Navigation and Control Design for the CanX-4/-5 Satellite Formation Flying Mission

by

Niels Henrik Roth

A thesis submitted in conformity with the requirements for the degree of Master of Applied Science
Institute for Aerospace Studies
University of Toronto

Copyright © 2010 by Niels Henrik Roth
Abstract

Navigation and Control Design for the CanX-4/-5 Satellite Formation Flying Mission

Niels Henrik Roth
Master of Applied Science
Institute for Aerospace Studies
University of Toronto
2010

CanX-4/-5 is a formation flying technology demonstration mission that shall demonstrate sub-meter formation tracking control. The key to this precision control is carrier phase differential GPS state estimation, which enables centimeter-level relative state estimation. In this thesis, the formation flying controller design is reviewed in detail, and an innovative closed-loop formation reconfiguration strategy is presented. In addition, the designs of both coarse- and fine-mode relative state estimators are presented. Formation flying simulations demonstrate the efficacy of the proposed control and coarse estimation. Furthermore, hardware tests are performed to test the computational efficiency of the control algorithms and to validate the fine-mode relative navigation filter.
Acknowledgements

Without the aid and support of my supervisors and colleagues, this work would not be possible. I would like to thank Dr. Chris Damaren for his invaluable insight to the challenges faced during this work. Jakob Lifshits deserves thanks for enduring my endless computer-related questions, as do the entire SFL staff for their support in all hardware related aspects of the work. In addition I would like to thank Dr. Kyle O’Keefe from the University of Calgary Geomatics Department for providing me with valuable knowledge and answering my technical questions relating to GPS-based relative navigation. Lastly I would like to thank Dr. Robert Zee for giving me the opportunity to work alongside the SFL staff on the CanX-4/-5 mission, and my peers working on other aspects of the mission.
## Contents

1 Introduction ................................................. 1
   1.1 Relative Navigation Review ........................... 3
   1.2 Formation Control Review ............................. 5
   1.3 Thesis Overview ...................................... 7

2 CanX-4/-5 Overview ........................................ 8
   2.1 Communication Hardware ............................. 9
   2.2 Onboard Computers ................................... 10
   2.3 Navigation and Control Hardware .................... 11
      2.3.1 Attitude ......................................... 11
      2.3.2 Orbit Determination ............................ 11
      2.3.3 Formation Control .............................. 11

3 Orbital Dynamics .......................................... 13
   3.1 Time Systems ......................................... 13
   3.2 Coordinate Frames .................................... 14
      3.2.1 Geocentric/J2000 Inertial Frame ............... 14
      3.2.2 Earth-Centered-Earth-Fixed Frame ............. 14
      3.2.3 Local Orbital/Hill Frame ....................... 14
      3.2.4 Spacecraft Body Frame ........................ 15
      3.2.5 Coordinate Transformations .................... 15
         3.2.5.1 J2000 to ECEF ............................. 15
         3.2.5.2 ECEF to Local Orbital .................... 16
         3.2.5.3 Body to J2000 .............................. 17
   3.3 Equations of Motion ................................ 17
      3.3.1 Forces .......................................... 17
         3.3.1.1 Earth Gravity .............................. 18
         3.3.1.2 Third Body Gravity ......................... 19
5.1.2 Implicit Tracking Control ................................................. 51
5.1.3 Weighting Matrix Selection ............................................ 52
5.1.4 Reference Trajectories .................................................. 52

5.2 Reconfiguration and Recovery ............................................ 53
5.2.1 Preliminaries .............................................................. 54
5.2.2 Continuous Solution ..................................................... 55
5.2.3 Discrete (Impulsive) Multiple-Thrust Solution ....................... 57
5.2.4 Equivalence of Discrete and Continuous Solutions .................. 59
5.2.5 Robust Reconfigurations ................................................. 60

5.3 Attitude Slews ............................................................... 61
5.3.1 Method 1 ................................................................. 62
5.3.2 Method 2 ................................................................. 63

6 Simulation and Hardware Test Results .................................... 65
6.1 FIONA ................................................................. 65

6.2 Reconfiguration Algorithm Test ......................................... 68
6.2.1 Case 1 - ATO to PCO ...................................................... 70
6.2.2 Case 2 - ATO to ATO ...................................................... 71
6.2.3 Case 3 - Recovery Maneuver ............................................. 73
6.2.4 Case 4 - Varying Transfer Duration .................................... 74
6.2.5 Reconfiguration Test Conclusions ..................................... 75

6.3 Flight Hardware Implementation ......................................... 76
6.3.1 Function Timing .......................................................... 76
6.3.1.1 Coarse Mode EKF ..................................................... 77
6.3.1.2 Reconfiguration Algorithm .......................................... 78
6.3.1.3 FIONA Execution Times ............................................. 78

6.4 Simulation Environment .................................................. 81
6.4.1 Blackout Simulation ...................................................... 82

6.5 CanX-4/-5 Reconfiguration Simulations .................................. 83

6.6 Mission Simulation ......................................................... 86
6.6.1 Coarse EKF Performance ................................................ 88
6.6.2 Case 1 - No Control During Blackouts ................................. 91
6.6.2.1 Reconfiguration Performance ....................................... 91
6.6.2.2 Formation Keeping Performance ................................... 93
6.6.3 Case 2 - Coarse Control During Blackouts ............................. 96
6.6.3.1 Reconfiguration Performance ....................................... 96
# List of Tables

4.1 Transmitted data for fine mode relative navigation. ........................................ 45

5.1 Choice of constants for circular PCO and ATO reference trajectories obtained from Eq. (3.26). ............................................................ 53

5.2 Choice of constants for elliptical PCO and ATO reference trajectories obtained from Eq. (3.24). ............................................................ 53

6.1 Initial conditions for Chief spacecraft in reconfiguration algorithm tests, where $h_a$ is the altitude at periapsis. .................................................. 68

6.2 Deputy initial condition parameters in reconfiguration algorithm tests. ........ 69

6.3 Fuel usage [cm/s] for Case 1 reconfiguration maneuver. .......................... 71

6.4 Fuel usage [cm/s] for Case 2 reconfiguration maneuver. .......................... 73

6.5 Fuel usage [cm/s] for Case 3 reconfiguration maneuver. .......................... 74

6.6 Final errors and fuel usages for Case 2 reconfiguration with varying transfer times. 74

6.7 EKF update step timing for various methods - five run average. ............... 77

6.8 EKF covariance prediction step timing for various STM expansion orders - three run average. ............................................................ 77

6.9 EKF state prediction step timing for various gravity models and RK4 time steps - three run average for a 5 second integration period. .................. 78

6.10 FIONA execution times for various operational modes. .......................... 80

6.11 FIONA reconfiguration simulation - truth model parameters. ..................... 83

6.12 FIONA reconfiguration simulation - FIONA parameters. ......................... 84

6.13 Results of reconfiguration simulations with FIONA. ............................. 86

6.14 FIONA mission simulation - FIONA parameters. .................................. 87

6.15 Coarse mode estimation errors (RMS). .................................................. 88

6.16 Case 1 mission simulation reconfiguration final errors. .......................... 91

6.17 Case 1 mission simulation RMS tracking error summary. ........................ 94

6.18 Blackout-free tracking error summary; 10 orbits per formation. ............... 94

6.19 Case 1 mission simulation fuel use breakdown. ....................................... 94
6.20 Improvement in tracking errors and fuel use with attitude targeting method 2.

All fields record the difference Method 1-Method 2. ........................................ 96

6.21 Case 2 mission simulation reconfiguration final errors. ................................. 97

6.22 Case 2 mission simulation RMS tracking error summary. ............................. 97

6.23 Case 2 mission simulation fuel use breakdown. ............................................. 98

6.24 Case 3 mission simulation reconfiguration final errors. ................................. 99

6.25 Case 3 mission simulation tracking error summary. ...................................... 99

6.26 Case 3 mission simulation fuel use breakdown. ............................................. 100

6.27 Static baseline differential navigation test EKF parameters. ........................... 106

D.1 Simulation constants. .......................................................... 128
# List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Perspective view of the CanX-4 satellite [1]</td>
<td>9</td>
</tr>
<tr>
<td>2.2</td>
<td>Reverse perspective view of the CanX-4 satellite [1]</td>
<td>10</td>
</tr>
<tr>
<td>3.1</td>
<td>The three major coordinate frames. The J2000 frame is shown with a solid black line, the ECEF frame is shown with a dashed line, and the Local Orbital frame is shown with a dotted line.</td>
<td>15</td>
</tr>
<tr>
<td>3.2</td>
<td>Gravitational influence of the Earth and third body on a satellite.</td>
<td>18</td>
</tr>
<tr>
<td>3.3</td>
<td>The angular components of the orbital elements.</td>
<td>21</td>
</tr>
<tr>
<td>3.4</td>
<td>Relative position definition.</td>
<td>23</td>
</tr>
<tr>
<td>4.1</td>
<td>Differential GPS measurement concept, where ( p^r_s ) denotes a measurement made from receiver ‘r’ to satellite ‘s’.</td>
<td>35</td>
</tr>
<tr>
<td>4.2</td>
<td>Position of the GPS antenna phase center in the satellite body frame.</td>
<td>37</td>
</tr>
<tr>
<td>4.3</td>
<td>Main GPS data processing algorithm, including data transmission.</td>
<td>46</td>
</tr>
<tr>
<td>5.1</td>
<td>Thrust recalculation procedure to greatly increase robustness.</td>
<td>60</td>
</tr>
<tr>
<td>5.2</td>
<td>Depiction of ACS target prediction scheme.</td>
<td>61</td>
</tr>
<tr>
<td>5.3</td>
<td>Attitude spikes resulting from prediction scheme. The three plots denote the X, Y, and Z components of the target attitude vector in the GCI frame. The red lines denote a spike-free series of targets.</td>
<td>62</td>
</tr>
<tr>
<td>6.1</td>
<td>FIONA - high level program flow.</td>
<td>66</td>
</tr>
<tr>
<td>6.2</td>
<td>Reconfiguration thrust example.</td>
<td>69</td>
</tr>
<tr>
<td>6.3</td>
<td>Final errors for Case 1 reconfiguration maneuver. The arrows denote the improvement in final accuracy achieved by using the robust transfer method.</td>
<td>71</td>
</tr>
<tr>
<td>6.4</td>
<td>Final errors for Case 2 reconfiguration maneuver. The arrows denote the improvement in final accuracy achieved by using the robust transfer method.</td>
<td>72</td>
</tr>
<tr>
<td>6.5</td>
<td>Final errors for Case 3 reconfiguration maneuver. The arrows denote the improvement in final accuracy achieved by using the robust transfer method.</td>
<td>73</td>
</tr>
</tbody>
</table>
6.6 Reconfiguration algorithm execution time versus number of thrusts. 79
6.7 Differential integration error for the RK4 method, with 0.1 second time step baseline. 80
6.8 Simulation environment block diagram. 81
6.9 Depiction of pertinent blackout simulation parameters. 83
6.10 Position and velocity errors throughout the three reconfiguration orbits. 85
6.11 Sample of coarse EKF performance for Deputy satellite. The dashed lines represent estimated $\pm 3\sigma$ error bounds. 89
6.12 Sample of coarse relative state estimation performance. The dashed lines represent estimated $\pm 3\sigma$ error bounds. 90
6.13 Case 1 mission simulation tracking errors. 93
6.14 Sample thrust profile with GPS blackouts. 95
6.15 Case 3 mission simulation tracking errors. 100
6.16 Tracking error as coarse mode changes to fine mode. 102
6.17 Absolute navigation test errors. 105
6.18 Differential positioning test - antenna setup. 107
6.19 East-North-Up frame. 108
6.20 Static baseline fine mode test - comparison with coarse solution. 108
6.21 Static baseline fine mode test - relative position errors in ECEF frame. 109
6.22 Static fine mode test - relative velocity errors in ECEF frame. 110
B.1 Flowchart depicting the relative navigation algorithm. 124
B.2 Flowchart depicting the double difference ambiguity resolution algorithm. 124
B.3 Flowchart depicting the parsing of the RANGECMP GPS log. 124
B.4 Flowchart depicting the data quality check. 125
Notation

General

\( \mathcal{F}_A \) = Matrix whose columns are the basis vectors of frame A.
\( c \) = Scalar quantity.
\( \mathbf{x} \) = Vector quantity.
\( A, a \) = Matrix or vector quantity.
\( \hat{a} \) = State estimate or unit vector.
\( \hat{a}_A \) = Scalar components of a vector expressed in frame A.
\( \mathbf{I}_{n \times n} \) = n by n identity matrix.
\( (\cdot) \) = Time derivative of a vector in the inertial frame.
\( (\cdot)_o \) = Time derivative of a vector in a moving frame.
\( C_{12} \) = Rotation matrix from frame 2 to frame 1.
\( \| \cdot \| \) = Two-norm of a vector.
\( \mathcal{N}(\mathbf{a}, \mathbf{P}) \) = Normal distribution with mean \( \mathbf{a} \) and covariance \( \mathbf{P} \).
\( (\cdot)_k^- \) = Predicted quantity at time \( k \).
\( (\cdot)_k^+ \) = Corrected quantity at time \( k \).

Subscripts

\( (\cdot)_E \) = Earth-centered-Earth-fixed frame.
\( (\cdot)_I \) = Inertial frame.
\( (\cdot)_L \) = Local orbital frame.
\( (\cdot)_B \) = Spacecraft body frame.
\( (\cdot)_\oplus \) = Relating to Earth.
\( (\cdot)_c \) = Relating to Chief satellite.
\( (\cdot)_d \) = Relating to Deputy satellite.
\( (\cdot)_r \) = Relating to a GPS receiver.
\( (\cdot)_s \) = Relating to a GPS satellite.

Math

Let \( \mathbf{a} = \begin{bmatrix} a_1 & a_2 & a_3 \end{bmatrix}^T \), then

\[
\mathbf{a}^\times = \begin{bmatrix}
0 & -a_3 & a_2 \\
-a_3 & 0 & -a_1 \\
-a_2 & a_1 & 0
\end{bmatrix}.
\]
Chapter 1

Introduction

Satellite formation flying can be defined as the active control of relative position and velocity between two or more space vehicles. There is no restriction on the sizes or distributions of the vehicles: one vehicle may be massive (such as the International Space Station), while the others may be micro- or nanosatellites, and they can be separated by any distance. Recently this topic has been given much attention in the literature, as more people recognize the benefits of formation flying and design more missions around this concept. In principle the formation flying concept holds several main advantages over conventional mission designs. Since a large satellite could potentially be replaced with several smaller satellites flying in formation, tremendous development time and cost reductions can be achieved. Greater mission flexibility and robustness are also possible, because contrary to a monolithic spacecraft, the failure of one satellite still allows the mission to continue. The mission can also be augmented by launching more satellites and expanding or reconfiguring the formation, which will lead to new and innovative applications. Lastly, distributed instruments in formation provide more accurate scientific data than otherwise possible. These ideas have already been validated in several past missions, while future planned missions aim to push the boundary of what is achievable in formation flying.

Ideas such as inspection of the International Space Station or space shuttles, and on-orbit servicing of defunct or aging satellites have already been validated by the Orbital Express mission [2], which successfully demonstrated autonomous rendezvous, inspection, capture, refuelling, and component exchange with a passive (non-thrusting) spacecraft. The autonomous rendezvous maneuvers were notable, in that they were demonstrated from both short (< 200 m) and long ranges (> 200 km). In this case, both satellites were massive, each weighing over 227 kg. Given the technology is already available, a future on-orbit servicing mission which would provide refuelling and graveyard orbit parking services is being planned [3].

Enhanced on-orbit measurements have already been demonstrated in the Earth observing one (EO-1) mission [4] and the GRACE mission [5]. While by definition the EO-1 mission did
not use the FF concept, it was placed into a 450 km along-track orbit with existing imaging satellites and was equipped with a suite of imaging sensors to augment and complement existing capabilities, thereby demonstrating one of the major FF benefits. Only coarse formation keeping strategies were applied to the EO-1 satellite. The GRACE mission used two massive satellites flying in roughly the same orbit with a rough 200 km separation to obtain highly accurate estimates of the Earth’s gravitational field and changes thereof, by accurately measuring the changes in inter-satellite range [6]. For GRACE, coarse formation keeping was applied to keep the satellites within 50 km of their desired separation [7]. However, for this mission, it was the post-processed absolute and relative positions that mattered. The TanDEM-X mission, planned to begin operations in the near future, will augment the measurement capabilities of the TerraSAR-X satellite by initiating a close proximity (< 1 km) helical formation, to allow highly accurate synthetic aperture radar (SAR) measurements to obtain digital elevation maps of the Earth [8], [9].

There exists several other planned formation flying missions, whose aims range from demonstrating the formation flying technology on small platforms to providing useful and unprecedented on-orbit measurements. The recently launched PRISMA mission [7] is a technology demonstration mission that will perform real time sub-meter relative orbit determination and to enable low Earth orbit (LEO) formation flight with a passive target spacecraft with an error < 10 m. The active (thrusting) spacecraft is about 150 kg while the passive (attitude stabilized) craft is about 40 kg. The PROBA-3 mission [10] will demonstrate two satellites formation flight in high Earth orbit (HEO) at a distance of roughly 150 m, acting as a space-based coronagraph. The JC2SAT mission [11] will demonstrate formation flying between two nanosatellites using only differential drag as a means of actuation. The satellites will also be equipped with miniature far infrared radiometers to provide additional mission value. The CanX-4/-5 mission [12] is another LEO formation flying technology demonstration mission that will show that sub-meter formation flying is possible on the nanosatellite scale (see Chap. 2). A sub-meter formation tracking error would represent the most accurate control to date by an order of magnitude. The aforementioned missions will pave the way for even more ambitious on-orbit instruments which include the Simbol-X X-ray telescope [13], the MAX gamma ray telescope [14], and the LISA interferometer [15], to name a few. See [16] for more examples of formation flying science missions.

Behind any formation flying mission, there are two major aspects which enable the completion of the scientific objectives: formation control, and absolute/relative navigation. The topic of formation control encompasses the theoretical basis for maintaining and reconfiguring the satellite formations, while navigation is the process by which the state (e.g. positions, velocities, attitudes, etc.) of each vehicle in the formation is determined. This thesis is concerned with
the formation control and relative navigation as it relates to the CanX-4/-5 mission.

1.1 Relative Navigation Review

There are several examples in the literature of relative navigation for LEO formation flying spacecrafts, ranging from offline theoretical investigations to demonstrations of online filters that are validated using hardware-in-the-loop demonstrations. While all the studies discussed below include absolute spacecraft navigation, only the relative navigation components are elucidated.

In [17], the absolute position and velocity of two spacecrafts are estimated using an iterated extended Kalman filter (EKF) and are directly differenced to obtain the relative state. The state vector contains Chief and Deputy position and velocity, drag coefficients, unmodeled accelerations, clock errors, ionospheric error, user range error, and double difference ambiguity states, while the measurements are comprised of pseudoranges and double difference carrier phases. Unlike more recent work, only the float double difference ambiguities are estimated. This work is unique in that the measurement equations are linearized about the received measurement time, so that the range rate is explicitly included, which avoids the problem of recursively solving for the transmission time. The estimation scheme is validated in both open loop and rendezvous hardware-in-the-loop simulations, and demonstrates an achievable accuracy on the order of cm and mm/s, for relative position and velocity, respectively. For this study, the computations were performed using a desktop computer, which is typically much more powerful than flight hardware for micro- or nanosatellites. A more succinct account of this work can be found in [18].

In [19], a relative navigation filter is designed to operate in real time on flight hardware. The filter processes single frequency, single difference, pseudorange, carrier phase, and Doppler data using a single-point recursive least-squares (kinematic) filter. The state is comprised of relative position and velocity, and clock and frequency error terms. The method is validated in a closed-loop rendezvous scenario with a GPS signal simulator, and demonstrates an accuracy less than 1 m in relative position and on the order of 1 cm/s in relative velocity. It was found in this study that the receiver-measured Doppler shifts are quite noisy, and better results are achievable by using a finite difference approximation of the carrier phase observables.

