s/e}(t_0)||}\right) \tag{B.19}$$

where $T_I^{LVLH}(t_0)$ is the rotation matrix from the inertial frame to the LVLH frame at time t_0.

APPENDIX C
ROCKET SHIP EQUATIONS

The equations for a Monte Carlo and linear covariance analysis of a simplified rocket ship example are derived to demonstrate the capabilities of the two methods. The results from the two models are described in Section 3.3.

C.1 Monte Carlo Analysis

The true state for the rocket ship example consists of the position of the rocket ship relative to the planet, r, and its velocity, v.

$$\mathrm{x} = \begin{bmatrix} r \\ v \end{bmatrix} \tag{C.1}$$

Truth Dynamics

The continuous truth dynamics are:

$$\dot{\mathrm{x}} = F\mathrm{x} + Ww \tag{C.2}$$

$$\begin{bmatrix} \dot{r} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} r \\ v \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} w \tag{C.3}$$

where:

$$E[w(t)] = 0 \tag{C.4}$$

$$\sigma_w^2 = 0.000001 \text{ au}^2/\text{day}^3 \tag{C.5}$$

and $\mathrm{r}_0 = 100$ au, $\mathrm{v}_0 = -1$ au/day.

Sensor Model

The range sensor model is:

$$\tilde{\mathrm{y}} = r + \nu = H\mathrm{x} + \nu \tag{C.6}$$

$$\tilde{\mathrm{y}} = \begin{bmatrix} 1 & 0 \end{bmatrix} \mathrm{x} + \nu \tag{C.7}$$

where:

$$E[\nu] = 0 \tag{C.8}$$

$$\sigma_r^2 = 0.0001 \text{ au}^2 \tag{C.9}$$

Actuator Model

The filter state for the scenario consists of the filter estimated position, r, and velocity, v.

$$\hat{\mathrm{x}} = \begin{bmatrix} \hat{r} \\ \hat{v} \end{bmatrix} \tag{C.10}$$

The actuator model is:

$$\Delta V = \Delta \hat{V} \tag{C.11}$$

where $\Delta \hat{V}$ is the change in velocity calculated by the onboard guidance algorithm. Note that, for simplicity, there is no noise associated with the actuator in this example.

Navigation Filter Dynamics

The navigation measurement is processed by the onboard Kalman filter with the following state dynamics:

$$\hat{\mathrm{x}}_{i+1} = \hat{\Phi}(\Delta t)\hat{\mathrm{x}}_i \tag{C.12}$$

$$\begin{bmatrix} \hat{\mathrm{r}}_{i+1} \\ \hat{\mathrm{v}}_{i+1} \end{bmatrix} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \hat{\mathrm{r}}_i \\ \hat{\mathrm{v}}_i \end{bmatrix} \tag{C.13}$$

$$\hat{\mathrm{r}}_0 = 100 \text{ au} \tag{C.14}$$

$$\hat{\mathrm{v}}_0 = -1 \text{ au/day} \tag{C.15}$$

and covariance dynamics:

$$\hat{P}_{i+1} = \hat{\Phi}\hat{P}_i\hat{\Phi}^T + Q_d \tag{C.16}$$

$$\hat{Q}_d \approx \hat{\Phi}\hat{W}\hat{Q}_w\hat{W}^T\hat{\Phi}^T\Delta t = \begin{bmatrix} \Delta t^3 & \Delta t^2 \\ \Delta t^2 & \Delta t \end{bmatrix} \sigma_w^2 \tag{C.17}$$

$$P_0 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \tag{C.18}$$

Navigation Filter Update

The navigation filter processes range measurement using the standard Kalman filter update equations:

$$\hat{\mathrm{x}}^+ = \hat{\mathrm{x}}^- + \hat{K}\left[\tilde{\mathrm{y}} - \hat{H}\hat{\mathrm{x}}^-\right] \tag{C.19}$$

$$\hat{P}^+ = \left[I_{2\times 2} - \hat{K}\hat{H}\right]\hat{P}^-\left[I_{2\times 2} - \hat{K}\hat{H}\right]^T + \hat{K}\sigma_r^2\hat{K}^T \tag{C.20}$$

where:

$$\hat{K} = \hat{P}^-\hat{H}^T\left[\hat{H}\hat{P}^-\hat{H}^T + \sigma_r^2\right]^{-1} \tag{C.21}$$

$$\hat{H} = H = \begin{bmatrix} 1 & 0 \end{bmatrix} \tag{C.22}$$

Guidance Correction

The desired maneuver is calculated based on the estimated range and the time remaining until arrival.

$$\Delta\hat{V} = \hat{G}\hat{\mathrm{x}}^{-c} \tag{C.23}$$

$$\Delta\hat{V} = \begin{bmatrix} -1/(100-t) & -1 \end{bmatrix}\hat{\mathrm{x}}^{-c} \tag{C.24}$$

This ΔV command is sent to the thruster and the navigation state and covariance are corrected as:

$$\hat{\mathrm{x}}^{+c} = \hat{\mathrm{x}}^{-c} + \hat{B}\Delta\hat{V} \tag{C.25}$$

$$\hat{\mathrm{x}}^{+c} = \hat{\mathrm{x}}^{-c} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \Delta \hat{V} \tag{C.26}$$

or equivalently as:

$$\hat{\mathrm{x}}^{+c} = \hat{\mathrm{x}}^{-c} + \hat{B}\hat{G}\hat{\mathrm{x}}^{-c} \tag{C.27}$$

$$\hat{\mathrm{x}}^{+c} = \hat{\mathrm{x}}^{-c} + \begin{bmatrix} 0 & 0 \\ -1/(100-t) & -1 \end{bmatrix} \hat{\mathrm{x}}^{-c} \tag{C.28}$$

Truth Maneuver Dynamics

The truth dynamics are also corrected with the calculated maneuver as:

$$\mathrm{x}^{+c} = \mathrm{x}^{-c} + B\Delta\hat{V} = \mathrm{x}^{-c} + B\hat{G}\hat{x}^{-c} \tag{C.29}$$

$$\mathrm{x}^{+c} = \mathrm{x}^{-c} + \begin{bmatrix} 0 & 0 \\ -1/(100-t) & -1 \end{bmatrix} \hat{\mathrm{x}}^{-c} \tag{C.30}$$

C.2 Linear Covariance Analysis

The augmented state for this example consists of the true and estimated state deviations from the nominal trajectory

$$\mathrm{X} = \begin{bmatrix} \mathrm{x} - \bar{\mathrm{x}} \\ \hat{\mathrm{x}} - \bar{\mathrm{x}} \end{bmatrix} = \begin{bmatrix} \delta r \\ \delta v \\ \delta \hat{r} \\ \delta \hat{v} \end{bmatrix} \tag{C.31}$$

where the nominal trajectory consists of the linear path defined by the noise-free, constant velocity (e.g. $w = 0$).

$$\bar{r}(t) = 100 - t \tag{C.32}$$

$$\bar{v}(t) = -1 \tag{C.33}$$

The mean and covariance of the augmented state are:

$$E[\mathrm{X}] = 0_{4\times 1} \tag{C.34}$$

$$E[\mathrm{X}\mathrm{X}^T] = C_{4\times 4} \tag{C.35}$$

The scenario begins with perfect knowledge of both position and velocity:

$$C_0 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{C.36}$$

Augmented State and Covariance Dynamics

$$\mathrm{X}_{i+1} = \mathcal{F}\mathrm{X}_i + \mathcal{W}w \tag{C.37}$$

$$C_{i+1} = \mathcal{F}C_i\mathcal{F}^T + \mathcal{W}\sigma_w^2\mathcal{W}^T \tag{C.38}$$

$$\mathcal{F} = \begin{bmatrix} 1 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathcal{W} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \tag{C.39}$$

Augmented State Covariance Update

$$\mathrm{X}_j^+ = \mathcal{A}\mathrm{X}_j^- + \mathcal{B}\nu_j \tag{C.40}$$

$$C^+ = \mathcal{A}C^-\mathcal{A}^T + \mathcal{B}\sigma_r^2\mathcal{B}^T \tag{C.41}$$

$$\mathcal{A} = \begin{bmatrix} I_{2\times 2} & 0_{2\times 2} \\ \hat{K}H & I_{2\times 2} - \hat{K}\hat{H} \end{bmatrix}, \quad \mathcal{B} = \begin{bmatrix} 0_{2\times 1} \\ \hat{K} \end{bmatrix} \tag{C.42}$$