In [20], a thorough investigation of various relative navigation filters using single-frequency measurements was performed. Least squares, linear and nonlinear Kalman filters, with constant velocity, linear, or nonlinear relative dynamics and single or double difference pseudorange and carrier phase measurements are compared and contrasted to assess which filters perform best with realistic data from a GPS signal simulator. The filter’s state vector is comprised of relative position and velocity, relative clock error and clock error drift, ionospheric error,
and single or double difference ambiguity states. Only float double difference ambiguities are resolved. The actual relative estimation is performed offline, after the data is collected and saved. A unique contribution of this research is the use of an adaptive filter to estimate the process and measurement noise covariance matrices. From this trade study, several important conclusions are made. First, the nonlinear Kalman filters perform much better than the linear filters for baselines greater than 10 km. Second, the nonlinear dynamics model provides the best estimates given large baselines and large filter time steps (> 20 s). Third, the Doppler measurements are not found to have a significant impact on the estimated relative velocity. Lastly, it is found that filters operating with single and double difference measurements have equivalent performance in steady state, while the single difference filters have better transient performance. Overall, it is demonstrated that cm- and mm/s-level relative position and velocity determination is achievable.

In [21], the work performed in [20] was extended in two ways. First, an unscented Kalman filter (UKF) for relative navigation was designed, and it was shown that it outperforms the EKF for large relative baselines (> 1 km) and large filter time steps (> 20 s). Second, the filter was tested in hardware-in-the-loop simulations with active formation control, thus showing that the filters can be operated in real time. However, unlike other work, the navigation and control functions were not executed on representative flight hardware. Nonetheless, this represents, to the best of the author’s knowledge, the first UKF implementation for relative navigation.

In [22], real time relative navigation for a four satellite non-maneuvering formation is demonstrated in hardware-in-the-loop simulations using representative flight hardware. The state vector is comprised of relative position and velocity, ionospheric error, and double difference carrier phase ambiguities, while only double differenced single-frequency carrier phase measurements are used. Two important ideas are introduced here: using ‘pseudo’ relative dynamics to propagate the state vector using the nonlinear equations of motion (which should include at least $J_2$), and using scalar filter state updates rather than matrix-vector updates. It is demonstrated that under a best case scenario, relative position and velocity can be resolved on the order of mm and $\mu$m/s, respectively. It is important to note that these tests were performed using a GPS signal simulator, GPS receivers, and inter-satellite communication hardware, making it one of the most representative demonstrations to date.

In [23], an offline filter for post-processing absolute and relative navigation data from the GRACE mission was designed and implemented. The relative navigation filter design is very rigorous due to the requirement for very high accuracy. The relative state vector is comprised of relative position and velocity, relative clock bias and clock bias drift errors, relative drag and solar radiation pressure coefficients, relative empirical accelerations, ionospheric errors, and single difference ambiguities. Single difference pseudorange and carrier phase observations are used.
The filter state is propagated using ‘pseudo’ relative dynamics as in [22], and employs a high fidelity orbit model including a 120x120 gravity model, third body perturbations, atmospheric drag with a Jacchia ‘71 density model, solar radiation pressure, and solid and ocean tides. This work introduces the idea of adding the double difference ambiguity resolution step external to the filter operation, which allows highly accurate relative state determination without explicit inclusion of double difference ambiguities in the state vector. One important conclusion is that a nonlinear Kalman filter is essential for this problem, and that kinematic filters lead to insufficient accuracy and may even diverge when using real measurement data. Relative positions and velocities are determined to mm- and mm/s-level, which is validated using laser range measurements.

In [24] and [25], cm- and cm/s-level relative position and velocity solutions were obtained in an offline estimator, with uncontrolled vehicles in along-track formations with baselines up to 5000 m. The filter states were comprised of absolute position and velocity of the deputy expressed in a spherical coordinate frame, and double difference ambiguities. Dual-frequency double difference pseudorange, carrier phase, and Doppler measurements generated from a GPS signal simulator were processed offline in an EKF using constant velocity dynamics, and the double difference ambiguities were resolved using the LAMBDA method. No control forces were considered in this work.

In [7], the navigation concept in [22] was extended to provide accurate relative state estimates for closed-loop feedback control applied to the PRISMA mission. One innovation was the use of a single monolithic state vector, in which both absolute and relative satellite states are resolved using single-frequency pseudorange and double difference carrier phase measurements. This scheme has the advantage that the relative state can be accurately determined even when the same GPS satellites are not in view.

In [26], a comprehensive navigation and control scheme for the TanDEM-X mission is presented. Rather than constructing a navigation filter from raw GPS measurements, the single-frequency GPS receiver solutions are used as inputs to an observer, which estimates the differential orbital elements with sub-meter accuracy to provide inputs for the impulsive navigation scheme. The proposed strategy is validated with hardware-in-the-loop simulations using flight hardware.

1.2 Formation Control Review

It suffices to say that the available literature on satellite formation keeping and reconfiguration is vast; the earliest studies date back to the 1960’s, when orbital rendezvous maneuvers were first being investigated. Instead, just a broad overview of the existing work will be presented,
with a focus on the work that is most related to the control strategies presented in this thesis. Formation control schemes can be divided into active and passive control (i.e., control through initial conditions). Active control can be either continuous, or impulsive, and each variant can be in Cartesian or orbital element space. Furthermore, controllers can be either linear, or nonlinear. There is also a distinction between control for formation reconfiguration (or rendezvous), and control for maintaining a formation.

Examples of passive formation control include [27], where analytic corrections to Hill’s initial conditions are found via Melton’s time-explicit relative solution [28] to achieve bounded relative motions with nonlinearities and non-zero eccentricity. Similarly, in [29] conditions for bounded relative motion with eccentric Chief orbit are derived from Carter’s relative solution [30] by enforcing periodicity and by using energy matching constraints. In [31], conditions for bounded relative orbits under $J_2$ disturbances are found by matching the secular drifts of the mean orbital elements. Quasi-periodic relative orbits subject to $J_2$ are presented in [32].

In [33], a continuous control strategy based on Lyapunov stability theory and the classic orbital elements is presented for formation establishment and station keeping. A continuous strategy for formation keeping that minimizes fuel use is given in [32]. Several control strategies using both Cartesian and orbital element feedback based on the Lyapunov method are presented in [34]. In [35], a nonlinear adaptive Lyapunov-based controller is designed for formation keeping. In [36] impulsive/discrete control laws for formation keeping based on Gauss’ variational equations are designed. A twice per orbit impulsive formation maintenance strategy is presented in [37]. In [38], a near-optimal controller is designed for formation keeping, and it is shown that excellent tracking performance is obtained for a much smaller control effort than the baseline LQR controller. The LQR control strategy used in this work is presented in [39], and essentially applies the continuous-time control gain to the discrete system. As shown in Chap. 6, this scheme still leads to stable formations, though often suffering from large fuel cost given large estimation errors.

A comprehensive review of work done on impulsive control for formation reconfiguration up until 1998 is presented in [40], which discusses methods which assume linearity in the relative motion, with both circular and elliptical Chief orbit assumptions. This work also presents a state-transition matrix based approach to reconfiguration. In [41], an analytic solution to the formation reconfiguration problem is presented. In [42] an impulsive minimum energy strategy for formation reconfiguration is presented, which is accurate when subject to relative perturbations. In [43], two-impulse state transition matrix based reconfiguration strategies are investigated, showing that better relative dynamics models lead to more accurate results. A generalized analytic continuous-time reconfiguration solution for linearized relative dynamics is presented in [44]. The reconfiguration problem can also be cast into a numerical optimization
framework, such as in [45] (mixed integer linear programming), [46] (genetic algorithms), or [47] (sequential quadratic programming). The advantage of an optimization-based strategy is that fuel-optimal solutions that satisfy collision avoidance constraints can be found. However, these methods also have the disadvantage that they are not feasible to run in real-time on computationally restricted flight hardware.

1.3 Thesis Overview

In this thesis the formation control and relative navigation as it relates to the CanX-4/-5 mission is discussed. A general overview of the CanX-4/-5 satellites is given in Chap. 2. In Chap. 3, the fundamentals of satellite orbital dynamics are outlined. In Chap. 4, the fundamentals of state estimation and GPS measurements are reviewed. Further, the design of coarse and fine relative navigation filters for the mission is presented. Implementation aspects such as filter initialization and data transfer are also discussed. Building upon Chap. 3, Chap. 5 discusses the formation control laws for the CanX-4/-5 mission that were originally designed in [39]. This includes a review of the discrete and continuous linear quadratic regulator for formation keeping, and an impulsive thrusting scheme for formation reconfiguration. In addition, a novel strategy for accurate, computationally efficient, multiple-impulse formation reconfiguration maneuvers is presented. This robust closed-loop method greatly increases the accuracy of these formation transfers. Lastly, in Chap. 6 the formation control software and simulation environments are outlined, and simulations of the control algorithm performance are presented. Further, hardware tests verifying the functionality and performance of the relative navigation algorithm design are presented.
Chapter 2

CanX-4/-5 Overview

CanX-4/-5 are a pair of identical nanosatellites designed by the Space Flight Lab (SFL) at the University of Toronto Institute for Aerospace Studies. Their design is based on the “generic nanosatellite bus” (GNB) concept, which allows the requirements for numerous different missions to be achieved with a single adaptable nanosatellite template. This in turn reduces design and development time for the overall structure of the satellite and allows one to focus on the mission specific objectives. GNB satellites are 20 cm cubes, weigh approximately 7 kg, can generate at least 5 W of power, and all come equipped with certain standard components, including computers for satellite task management and attitude determination and control, communication antennas, magnetometer and sun sensors for attitude determination, and place for three reaction wheels and magnetorquers for fine/coarse attitude control. In addition, the GNB has room for a payload with restricted mass, volume, and power consumption. This payload could include scientific instruments (e.g. spectrometer, telescope), additional sensors (e.g. GPS antenna, startracker), communication devices such as antennas, or additional computers. Since the CanX-4/-5 mission is primarily a technology demonstration, its payload comprises an additional computer and GPS for precise orbit determination.

The goal of the mission is to demonstrate precise, autonomous formation flying in four different configurations: a 1000 m along-track orbit (ATO), a 500 m ATO, a 50 m projected circular orbit (PCO), and a 100 m PCO. The formation control error is to be within 1 m, which is to be realized through differential carrier phase relative state estimation, accurate to the centimeter level. At least ten orbits will be flown in each formation, though fifty orbits in each formation is desired. They will be launched and commissioned in a joined configuration, and will separate to begin the mission once all satellite systems have been commissioned. The intersatellite separation system (ISS) is the device which initially holds the two satellites together with a mechanical bond that is weakened with an applied potential of 12 V. Upon weakening, a preloaded spring will break the bond and separate the satellites with a relative velocity
somewhere in the range of 0.07 – 0.10 m/s [48].

Figure 2.1: Perspective view of the CanX-4 satellite [1].

Figs. 2.1 and 2.2 give perspective views of the CanX-4 nanosatellite. Each face of the satellite has three solar cells for power generation. Though more solar cells would be desirable, the potential surface area is occupied by the GPS and inter-satellite link antennas. It should be noted that the imager labelled in Fig. 2.2 will likely be replaced with a test unit of a new star tracker specially designed for small satellites. The remainder of the satellites’ components are discussed in more detail below.

2.1 Communication Hardware

Each spacecraft is equipped with VHF, UHF, and S-Band antennas. The VHF antenna is used as a beacon for satellite identification. Four UHF antennas provide near omni-directional coverage for data uplink from the ground station at a rate of about 4000 bits per second (bps). This allows operators to upload new software or instructions to improve mission performance or fulfill science goals. Two S-Band patch antennas mounted on perpendicular faces allow data downlink to the ground station at a rate of about 32-256 kbps. Another two S-Band antennas are mounted on opposing faces, providing omni-directional coverage for inter-satellite
communications at a rate of about 10 kbps up to a maximum range of 5 km [12]. This is termed the inter-satellite link (ISL).

2.2 Onboard Computers

A total of three ARM7 microcontrollers are present on each satellite. One housekeeping computer to manage the overall operation of the spacecraft, one computer dedicated for attitude estimation and control, and a third for GPS-based relative navigation and formation-flying control. These computers have a maximum clock speed of 40 MHz [49]. Each onboard computer (OBC) runs CANOE, or the Canadian nanosatellite operating environment, which handles all task scheduling, routing of inputs and outputs, inter-computer communication, memory management, etc.
2.3 Navigation and Control Hardware

2.3.1 Attitude

Attitude determination sensors include six fine/coarse sun sensor pairs mounted on each face of the satellite, a magnetometer mounted on a boom external to the satellite, and three rate gyros for three-axis angular rate measurements. Attitude actuation is provided by three orthogonal pairs of reaction wheels and magnetorquers. The magnetorquers are used to de-tumble the spacecraft and to dump momentum from the reaction wheels so they never reach saturation, while only the reaction wheels are used to point the spacecraft [50]. Based on preliminary analysis, the spacecraft should have a worst case pointing accuracy of about three degrees RMS, which includes errors due to actuation and state determination [50]. The attitude control system (ACS) will be responsible for all attitude control and determination tasks, including the execution of an algorithm which maximizes GPS constellation coverage, while minimizing the risk of frequently dropped satellites.

2.3.2 Orbit Determination

Each satellite will have one single-frequency NovAtel OEMV-1G GPS receiver and antenna pair, serving to collect absolute position information about each satellite in addition to raw data to be used for relative positioning. The primary reason for selecting these receivers over the OEM4-G2 or OEM4-G2L, which was flown aboard CanX-2 and has undergone extensive testing by a number of groups (see [51] and [52]), is that NovAtel stopped their production. However, it has been shown by several authors (see Sec. 1.1) that single frequency receivers are adequate for providing accurate relative navigation solutions. The OEMV-1G receivers have the additional benefit of 14 channels instead of 12, and consume much less power than other GPS receivers in its class [53].

Since the GPS antennas will be mounted on opposite sides of the joined satellites, there will be no opportunity to test the carrier phase differential navigation algorithm prior to separation, since the same satellites will never be in view. However, it will be possible to calibrate the coarse relative navigation strategy, since both receivers will form independent state estimates that can be subtracted to yield the relative position, which is known precisely.

2.3.3 Formation Control

The Canadian Nanospace Propulsion System (CNAPS) is the cold gas propulsion system used onboard CanX-4/-5. It consists of four independently controlled thrust nozzles mounted in a cruciform formation on one face of the satellite. Its fuel is liquid SF$_6$, and the tank has
a maximum capacity of 300 mL. Each nozzle will provide a constant thrust magnitude of 5 mN, though this should be configurable through back-pressure regulation. Current estimates indicate a total $\Delta V$ of approximately 13 m/s per satellite. The thrust system will also perform the auxiliary task of dumping reaction wheel momentum when it can. This algorithm will reside aboard the attitude computer, and shall not impact the formation control in any way. CNAPS will be a passive system, which simply thrusters when it is told to and provides a report of each completed thrust.
Chapter 3

Orbital Dynamics

In this chapter, the fundamental orbital dynamics of the formation flying problem are reviewed. The information given here is provided in more detail in [54] and [55].

3.1 Time Systems

This section gives a brief description of the various time systems associated with satellite orbit estimation.

- **Universal Time** (UT) - A time system based on the motion of a fictitious ‘mean sun’ which passes across the sky at a fixed rate. By default this time system accounts for the gradual slowing of the Earth’s rotation. A specific realization, UT1, is used to compute the Greenwich apparent sidereal time which is important for switching between coordinate frames.

- **International Atomic Time** (TAI) - An atomic time scale whose second is defined by a fixed number of oscillations of the cesium-133 atom. Data from hundreds of stations are statistically combined to maintain this time scale.

- **Coordinated Universal Time** (UTC) - Another atomic time scale, which lags TAI by some number of leap seconds. It is periodically adjusted with leap seconds so that its calendar days remain consistent with the mean solar time, UT. The International Earth Rotation Service (IERS) publishes the offsets of UTC from UT1.

- **Terrestrial Time** (TT) - The theoretical time scale associated with celestial motions as observed from Earth. It leads TAI by exactly 32.184 s.

- **GPS Time** - An atomic time system used by the GPS satellites. It was initialized to 0\(^h\) UTC on Jan. 6 1980, and its offset from UTC is given by a fixed integer number of
3.2 Coordinate Frames

In this section, the major coordinate frames relevant to this problem are discussed. Refer to Fig. 3.1 for a graphical representation of these frames.

3.2.1 Geocentric/J2000 Inertial Frame

The Geocentric Inertial (GCI) frame has its origin at the center of the Earth, with its X-axis pointing towards the vernal equinox (the constellation Aries), its Z-axis pointing towards the geographic north pole, and its Y-axis completing the orthogonal triad. In actuality, there are many different realizations of the GCI frame. A popular realization is the J2000 Inertial Frame, whose X-axis points at the vernal equinox at precisely 12:00 January 1st 2000, TT.

3.2.2 Earth-Centered-Earth-Fixed Frame

The Earth-Centered-Earth-Fixed frame is a non-inertial frame in which all GPS-based navigation is performed. Its origin is at the center of the Earth, with its X-axis pointing towards the Greenwich prime meridian, its Z-axis pointing towards the geographic north pole, with its Y-axis completing the orthogonal triad. This reference frame rotates at the angular velocity of the Earth, which is approximately constant. To be consistent with the GPS constellation, the WGS-84 realization of this frame is used throughout this work.

3.2.3 Local Orbital/Hill Frame

In this work, the local orbital (LO) frame is used for all control calculations. It is a rotating reference frame centered on the Chief satellite, whose axes are given by the following relationships:

\[
\hat{X}_L = \frac{R}{\|R\|}, \quad \hat{Z}_L = \frac{R \times V}{\|R \times V\|}, \quad \hat{Y}_L = \hat{Z}_L \times \hat{X}_L, \quad (3.1)
\]

where \(R\) and \(V\) are the absolute position and velocity of the Chief. When the Chief position and velocity vectors used to construct this reference frame are expressed in an inertial frame, the local orbital frame is known as the Hill frame. Under the assumptions of circular Chief orbit and small relative distances between Chief and Deputy, it can be shown that Hill’s equations of relative motion hold for any local orbital frame. This fact will be exploited later when considering the control and estimation algorithms.
3.2.4 Spacecraft Body Frame

This is the frame in which spacecraft measurements (magnetometer, body rates, sun vectors, GPS) are expressed. Its X, Y, and Z axes are perpendicular to the geometric faces of the satellite, and are not assumed to be aligned with the principal axes of inertia [50]. The body frame orientation is typically defined with respect to the inertial or local orbital frame.

![Diagram of coordinate frames](image)

Figure 3.1: The three major coordinate frames. The J2000 frame is shown with a solid black line, the ECEF frame is shown with a dashed line, and the Local Orbital frame is shown with a dotted line.

3.2.5 Coordinate Transformations

This section discusses the transformations between the coordinate frames used in this work.

3.2.5.1 J2000 to ECEF

This transformation is otherwise known as the FK5 reduction, and consists of four rotations. The first transformation accounts for the precession of the Earth’s axes away from J2000 due to the gravitational influence of the Sun and Moon. The second transformation accounts for the nutation of the Earth’s axes due to similar effects. The third transformation accounts for the rotation of the Earth, and is a function of the Greenwich Apparent Sidereal Time (GAST). The last transformation accounts for minute perturbations to the Z-axis, known as Polar Motion. In
the flight code, this transformation is implemented using the algorithm in [54], but employing only the 20 most dominant terms in the 1980 IAU theory of nutation. Further information on the FK5 reduction can be found in [54], [55], and [56]. The full expression can be written as

$$ R_E = (\Phi SNP) R_I = C_{EI} R_I, \quad (3.2) $$

where $R_j$ is the position vector expressed in frame $j$, and $\Phi$, $S$, $N$, and $P$ are the rotation matrices for polar motion, sidereal time, nutation, and precession, respectively.

The time derivative of $C_{EI}$ is approximated by

$$ \dot{C}_{EI} = \Phi \dot{SNP} = -\Phi \omega_{\oplus \times} SNP, \quad (3.3) $$

where $\omega_{\oplus} = \begin{bmatrix} F_{\oplus E}^T & 0 & 0 \end{bmatrix}$ is the angular velocity of the Earth. Here it is assumed that the precession, nutation, and polar motions occur on a much slower time scale than the Earth’s rotation, so that their time derivatives are negligible.