Augmented State Covariance Correction

$$\mathrm{X}^{+c} = \mathcal{D}\mathrm{X}^{-c} \tag{C.43}$$

$$C^{+c} = \mathcal{D}C^{-c}\mathcal{D}^{T} \tag{C.44}$$

$$\mathcal{D} = \begin{bmatrix} I_{2\times2} & \hat{B}\hat{G} \\ 0_{2\times2} & I_{2\times2} + \hat{B}\hat{G} \end{bmatrix} \tag{C.45}$$

True Position Dispersion

$$D_{\delta r,i} = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} C_i \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}^T \tag{C.46}$$

Filter Position Dispersion

$$D_{\delta \hat{r},i} = \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix} C_i \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix}^T \tag{C.47}$$

True Navigation Position Error

$$P_{\delta \hat{r},i} = \begin{bmatrix} -1 & 0 & 1 & 0 \end{bmatrix} C_i \begin{bmatrix} -1 & 0 & 1 & 0 \end{bmatrix}^T \tag{C.48}$$

True Maneuver Dispersion

$$D_{\Delta V} = \begin{bmatrix} 0 & 0 & \frac{-1}{100-t} & -1 \end{bmatrix} C^{-c} \begin{bmatrix} 0 & 0 & \frac{-1}{100-t} & -1 \end{bmatrix}^T \tag{C.49}$$

APPENDIX D

ANGLES-ONLY MEASUREMENT SENSITIVITY MATRIX

The camera measurement model is given as:

$$\tilde{\mathrm{y}}_j = h(\mathrm{x}_j) + \nu_j \tag{D.1}$$

$$\tilde{\mathrm{y}}_j = \begin{bmatrix} r_x^{cam}/r_z^{cam} \\ r_y^{cam}/r_z^{cam} \end{bmatrix} + \nu_j^{cam} \tag{D.2}$$

where r_{RSO}^{cam} is the position vector of the RSO relative to the imaging vehicle's camera frame. This vector is defined as:

$$\mathrm{r}_{RSO}^{cam} = \begin{bmatrix} r_x^{cam} \\ r_y^{cam} \\ r_z^{cam} \end{bmatrix} = -T_{lvlh}^{cam}\mathrm{r}^{lvlh} \tag{D.3}$$

where r^{lvlh} is the position vector of the vehicle relative to the RSO in the LVLH frame, and T_{lvlh}^{cam} is the rotation matrix defining the orientation of the vehicle camera frame relative to the LVLH frame.

The measurement sensitivity matrix is defined as:

$$H = \left.\frac{\partial \mathrm{h}(\mathrm{x})}{\partial \mathrm{x}}\right|_{\bar{\mathrm{x}}} \tag{D.4}$$

where x is the filter state vector. This partial derivative can be expanded using the chain rule:

$$H = \frac{\partial \mathrm{h}(\mathrm{x})}{\partial \mathrm{r}_{RSO}^{cam}} \frac{\partial \mathrm{r}_{RSO}^{cam}}{\partial \mathrm{x}} \tag{D.5}$$

The first partial derivative of Equation (D.5) is:

$$\frac{\partial}{\partial \mathrm{x}}\begin{bmatrix} r_x^{cam}/r_z^{cam} \\ r_y^{cam}/r_z^{cam} \end{bmatrix} = \begin{bmatrix} 1/r_z^{cam} & 0 & -r_x^{cam}/(r_z^{cam})^2 \\ 0 & 1/r_z^{cam} & -r_y^{cam}/(r_z^{cam})^2 \end{bmatrix} \tag{D.6}$$