### 3.2.5.2 ECEF to Local Orbital

First, one must evaluate Eq. (3.1) using inertial position and velocity vectors expressed in the ECEF frame. Given that the navigation outputs are position and velocity as seen in the ECEF frame ($R_E$ and $\dot{R}_E$), the inertial velocity is given by $V_E = \dot{R}_E + \omega_{\oplus \times} R_E$, where $\omega_{\oplus}$ is the angular velocity of the Earth, which is taken to be a constant for simplicity. Then, the rotation matrix from the ECEF to the LO frame is given by

$$ C_{LE} = \begin{bmatrix} \hat{X}_{L,E} & \hat{Y}_{L,E} & \hat{Z}_{L,E} \end{bmatrix}^T, \quad (3.4) $$

where the $E$ in the subscript emphasizes the fact that these quantities (given in Eq. (3.1)) are evaluated in the ECEF frame. The time derivative of this rotation matrix (needed for transforming velocities) is given by

$$ \dot{C}_{LE} = -\omega_{LE}^\times C_{LE}, \quad (3.5) $$

where $\omega_{LE}$ is the instantaneous angular velocity of the LO frame with respect to the ECEF frame, and is given by [19]

$$ \omega_{LE} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T \frac{\dot{Y}_{L,E}^T V_E}{\|R_E\|}. \quad (3.6) $$
3.2.5.3 Body to J2000

It is assumed that these quantities will be calculated as part of the regular operation of the attitude determination and control subsystem, and that they will be made available to the payload computer upon request. Based on [50], the attitude determination error is assumed to be on the order of 1°, while the rate sensor error is assumed to be about 0.05°/s per axis.

3.3 Equations of Motion

The Keplerian equation of motion for the spacecraft about a body is written as

$$\ddot{r} - \frac{\mu}{\|r\|^3} r = 0,$$  \hspace{1cm} (3.7)

where $r$ is the spacecraft position vector and $\mu$ is the Earth’s gravitational parameter. The gravitational force expressed in this equation assumes the Earth is a point mass. The solution to this equation is well-known, and describes an ellipse with the Earth at one focus. It can be written as

$$r = a \frac{1 - e^2}{1 + e \cos f},$$  \hspace{1cm} (3.8)

where $a$ and $e$ are the semi-major axis and eccentricity of the ellipse, and $f$ is the true anomaly, which gives the angular position of the satellite in the orbit, as measured from the periapsis. Another result of Eq. (3.7) is that the angular momentum of an orbit is constant, which is expressed as $h = r^2 \dot{f}$.

3.3.1 Forces

Aside from the central-body gravitational force acting on a satellite, there are many other perturbative forces acting upon the satellite which cause it to deviate from its nominal Keplerian trajectory. Subject to these additional forces, the results from the previous section no longer hold. For satellites in LEO (such as CanX-4/-5), the most important effects are the high order gravitational effects arising from the inherently “lumpy” shape of the Earth, atmospheric drag [55], third-body effects, and the control forces. Brief descriptions of these forces are provided below.
3.3.1.1 Earth Gravity

Referring to Fig. 3.2, the gravitational potential energy on a satellite due to an infinitesimal mass $dm$ located on Earth is given by

$$dV = -\frac{Gm_s dm}{\|r - r'\|} = -\frac{Gm_s \rho (r') dV}{\|r - r'\|},$$

where $dV$ is an infinitesimal volume element, $G$ is the universal gravitational constant, $\rho (r')$ is the density as a function of position, and $m_s$ is the satellite mass. Integrating around volume of the Earth yields the potential due to the entire Earth. Then, the gravitational force on a satellite is given by the negative gradient of the potential

$$f_{g} = -\nabla V,$$

where the potential can be expanded into spherical harmonics as [54]

$$V = \frac{\mu_\oplus}{r} \left[ 1 + \sum_{n=2}^{\infty} \sum_{m=0}^{n} \left( \frac{R_\oplus}{r} \right)^n P_{n,m} \sin \phi \left( C_{n,m} \cos (m\lambda) + S_{n,m} \sin (m\lambda) \right) \right].$$

In Eq. (3.11), $\mu_\oplus$ is the Earth’s gravitational parameter, $R_\oplus$ is the radius of the Earth, $r$ is the magnitude of the satellite’s position vector, $\lambda$ is the longitude, $\phi$ is the latitude, $P_{n,m}$ are the associated Legendre polynomials, and $C_{n,m}$ and $S_{n,m}$ are experimentally determined coefficients which are particular to each gravity model. The coefficients $n$ and $m$ are termed the gravity model’s degree and order, respectively. The zonal harmonics ($J_n$ terms) are related to this
expression by: $-C_{n,0} = J_n$.

### 3.3.1.2 Third Body Gravity

Again referring to Fig. 3.2, the gravitational acceleration due to a third body with position vector $\mathbf{r}_{s}$ is given by

$$\mathbf{a}_{tb} = -\mu_s \left[ \frac{\mathbf{r} - \mathbf{r}_s}{\|\mathbf{r} - \mathbf{r}_s\|^3} - \frac{\mathbf{r}_s}{\|\mathbf{r}_s\|^3} \right],$$

(3.12)

where $\mu_s$ is the gravitational parameter of the third body (e.g. Sun, moon, Venus, Jupiter).

Note that the acceleration of the Earth due to the third body is subtracted from that of the satellite. This is because the third body force acts on the Earth-satellite system, and we must subtract to isolate its effect on the satellite.

### 3.3.1.3 Atmospheric Drag

Based on the classical development of drag force models from aeronautics, the acceleration due to atmospheric drag is given by

$$\mathbf{a}_{\text{drag}} = -\frac{1}{2} \rho C_D A m_s \mathbf{V}_{\text{rel}}^T \mathbf{V}_{\text{rel}} \mathbf{V}_{\text{rel}},$$

(3.13)

where $\rho$ is the atmospheric density, $C_D$ is the drag coefficient ($\approx 2.2$), $A$ is the total frontal area projected into the velocity direction, $m_s$ is the satellite mass, and $\mathbf{V}_{\text{rel}}$ is the velocity relative to the rotating atmosphere, which is given by

$$\mathbf{V}_{\text{rel}} = \mathbf{V}_I - \omega_\oplus \times \mathbf{R}_I,$$

(3.14)

where $\mathbf{R}_I$ and $\mathbf{V}_I$ are the position and velocity of the satellite. The main difficulty in calculating a drag force is obtaining an accurate estimate of the density, which is a function of altitude, temperature (solar activity), distribution of chemical elements, and magnetic field. For simplicity, in this study the density is approximated using an exponential model [54]:

$$\rho = \rho_0 e^{-\frac{(h-h_0)}{H}},$$

(3.15)

where $h$ is the satellite altitude, and $\rho_0$, $h_0$, and $H$ are tabulated constants that vary with altitude.
3.3.1.4 Control Forces

In this work, a simplistic model of the applied control forces is used. In general, the control thrusts are treated as finite width pulses, with the transient regions ignored. The general relationship used to apply calculated control accelerations is given by [39]

\[
\frac{\|u_c\|}{U_{max}} = \frac{t_{on}}{T_{pwm}},
\]  

where \(u_c\) is the calculated control acceleration per unit mass, \(U_{max}\) is the maximum acceleration per unit mass available to the satellite, \(t_{on}\) is the desired thruster on-time, and \(T_{pwm}\) is the pulse width modulation (PWM) period. In this case, \(u_c\) is calculated from the station-keeping LQR control law given in Sec. 5.1.1.

For impulsive reconfiguration thrusts, which have units of m/s, the on-time is calculated directly as

\[
t_{on} = \frac{\Delta V_c}{U_{max}},
\]

where \(\Delta V_c\) is the controlled change in velocity. This relationship effectively gives the width of the finite pulse of amplitude \(U_{max}\) that matches the \(\Delta V\) provided by the impulsive thrust.

3.3.1.5 Other Forces

Additional perturbations arise due to gravitational forces from solid and ocean tides, solar radiation pressure, and general relativistic effects. These effects can be safely ignored in most LEO applications.

3.3.2 Orbital Elements

The classical orbital elements are a set of six independent quantities which, equivalently to position and velocity, fully specify the orbit of a satellite. They are typically used rather than position and velocity vectors because they have a nicer geometric interpretation, as shown in Fig. 3.3. The orbital elements can be computed from position and velocity at any point in time using Algorithm 9 in [54], which is a non-singular implementation. The first orbital element is the semi-major axis of the orbit ellipse, denoted by \(a\). Next is the orbit’s eccentricity, \(e\). Third is the inclination of the orbital plane with respect to the equatorial plane, given by \(i\). The longitude/right ascension of the ascending node, \(\Omega\), is the angle from the X-axis at which the orbit passes from below to above the equatorial plane. The line formed by the intersection of the orbital and equatorial plane is called the ‘line of nodes’ and is denoted by \(\hat{n}\). The argument of the perigee, \(\omega\), is the angle from the line of nodes to the periapsis of the orbit, measured in the orbital plane. Lastly, the initial mean anomaly \(M_0\) is the mean anomaly at the initial
time $t_0$. The mean anomaly is the way of describing the satellite’s current position in an orbit. It increases at a fixed rate, as if the satellite was travelling uniformly along a circular orbit of radius $a$. The mean anomaly at any time is given by

$$M = M_0 + n(t - t_0),$$

where $n = \sqrt{\frac{\mu}{a^3}}$ is the mean orbital rate.

If one were to calculate the orbital elements from position and velocity once every second, one would find that there exists both short and long period variations. For this reason, the classical orbital elements are often called the osculating elements. It is possible to remove the oscillatory components through certain transformations [34], resulting in the mean orbital elements. Under the Keplerian orbit assumption, the mean elements remain constant; there is no variation, on average. However, due to the action of the perturbative forces discussed in Sec. 3.3.1, the mean components of the orbital elements experience a secular drift. Most notably, the $J_2$ zonal harmonic induces a secular drift in $\Omega$, $\omega$, and $M_0$, while the cumulative effect of atmospheric drag reduces the orbital eccentricity and the semi-major axis. Unless otherwise stated, the osculating elements are used in this thesis.
3.4 Relative Dynamics

As shown in Fig. 3.4, the relative position vector is defined as \( \Delta r = R_d - R_c \). Then using Eq. (3.7), the un-perturbed equations of relative motion are written as

\[
\Delta \ddot{r} = -\mu \left( \frac{R_d}{\|R_d\|^3} - \frac{R_c}{\|R_c\|^3} \right) = -\mu \left( \frac{R_c + \Delta r}{\|R_c + \Delta r\|^3} - \frac{R_c}{\|R_c\|^3} \right). \tag{3.19}
\]

From here it is typical to assume that \( \|\Delta r\| \ll \|R_c\| \), so that Eq. (3.19) can be linearized about small \( \Delta r \). This assumption holds for the CanX-4/-5 mission since the relative separation shall be less than 5 km. Performing the linearization and dropping terms of order \( \Delta r^2 \), one can write

\[
\Delta \ddot{r} = -\mu \left( \frac{R_c T_{R_c} I_{3 \times 3} - 3 R_c R_c^T}{\|R_c\|^5} \right) \Delta r. \tag{3.20}
\]

Then, using the expression for the time derivative of a vector in a rotating frame, we can write

\[
\ddot{r} = \ddot{r}^0 + \omega \times \dot{r} + 2 \omega \times \dot{r} + \omega \times \omega \times r, \tag{3.21}
\]

where \( \omega \) is the angular velocity of the rotating frame with respect to the GCI frame. If we then express all quantities in the LO frame so that \( \omega = \begin{bmatrix} 0 & 0 & \dot{f} \end{bmatrix}^T \), \( R_c = \begin{bmatrix} -R_c & 0 & 0 \end{bmatrix}^T \), and \( \Delta r = \begin{bmatrix} x & y & z \end{bmatrix}^T \), using Eq. (3.21) we can write Eq. (3.20) as

\[
\begin{align*}
\ddot{x} &= \left( \frac{2\mu}{R_c^5} + \dot{f}^2 \right) x + \dot{f} y + 2 \dot{f} \dot{y} \\
\ddot{y} &= \left( -\frac{\mu}{R_c^5} + \dot{f}^2 \right) y - \dot{f} x - 2 \dot{f} \dot{x} \\
\ddot{z} &= -\frac{\mu}{R_c^5} z.
\end{align*} \tag{3.22}
\]

Note that these equations are linear in \( \Delta r \) and can be written in first-order form as

\[
\dot{x} = A(t)x(t). \tag{3.23}
\]

The solution to these equations are well known and are given by [29]

\[
\begin{align*}
x(f) &= \sin f \left( d_1 e + 2d_2 e^2 H(f) \right) - \cos f \left( \frac{d_3}{\rho} e + d_3 \right) \\
y(f) &= \left( d_1 + \frac{d_3}{\rho} + 2d_2 e H(f) \right) + \sin f \left( \frac{d_4}{\rho} + d_3 \right) + \cos f \left( d_1 e + 2d_2 e^2 H(f) \right) \\
z(f) &= \sin f \left( \frac{d_5}{\rho} \right) + \cos f \left( \frac{d_6}{\rho} \right)
\end{align*} \tag{3.24}
\]

where \( \rho = 1 + e \cos f \), the \( d_i \) are constants of integration, and

\[
H(f) = \int_0^f \frac{\cos f}{\rho^2} df = -(1 - e^2)^{-2.5} \left( 1.5 e E - (1 + e^2) \sin E + e \sin 2E + d_H \right),
\]

and
where \( E \) is the eccentric anomaly, and \( d_H \) is obtained from \( H(f_0) = 0 \). If the Chief satellite is assumed to be in a circular orbit, then Eq. (3.22) reduces to

\[
\begin{align*}
\ddot{x} &= 2n\dot{y} + 3n^2x \\
\ddot{y} &= 2n\dot{x} \\
\ddot{z} &= -n^2z,
\end{align*}
\]

where \( n \) is the mean orbital rate. These are the Hill-Clohessy-Wiltshire (HCW) equations of motion, and their solution is given by [34]

\[
\begin{align*}
x(t) &= d_1 \cos (nt + \alpha) + d_2 \\
y(t) &= -2d_1 \sin (nt + \alpha) - 1.5d_2nt + d_3 \\
z(t) &= d_4 \cos (nt + \beta),
\end{align*}
\]

where the \( d_i \) are constants of integration, and \( \alpha \) and \( \beta \) are phase angles. By selecting the initial conditions appropriately, i.e., \( d_2 = 0 \), the relative motion described by the HCW equations can be made bounded. This is discussed further in Sec. 5.1.4.

![Figure 3.4: Relative position definition.](image)

### 3.4.1 State Transition Matrices

Simply put, a state transition matrix (STM) is a matrix that maps a state from one time to another. For instance, in absence of control, the relative state at some time \( t \) can be determined given the initial state \( x(t_0) \), and the state transition matrix from \( t_0 \) to \( t \). That is,

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

A good review of the mathematical properties of the state transition matrix is provided in [57]. They are typically derived using the linearly independent solutions of the ordinary differential
equation $\dot{x}(t) = A(t)x(t)$. In this work, three different state transition matrices are used: Hill-Clohessy-Wiltshire (HCW) [40], Ankersen-Yamanaka (AY) [58], and Gim-Alfriend (GA) [59]. The HCW STM is the simplest of the three, and is derived under the same assumptions as the HCW equations of relative motion. Consequently, it is also the least accurate at predicting the relative motion in a more realistic scenario with non-zero eccentricity and other orbital perturbations [60]. The AY STM assumes small relative distances between the two spacecraft but any Chief orbit. Unlike other elliptical relative motion theories, it remains non-singular for all Chief eccentricity values. Finally, the most accurate is the GA STM, which assumes a small relative distance, but is general enough to include any eccentricity, and the mean effect of the $J_2$ perturbation to first order.
Chapter 4

Satellite State Estimation

This chapter outlines the methods for estimating the Deputy satellite’s state vector in the LO frame. There are two different positioning methods used, which will be referred to as coarse mode and fine mode. Coarse mode (Sec. 4.2) is a method which takes the difference between two absolute state estimates to estimate the relative state. In contrast, fine mode (Sec. 4.3) uses raw GPS observables to directly estimate the relative satellite state. Coarse mode is much less accurate than fine mode [20], but holds the advantage that it can be used even when the Chief and Deputy spacecraft are locked onto different GPS satellites. Fine mode can only be used when the two satellites are locked onto at least 4 of the same GPS satellites. Prior to discussing coarse and fine navigation modes, the extended Kalman filter (EKF) is reviewed, since it is used exclusively for the relative navigation tasks. Since the GPS constellation works with the WGS-84 realization of the ECEF frame [61] for its coordinate system, from this point it is assumed that all positions and velocities are expressed as seen in the ECEF frame.

4.1 Extended Kalman Filter

The general idea of the EKF is to obtain a recursive estimate for a state, $x_k$ whose dynamics are governed by nonlinear equations of motion, using measurements, $y_k$, that are nonlinearly related to the state. This is contrary to the Kalman filter (KF), where the dynamic and measurement processes are governed by linear time-varying processes. In discrete form, the measurement and motion equations are written as

$$
x_k = f(x_{k-1}, u_{k-1}, w_{k-1})
$$
$$
y_k = h(x_k, v_k),
$$

(4.1)

where the subscript ‘k’ indicates the time index, $x_k \in \mathbb{R}^n$ is the state, $u_k$ is the control force, $w_k$ is the process noise, $y_k \in \mathbb{R}^m$ is the measurement, and $v_k$ is the measurement noise. The
discrete, rather than continuous, form of these equations will be used since all real estimation systems are actually implemented in discrete time. Unlike the KF, the EKF is not an optimal estimator [62]. Rather, it is an approximation of the Bayes filter derived under the assumption that the probability density function (PDF) of $x_k$ is normally distributed, i.e.,

$$x_k \sim \mathcal{N}\left(\hat{x}_k, \hat{P}_k\right).$$ (4.2)

In addition, it is assumed that the process and measurement noise are normally distributed with zero mean, and are white noise sources. Specifically, these distributions are given by

$$w_k \sim \mathcal{N}(0, Q_k)$$
$$v_k \sim \mathcal{N}(0, R_k).$$ (4.3)

It should be noted that an underlying assumption of the EKF is that the state has the Markov property, i.e., the state $x_{k+1}$ is solely a function of quantities at time ‘k’: $x_k$, the control $u_k$, and noise $w_k$. While this allows the filter to take on a recursive form, it also means some accuracy is lost because in reality, $x_{k+1}$ will be a function of all the states, controls and disturbances that came before it. There are two steps to the EKF, the prediction step and the correction step. The prediction step takes the current state estimate and projects it forward in time to predict what it will be at the next instant a measurement is made. Then, the difference between the predicted measurement and the actual measurement, otherwise known as the innovation, is used to update the state estimate during the correction step. The prediction step is given by [62]

$$\hat{x}_k^- = f\left(\hat{x}_{k-1}^+, u_{k-1}, 0\right)$$
$$\hat{P}_k^- = F_{x,k} \hat{P}_{k-1}^+ F_{x,k}^T + F_{w,k} Q_k F_{w,k}^T,$$ (4.4a)

and the correction step is given by

$$\hat{x}_k^+ = \hat{x}_k^- + K_k \left( y_k - h\left(\hat{x}_k^-, 0\right)\right)$$
$$\hat{P}_k^+ = (I - K_k H_{x,k}) \hat{P}_k^-,$$ (4.5a)

where

$$F_{x,k} = \frac{\partial f}{\partial x_{k-1}}|_{\hat{x}_{k-1}^+, u_{k-1}, 0}$$
$$F_{w,k} = \frac{\partial f}{\partial w_{k-1}}|_{\hat{x}_{k-1}^+, u_{k-1}, 0}$$
$$H_{x,k} = \frac{\partial h}{\partial x_k}|_{\hat{x}_k^-, 0}$$
$$H_{v,k} = \frac{\partial h}{\partial v_k}|_{\hat{x}_k^-, 0}$$ (4.6)

and the Kalman gain matrix $K_k$ is given by

$$K_k = \hat{P}_k^- H_{x,k}^T \left(H_{x,k} \hat{P}_k^- H_{x,k}^T + H_{v,k} R_k H_{v,k}^T\right)^{-1}.$$ (4.7)
While not strictly true, it is herein assumed that the process and measurement noise $\mathbf{w}_k$ and $\mathbf{v}_k$ are merely additive rather than propagated nonlinearly through the motion and measurement equations $\mathbf{f}(\cdot, \cdot, \cdot)$ and $\mathbf{h}(\cdot, \cdot)$. This assumption means that $\mathbf{F}_{w,k}$ and $\mathbf{H}_{v,k}$ equal the identity matrix, thereby simplifying the EKF equations considerably. The main motivation behind this assumption for the CanX-4/-5 mission is to reduce the computational cost of the navigation filter. Note however, that the difference can be made up by properly designing the $\mathbf{Q}_k$ and $\mathbf{R}_k$ matrices.