The second partial derivative of Equation (D.5) is:

$$\frac{\partial}{\partial \mathrm{x}}\mathrm{r}_{RSO}^{cam} = \begin{bmatrix} \frac{\partial \mathrm{r}_{RSO}^{cam}}{\partial \mathrm{r}^{lvlh}} & \frac{\partial \mathrm{r}_{RSO}^{cam}}{\partial \mathrm{v}^{lvlh}} \end{bmatrix} \tag{D.7}$$

where the position of the RSO in the camera frame is:

$$\mathrm{r}_{RSO}^{cam} = -T_{lvlh}^{cam}\mathrm{r}^{lvlh} \tag{D.8}$$

Equation (D.7) is then written as:

$$\frac{\partial}{\partial \mathrm{x}}\mathrm{r}_{RSO}^{cam} = \begin{bmatrix} -T_{lvlh}^{cam}\frac{\partial \mathrm{r}^{lvlh}}{\partial \mathrm{r}^{lvlh}} & -T_{lvlh}^{cam}\frac{\partial \mathrm{r}^{lvlh}}{\partial \mathrm{v}^{lvlh}} \end{bmatrix} \tag{D.9}$$

$$\frac{\partial}{\partial \mathrm{x}}\mathrm{r}_{RSO}^{cam} = \begin{bmatrix} -T_{lvlh}^{cam} & 0_{3\times 3} \end{bmatrix} \tag{D.10}$$

Substituting Equations (D.6) and (D.10) back into Equation (D.5) yields:

$$H = \begin{bmatrix} 1/r_z^{cam} & 0 & -r_x^{cam}/(r_z^{cam})^2 \\ 0 & 1/r_z^{cam} & -r_y^{cam}/(r_z^{cam})^2 \end{bmatrix}\begin{bmatrix} -T_{lvlh}^{cam} & 0_{3\times 3} \end{bmatrix} \tag{D.11}$$

The nominal camera pointing places the RSO directly on the camera boresight such that:

$$\mathrm{r}_{RSO}^{cam} = \begin{bmatrix} 0 \\ 0 \\ r_z^{cam} \end{bmatrix} \tag{D.12}$$

The measurement partial evaluated at the nominal state can now be given as:

$$H|_{\bar{x}} = \begin{bmatrix} 1/r_z^{cam} & 0 & 0 \\ 0 & 1/r_z^{cam} & 0 \end{bmatrix} \begin{bmatrix} -T_{lvlh}^{cam} & 0_{3\times 3} \end{bmatrix} \tag{D.13}$$

APPENDIX E

MATLAB GENETIC ALGORITHM PARAMETER SETTINGS

The following settings are applied to the MATLAB Genetic Algorithm through the *optimoptions* function.

Table E.1: Parameters and values used for MATLAB *optimoptions*.

Parameter Name	Value
`ConstraintTolerance`	`1e-6`
`FunctionTolerance`	`1e-6`
`MaxTime`	9000
`CrossoverFraction`	`0.8`
`PopulationSize`	`100`
`MaxGenerations`	`200`
`MaxStallGenerations`	`50`
`MutationFcn`	{`@mutationgaussian,10,0.05`}
`UseParallel`	`true`
`Display`	`iter`

`PopulationSize`: The population size represents the number of unique individuals in each generation and is a key parameter for tuning the GA. While having larger population sizes is generally preferable for determining the global optimum, this may not be feasible due to computational time. A general rule of thumb is the have a population 10 to 30 times larger than the number of optimization variables. For both the two and three-impulse path planning problem, it was generally found that a population size of 100 was adequate to achieve convergence and limit execution time. For multiple-impulse sequences, population sizes of 150 to 300 individuals were more reliable for achieving convergence. Additionally, for the two-impulse problem that has only two optimization variables, using a population size of 50 was able to reliably converge to the desired solution. The solution, however, typically required more generations (and time) to find the solution.

`MaxGenerations`: The maximum number of generations determines the maximum number of iterations allowed before stopping the algorithm. Each generation after the initial population allows for the GA to search for the global optimum through the processes of selection, crossover, mutation, and elitism. This setting is strongly integrated with the population size in tuning the GA. Generally, larger population sizes require fewer generations. Through testing it was observed that a minimum of 100 generations should be allowed for the algorithm. Using 200 generations for the path planning problem provided a consistent number of iterations to identify the optimal solution. It is important to use the maximum stall generations setting to identify when the algorithms should be stopped at an earlier iteration.