### 4.1.1 Inverse-Free Measurement Update

Based on Eq. (4.7), the measurement update step requires the inverse of an $m \times m$ matrix. For computationally restricted systems such as CanX-4/-5, it may be undesirable to perform this inverse at each time step. Fortunately, it is possible to obtain the Kalman gain matrix with $m$ sequential updates requiring only division and outer products to avoid the problem of taking large matrix inverses [63]. The sequence of updates is algebraically equivalent to the single matrix update if the measurement noise covariance matrix $\mathbf{R}_k$ is diagonal, i.e., the measurements are uncorrelated. Even if this is not the case, it is possible to linearly transform the measurements such that they are uncorrelated, and then perform the sequential update. This process is discussed in detail in [63]. Improved performance is observed since the sequential update algorithm is $O(m^2)$, while the traditional update is $O(m^3)$ owing to the matrix inverse.

To show that the sequential scalar and matrix EKF updates are equivalent, it is much easier to work with the inverse covariance form of the EKF, also called the information filter. The information filter is equivalent to the form the EKF presented in Section 4.1, except it uses the inverse of the covariance matrix $\hat{\mathbf{P}}_k$. First, the equivalence between the two forms of the covariance update is demonstrated. Then, it is shown that using $m$ scalar updates, the covariance matrix can be updated equivalently to using a single matrix update. For simplicity, the subscript ‘$k$’ and the $(\cdot)$ identifiers are omitted from the following derivation. Beginning with Eq. (4.7) and post-multiplying by $(\mathbf{HP} - \mathbf{H}^\top + \mathbf{R})$ we obtain

$$
\mathbf{K} (\mathbf{HP} - \mathbf{H}^\top + \mathbf{R}) - \mathbf{P}^{-1} \mathbf{H}^\top = 0
$$

$$
- (\mathbf{I} - \mathbf{KH}) \mathbf{P}^{-1} \mathbf{H}^\top + \mathbf{KR} = 0.
$$

Then using Eq. (4.5b), we obtain

$$
\mathbf{K} = \mathbf{P}^+ \mathbf{H}^\top \mathbf{R}^{-1}.
$$

(4.8)

Substituting this back into Eq. (4.5b), factoring out $\mathbf{P}^+$, pre-multiplying by $(\mathbf{P}^+)\mathbf{H}^\top \mathbf{R}^{-1}$, and post-
multiplying by \((P^-)^{-1}\) results in

\[
(P^+)^{-1} = (P^-)^{-1} + H^T R^{-1} H, \tag{4.9}
\]

which is the information filter covariance update equation.

**Claim:** If \(R\) is a diagonal matrix, i.e., the measurements are uncorrelated, the corrected covariance matrix \(P^+\) can be calculated one measurement at a time using the rows of \(H\). The state correction step shown in Eq. (4.5a) can then be performed, without the matrix inverse. This process is fully equivalent to the matrix update (Eqs. (4.7) and (4.5)).

**Proof:** Let \(H \in \mathbb{R}^{m \times n}\), \(P \in \mathbb{R}^{n \times n}\), \(R^{-1} \in \mathbb{R}^{m \times m}\), \(P_0 = (P^-)^{-1}\), and \(P_m = (P^+)^{-1}\). Further, let \(R^{-1} = \text{diag}\{r_1, r_2, \cdots, r_m\}\), and

\[
H = \begin{bmatrix}
h_{11} & \cdots & h_{1n} \\
\vdots & \ddots & \vdots \\
h_{m1} & \cdots & h_{mn}
\end{bmatrix}.
\]

Lastly, let \(h_i^T = [h_{i1} \cdots h_{in}]\) denote the \(i^{th}\) row of \(H\), and \(P_i\) denote the \(i^{th}\) sequential update of \(P\) using the \(i^{th}\) measurement. Then, using Eq. (4.9) with single rows of \(H\)

\[
P_m = P_{m-1} + r_m h_m h_m^T \\
= P_{m-2} + r_{m-1} h_{m-1} h_{m-1}^T + r_m h_m h_m^T \\
= \cdots = P_0 + \sum_{k=1}^m r_k h_k h_k^T. \tag{4.10}
\]

This is equivalent to updating the covariance matrix using \(H\) all at once since

\[
\sum_{k=1}^m r_k h_k h_k^T = H^T R^{-1} H,
\]

which can be seen from the fact that the \((p, q)\) entry in both matrices equals \(\sum_{k=1}^m h_{kp} r_k h_{kq}\).

However, since it was shown that Eq. (4.9) is equivalent to Eqs. (4.7) and (4.5b), it must be possible to update the covariance matrix using the following procedure:

1. Let \(P_0 = \hat{P}_k^-\). Go to to step 2 with \(i = 1\).
2. Calculate \(k_i = P_{i-1} h_i^T (h_i^T r_i h_i)^{-1}\).
3. Calculate \(P_i = (I - k_i h_i^T) P_{i-1}\).
4. Check if \(i = m\). If false, \(i = i + 1\) and go to step 2. If true, go to step 5.
5. Calculate the gain matrix using Eq. (4.8) with \(P^+ = P_m\).
6. Update state estimate using Eq. (4.5a).

Note that in the linear case, it is possible to update the state along with the covariance in step 4 and stop the procedure when \( i = m \). However, this procedure of sequential updates is not, to the author’s knowledge, equivalent to the matrix update in the nonlinear case due to the \( h(x_k^-, 0) \) term in the correction step.

4.2 Coarse Mode EKF

Since the CanX-4/-5 mission goal is to control the satellites in formation flight to within 1 m of their reference trajectories, some method of determining the current relative state is required. The coarse mode EKF was designed to dynamically filter the estimated states coming from the GPS receivers [39], which are calculated using purely kinematic positioning methods [64]. By “purely kinematic methods” it is meant that the receiver utilizes a special processing technique which does not require the inclusion of system dynamics to obtain an accurate state estimate. For the two satellites, a total of 12 states are required to fully describe the system (six position, six velocity). After the filtering prediction and correction steps, the relative position and velocity are obtained by taking the direct difference between the two absolute state vectors: \( \Delta r = R_d - R_c, \Delta v = V_d - V_c \). The coarse EKF removes some of the noise characteristics from the “measured” position and velocity to obtain a better relative state estimate than possible without the dynamic filtering. The state vector is given by

\[
x_k = [R_c^T \ V_c^T \ R_d^T \ V_d^T]^T_k,
\]

(4.11)

where \( R \) and \( V \) are position and velocity, respectively. Again, it is assumed that these quantities are expressed as seen in the ECEF frame. The measurement model is linear, and is given by

\[
y_k = Hx_k + v_k.
\]

(4.12)

These measurements will be extracted from the BESTXYZ log of the GPS receiver, which is received at a fixed rate. Included in the BESTXYZ are flags giving the reliability of the solution (P-Sol status and V-Sol status), and estimates of the standard deviation on each state, which come from the receiver’s positioning filter. The position and velocity status flags indicate when these values can be used inside the coarse filter. Any value aside from zero indicates an unreliable estimate. Based on these flags, the dimension of \( H \) is varied. For example, if the
Deputy velocity is unreliable the design matrix is written as
\[ H = \begin{bmatrix} I_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & I_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & I_{3 \times 3} & 0_{3 \times 3} \end{bmatrix}. \] (4.13)

However, if the Chief velocity is unreliable the design matrix is written as
\[ H = \begin{bmatrix} I_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & I_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & I_{3 \times 3} \end{bmatrix}. \] (4.14)

If a position measurement is not reliable, that satellite’s state is not updated since the position is not observable from a velocity measurement alone. If no measurements are available at a particular time step, there is no correction performed so that
\[ \hat{x}_k^+ = \hat{x}_k^- \quad \hat{P}_k^+ = \hat{P}_k^- . \]

This ensures that the covariance estimate increases with no measurements, i.e., our confidence in the solution decreases.

Within the EKF correction step, the measurement noise covariance matrix is assumed to be diagonal, which is not strictly true, since the measured states of the Chief and Deputy will be correlated in some way. This does, however, greatly simplify the correction step since the method of Section 4.1.1 can be used. Moreover, the exact correlation cannot be explicitly determined from the given information, thus the inclusion of correlated noise would be somewhat arbitrary. The diagonals of the measurement covariance matrix, \( R_k \), are chosen to be the squares of the standard deviation values extracted from the BESTXYZ log. This ensures that the correct weighting is applied when updating the state estimate. The Chief and Deputy motion models are given separately as
\[ \ddot{R}_E = -\mu \frac{R_E}{||R_E||^3} + f_p(R_E) + u - 2\omega_\oplus \times \dot{R}_E - \omega_\oplus \times \omega_\oplus \times R_E , \quad \dot{x} = g(x, t) . \] (4.15)

where \( x = \left[ R_E^T, \dot{R}_E^T \right]^T \), \( \mu \) is the Earth’s gravitational parameter, \( f_p(R) \) is the perturbative force acting on the satellite due to higher order gravitational effects (no other effects mentioned in Section 3.3.1 are used), \( u \) is the control force (which is 0 for the Chief spacecraft), and \( \omega_\oplus = \left[ 0 \ 0 \ \omega_\oplus \right]^T \) is the angular velocity of the ECEF frame with respect to the GCI frame, which is assumed to be constant. The flight code implementation of \( f_p(R_E) \) uses the recursive
algorithm in [55] to calculate the gravitational acceleration due to an aspherical primary up to a maximum degree and order of 20.

Eq. (4.15) is integrated with a fourth-order Runge-Kutta scheme, which is a popular choice for onboard navigation schemes [65]. This four-stage numerical integration scheme can be written as

\[ x_{k+1} = x_k + \frac{\Delta t}{6} (k_1 + 2k_2 + 2k_3 + k_4), \]  

(4.16)

where

\[ k_1 = g(x_k, t_k) \]
\[ k_2 = g(x_k + \frac{\Delta t}{2}k_1, t_k + \frac{\Delta t}{2}) \]
\[ k_3 = g(x_k + \frac{\Delta t}{2}k_2, t_k + \frac{\Delta t}{2}) \]
\[ k_4 = g(x_k + \Delta t k_3, t_k + \Delta t) \]  

(4.17)

and \( \Delta t = t_{k+1} - t_k \). Eq. (4.16) then becomes the discrete time equations of motion that were expressed more generally in Eq. (4.4a). To be perfectly rigorous and consistent with our discretization of the continuous time motion equations, the matrix \( F_{x,k} \) used to propagate the covariance from one time step to the next would be given by

\[ F_{x,k} = I_{3 \times 3} + \Delta t A + \frac{\Delta t^2}{2!} A^2 + \frac{\Delta t^3}{3!} A^3 + \frac{\Delta t^4}{4!} A^4, \]  

(4.18)

where \( A = \frac{\partial g(x,t)}{\partial x} \bigg|_{x_k,t_k} \). However, two approximations are made in order to reduce computational time. First, only the central gravity term of \( g(x,t) \) is included when taking its partial derivative, so that the expressions do not become too cumbersome. Second, only second order terms are retained in Eq. (4.18).

As an aside, it is interesting to note that the expression given in Eq. (4.18) is simply a truncated series expansion of the matrix exponential \( e^{A \Delta t} \), which is the STM from \( k \) to \( k + 1 \) assuming a zero-order hold of the continuous time dynamics \( \dot{x} = g(x,t) \). This is typically how the quantity \( F_{x,k} \) would be calculated in continuous time. Moreover, in continuous time one would not evaluate \( e^{A \Delta t} \) exactly, but use a truncation of its power series. This is exactly what the discrete time procedure has yielded.

The process noise covariance matrix \( Q_k \) is set to be a constant diagonal matrix. Its diagonal entries can be adjusted to yield the desired filtering results.

### 4.3 GPS EKF (Fine Mode)

The purpose of fine mode is to obtain a very accurate estimate of the relative satellite state using raw GPS observables. Accuracies on the order of cm and mm/s for relative position and relative velocity, respectively, are required to achieve the desired formation control accuracy. For
this mode, it is assumed that an accurate estimate of the Chief (reference) satellite is available at every time step, since this is required to evaluate the measurement equations. While this absolute state estimate is briefly discussed in Sec. 4.3.3, the development of a Kalman filter for this purpose was considered beyond the scope of this thesis.

4.3.1 GPS Observables

In what follows the three main GPS observables are described, along with single differences and double differences, which are linear combinations of the observables taken from two different spacecrafts. The majority of the information in this section can be found in [56], [66], and [23].

Before discussing the observables, it is helpful to understand the way in which positioning information is transmitted from GPS satellites. Each GPS satellite generates a unique pseudo-random noise code (PRN code). This code is composed of a sequence of ones and zeros that identifies a satellite. Each vehicle’s code is unique enough so that it cannot be confused with that of another vehicle, thus the GPS satellites are identifiable by their PRN code number. There are two versions of this code, the coarse acquisition (C/A) code and the precision (P) code. The C/A code is composed of 1023 chips (a chip is the same as a bit, except it carries no information), and is repeated every millisecond. On the other hand, the P code repeats every 266 days, and each satellite is assigned a unique 7 day section of the P code, to be repeated weekly. Positioning accuracy is essentially determined by the width of a chip. The C/A code has a chip width of 1 microsecond, while the P code has a chip width of 0.1 microseconds. Each GPS satellite transmits its PRN codes, while each GPS receiver generates its own version of the code. By finding the offset between the generated and received codes, the signal travel time can be calculated. Multiplying the travel time by the speed of light yields the geometric distance between the satellite when the signal was emitted and the receiver when the signal was received. Consequently, the C/A code is accurate to about 300 m, while the P code is accurate to about 30 m. Unfortunately, the U.S. military encrypts the P code on all newer GPS satellites, so only authorized users can benefit from its accuracy. This procedure is known as anti-spoofing.

The C/A code is modulated onto the L1 carrier frequency (1575.42 MHz), while the P code is modulated onto both the L1 and L2 (1227.60 MHz) carrier frequencies. Using dual-frequency processing techniques, one is able to obtain a more accurate state estimate since the effect of the ionosphere can be mitigated through measurement combinations; however, the GPS receivers onboard CanX-4/-5 can only use the L1 frequency.
4.3.1.1 Pseudorange

The first GPS observable is called the pseudorange (PSR), which has units of m. It is generated by aligning the PRN code generated on the receiver to that received from a GPS satellite. The time offset of the two signals gives the time of signal travel, which can be turned into a range measurement. This PRN code alignment is performed on the receiver in a code-locked-loop (CLL). Of course, this range measurement is subject to many errors, such as the receiver clock error, the GPS satellite clock error, the errors in travel time caused by the signal refracting through the ionosphere, thermal noise errors, errors in the knowledge of the GPS satellite position, relativistic errors, and other higher-order errors. A nice derivation of the PSR observable can be found in [56], which can be written as

\[ P_s^r(t_r, t_e) = \rho_s^r(t_r, t_e) + c(\delta t_r - \delta t_s) + \delta I + \epsilon_P, \] (4.19)

where \( t_r \) is the time of signal reception, \( t_e \) is the time of signal emission, \( \rho_s^r(t_r, t_e) = \|R_r(t_r) - R_s(t_e)\| \) is the geometric range between the receiver (GPS antenna) at \( t_r \) and the GPS satellite at \( t_e \), \( c \) is the speed of light, \( \delta t_r \) is the receiver clock error, \( \delta t_s \) is the GPS satellite clock error, \( \delta I \) is the ionospheric error, and \( \epsilon_P \) encompasses all the other sources of error.

4.3.1.2 Carrier Phase

The second GPS observable is the carrier phase (CP) or accumulated doppler range (ADR), which has units of cycles or m. This is a measure of the difference between the phase of the receiver-generated code carrier wave at \( t_r \), and the phase of the transmitted carrier wave at \( t_e \). The CP observable can be thought of as another, more accurate, measure of the geometric range between the receiver and the GPS satellite, which aligns the carrier wave itself, rather than the code. The accuracy comes from the fact that the L1 wavelength is about 19 cm, and the carrier waves can be aligned to within a few percent of a wavelength. Unfortunately, upon receiving the carrier wave, the GPS receiver only aligns the two waves to within a fraction of a cycle; it cannot distinguish between the additional integer number of cycles remaining. This integer number of cycles is referred to as the “integer ambiguity”. Estimating this integer ambiguity correctly allows one to make the most precise state estimates.

The carrier phase observable can be written as

\[ \Phi_s^r(t_r, t_e) = \rho_s^r(t_r, t_e) + c(\delta t_r - \delta t_s) + \lambda N - \delta I + \epsilon_\Phi, \] (4.20)

where \( \lambda \) is the carrier wavelength, \( N \) is the float ambiguity, and \( \epsilon_\Phi \) is the error due to all other sources. Note that the ionospheric error \( \delta I \) is opposite in sign to that in Eq. [4.19]. This is
due to the fact that the carrier waves move at the phase velocity, and the CP is advanced by
the ionosphere whereas the modulated information (code) is delayed because it travels at the
group velocity. Also note that the CP observable is very similar to the PSR observable, mainly
differing by the ambiguity term.

As with the CLL for the PSR observable, the GPS receiver tracks the carrier phase using
a phase-locked-loop (PLL). The CP observable is actually generated by measuring the initial
phase offset, and then keeping track of and summing the instantaneous changes in carrier
phase, which are due to the Doppler shifts between the receiver and the satellite. This is why
the measurement is sometimes referred to as the “accumulated Doppler range”.

4.3.1.3 Doppler

The third and final GPS observable is the Doppler frequency shift, which has units of Hz or
m/s. This is a measure of the instantaneous frequency shift of the carrier wave (hence, the
range rate), and is measured by the sampling the frequency of the digitally controlled oscillator
in the PLL [19]. The Doppler observable can be thought of as the time derivative of Eq. (4.20),
though some authors prefer to make the distinction between the rate of change of the clock
error and the frequency error, as in [19]. However, here we follow the formulation in [20] and
write the Doppler observable as

$$\dot{\Phi} = \dot{\rho}_c(t_r, t_e) + c(\delta \dot{t}_r - \delta \dot{t}_s) - \delta \dot{I} + \epsilon \dot{\Phi},$$ (4.21)

which is just the time derivative of Eq. (4.20), with a different error term, $\epsilon \dot{\Phi}$. As discussed
in [19], the Doppler observable can be equivalently formulated as the three-point finite differ-
ence of the carrier phase observables. This improves the error characteristics of the Doppler
measurement and can potentially lead to better state estimates.

4.3.2 Measurement Combinations

In the theory of GPS positioning, there exist many different linear combinations of the afore-
mentioned observables (see [56] or [66]). They are used primarily to reduce the positioning
error induced by GPS clock errors, receiver clock errors, ionospheric errors, etc. They can also
be quite helpful when performing differential state estimation. Here only the single difference
and double difference measurement combinations will be discussed.

4.3.2.1 Single Difference

The single difference (SD) measurement combination is a way of removing the GPS clock error
term from the measurement. Referring to Fig. 4.1, consider two measurements made from the
Figure 4.1: Differential GPS measurement concept, where \( \rho^s_r \) denotes a measurement made from receiver ‘r’ to satellite ‘s’.

same GPS satellite but at different receivers: \( \rho^i_c \) and \( \rho^i_d \) (they must be the same observable). The corresponding single difference measurement is then given by

\[
\Delta \rho_i = \rho^i_d - \rho^i_c. \tag{4.22}
\]

For the case of a pseudorange observable, the single difference will be given by

\[
\Delta P_i = (\rho^i_d(t_r, t_{e,d}) + c(\delta t_d - \delta t_i) + \delta I_d + \epsilon_{P,d}) - (\rho^i_c(t_r, t_{e,c}) + c(\delta t_c - \delta t_i) + \delta I_c + \epsilon_{P,c})
\]

\[
= \rho^i_d(t_r, t_{e,d}) - \rho^i_c(t_r, t_{e,c}) + c(\delta t_r,d - \delta t_r,c) + \Delta I + \Delta \epsilon_P
\]

\[
\Delta P_i = \rho^i_d(t_r, t_{e,d}) - \rho^i_c(t_r, t_{e,c}) + \Delta b + \Delta I + \Delta \epsilon_P \tag{4.23}
\]

where \( \Delta b = c(\delta t_r,d - \delta t_r,c) \) is defined to be the difference between the Chief and Deputy receiver clock errors (i.e., relative clock error), \( \Delta I \) is the combined ionospheric error, and \( \Delta \epsilon_P \) is the combination of additional error sources from both receivers. Note that when using this measurement, \( \Delta b \) must also be estimated.

Likewise, the SD CP measurement is given by

\[
\Delta \Phi_i = \rho^i_d(t_r, t_{e,d}) - \rho^i_c(t_r, t_{e,c}) + \Delta b + \Delta N_i - \Delta I + \Delta \epsilon_\Phi, \tag{4.24}
\]

where \( \Delta N_i \) is the SD ambiguity for GPS satellite ‘i’. When using this measurement type, the SD ambiguity for each satellite must also be estimated.
Lastly, following [20], the SD Doppler measurement is given by

\[
\Delta \dot{\Phi}_i = \dot{\rho}_d(t_r, t_{e,d}) - \dot{\rho}_c(t_r, t_{e,c}) + \Delta \dot{b} - \Delta \dot{I} + \Delta \epsilon \dot{\Phi},
\] (4.25)

where \(\Delta \dot{b} = c (\delta t_{r,d} - \delta t_{r,c})\) is the differential clock error drift, which must also be estimated if using this measurement type.

Note how the GPS satellite clock error term, \(c \cdot \delta t_i\), falls out due to the difference across receivers. The disadvantage of the SD measurement is the increased measurement noise.

### 4.3.2.2 Double Difference

The double difference (DD) measurement combination is a way of removing the relative clock error term from the measurement equation. This is achieved by subtracting two single difference measurement equations. Referring again to Fig. 4.1, consider all four measurements made between satellites and receivers: \(\rho_{i,c}\), \(\rho_{i,d}\), \(\rho_{j,c}\), and \(\rho_{j,d}\). The double difference combination is then defined to be

\[
\nabla \Delta \rho_{ij} = \Delta \rho_i - \Delta \rho_j = \left(\dot{\rho}_d^i - \dot{\rho}_c^i\right) - \left(\dot{\rho}_d^j - \dot{\rho}_c^j\right).
\] (4.26)

Considering the specific case of the pseudorange observable, it is clear from Eq. (4.23) that the relative clock error term \(\Delta b\) is cancelled out. Again, the disadvantage is the further increased measurement noise.

Similarly to the case of SD CP measurements, when using DD CP, the DD ambiguity \(\nabla \Delta N_{ij}\) for each DD measurement must be estimated. These DD ambiguities are, in theory, integers [56], and the resolution of these integer ambiguities enables the most precise position estimates.

When using DD measurements, one must typically select a reference satellite from which all double differences are calculated. This reference satellite can be set to be that with the highest signal to noise ratio, or the highest elevation [23] [17]. Alternatively, given a vector \(y = \left[ \Delta \rho_1 \cdots \Delta \rho_m \right]^T\) of single difference measurements, the double difference measurements could be formed by subtracting the second measurement from the first, the third from the second, and so on, with no regard for noise levels or elevation. This last method would be used only in computationally or time limited systems.

### 4.3.3 Absolute State Estimation

An accurate state estimate of the Chief satellite is required for two primary reasons. First, this quantity is used to evaluate the EKF measurement equations and forms the reference point for the relative state estimate. Second, the Chief position and velocity vectors are required to form
the rotation matrix from the ECEF frame to the LO frame (Sec. 3.2.5.2), which is used to express
the calculated control force in the ECEF frame (see Sec. 5.3). Based on GPS signal simulator
tests performed in [51], using a receiver model very similar to that being flown on CanX-4/-5, the
position and velocity solutions calculated by the receiver software are accurate to approximately
10 m and 0.07 m/s, in the best case (no ionospheric delays or broadcast ephemeris errors). This
inaccuracy is primarily due to two sources: the application of a tropospheric refraction model
to correct for signal delay that does not actually occur, and a high level of noise on the Doppler
measurements. A real-time absolute filter has already been demonstrated in hardware-in-the-
loop tests with a GPS signal simulator using both kinematic [19] and dynamic [22] filters. If it is
found that the desired control accuracy is not achievable using the receiver-estimated absolute
states, a more accurate absolute filter should be pursued.

4.3.4 Antenna Location Correction

One very important point is that the absolute position and velocity recorded by the GPS receiver
are those of the GPS antenna phase center, whereas the position and velocity of the center of
mass are required. As a result, an additional step of correcting for this offset is required. This
process is described below.

Referring to Fig. 4.2 the position and velocity of the GPS antenna phase center are given
by \( \mathbf{R} \) and \( \mathbf{V} \), respectively. The position and velocity of the center of mass of the satellite are
given by \( \mathbf{R}_i \) and \( \mathbf{V}_i \), respectively. Finally, the position of the GPS antenna relative to the
center of mass of the satellite is given by \( \mathbf{R}_a \).

From the diagram, the position of the center of mass is given by \( \mathbf{R}_i = \mathbf{R} - \mathbf{R}_a \). In terms
of known quantities, this is written as

\[
\mathbf{F}_{E}^{T} \mathbf{r}_{i,E} = \mathbf{F}_{E}^{T} (\mathbf{r}_{E} - \mathbf{C}_{EI} \mathbf{C}_{IB} \mathbf{r}_{a,B}),
\]  

(4.27)

where \(\mathbf{C}_{EI}\) is the rotation matrix from the GCI to the ECEF frame as calculated in Sec. 3.2.5.1, and \(\mathbf{C}_{IB}\) is the rotation matrix from the body frame to the GCI frame, which is estimated by the attitude control system. The velocity as seen in the ECEF frame is given by the time derivative of Eq. (4.27), or

\[
\dot{\mathbf{r}}_{i,E} = \dot{\mathbf{r}}_{E} - \left( \mathbf{C}_{EI} \dot{\mathbf{C}}_{IB} + \dot{\mathbf{C}}_{EI} \mathbf{C}_{IB} \right) \mathbf{r}_{a,B},
\]  

(4.28)

where \(\dot{\mathbf{C}}_{IB} = \mathbf{C}_{IB} \mathbf{\omega}_{BI} \times\), and \(\mathbf{\omega}_{BI}\) is the angular velocity of the body frame with respect to the GCI frame, which is also estimated by the attitude control system.

### 4.3.5 Relative Navigation Filter Design

The design of the relative navigation filter presented below draws upon many different sources which have previously investigated relative navigation filters for spacecraft formation flying in both real-time and offline environments. The design is meant to use only the desirable qualities from certain filters, discarding the rest. The contributions from the work of [24], [23], [17], [20], [22], and [19] are gratefully acknowledged. It should also be noted that significant departures meant to improve filter performance have been taken from [24], where the preliminary filter design and analysis was performed.

#### 4.3.5.1 Measurement Equations

Single difference pseudorange, carrier phase, and Doppler measurements have been selected for the relative navigation filter. Single differences were selected over double differences primarily because of the reduced computational overhead linked to bookkeeping, and the increased chance of filter convergence. It is expected that throughout the mission, frequent attitude slewing maneuvers will be required for control purposes. During a slew, one or more GPS satellites may drop from the Chief or Deputy view. If even a single satellite drops from view of either the Chief or Deputy, then the corresponding DD measurements must be discarded, and the DD ambiguity states must be re-initialized. A state vector with frequently changing states may not allow the solution to converge, resulting in a poor navigation solution. A secondary reason for selecting SD is that more measurements are available because no reference measurement is needed.

It is also assumed that the differential ionospheric error will be very small for relative
separations on order of 1 km, which justifies the exclusion of the ionospheric error term. Any
loss in performance can be corrected by adjusting the measurement noise covariance terms.

This set of nonlinear measurements is assumed to take the form \( y_k = h(x_k) + v_k \), where it
is assumed that the measurement noise is purely additive. It is further assumed that the mea-
surement noise covariance \( R_k = E[v_k v_k^T] \) is diagonal, i.e., the measurements are uncorrelated. This may in fact be a poor assumption, however, it greatly simplifies the problem. Given ‘n’
commonly observed GPS satellites, the vector of measurements is written explicitly as
\[
\begin{align*}
        y_k &= h(x_k) + v_k = \begin{bmatrix} \Delta P_1 & \cdots & \Delta P_n & \Delta \Phi_1 & \cdots & \Delta \Phi_n & \Delta \dot{\Phi}_1 & \cdots & \Delta \dot{\Phi}_n \end{bmatrix}^T + v_k \\
\end{align*}
\]

During the EKF update step, the state and covariance are updated using the update scheme
presented in Sec. 4.1.1. The partial derivative of the measurement equation \( h(x_k) \) w.r.t. the
state vector \( x_k \) is denoted by \( H_k \) and is given in App. A. The diagonal elements of the matrix
\( R_k \) are set to be constant. During actual operation, these values will be tuned in order to
optimize filter performance.

### 4.3.5.2 State Vector and Dynamic Equations of Motion

Given that all three SD measurement types are used, the state vector is selected to be
\[
\begin{align*}
    x_k &= \begin{bmatrix} \Delta x^T & \Delta b & \Delta \dot{b} & \Delta N_1 & \cdots & \Delta N_n \end{bmatrix}_k^T, \\
\end{align*}
\]

where \( \Delta x_k = [\Delta r^T \Delta \dot{r}^T]_k^T \), \( \Delta r \) is the relative position in the ECEF frame, \( \Delta \dot{r} \) is the relative
velocity in the ECEF frame, \( \Delta b \) is the relative clock error, \( \Delta \dot{b} \) is the relative clock error drift,
and \( \Delta N_i \) is the \( i \)th ambiguity. The size of this state vector is \( 8 + n \), for \( n \) commonly observed
satellites. Given that the OEMV-1G has 14 channels, the maximum number of states is 22.
Note that the size of this state vector will vary with the number of commonly observed GPS
satellites.

Following [23] and [22], the relative states are propagated using “pseudo” relative dynamics.
That is, given a Chief position and velocity at time \( t_k \): \( R_{c,k} \) and \( \dot{R}_{c,k} \), as in Sec. 3.4, the Deputy
position and velocity at time \( t_k \) are given by \( R_{d,k} = R_{c,k} + \Delta r_k \) and \( \dot{R}_{d,k} = \dot{R}_{c,k} + \Delta \dot{r}_k \),
respectively. Then, the relative state at time \( t_{k+1} \) is obtained by integrating Eq. (4.15) (again
using the RK4) with the Chief and Deputy states and then subtracting the result such that
\( \Delta r_{k+1} = R_{d,k+1} - R_{c,k+1} \) and \( \Delta \dot{r}_{k+1} = \dot{R}_{d,k+1} - \dot{R}_{c,k+1} \). In this way, the existing code for the
coarse EKF can be re-used.
Following [20], the differential clock error drift is modeled as a constant velocity process:

\[
\begin{bmatrix}
\Delta b \\
\Delta \dot{b}
\end{bmatrix}_{k+1} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \begin{bmatrix}
\Delta b \\
\Delta \dot{b}
\end{bmatrix}_k + \begin{bmatrix} w_{\Delta b} \\
w_{\Delta \dot{b}}
\end{bmatrix},
\]

(4.31)

where \(w_{\Delta b}\) and \(w_{\Delta \dot{b}}\) are Gaussian white noise sources, and are not actually included as a forcing term during the integration.

The SD ambiguity states are assumed to be constant between measurement epochs, so that

\[
\Delta N_{k+1} = \Delta N_k.
\]

(4.32)

The covariance matrix is propagated using

\[
P_{k+1}^- = \Phi_k P_k^+ \Phi_k^T + Q_w,
\]

(4.33)

which is equivalent to Eq. (4.4b) with purely additive noise. Since the position, clock, and ambiguity states are decoupled, the matrix \(\Phi_k\) is written as

\[
\Phi_k = \begin{bmatrix}
\Phi_{x,k} & 0_{6\times2} & 0_{6\times n} \\
0_{2\times6} & \Phi_b & 0_{2\times n} \\
0_{n\times6} & 0_{n\times2} & I_{n\times n}
\end{bmatrix},
\]

(4.34)

where \(\Phi_{x,k}\) is given by the second order Taylor series expansion of the matrix exponential \(e^{A(R_{\Delta,k})\Delta t}\) with \(A(R)\) given in App.\ A. \(\Phi_b\) is given by

\[
\Phi_b = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix},
\]

(4.35)

and the identity matrix is a trivial consequence of Eq. (4.32).

Similarly, the process noise covariance matrix \(Q_w\) is assumed to take the form

\[
Q_w = \begin{bmatrix}
Q_x & 0_{6\times2} & 0_{6\times n} \\
0_{2\times6} & Q_b & 0_{2\times n} \\
0_{n\times6} & 0_{n\times2} & 0_{n\times n}
\end{bmatrix},
\]

(4.36)

where \(Q_x = Q_{w,x}\) and \(Q_b\) are given in App.\ A. Note that there is no noise term associated with the ambiguity states.
4.3.5.3 Evaluating the Measurement Equations

There are a few important points regarding the nonlinear measurement equations. First, note how the geometric range $\rho$ and range rate $\dot{\rho}$ in Eqs. (4.23), (4.24), and (4.25) are functions of the receiver state at the time of signal reception, and the GPS satellite state at the time of signal emission. The time of signal reception ($t_r$) is known, and is the same for both the Chief and Deputy satellites. However, the time of signal emission ($t_e$) is unknown, and is different for each GPS satellite. As a result, one must iteratively solve for the emission time from each GPS satellite. This is done using Algorithm 1 in [20].

Second, once $t_e$ has been calculated, the GPS satellite position and velocity in the ECEF frame must be obtained. Using the broadcast ephemeris parameters obtained from the RAWEPEM GPS log [64], the position is calculated using the algorithm presented in [61], and the velocity is calculated using the equations provided in [67].

Lastly, it is important to note that the SD measurement equations can only be evaluated if the Chief and Deputy measurement times are the same. This is true as long as both receivers are set to FINE or FINESTEERING mode, which indicates that their clocks are aligned to GPS time to within 1 $\mu$s [64].

4.3.5.4 Data Quality Verification

Before the measurements from a given GPS satellite can be used, a series of data quality checks must be performed. This ensures that only valid data is processed in the filter outlined in Sec. 4.3. If any one of the conditions below fails, then the measurement data from the GPS satellite is rejected, i.e., the satellite is not considered to be in view. These checks have been adapted from [24], and are given as follows:

1. Phase Locked Flag = TRUE – Receiver is tracking the carrier phase.
2. Code Locked Flag = TRUE – Receiver is tracking the code.
3. Lock Time > 0 – Satellite has been locked for more than one instant.
4. Tracking State = L1 Phase Lock Loop – Receiver is tracking the carrier phase.
5. Carrier to Noise Ratio > $\text{CNR}_{min}$ – Signal strength is adequate. Currently $\text{CNR}_{min}$ is set to 32.
6. Parity Known Flag = TRUE – Half-cycle carrier phase ambiguity has been resolved.

These checks ensure that the PSR, CP, and Doppler measurements can all be used. For this filter design, either 3 or 0 measurements are made available. The flight code, however, has been generalized so that varying numbers of each measurement are possible.
4.3.6 Integer Ambiguity Resolution and Validation

The aforementioned filter design supports the estimation of single difference ambiguity states. However, it is well known that DD integer ambiguity resolution is required to obtain the most accurate relative state estimates. Following the development of [23], this extra step of resolving the DD ambiguities is performed exterior to the nominal filter operation – after the prediction/correction step. If the DD ambiguities can be resolved, the estimated relative state is corrected to account for this, and is denoted as the fixed solution.

The problem of DD integer ambiguity resolution can be stated as

$$\arg \min_a J(a) \text{ s.t. } a \in \mathbb{Z}^n$$

$$J(a) = (\hat{a} - a)^T Q_{\hat{a}}^{-1} (\hat{a} - a),$$

where $\hat{a}$ is the vector of float DD ambiguities, $a$ is the vector of integer DD ambiguities, and $Q_{\hat{a}}$ is the float DD ambiguity covariance matrix. This problem is, in fact, very difficult to solve [56]. A secondary issue is that of validating the solution, since improperly fixed integer ambiguities generally lead to poor state estimates and filter divergence [23].

The best method of solving this problem is through sequential conditional least squares, the basic premise of which is simple. The search begins by finding the DD ambiguity closest to an integer, and simply rounding it. Then, the second DD integer ambiguity is estimated, conditioned on the fact that the first ambiguity is an integer. Next, the third DD integer ambiguity is estimated, conditioned on the fact that the first and second ambiguities are integers. This process continues until all the ambiguities have been estimated. In this manner, all possible sets of integer ambiguities are estimated, and the vector of ambiguities which solves Eq. (4.37), is selected as the best solution.

The least-squares ambiguity decorrelation adjustment (LAMBDA) method [68] was selected to perform the ambiguity resolution. This method is an optimized version of sequential conditional least squares, transforming the search domain to enable faster searches than otherwise possible. The LAMBDA method has been implemented using the algorithms outlined in [69], where the size of the search ellipsoid is determined so that at least two integer ambiguity vectors are determined. Having two solutions (the best and second best) allows the validation step to be performed.

Drawing on the validation procedure outlined in Sec. 4.1.2 of [23], there are two ambiguity validation steps performed. The first is to check the lower bound on the success rate of the LAMBDA method. If the success rate is below a specified tolerance (currently 95%), the estimated ambiguity vector is rejected. The second validation step is called the ratio test.
Given the float DD ambiguity $A_{dd}$, the best integer ambiguity estimate $N_{dd,b}$, the second best integer ambiguity estimate $N_{dd,sb}$, and the float DD ambiguity covariance $Q_a$, the ratio

$$R = \frac{\left( A_{dd} - N_{dd,sb} \right)^T Q_a^{-1} \left( A_{dd} - N_{dd,sb} \right)}{\left( A_{dd} - N_{dd,b} \right)^T Q_a^{-1} \left( A_{dd} - N_{dd,b} \right)}$$

is calculated. The best integer ambiguity estimate is accepted if

$$R > k_{sb}$$

and rejected otherwise. As in [23], the value $k_{sb} = 3$ is used.

Intuitively, passing the ratio test means that $N_{dd,b}$ is sufficiently far away from every other integer ambiguity estimate that it is likely to be correct.

### 4.3.6.1 Obtaining the Fixed Solution

As discussed in Sec. 4.3.5 only the SD ambiguities and their covariances are estimated. Before the methods discussed in Sec. 4.3.6 can be applied, the ambiguities and their covariances must be transformed.

First, the EKF state vector is written as

$$x_k^+ = \left[ \hat{b}^T \quad \hat{a}^T \right]^T,$$

where the relative position, velocity, and clock error states are lumped into $\hat{b}$, also called the ‘baseline parameters’, and the SD ambiguity states are given by $\hat{a}$.

Next, the EKF covariance matrix $P_k^+$ is written as

$$P_k^+ = \begin{bmatrix} P_{\hat{b}} & P_{\hat{b} \hat{a}} \\ P_{\hat{a} \hat{b}} & P_{\hat{a}} \end{bmatrix},$$

where $P_{\hat{b}}$ is the covariance associated baseline parameters, $P_{\hat{a}}$ is the covariance associated with the SD ambiguity states, and $P_{\hat{b} \hat{a}}$ is the cross-covariance.

The SD ambiguities are transformed into DD ambiguities via the linear transformation

$$\hat{A} = T \hat{a},$$

(4.41)
where $T \in \mathbb{R}^{n-1 \times n}$ is given by

$$
T = \begin{bmatrix}
1 & -1 & 0 & \cdots & 0 \\
0 & 1 & -1 & \cdots & 0 \\
\vdots & \ddots & \ddots & \ddots & \vdots \\
0 & \cdots & 0 & 1 & -1
\end{bmatrix}.
$$

(4.42)

The covariances are then transformed as

$$
P_{\hat{A}} = TP_{\hat{a}}T^T \tag{4.43a}
$$

$$
P_{b\hat{A}} = P_{b\hat{a}}T^T, \tag{4.43b}
$$

where $P_{\hat{A}}$ is the float DD ambiguity covariance and $P_{b\hat{A}}$ is the cross covariance between the baseline parameters and the DD ambiguities.

Then, assuming the DD integer ambiguity $N$ has been estimated and validated using the procedure in Sec. 4.3.6, the fixed solution is then calculated as [70]:

$$
\hat{b} = \hat{b} - P_{b\hat{A}}P_{\hat{A}}^{-1}(\hat{A} - N),
$$

(4.44)

where the fixed solution for the baseline parameters (relative states) is given by $\hat{b}$.

### 4.4 Filter Initialization

The coarse EKF states are initialized using the first absolute states recorded by the GPS receiver, while the covariance matrix is initialized to a constant diagonal matrix. The initial position variance is set to 100 m$^2$ for each coordinate, while the initial velocity variance is set to 10 m$^2$ for each coordinate. This initialization scheme prevents divergence by starting at the best estimate of the state.