`MaxStallGenerations`: A stall generation is an iteration of the GA where there is no improvement to the top individual's fitness function. In other words, if the current generation has no improvement over the previous generation, the algorithm has stalled on its way to the optimal solution. Should the algorithm find the global optimum, there is no further ability to improve the fitness function and the algorithm stalls. Setting a maximum number of stall generations helps to stop the algorithm when the optimum has been found or if the algorithm is stuck in local minimum. When set at 50 stall generations, this setting does not significantly impact run time and only stops the algorithm early on unique occasions.

`CrossoverFraction`: The crossover fraction indicates the population fraction of a subsequent generation that are created through the crossover function. Set at 0.8 (80%), the MATLAB default value, this setting allows for efficient exploration of the solution space. If set to 0, new populations will only differ from the previous population by the mutation process.

`MutationFcn`: Mutation allows for genetic diversity in GA children and allows the algorithms to search a broader solution space. Evaluations done by the author showed that the `mutationgaussian` function provided the best results for the path planning problem. This function operates by applying a Gaussian random mutation to the genome that reduces in magnitude over generations. The standard deviation of the mutation for a given iteration, k, is determined in the following recursive algorithm:

$$\sigma_k = \sigma_{k-1}\left[1 - \alpha\frac{k}{N}\right] \tag{E.1}$$

where α represents the shrink factor and N the total number of generations. For the function call in the path planning, the initial standard deviation, σ_1, is set to 10 and the shrink factor is set at $\alpha = 0.05$. These settings allows the mutation standard deviation to remain large for several generations before slowly reducing towards the final generation. Smaller mutations in the final generations allow for finer adjustments in the search for the optima.

`UseParallel`: If available, GA users should employ parallel computing resources for the GA. Each individual of the population is independent allowing for efficient parallel computation within each generation.

`MaxTime`: The maximum run time should only be used to limit computational problems with running the GA and should not be used as a nominal stopping method. The value of 9000 seconds was set for the three-impulse scenario and represents the maximum expected run-time when including the tree analysis and safety constraint. Scenarios were run on a laptop with the following hardware and configuration:

- CPU: Intel(R) Core(TM) i7-7820HQ (2.90GHz)
- RAM: 32 GB DDR4
- MATLAB: 9.9.0.1570001 (R2020b) Update 4
- Parallel Computing: 4 parallel workers (Quad core)

`ConstraintTolerance` and `FunctionTolerance`: These tolerance values are used for determining the stopping condition of the optimization routines. These values were not observed to affect the performance of the GA and were, instead, primarily used to define the stopping conditions for *fmincon*. While the GA maximum time was set to 9000 seconds, the *fmincon* step of the hybrid optimization approach typically converged in less than 30 seconds.

`Display`: When learning to run a GA, the user should monitor key parameters such as the fitness function at each iteration. Key parameters can also be viewed through `PlotFcn`. The author preferred to plot the following functions while developing the GA implementation:

- `gaplotstopping`
- `gaplotscores`
- `gaplotscorediversity`
- `gaplotbestf`
- `gaplotbestindiv`
- `gaplotexpectation`

Displaying or plotting iteration data was not observed to impact run-time.

CURRICULUM VITAE

Nathan Bohus Stastny

EDUCATION

Ph.D.	**Aerospace Engineering**
May 2022	Utah State University, Logan, UT
	Major Advisor: David Geller
	Dissertation: *Optimal Relative Path Planning for Constrained Stochastic Space Systems*
M.S.	**Mechanical Engineering**
Dec 2006	Utah State University, Logan, UT
	Advisor: David Geller
	Thesis: *Autonomous Optical Navigation at Jupiter: A Linear Covariance Analysis*
B.S.	**Mechanical Engineering**
Aug 2002	Brigham Young University, Provo, UT
	Defense Acquisition University
	Science & Technology Management - Level 2
	Systems Engineering - Level 1

PROFESSIONAL EXPERIENCE

Space Dynamics Laboratory (2018-Present)
Utah State University, North Logan, UT

- Engineer - Guidance, Navigation, and Control
- Systems Engineering Group - GNC Sub-group Lead
- Satellite Bus Team - GNC Lead

Air Force Research Laboratory, Systems Technology Office (2016-2018)
Wright-Patterson AFB, Dayton, OH