The fine EKF is initialized using the scheme proposed in [22]. This scheme uses the single-point least squares solution presented in [19] to initialize the relative position and velocity states. The relative position and differential clock error states are updated with an iterative least squares solution using only SD PSR measurements, starting with an initial guess of 0. The relative velocity and differential clock error drift are then determined with a least squares solution using only SD Doppler measurements, where the measurement equations are evaluated assuming initial velocities of 0. The velocity solution is not iterated since the measurements are linear in relative velocity, with the relative position assumed known from the previous step.
The portion of the covariance matrix related to these states is assumed to be diagonal with apriori values assigned.

The initial SD ambiguity states are initialized to be the difference between the measured carrier phase and the calculated pseudorange [20]. This method differs slightly from [22], where measured pseudorange is used. The initial SD ambiguity covariance is set to an apriori value, as in [23] and [22].

4.5 Filter Operation with ISL/Data Communication

Fig. 4.3 depicts the operation of the relative navigation filter with both satellites and the control algorithm in the loop. The rate \( f_c \) is the rate at which the data is collected from the GPS receivers, and likely be set at around 0.2 Hz. The rate \( f_s \) is the rate at which the data is sent across the ISL, which is about 1250 bytes per second. As shown in Tab. 4.1, the maximum amount of data to be transmitted from the Chief to the Deputy is about 550 bytes. Including margin on processing time, the Deputy should begin the filtering step less than one second after the messages are initially produced. The remaining tables depicting the relative navigation filter action are shown in App. B.

<table>
<thead>
<tr>
<th>Data</th>
<th>Variable</th>
<th>Data Type</th>
<th>Size (bytes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time</td>
<td>GPS week</td>
<td>uint32</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>GPS sec.</td>
<td>int32</td>
<td>4</td>
</tr>
<tr>
<td>Chief State</td>
<td>( R, V )</td>
<td>double</td>
<td>6 * 8 = 48</td>
</tr>
<tr>
<td></td>
<td>( \sigma^2_R, \sigma^2_V )</td>
<td>float</td>
<td>6 * 4 = 24</td>
</tr>
<tr>
<td></td>
<td>( R ) and ( V ) reliability</td>
<td>uchar</td>
<td>2 * 1 = 2</td>
</tr>
<tr>
<td></td>
<td>attitude - orientation and rate</td>
<td>float</td>
<td>7 * 4 = 28</td>
</tr>
<tr>
<td></td>
<td>attitude - mode</td>
<td>uchar</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>attitude - error flags</td>
<td>uint32</td>
<td>4</td>
</tr>
<tr>
<td>Meas. Data</td>
<td>Num. Meas.</td>
<td>uchar</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>pseudorange</td>
<td>double</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>carrier phase</td>
<td>double</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>Doppler</td>
<td>float</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>PSR and CP variance</td>
<td>float</td>
<td>2 * 4 = 8</td>
</tr>
<tr>
<td></td>
<td>measurement flags</td>
<td>uchar</td>
<td>3</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td></td>
<td>116 + 31 \cdot \text{NumMeas}</td>
</tr>
<tr>
<td>Max. Total</td>
<td></td>
<td></td>
<td>550</td>
</tr>
</tbody>
</table>

Table 4.1: Transmitted data for fine mode relative navigation.

It should be noted here that the data quality checks discussed in Sec. 4.3.5.4 are applied as part of the data parsing steps in Fig. 4.3. This way, the data transmission time is improved by removing erroneous information before it is sent.
4.6 Effect of Attitude Slews on Navigation

One very important consideration for the operation of this navigation filter is the effect that the attitude slew maneuvers will have on relative state estimation. Unfortunately, this effect has not yet been analyzed in detail, so no quantitative results are available. For missions such as PRISMA and GRACE, this is not such a concern since multiple GPS antennas are mounted to the spacecrafts. However, in the case of CanX-4/-5, each satellite has only one antenna. For argument’s sake, consider a 90° attitude maneuver for the Deputy spacecraft, required to orient the thrusters for a formation keeping burn. In this case, many of the satellites that it was tracking will be lost from view. The receiver will then take some time before new satellites can be tracked. This exact time will depend whether or not almanac/ephemeris data is available for the new satellites, but is expected to be approximately 60 to 120 seconds. After the new satellites are acquired, the relative navigation filter could take upwards of five minutes to converge. No control maneuvers should be performed during this time, since navigation data will not be available. Thus, the overall effect of large attitude maneuvers is to increase the minimum time between control thrusts, which will have a large impact on formation keeping accuracy. It is
highly recommended that the effect of attitude maneuvers are investigated in further detail, and that a comprehensive strategy for mitigating the reduction of control accuracy is devised.
Chapter 5

Formation Control

This chapter discusses the formation control methods used onboard CanX-4/-5. There are two modes: formation keeping, and reconfiguration.

5.1 Formation Keeping

Here formation keeping is defined as the active control of a desired formation. The formation keeping method for the formation flying integrated onboard navigation algorithm (FIONA) was designed and implemented in [39]. Its details are repeated here for completeness.

5.1.1 Linear Quadratic Regulator

The backbone of the formation keeping algorithm is the linear quadratic regular (LQR). LQR controllers are optimal controllers in that their design is accomplished through the minimization of a functional – the sum of weighted norms of the state and control is minimized. FIONA actually uses both the continuous and discrete steady-state forms of the LQR. The continuous formulation is used primarily during the first and last orbits of each formation. This is to stabilize the formation before and after reconfiguration maneuvers. Its other use is when a preset maximum tracking error is exceeded. This ensures that the formation tracking error remains small throughout the active formation keeping portions of the mission. All other formation keeping orbits utilize the control gain calculated from the discrete formulation, enabling longer periods with no control actuation, which would be used for observations. The continuous laws are not used in these cases since it was found that excessive ΔV penalties are incurred when the continuous controller is used with a period greater than 2.5 minutes [71].
5.1.1.1 Continuous LQR

Given a linear time-varying (LTV) system

$$\dot{x}(t) = A(t)x(t) + B(t)u(t), \quad (5.1)$$

where $x(t)$ is the state and $u(t)$ is the control, the continuous time, finite horizon LQR problem is formed as:

$$\arg\min_{u, \lambda} J(u, \lambda) \quad (5.2a)$$

$$J(u, \lambda) = \frac{1}{2}x^T(T)Mx(T) + \int_0^T \frac{1}{2}x^T(t)Qx(t) + \frac{1}{2}u^T(t)Ru(t) - \lambda^T(t)(\dot{x}(t) - A(t)x(t) + B(t)u(t)) dt, \quad (5.2b)$$

where $M$, $Q$ and $R$ are symmetric weighting matrices satisfying $M, Q \geq 0$ and $R > 0$, $T$ is the control horizon, and $\lambda(t)$ is a Lagrange multiplier that has been introduced in order to enforce the system dynamics of Eq. (5.1) as a constraint. Different controllers are obtained by varying the weighting matrices so that more or less emphasis is placed on regulating the system, or the control force magnitude. Weighting matrix selection is discussed more in Sec. 5.1.3.

The problem in Eq. (5.2) is solved using techniques of variational calculus. Note that in what follows, the time argument is omitted. Taking the first variation of Eq. (5.2) and setting the result equal to zero, we obtain

$$\delta J = x^T(T)M\delta x(T) + \int_0^T x^TQ \delta x + u^TR \delta u + \lambda^T(\dot{x} - A\delta x + B\delta u) - \lambda^T(\dot{x} - A\delta x + Bu) dt = 0. \quad (5.3)$$

Using integration by parts on the term $\lambda^T \delta \dot{x}$, this reduces to

$$\int_0^T \left( x^T Q + \lambda^T A + \lambda^T \dot{x} \right) \delta x + \left( u^T R + \lambda^T B \right) \delta u - \delta \lambda^T(\dot{x} - A\delta x + Bu) dt - \lambda^T \delta x|_0^T = 0, \quad (5.4)$$

where the final term becomes represents our boundary condition. Since this relationship must hold for arbitrary $\delta x$, $\delta u$, and $\delta \lambda$, we must have

$$\dot{x} = Ax + Bu$$

$$-\dot{\lambda} = A^T \lambda + Qx, \quad \lambda(T) = Mx(T)$$

$$u = -R^{-1}B^T \lambda, \quad (5.5)$$

where we have eliminated one boundary condition since the initial condition is known, hence $\delta x(0) = 0$. 
Setting $\lambda = P x$, motivated by the boundary condition at $T$, reduces the system in Eq. (5.5) to the matrix Riccati equation

$$- \dot{P} = PA + A^T P - PBR^{-1}B^T P + Q,$$

(5.6)

with boundary condition $P(T) = M$. If one makes the additional assumption of a steady state process so that $\dot{P} = 0$, the problem reduces to solving the algebraic Riccati equation (ARE). The optimal control function $u$ can then be written as

$$u(t) = - R^{-1} B^T P x(t).$$

(5.7)

The benefit of the ARE is that rather than solving a differential equation for $P$, the solution to the ARE is obtained offline thus saving computational effort and memory. There may be a slight degradation in regulation performance in transient regions, but this is tolerable.

In practice, these controllers are calculated using the MATLAB function `lqr(A, B, Q, R)`, where the plant dynamics are given by the HCW equations (Eq. (3.25)), and the matrix $B$ is given by $B = [0_{3\times3} I_{3\times3}]^T$.

5.1.1.2 Discrete LQR

Using a discretization procedure such as a zero order hold, the discretized form of the LTV system in Eq. (5.1), can be written as

$$x_{k+1} = A_k x_k + B_k u_k.$$  

(5.8)

Then, the discrete LQR problem is formulated as [63]

$$\arg \min_{u_k,\lambda_k} J(u_k, \lambda_k)$$

(5.9a)

$$J(u_k, \lambda_k) = \frac{1}{2} x_N^T M x_N + \sum_{k=0}^{N-1} \frac{1}{2} x_k^T Q_k x_k + \frac{1}{2} u_k^T R_k u_k - \lambda_{k+1}^T (x_{k+1} - A_k x_k - B_k u_k),$$

(5.9b)

where the interpretations of $M, Q, R,$ and $\lambda$ are the same as in the continuous case. Eq. (5.9) is minimized with respect to the discrete state variables by taking partial derivatives with
respect to $x_k$, $u_k$, $\lambda_k$ and setting the result equal to zero. This process yields

$$
\begin{align*}
\lambda_k &= A_k^T \lambda_{k+1} + Q_k x_k \\
R_k u_k &= B_k^T \lambda_{k+1} \\
x_{k+1} &= A_k x_k + B_k u_k
\end{align*}
$$

subject to the boundary condition $\lambda_N = M x_N$. Motivated by this boundary condition, we guess that

$$
\lambda_k = S_k x_k,
$$

where $S_N = M$. Substituting this into Eq. (5.10), one finds the optimal control function and the recursive equation for the matrix $S_k$:

$$
\begin{align*}
\mathbf{u}_k &= - \left( \mathbf{R}_k + \mathbf{B}_k^T \mathbf{S}_{k+1} \mathbf{B}_k \right)^{-1} \mathbf{B}_k^T \mathbf{S}_{k+1} \mathbf{A}_k x_k \\
\mathbf{S}_k &= \mathbf{A}_k^T \mathbf{S}_{k+1} \mathbf{A}_k - \mathbf{A}_k^T \mathbf{S}_{k+1} \mathbf{B}_k \left( \mathbf{R}_k + \mathbf{B}_k^T \mathbf{S}_{k+1} \mathbf{B}_k \right)^{-1} \mathbf{B}_k^T \mathbf{S}_{k+1} \mathbf{A}_k,
\end{align*}
$$

where with boundary condition $S_N = M$. Thus, Eq. (5.11b) must be solved backwards in time, and then the optimal control at each step can be found using Eq. (5.11a).

If we have a linear time-invariant system such that matrices $A_k$, $B_k$, $Q_k$, and $R_k$ are constant, then we can assume that $S_{k+1} = S_k = S$, which results in the discrete algebraic Riccati equation. Eq. (5.11b) is then solved once for $S$ and the optimal control at each step in time is given by

$$
\mathbf{u}_k = - \left( \mathbf{R}_k + \mathbf{B}_k^T \mathbf{S}_k \mathbf{B}_k \right)^{-1} \mathbf{B}_k^T \mathbf{S}_k \mathbf{A}_k x_k = -\mathbf{K} x_k.
$$

This steady state assumption is made in the design of the formation flying controller, which is in practice calculated using the MATLAB function lqrd($\mathbf{A}$, $\mathbf{B}$, $\mathbf{Q}$, $\mathbf{R}$, $T$), which discretizes the continuous plant using a zero order hold with sample time $T$.

### 5.1.2 Implicit Tracking Control

For FIONA, the LQR controllers have been implicitly designed for tracking by cleverly defining the LQR cost function [71]. First, define $\mathbf{\hat{x}} = \mathbf{x} - \mathbf{x}_{ref}$, where $\mathbf{x}_{ref}$ is a solution to the unforced relative motion equations. That is

$$
\dot{\mathbf{x}}_{ref} = \mathbf{A} \mathbf{x}_{ref}.
$$

Then the LQR cost function can be formulated identically to Eq. (5.2), with $\mathbf{\hat{x}}$ replacing $\mathbf{x}$. It can then be shown that the optimal control function is given by Eq. (5.7), with $\mathbf{\hat{x}}$ replacing $\mathbf{x}$. This formulation avoids the inclusion of auxiliary variables typically associated with tracking LQR controllers. Note that $\mathbf{x}_{ref}$ is selected to be a PCO or ATO orbit, which are discussed in
Sec. 5.1.4 The discrete time implicit tracking controller is formulated in an analogous manner.

5.1.3 Weighting Matrix Selection

Using the LQR control formulations presented in Sec. 5.1.1.1 and 5.1.1.2, the controller design procedure is reduced to the selection of the weighting matrices $Q$ and $R$, and the discretization time period $T$. The discretization period is a configurable parameter, representing our desired time in between control thrusts. This must be large enough to provide adequate time between thrusts for observations, but not so large that the formation is unstable. The weighting matrix selection is a little more difficult, since one cannot predict closed-loop performance of the nonlinear system given a particular set of weights. A purely diagonal form of the weighting matrices is chosen here such that

$$Q = \begin{bmatrix} I_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & I_{3 \times 3} \end{bmatrix} q_r,$$

(5.13)

where $q_r$ and $q_v$ are the position and velocity weights, respectively, and

$$R = I_{3 \times 3} r_u,$$

(5.14)

where $r_u$ is the control weighting.

Increasing the magnitude of $Q$ places a larger penalty on state errors, thus the controller is forced to track the reference solutions more closely. Conversely, decreasing its magnitude relaxes the state error constraint, leading to coarser tracking performance. Increasing the magnitude of $R$ places a larger penalty on the control effort, which reduces the control authority. Decreasing the control weighting results in greater fuel use, since large impulses are not penalized. Here we wish to select these matrices to provide sub-meter tracking control, while minimizing the fuel use – two competing objectives.

No set values for the weights are presented here, since these generally need to be tuned depending on the fidelity of the simulation, and the desired performance. This tuning is especially critical before on orbit operations. It is highly recommended that before any actual thrusts are performed, the calculated controls are monitored and verified to be of reasonable magnitude, and would result in stable formation keeping.

5.1.4 Reference Trajectories

There are two main types of reference trajectories planned for CanX-4/-5, the projected circular orbit (PCO) and the along-track orbit (ATO). When observed from Earth, satellites in a PCO appear to be in a circular orbit of radius $\rho$ about the reference (Chief) satellite. The ATO
Table 5.1: Choice of constants for circular PCO and ATO reference trajectories obtained from Eq. (3.26).

<table>
<thead>
<tr>
<th>Formation Type</th>
<th>$d_1$ (m)</th>
<th>$d_2$ (m)</th>
<th>$d_3$ (m)</th>
<th>$d_4$ (m)</th>
<th>$\alpha$ (rad)</th>
<th>$\beta$ (rad)</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCO</td>
<td>$\rho$</td>
<td>0</td>
<td>0</td>
<td>$-2d_1$</td>
<td>$\alpha$</td>
<td>$\alpha$</td>
</tr>
<tr>
<td>ATO</td>
<td>0</td>
<td>0</td>
<td>$d$</td>
<td>0</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 5.2: Choice of constants for elliptical PCO and ATO reference trajectories obtained from Eq. (3.24).

<table>
<thead>
<tr>
<th>Formation Type</th>
<th>$d_1$ (m)</th>
<th>$d_2$ (m)</th>
<th>$d_3$ (m)</th>
<th>$d_4$ (m)</th>
<th>$d_5$ (m)</th>
<th>$d_6$ (m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCO</td>
<td>0</td>
<td>0</td>
<td>$-\frac{1}{2}\rho$</td>
<td>0</td>
<td>0</td>
<td>$\rho$</td>
</tr>
<tr>
<td>ATO</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>$d$</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

corresponds to maintaining a fixed relative separation, with zero relative velocity. The PCO and ATO reference orbits can be designed using either the solutions to the elliptical equations of relative motion (Eq. (3.24)), or the solutions to the HCW equations (Eq. (3.26)). The selection of constants for these reference trajectories are given in Tabs. 5.1 and 5.2.

5.2 Reconfiguration and Recovery

Here reconfiguration is defined as the method used to switch between spacecraft formations, i.e., reach a final desired relative state from any initial relative state. Recovery is a specific reconfiguration maneuver which is used to re-initialize a formation that was lost because the formation keeping control was halted for some reason (e.g. loss of communication, loss of attitude control, loss of power, etc.). Previously, the reconfiguration maneuvers for CanX-4/-5 were designed using the method outlined in [47], which employed a gradient-based optimization technique to seek out fuel optimal (minimum $\Delta V$) two impulse maneuvers with constraints such that the final desired state was reached in the desired amount of time. This approach was selected rather than a conventional analytical technique such as in [41], because it was found that these methods did not work well in a perturbed orbital environment. The control thrusts were calculated using the HCW STM, which typically leads to inaccurate maneuvers in a high fidelity simulation. However, this fact was compensated by the optimization.

The pitfalls of this technique are the computational time and resources required to compute the maneuver, which requires several iterations of an SQP algorithm. This would not allow the maneuvers to be calculated onboard the satellites in real time. Further, there is a risk of the optimizer exploiting flaws in the employed orbit model so that maneuver accuracy decreases when subject to a different orbit environment, or that the final solution is sensitive to initial conditions.

As a result, a better reconfiguration strategy was required: one that addressed the issue of
accuracy, yet was robust to errors of all types and could be calculated in real time. To meet this need, a fuel-optimal, computationally efficient reconfiguration algorithm based on relative motion STMs was developed. This algorithm assumes that the thrusts are impulsive in nature, i.e., the velocity changes instantaneously. Robustness is achieved by recalculating the remaining thrusts in the maneuver after a thrust has been performed, which essentially closes the control loop and compensates for hardware and relative dynamic modeling errors. Furthermore, since the formulation applies to any STM, the algorithm’s performance can be assessed using several different motion models.

In addition, a continuous time version of the solution is developed, and it is shown that the continuous time solution can be derived from the discrete impulsive solution as the number of impulsive thrusts approaches infinity. This solution is purely of theoretical interest, since the flight hardware prevents continuous maneuvers from being applied in practice. Both continuous and discrete time controllers are obtained by solving

$$\arg\min_{u,\lambda} J(u, \lambda),$$

where the cost function $J(u, \lambda)$ is appropriately defined for each problem.

### 5.2.1 Preliminaries

The relative dynamics of two satellites in formation flight can be written as a linear time-varying system of equations, or

$$x' = A(\alpha)x(\alpha) + B(\alpha)u(\alpha), \quad (5.15)$$

where $(\cdot)'$ denotes the first derivative with respect to the independent variable $\alpha$ (time or true anomaly). The general solution to Eq. (5.15) can be written as

$$x(\alpha) = \Phi(\alpha, \alpha_0)x(\alpha_0) + \int_{\alpha_0}^{\alpha} \Phi(\alpha, s)B(s)u(s) \, ds, \quad (5.16)$$

where $\Phi(\alpha, \alpha_0)$ is STM from the initial state to some final state.

The satellite formation reconfiguration problem can be stated as follows: Given any initial relative state $x_0$, any final desired relative state $x_f$ (both expressed in the LO coordinate frame), and a final transfer time (expressed in either radians or seconds – depending on the independent variable), what is the optimal control thrust $u(\alpha)$ such that the final state is achieved? Before the solution is presented, some definitions are required.

Let $\mathbb{R}^n$ be an n-dimensional vector space, and let $L^2[\alpha_0, \alpha_f]$ be the vector space of continuous, square-integrable functions defined on the interval $[\alpha_0, \alpha_f]$. Define the inner products in
\[ \mathbb{R}^n \text{ and } L^2[\alpha_0, \alpha_f] \] to be
\[
\langle u, v \rangle_{\mathbb{R}^n} = u^T v, \quad u, v \in \mathbb{R}^n
\]
\[
\langle x, y \rangle_{L^2} = \int_{\alpha_0}^{\alpha_f} x(s)^T y(s) \, ds, \quad x, y \in L^2[\alpha_0, \alpha_f]
\]

Now, let the linear operator \( W \) be defined as
\[
W : L^2[\alpha_0, \alpha_f] \longrightarrow \mathbb{R}^n.
\]
That is, \( W \) is an operator which maps continuous functions on the interval \([\alpha_0, \alpha_f]\) to an \( n \)-dimensional vector. Specifically, for a function \( u \in L^2[\alpha_0, \alpha_f] \), we have
\[
Wu = \int_{\alpha_0}^{\alpha_f} \Phi(\alpha_f, s) B(s) u(s) \, ds.
\]
The adjoint of \( W \) is defined as
\[
W^* : \mathbb{R}^n \longrightarrow L^2[\alpha_0, \alpha_f].
\]
Hence, \( W^* \) is itself an operator which maps an \( n \)-dimensional vector to a continuous function of the interval \([\alpha_0, \alpha_f]\). The adjoint operator is derived as follows, using the relation
\[
\langle Wu, y \rangle_{\mathbb{R}^n} = \langle u, W^* y \rangle_{L^2},
\]
\[
\langle Wu, y \rangle_{\mathbb{R}^n} = \left( \int_{\alpha_0}^{\alpha_f} \Phi(\alpha_f, s) B(s) u(s) \, ds \right)^T y
\]
\[
= \int_{\alpha_0}^{\alpha_f} u(s)^T B(s)^T \Phi(\alpha_f, s)^T y \, ds
\]
but,
\[
\langle u, W^* y \rangle_{L^2}
\]
\[
= \int_{\alpha_0}^{\alpha_f} u(s)^T W^* y \, ds
\]
\[
\therefore W^* = B(s)^T \Phi(\alpha_f, s)^T
\]

5.2.2 Continuous Solution

For an initial relative state \( x_0 \), and a final desired relative state \( x_f \) with corresponding independent variable \( \alpha_f \), Eq. (5.16) can be used to obtain
\[
b = x_f - \Phi(\alpha_f, \alpha_0) x_0 = \int_{\alpha_0}^{\alpha_f} \Phi(\alpha_f, s) B(s) u(s) \, ds,
\]
or

\[ \mathbf{b} = \mathbf{Wu}, \quad (5.23) \]

Note that \( \mathbf{u}(\alpha) \in L^2[\alpha_0, \alpha_f] \). The constraint given by Eq. (5.23) ensures that the final state is attained through application of the control thrust \( \mathbf{u} \).

Since a minimum-fuel solution is required, the cost function for this problem is chosen to be:

\[ J(\mathbf{u}, \lambda) = \int_{\alpha_0}^{\alpha_f} \frac{1}{2} \mathbf{u}(\tau)^T \mathbf{u}(\tau) \, d\tau + \lambda^T (\mathbf{b} - \mathbf{Wu}), \quad (5.24) \]

where the Lagrange multiplier \( \lambda \) has been introduced to enforce the constraint. Note that here \( \lambda \) is a constant vector because the constraint is only enforced at one point in time. Also note that since \( \mathbf{u}(\alpha) \) has units of acceleration, this can be thought of a minimum-power reconfiguration maneuver. Before taking the first variation, we rewrite \( J(\mathbf{u}, \lambda) \) as

\[
J(\mathbf{u}, \lambda) = \int_{\alpha_0}^{\alpha_f} \frac{1}{2} \mathbf{u}(\tau)^T \mathbf{u}(\tau) \, d\tau + \langle \lambda, \mathbf{Wu} - \mathbf{b} \rangle_{\mathbb{R}^n} \\
= \int_{\alpha_0}^{\alpha_f} \frac{1}{2} \mathbf{u}(\tau)^T \mathbf{u}(\tau) \, d\tau + \langle \lambda, \mathbf{Wu} \rangle_{\mathbb{R}^n} - \langle \lambda, \mathbf{b} \rangle_{\mathbb{R}^n} \\
= \int_{\alpha_0}^{\alpha_f} \frac{1}{2} \mathbf{u}(\tau)^T \mathbf{u}(\tau) \, d\tau + \langle \mathbf{W}^* \lambda, \mathbf{u} \rangle_{L^2} - \langle \lambda, \mathbf{b} \rangle_{\mathbb{R}^n} 
\]

Then, the first variation of \( J \) is given by:

\[
\delta J = \int_{\alpha_0}^{\alpha_f} \left( \mathbf{u}^T \delta \mathbf{u} + (\mathbf{W}^* \lambda)^T \delta \mathbf{u} + \mathbf{u}^T \mathbf{W}^* \delta \lambda \right) \, ds - \mathbf{b}^T \delta \lambda = 0 \quad (5.26)
\]

After some manipulation, \( \delta J \) can be re-written as

\[
\delta J = \int_{\alpha_0}^{\alpha_f} (\mathbf{u} + \mathbf{W}^* \lambda)^T \delta \mathbf{u} \, ds + (\mathbf{Wu} - \mathbf{b})^T \delta \lambda = 0 \quad (5.27)
\]

Since \( \delta \mathbf{u} \) and \( \delta \lambda \) are arbitrary, we must have

\[ \mathbf{u} = -\mathbf{W}^* \lambda \quad (5.28) \]

and

\[ \mathbf{Wu} = \mathbf{b} \quad (5.29) \]

Pre-multiplying Eq. (5.28) by \( \mathbf{W} \) and applying Eq. (5.29) gives

\[ \mathbf{b} = -\mathbf{WW}^* \lambda \quad (5.30) \]
At this point, note that
\[ WW^* = \int_{\alpha_0}^{\alpha_f} \Phi(\alpha_f, s) B(s) B(s)^T \Phi(\alpha_f, s)^T \, ds \] (5.31)

is the controllability Gramian. This matrix will be invertible so long as the system is controllable.

The argument for controllability is as follows: Assuming the spacecraft in question has enough fuel, any relative state of interest can be reached in a finite amount of time. Since any relative state can be reached in finite time, the system must be controllable, by definition. Since the system is controllable, then the controllability Gramian must have full rank, since both statements are equivalent. As a result, the inverse of the controllability Gramian used in this solution will always exist and a continuous control force for reconfiguration can always be calculated.

Now, inverting Eq. (5.30) to solve for \( \lambda \) and substituting the result into Eq. (5.28) we arrive at the final form of the optimal control function:

\[
\begin{align*}
\mathbf{u}(\alpha) &= W^* (WW^*)^{-1} \mathbf{b} \\
\mathbf{u}(\alpha) &= \mathbf{B}(\alpha)^T \Phi(\alpha_f, \alpha)^T \left( \int_{\alpha_0}^{\alpha_f} \Phi(\alpha_f, s) B(s) B(s)^T \Phi(\alpha_f, s)^T \, ds \right)^{-1} \cdot (\mathbf{x}_f - \Phi(\alpha_f, \alpha_0) \mathbf{x}_0) 
\end{align*}
(5.32)
\]

### 5.2.3 Discrete (Impulsive) Multiple-Thrust Solution

In many formation flying missions, continuous thrust may not be feasible or even practical. For one, the thrust magnitudes may be smaller than the thrust system is capable of providing. Second, thrust direction may be constrained in the absence of full three-axis control. This is the case of CanX-4/-5, which has thrusters mounted on only one face of the satellite. Thrust vectoring is achieved by orienting the satellite along the desired thrust direction using the attitude control system. In this situation, purely impulsive thrusts are required. In what follows, the optimal reconfiguration thrust solution is modified to admit an arbitrary number (assumed to be \( \geq 2 \)) of impulsive thrusts. The major assumption here is that the reconfiguration control thrust takes the form

\[
\mathbf{u}(\alpha) = \mathbf{u}_1 \delta(\alpha - \alpha_1) + \mathbf{u}_2 \delta(\alpha - \alpha_2) + \cdots + \mathbf{u}_n \delta(\alpha - \alpha_n) \\
= \sum_{i=1}^{n} \mathbf{u}_i \delta(\alpha - \alpha_i),
(5.33)
\]

where \( \delta(\cdot) \) is the Dirac delta function, the \( \mathbf{u}_i \) are vectors along the thrust direction at \( \alpha_i \), and \( \alpha_0 < \alpha_1 < \cdots < \alpha_n < \alpha_f \).

Substituting Eq. (5.33) into Eq. (5.22) allows the integral term to be easily evaluated using the
sifting property of the delta function, which leads to

\[ \mathbf{b} = \mathbf{x}(\alpha_f) - \Phi(\alpha_f, \alpha_0) \mathbf{x}(\alpha_0) \]

\[ = \Phi(\alpha_f, \alpha_1) \mathbf{B}(\alpha_1) \mathbf{u}_1 + \cdots + \Phi(\alpha_f, \alpha_n) \mathbf{B}(\alpha_n) \mathbf{u}_n \]

\[ = \sum_{i=1}^{n} \Phi(\alpha_f, \alpha_i) \mathbf{B}(\alpha_i) \mathbf{u}_i \]  

(5.34)

Written in matrix/vector form this expression becomes

\[ \mathbf{b} = [\Phi(\alpha_f, \alpha_1) \mathbf{B}(\alpha_1) \Phi(\alpha_f, \alpha_2) \mathbf{B}(\alpha_2) \cdots \Phi(\alpha_f, \alpha_n) \mathbf{B}(\alpha_n)] \mathbf{u} \]

\[ = \mathbf{Q} \mathbf{u}, \]  

(5.35)

where \( \mathbf{u} \in \mathbb{R}^{3N} \) is a vector with N impulsive control thrusts stacked on top of one another, and \( \mathbf{Q} \in \mathbb{R}^{n \times 3N} \) is a matrix representing the system dynamics. Eq. (5.35) is the discrete analogue of Eq. (5.23), giving the constraint between the discrete control thrusts and the initial and final states.

Similarly to the continuous-time case, the cost function is defined as

\[ J(\mathbf{u}, \lambda) = \frac{1}{2} \mathbf{u}^T \mathbf{u} + \lambda^T (\mathbf{b} - \mathbf{Q} \mathbf{u}), \]  

(5.36)

since Eq. (5.35) has infinitely many solutions for the \( \mathbf{u}_i \) and we desire a minimum-thrust solution. Owing to the impulsive thrust assumption (instantaneous change in velocity), \( \mathbf{u}_i \) has units of velocity. Hence, this can be thought of as minimum \( \Delta V \) reconfiguration maneuver. It is possible to construct a cost function with a weighting on each of the control thrusts, but this is not done here since there would be no systematic way of selecting the weights. Eq. (5.36) can be minimized with respect to both \( \mathbf{u} \) and \( \lambda \) by using the first-order optimality conditions; differentiating with respect to each of these variables, setting the results to zero and solving the system of equations for the optimal control forces at each \( \alpha_i \). This process is shown below:

\[ \frac{\partial J}{\partial \mathbf{u}} = \mathbf{u}^T - \lambda^T \mathbf{Q} = 0 \]

\[ \mathbf{u} = \mathbf{Q}^T \lambda \]  

(5.37)

Pre-multiply Eq. (5.37) by \( \mathbf{Q} \) and use Eq. (5.35) to get

\[ \mathbf{b} = \mathbf{Q} \mathbf{Q}^T \lambda. \]  

(5.38)

Solving for \( \lambda \) and substituting back into Eq. (5.37) results in final form of the optimal N-thrust transfer maneuver. It should be said here that there exist cases where the inverse of \( \mathbf{Q} \mathbf{Q}^T \) does not exist when the HCW STM is used. One such case is a two-thrust maneuver when the thrust times are spaced apart by exactly one Chief orbital period. Since the portion of the HCW used
to form $Q$ contains only sinusoidal terms, the resulting matrix will be rank deficient, and no control force can be computed. However, if a three-thrust maneuver with thrusts at the start, end, and middle of on orbit are used, a solution will exist. Thus, these special cases must simply be avoided when implementing this method in practice.

$$u = Q^T \left( Q Q^T \right)^{-1} b$$  \hspace{1cm} (5.39)

Furthermore, the $i^{th}$ control thrust is given by

$$u_i = B(\alpha_i)^T \Phi (\alpha_f, \alpha_i)^T \left( \sum_{i=1}^{N} \Phi (\alpha_f, \alpha_i) B(\alpha_i) B(\alpha_i)^T \Phi (\alpha_f, \alpha_i)^T \right)^{-1} b.$$  \hspace{1cm} (5.40)

Note that due to the impulsive thrust assumption, these control thrusts have units of velocity.

5.2.4 Equivalence of Discrete and Continuous Solutions

Consider Eq. (5.40), which gives the instantaneous changes in velocity for each impulsive thrust. Furthermore, consider a situation with $M$ equally spaced thrusts distributed throughout the transfer interval, each with a pulse width of $\Delta t$. Multiplying both sides of Eq. (5.40) by $\frac{1}{\Delta t}$ we obtain

$$\frac{u_i}{\Delta t} = B(\alpha_i)^T \Phi (\alpha_f, \alpha_i)^T \left( \sum_{i=1}^{N} \Phi (\alpha_f, \alpha_i) B(\alpha_i) B(\alpha_i)^T \Phi (\alpha_f, \alpha_i)^T \right)^{-1} b \frac{1}{\Delta t}.$$  \hspace{1cm} (5.41)

Now recognize that if the term $\frac{1}{\Delta t}$ is brought inside the inverse and we take the lim$_{\Delta t \to 0}$ of both sides we obtain

$$a_i = B(\alpha_i)^T \Phi (\alpha_f, \alpha_i)^T \left( \int_{\alpha_0}^{\alpha_f} \Phi (\alpha_f, s) B(s) B(s)^T \Phi (\alpha_f, s)^T \, ds \right)^{-1} b,$$  \hspace{1cm} (5.42)

where $a_i$ is the instantaneous value of the acceleration at point $\alpha_i$ in the orbit. However, this is identical to the instantaneous acceleration one would compute from Eq. (5.32), the continuous thrust solution. Hence, both methods are equivalent. Using a similar argument, it is also possible to show that the overall fuel consumption ($\Delta V$) for each method is equivalent as the number of pulses approaches infinity; however, it is not yet known which method is more fuel efficient for a given final state error. The impulsive-thrust solution sometimes achieves lower $\Delta V$ at the cost of higher final state errors. Depending on the transfer accuracy requirement, this can be exploited to compute fuel-optimal reconfigurations with acceptable state errors by solving this same optimization problem with inequality constraints. However, this would necessitate the use of a constrained nonlinear equation solver which is not practically feasible.
onboard the CanX-4/-5 satellites.

5.2.5 Robust Reconfigurations

It was found in simulation that the multiple-thrust reconfigurations presented in Sec. 5.2.3 may actually lead to very large final errors, as a result of thrust magnitude errors, thrust alignment errors, and errors in the relative dynamics model. Consequently, a method to correct the calculated thrust solution to adjust for these errors was devised. At the same time, the method is computationally efficient since costly function evaluations are minimized.

Eq. (5.40) gives the expression for the $i^{th}$ optimal impulsive thrust in a sequence of $N$ thrusts. After the first thrust is applied, we form a new reconfiguration cost function

$$J(u', \lambda) = \frac{1}{2} u'^T u' + \lambda^T (b' - Q'u') ,$$

where

$$u' = \begin{bmatrix} u_2 & \cdots & u_N \end{bmatrix}^T$$

$$b' = x(\alpha_f) - \Phi(\alpha_f, \alpha_{rc}) x(\alpha_{rc})$$

$$Q' = \begin{bmatrix} \Phi(\alpha_f, \alpha_2) B(\alpha_2) & \cdots & \Phi(\alpha_f, \alpha_n) B(\alpha_n) \end{bmatrix}^T ,$$

where $\alpha_{rc}$ is the point at which the solution is calculated. It is important to note that the thrust times denoted by $\alpha_i$ remain the same as in the original problem, Eq. (5.36). Solving Eq. (5.43), we obtain a new $N - 1$ thrust solution

$$u' = Q'^T \left( Q'Q'^T \right)^{-1} b'.$$

After applying the second thrust, the $N - 2$ thrust solution is calculated in a similar manner. This procedure repeats until the two thrust solution, at which point no more recalculations are performed. This procedure is represented in Fig. 5.1.

![Diagram of thrust recalculation procedure](image)

Figure 5.1: Thrust recalculation procedure to greatly increase robustness.

There are two main benefits to this thrust recalculation strategy. First, it results in far
more accurate maneuvers than the nominal reconfiguration strategy, as will be shown in Chap. 6. This is because we are essentially ‘closing the loop’, so that future thrusts take into account past errors. Second, it is efficient since the $Q$ matrices need only be calculated once at the start of the maneuver and subsequently stored in memory, since they are dependently solely on time and Chief parameters. Future thrust recalculations can then directly use these values, which reduces the number of costly function evaluations.

### 5.3 Attitude Slews

Since the CanX-4/-5 satellites have only one axis of thrust, before each control thrust, it is necessary to align the thrust axis with the desired thrust direction. This is accomplished via the attitude control system, OASYS. However, since a finite amount of time is required to slew to the target attitude, one must know in advance the desired thrust direction. Referring to Fig. 5.2, in the past this was done by predicting the desired attitude ahead of time (at 1), by taking the current relative state estimate, projecting this forward in time by some $\Delta T_{\text{slew}}$ using the nonlinear equations of motion, and calculating the control/attitude vector based on this. During $\Delta T_{\text{slew}}$, OASYS maneuvers its way to the predicted target attitude. Then, when the $\Delta T_{\text{slew}}$ has elapsed (at 2), one uses the most recent relative state estimate to recalculate the control and provide an updated attitude estimate, the idea being the predicted and updated attitude states are very close to one another, so only a minor correction is required [39]. The thrust is performed at the desired time after the minor correction.

![Figure 5.2: Depiction of ACS target prediction scheme.](image)

Unfortunately, it was found that this attitude prediction scheme did not work very well in practice, in that the predicted and updated target attitudes were very far apart, thus leaving a very short time span to complete the remainder of the attitude maneuver to perform the required control thrust. An example of these attitude spikes is shown in Fig. 5.3. Each line denotes the normalized component of the commanded attitude vector in the GCI frame. Ideally the blue lines would be flat or mildly kinked, representing identical or similar predicted and actual attitude targets. The jumps in the blue lines represent the variation due to real-time updates of the commanded attitude, which are calculated using the current relative state estimate. These attitude spikes were found to be related to the accuracy of the relative velocity estimate, where perfect knowledge results in identical predicted and corrected values.
Figure 5.3: Attitude spikes resulting from prediction scheme. The three plots denote the X, Y, and Z components of the target attitude vector in the GCI frame. The red lines denote a spike-free series of targets.

A second argument against the original attitude targeting scheme is the fact that from a GPS state estimation standpoint, it is more advantageous to slew less often, since this increases the time spent observing the same GPS satellites thus improving both absolute and relative state estimates.

As a result, different schemes were sought so that the targeted attitude was never corrected. In this way, OASYS will always achieve the desired attitude in the given amount of time. The two methods presented below differ in how the control force is calculated. In method one, the control for the current time is calculated and applied later, while in method two the control for some future time is calculated and applied at the future time.

5.3.1 Method 1

Again referring to Fig. 5.2, the first new scheme simply calculates the control at time ‘k’ using the current state estimate, and sets the attitude target. This target is never changed. After the satellite has slewed to the desired attitude, the thrust is applied. This cycle then repeats.

Stability of the linear system can be shown by ensuring that the eigenvalues of the closed-loop system all lie within the unit circle. Consider the zero-order hold (ZOH) discretized linear time invariant plant (representing the HCW equations) with control $u_k = -K (x_{k-1} - x_{ref,k-1})$,
which can be written as

$$q_{k+1} = \begin{bmatrix} A_d & -B_dK \\ I_{6 \times 6} & 0_{6 \times 6} \end{bmatrix} q_k + \begin{bmatrix} B_dK \\ 0_{6 \times 6} \end{bmatrix} x_{ref,k-1} = A_{cl} q_k + \overline{B} x_{ref,k-1}, \quad (5.46)$$

where $q_k = [x_k^T, x_{k-1}^T]^T$ is the augmented state, and the ZOH discretization period is the PWM period. Then we need only check that the eigenvalues of $A_{cl}$ lie within the unit circle. Note that the gain matrix $K$ can be any gain matrix that has been designed via the methods of Sec. 5.1.1. Note that even if this check passes, one is not ensured closed-loop stability within the nonlinear system.