- Senior Aerospace Research Engineer
- Principal Investigator

Air Force Research Laboratory, Space Vehicles Directorate (2007-2016)
Kirtland AFB, Albuquerque, NM

- Aerospace Research Engineer
- Lead Mission Planner - ANGELS
- GNC Operations Lead - ANGELS
- Principal Investigator - Mycroft
- Principal Investigator - DSX Adaptive Control Experiment (ACE)

Utah State University (2005-2006)
Logan, UT

- Graduate Research Assistant

ATK Launch Systems (2006)
Promontory, UT

- Engineering Assistant

URS Corporation (2002-2004)
Las Vegas, NV

- Mechanical Design Engineer

AWARDS

United States Air Force (USAF)

- Air Force Exemplary Civilian Service Award (2018)

Air Force Research Laboratory, Space Vehicles Directorate (AFRL/RV)

- Junior Science and Technology Award (2010)
- Director's Cup (2012), TacSat-3 Team
- Team Award (2015), ANGELS Team
- Director's Cup (2019), Mycroft Team

Rotary National Award for Space Achievement (RNASA)

- Stellar Award - Early Career (2015)
- Stellar Award - Team (2016), ANGELS Team

Space Dynamics Laboratory

- SDL Ph.D. Fellowship (2019-2022)

JOURNAL ARTICLES

J3 Trajectory Optimization for Constrained Stochastic RPO Missions, N. Stastny and D. Geller, *AIAA Journal of Guidance, Control, and Dynamics*, (Submitted)

J2 Resource-Constrained Constellation Scheduling for Rendezvous and Servicing Operations, S. Cox, N. Stastny, G. Droge, and D. Geller, *AIAA Journal of Guidance, Control, and Dynamics*, March 2022

J1 Autonomous Optical Navigation at Jupiter: A Linear Covariance Analysis, Nathan B. Stastny and David K. Geller, *AIAA Journal of Spacecraft and Rockets*, vol. 45, no. 2, pp. 290-298, 2008.

CONFERENCE PAPERS

C8 Understanding Optimal Two-Burn Relative Trajectory Transfers with Maneuver Errors and Dispersion Requirements, N. Stastny and D. Geller, *AAS/AIAA Astrodynamics Specialist Conference*, South Lake Tahoe, CA, 2020

C7 Optimal Relative Trajectory Design with Mission Constraints and Performance Requirements, N. Stastny, D. Geller, and S. Shuster, *43rd Annual AAS Guidance, Navigation & Control Conference*, Breckenridge, CO, 2020.

C6 Constellation Planning Methods for Sequential Spacecraft Rendezvous Using Multi-Agent Scheduling, S. Cox, N. Stastny, G. Droge, and D. Geller, *AIAA/AAS Astrodynamics Specialists Conference*, Portland, ME, 2019.

C5 In-Situ O/N2 Ratios from the AFRL Mass Spectrometer on the TacSat-2 Satellite, J. O. Wise, J. O. Ballenthin, W. Thorn, E. Sutton, N. Stastny, F. Chavez, F. A. Marcos, D. Strickland, D., *48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, Orlando, FL, 2010.

C4 Localized Density/Drag Prediction for Improved Onboard Orbit Propagation, N. Stastny, F. Chavez, C. Lin, A. Lovell, R. Bettinger, and J. Luck, *Proceedings of the Advanced Maui Optical and Space Surveillance Technologies Conference*, Wailea, HI, pp. 51-58, 2009.

C3 Comparison of the Extended and Unscented Kalman Filters for Angles Based Relative Navigation, N. Stastny, R. Bettinger, and F. Chavez, *AIAA/AAS Astrodynamics Specialists Conference*, Honolulu, HI, 2008.

C2 LinCov Analysis of an Automated Celestial Inertial Navigation Approach for GEO Satellites, N. Stastny, R. Bettinger, and F. Chavez, *AIAA/AAS Astrodynamics Specialists Conference*, Honolulu, HI, pp. 184-196, 2008.

C1 Autonomous Optical Navigation at Jupiter: A Linear Covariance Analysis, Nathan B. Stastny and David K. Geller, *AIAA/AAS Astrodynamics Specialist Conference and Exhibit*, Keystone, CO, August 2006.