The main disadvantage of this new attitude targeting scheme is increased fuel use and tracking error. However, this is preferable to the old method because the attitude spikes mean the control would never actually be performed.

### 5.3.2 Method 2

The second new scheme is not really new: it involves keeping the old ACS prediction method without performing the real-time updates to the attitude target. That is, the desired attitude is calculated at time ‘$k$’ by taking the current relative state estimate, integrating it with the nonlinear equations of motion for the time span $\Delta T_{slew}$ and calculating the required control at this future time, then applying the predicted control at that time.

This method has the advantage of reducing the tracking error, since in Method 1 one ends up tracking a delayed reference trajectory causing non-zero mean errors. However, this method has an increased computational load associated with it, due to the integrations required for the prediction of the future relative state. With this scheme, it is important that the desired execution time of FIONA (approximately one second) is not exceeded.

For this case, stability in the linear case can be shown as follows: First, assume that the time line in Fig. 5.2 has a total length $\Delta T$ (distance between $k$ and $k + 1$) and that it is broken up into $n$ sub-intervals such that $n \cdot \Delta t = \Delta T$. Then, let $x_{k}^{i}$ be the state at the end of sub-interval $i$, where $x_{k}^{i} = x_{k+1}$. Now, we wish to show that the transition from $x_k$ to $x_{k+1}$ is stable given a control calculated some intermediate state $x_k^{i}$.

The transition from $x_k$ to $x_k^{i}$ is written as

$$x_k^{i} = A_d^i x_k, \quad (5.47)$$

where $A_d$ has been appropriately discretized based on the time step $\Delta t$. For the control step,

$$x_{k}^{i+1} = (A_d - B_dK) x_k^{i}, \quad (5.48)$$
Lastly,

\[ x_{k+1} = x_k^n = (A_d)^{n-(i+1)} x_{i+1}^k. \]  

(5.49)

This leads to

\[ x_{k+1} = (A_d)^{n-(i+1)} (A_d - B_d K) A_i^d x_k = A_{cl} x_k, \]  

(5.50)

so that stability can be verified by ensuring the eigenvalues of \( A_{cl} \) all lie within the unit circle.

Again, stability in the linear case does not guarantee stability within the nonlinear simulation, however it is a prerequisite.
Chapter 6

Simulation and Hardware Test

Results

This chapter discusses the expected performance of the formation control and estimation algorithms discussed in previous chapters. First, the formation flying integrated onboard navigation algorithm (FIONA), in which these algorithms have been implemented is discussed. Second, the reconfiguration methods derived in Chap. 5 are tested in four representative scenarios. Next, FIONA is tested on representative flight hardware to determine critical function timings. Then, FIONA’s performance during three full mission simulations under various conditions is discussed. Lastly, the differential GPS navigation algorithm is tested under static conditions to ascertain its functionality.

6.1 FIONA

FIONA is the control algorithm that was designed in [39] to perform the navigation and control tasks for the CanX-4/-5 mission. The current version of FIONA is not fully representative of its flight configuration. This is because during the mission, there will be a high level of interaction between the ground station operator and the two satellites to command changes between fine and coarse formation keeping, command formation reconfigurations, set control gains and EKF parameters, update reference trajectories, etc. Since this interaction is not practicable during development, the current implementation is designed to autonomously complete the mission sequence, with all preset parameters fixed for the simulation duration. A high level flowchart of the algorithm is given in Fig. 6.1. The algorithm described herein has been implemented in C, and is used in Secs. 6.3, 6.5 and 6.6.

Before operation, FIONA must be initialized to set the basic parameters that determine the desired mode, the EKF process and measurement noise, the fidelity of the orbit propagation
model, control thrust frequency, integration time steps, and control gains, among other things. During the mission, it will be possible to change any of these parameters from the ground, to fine tune FIONA’s control and estimation performance on-orbit, before formation flying begins.

After initialization and upon subsequent calls, FIONA executes a predefined set of commands based on the current mode of operation and set of inputs. Calls to FIONA must occur at a fixed rate, currently set to 0.2 Hz. The actual rate is not important (e.g. 0.21 Hz versus 0.19 Hz), as long as the input data itself corresponds to 0.2 Hz. The input data consists of absolute position and velocity of the Chief and Deputy and their estimated standard deviations (from the GPS receivers), a relative position and velocity (from the fine navigation algorithm) and flags denoting the reliability of these quantities.

At each time step, the formation tracking error is assessed: if the position or velocity tracking errors exceed a predefined value, the control authority is increased. If the tracking errors further increase beyond the stability bounds of the LQR, emergency hold mode (EHM) is activated and all further formation keeping actions are halted. EHM can also be activated by the ground station to halt FIONA’s operation for any other reason. To avoid activating EHM during reconfiguration mode, tracking errors are only calculated during formation keeping. During EHM, FIONA performs all calculations as under nominal operation, except no control thrusts or attitude slews are commanded.

Next the coarse EKF as described in Sec. 4.2 is executed. Currently the “prediction” step of the EKF occurs at the end of the current time step (i.e., after the mode executions) rather than at the beginning of the following time step. The only implication is that commanded thrusts are included in the state propagation before it is known whether or not they have occurred. If the commanded thrust does not occur, or ends up being longer or shorter than desired for whatever reason, the coarse mode state estimate will be negatively affected. After completion of the EKF correction step, the relative state is obtained. If an accurate relative state from the GPS fine mode algorithm (see Sec. 4.3) is available (i.e., fixed solution – corrected with integer DD ambiguities), it will be used. Otherwise, the state is obtained from a simple difference of the Chief and Deputy absolute state estimates. It is possible that in the future, the relative
state estimate will come from the fine mode filter, regardless of whether or not it has been fixed (i.e., use the SD ambiguity float solution of the relative state). This decision is left for the future, until the performance and accuracy of the relative navigation filter is assessed.

The next step is to transfer between operational modes, if required. Currently a transition to reconfiguration mode is performed if currently in formation keeping mode, the satellites have completed the desired number of orbits in the current formation, and EHM is not activated. A transition to formation keeping mode is performed from reconfiguration mode if the reconfiguration maneuver has been completed, and the final errors are below specified tolerances. If the reconfiguration has not been successful, it will automatically be attempted again. The current architecture supports commanded reconfiguration maneuvers, since any formation-altering action must be authorized by the ground station.

The FIONA mode dictates how and when the next control force will be commanded. If in separation mode, the satellites will drift uncontrolled until commanded to reconfigure into the first formation (1000 m ATO). At this point separation mode becomes indistinguishable from reconfiguration mode. During the first step of reconfiguration mode, the control thrusts (directions and magnitudes) for the maneuver are calculated as discussed in Sec. 5.2.3. Subsequent calls monitor the current time in relation to the desired thrust times and recalculate the solution (see Sec. 5.2.5) approximately one minute before the thrust, to allow the ACS to slew to the target attitude. If a commanded thrust is missed due to EHM, the entire maneuver still continues because it is possible to complete it if only some thrusts are missed. The commanded attitude, thrust start time, and thrust impulse are stored in a data structure to later be accessed by CNAPS to execute the thrust.

In formation keeping mode the thrusts occur at discrete intervals, depending on the current control period. At the beginning of each control period, the desired attitude target, thrust start time, and thrust impulse are calculated. The exact value of the control depends on the current formation, whether in coarse or fine estimation mode, whether or not more control authority is required to reduce tracking errors, and the current orbit number within a formation. The first and last orbits in each formation have a short control period (currently 75 s) to achieve fine formation control before and after reconfiguration maneuvers, helping to reduce overall control errors. Frequent thrusting will also occur if the tracking error exceeds a predefined limit, to ensure it never gets too large. The control actions during the short control periods are calculated using the steady state continuous time LQR presented in Sec. 5.1.1.1. All other orbits in a formation have a longer control period (currently 215 s) to increase the time between thrusts to allow time for the fine navigation filter to converge. Control actions during the long period are calculated using the steady-state discrete time LQR from Sec. 5.1.1.2. After setting a control thrust, the system monitors the current time in relation to the desired time and
commands a thrust when necessary.

Lastly, the FIONA status for the current time step is recorded. This includes the current time, desired relative separation (ATO separation or PCO radius), the current formation type, the number of orbits in the current formation, whether a thrust is commanded or not, GPS measurement status, and EHM status. At this point the output data is recorded and the algorithm stops and awaits the next call.

6.2 Reconfiguration Algorithm Test

The reconfiguration algorithms presented in Sec. 5.2 were implemented and simulated in MATLAB to assure their functionality before implementing the algorithm in C, to add to FIONA. These simulations also serve to evaluate the algorithms’ performance with varying numbers of impulsive thrusts and different relative motion models, to gain a better understanding of how to set these parameters in FIONA.

For all the test cases presented below, the orbits were integrated using a fourth order Runge-Kutta method with a one second time step. The initial osculating orbital elements for the Chief spacecraft are given in Tab. 6.1 and are consistent with the desired orbit for the CanX-4/-5 mission. The orbital dynamics employed the analytical expressions for the $J_2$-$J_6$ perturbations. The thrust level is constant at 15 mN (three active thrusters) and the spacecraft wet mass is 7 kg, which is assumed not to change over the course of a simulation. No thruster bias, pointing errors, thrust timing, or state estimation errors are assumed. Thus the dominant sources of error are the relative dynamics models, the non-impulsive nature of the thrusts, and the quality of the selected thrust times. See App. D for a list of the numerical values of the simulation constants.

<table>
<thead>
<tr>
<th>Chief Orbital Elements</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$h_a$</td>
<td>550 km</td>
</tr>
<tr>
<td>$a = \frac{h_a + R_\oplus}{1-e}$</td>
<td>7105781.54 m</td>
</tr>
<tr>
<td>$e$</td>
<td>0.025</td>
</tr>
<tr>
<td>$i$</td>
<td>97.6°</td>
</tr>
<tr>
<td>$\Omega$</td>
<td>99.56°</td>
</tr>
<tr>
<td>$\omega$</td>
<td>0.0°</td>
</tr>
<tr>
<td>$M_0$</td>
<td>0.0°</td>
</tr>
</tbody>
</table>

Table 6.1: Initial conditions for Chief spacecraft in reconfiguration algorithm tests, where $h_a$ is the altitude at periapsis.

The initial and final desired relative states are generated from the initial conditions from Tab. 5.1 which describe bounded relative motions for the HCW equations. The parameters describing the start and end conditions of each case are given in Tab. 6.2.
Table 6.2: Deputy initial condition parameters in reconfiguration algorithm tests.

<table>
<thead>
<tr>
<th>Simulation Case</th>
<th>Initial Parameters</th>
<th>Final Desired Parameters</th>
<th>Transfer Time</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(d_i) (m)</td>
<td>(\rho_i) (m)</td>
<td>(\alpha_i) (deg)</td>
</tr>
<tr>
<td>Case 1</td>
<td>500</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Case 2</td>
<td>3000</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Case 3</td>
<td>0</td>
<td>50</td>
<td>10</td>
</tr>
</tbody>
</table>

Each multiple-thrust maneuver consists of seven impulses: one at the beginning and end of the maneuver, and five spaced equally throughout the transfer interval. The thrust times for each of \(N\) equally spaced thrusts are given by

\[
t_i = (i - 0.5) \frac{t_f - t_0}{N},
\]

where \(t_0\) and \(t_f\) are the initial and final times, respectively. The multiple-impulse maneuvers are compared to the two-impulse maneuvers, where thrusts are applied only at the beginning and end of the transfer interval. An example of the computed reconfiguration thrusts is given in Fig. 6.2, which is taken from Case 1. Furthermore, the three state transition matrices discussed in Sec. 3.4.4 are compared for each maneuver.

Figure 6.2: Reconfiguration thrust example.

The ultimate goal of a reconfiguration maneuver is to provide “accurate” transfer between formations such that the LQR controller can stabilize the new formation without using too much fuel. As such, it is important to that the final position and velocity errors are as close
to zero as possible. In [47], the optimizations were constrained such that the final position and velocity errors are less than 2.5 m and 10 cm/s, respectively, since it was found that the LQR is able to stabilize the formation with these error levels. These are the figures of merit chosen in what follows.

It should be noted that the continuous thrust maneuvers were also executed, but are excluded from the results below for two main reasons. First, they were found to be equivalent in accuracy to the non-robust multiple thrust maneuvers. This result is surprising, since these maneuvers were expected to be the most accurate in the absence of thrust errors. The result may be due to relative dynamics modeling errors, but similar continuous time formulations in [42] and [44] encountered no such difficulties, so a programming error may be at fault. Further investigations were not performed, due to the second reason: the continuous time formulation cannot be applied to the CanX-4/-5 mission. Future investigations could examine the exact cause of the continuous thrust reconfiguration maneuver errors.

6.2.1 Case 1 - ATO to PCO

This case simulates a transfer between a 500 m ATO and a 50 m PCO over one and a half Chief orbits. It is assumed that the Deputy starts the transfer maneuver perfectly in formation, i.e., there are no initial relative state errors. Fig. 6.3 shows the magnitude of the position and velocity errors at the end of the reconfiguration maneuver. Mathematically, the final relative state errors are expressed as

$$\delta R = \| \delta R_{des} - \delta R(t_f) \|, \text{ and } \delta V = \| \delta V_{des} - \delta V(t_f) \|, \quad (6.2)$$

where \((\cdot)_{des}\) indicates the desired final state, and \(t_f\) is the final time, i.e., the end of the specified transfer period. If the final thrust was forced to exceed the transfer period because of actuator saturation, the errors are calculated at the end of the final thrust.

It can be seen from Fig. 6.3 that the Ankersen-Yamanaka (AY) and Gim-Alfriend (GA) STMs have lower position and velocity errors than the Hill-Clohessy-Wiltshire (HCW) STM, primarily because the latter does not account for the eccentricity of the Chief’s orbit. The reason why the HCW-based multiple impulse maneuver performs worse than the two-impulse maneuver is unknown, but it may again be related to its poor relative dynamics model. For the AY and GA STMs, the multiple thrust maneuvers alone produce more accurate maneuvers. For all STMs, the final position error is reduced by employing the recalculation scheme; however, the relative velocity error for the AY STM actually increases. Regardless of this increase, the final velocity error is acceptable, thus the exact cause was not investigated, though the finite-pulse nature of the thrusts may be the cause. Overall, the GA STM shows the best results.
Figure 6.3: Final errors for Case 1 reconfiguration maneuver. The arrows denote the improvement in final accuracy achieved by using the robust transfer method.

followed by the AY then HCW STMs, although all methods produce acceptable final errors using the proposed method.

The fuel use for these maneuvers is given in Tab. 6.3. The high fuel use for the two-impulse maneuvers compared to the multiple-impulse maneuvers is not justified, given the poorer performance. While it seems as though the two-impulse maneuvers are quite poor, it is important to keep in mind fuel use and final accuracy could be considerably improved through better thrust time selection. There is no associated fuel penalty for the robust method, save for the HCW STM which suffers from the aforementioned poor dynamic modeling.

### Table 6.3: Fuel usage [cm/s] for Case 1 reconfiguration maneuver.

<table>
<thead>
<tr>
<th>STM</th>
<th>Maneuver Type</th>
<th>Two-Impulse</th>
<th>Multi-Impulse</th>
<th>Robust</th>
</tr>
</thead>
<tbody>
<tr>
<td>HCW</td>
<td></td>
<td>39.6</td>
<td>8.8</td>
<td>18.0</td>
</tr>
<tr>
<td>AY</td>
<td></td>
<td>40.4</td>
<td>9.3</td>
<td>9.4</td>
</tr>
<tr>
<td>GA</td>
<td></td>
<td>39.3</td>
<td>9.3</td>
<td>9.3</td>
</tr>
</tbody>
</table>

6.2.2 Case 2 - ATO to ATO

This case simulates a transfer between a 3000 m ATO and a 500 m ATO over two Chief orbits. Again, it is assumed that the Deputy starts the transfer maneuver perfectly in formation, so...
that the initial relative velocity is identically zero. The final relative position and velocity error magnitudes for each method are given in Fig. 6.4. Again, the arrows denote the improvement in final accuracy achieved by using the thrust recalculation method presented in Sec. 5.2.5.

Similarly to Case 1, the AY and GA STMs give far better results than the HCW STM, both in terms of final position and velocity error. Again, this is primarily because of the HCW’s zero eccentricity assumption, which fails to capture the relative motion accurately. However, contrary to Case 1 the multiple-impulse maneuvers do not improve the final errors; they are at best equal to the nominal two-impulse maneuvers, while using more fuel (Tab. 6.4). Though the reason for this is not known, it may be that a two-impulse maneuver is in fact fuel optimal for this case. Further investigation using the primer vector theory, as shown in [72], would be required to determine this.

For the robust multiple-impulse maneuvers, there is a substantial (>50 m) improvement in final position error, while the final velocity error actually gets worse for the AY and GA STMs. This can only be attributed to the finite-pulse thrusts. Regardless, the increase is marginal and the final velocity error is acceptable given the performance criteria. Even with the thrust recalculations, the HCW STM falls short of the final position error requirement.

The fuel use for the multiple-impulse maneuvers, shown in Tab. 6.4, follows a similar trend to that in Case 1. The robust reconfiguration costs much more fuel for the HCW STM, but very
little or no more for the other STMs: there is very little fuel cost associated with the robust maneuvers. This indicates that the majority of the performance gain comes from changing the desired thrust direction, rather than the magnitude.

<table>
<thead>
<tr>
<th>STM</th>
<th>Maneuver Type</th>
<th>Two-Impulse</th>
<th>Multi-Impulse</th>
<th>Robust</th>
</tr>
</thead>
<tbody>
<tr>
<td>HCW</td>
<td>14.3</td>
<td>19.3</td>
<td>82.0</td>
<td></td>
</tr>
<tr>
<td>AY</td>
<td>16.6</td>
<td>21.2</td>
<td>21.2</td>
<td></td>
</tr>
<tr>
<td>GA</td>
<td>16.6</td>
<td>21.3</td>
<td>22.3</td>
<td></td>
</tr>
</tbody>
</table>

Table 6.4: Fuel usage [cm/s] for Case 2 reconfiguration maneuver.

### 6.2.3 Case 3 - Recovery Maneuver

This case simulates a recovery maneuver where a formation is re-established after a period of uncontrolled relative drift. At the initial time, the Deputy is in a 50 m PCO with a ten degree phase angle. At this point, the satellites are allowed to drift apart for 30000 seconds, or roughly five orbital periods. A reconfiguration is then initiated to recover the 50 m PCO over three Chief orbits.

![Final Position and Velocity Error](image)

Figure 6.5: Final errors for Case 3 reconfiguration maneuver. The arrows denote the improvement in final accuracy achieved by using the robust transfer method.

As shown in Fig. 6.5, without the recalculating the thrust solution, the multiple-impulse maneuvers offer little or no improvement over the baseline two-impulse maneuvers, though
it is interesting to note that less fuel is required for the multiple-impulse maneuvers (Tab. 6.5). Physically this makes sense as rather than attempting to abruptly change the relative velocity twice during a maneuver, which requires substantial control effort, the control effort is distributed throughout the transfer time domain and the natural dynamics of the problem are better incorporated, resulting in less fuel use.

Without recalculating the thrusts in the maneuvers, the final relative position errors are not acceptable. However, using the robust method the final position and velocity errors can be reduced by more than 45 m and 1 cm/s, respectively, using the robust method. This results in acceptable final errors for all STMs considered. As seen in Tab. 6.5, in this case the fuel increases by more than a factor of two for the robust maneuvers. However, this increase is justifiable given the performance increase. Further, the overall fuel use remains relatively small compared to Cases 1 and 2.

<table>
<thead>
<tr>
<th>STM</th>
<th>Maneuver Type</th>
<th>Two-Impulse</th>
<th>Multi-Impulse</th>
<th>Robust</th>
</tr>
</thead>
<tbody>
<tr>
<td>HCW</td>
<td>5.5</td>
<td>1.3</td>
<td>4.0</td>
<td></td>
</tr>
<tr>
<td>AY</td>
<td>6.1</td>
<td>1.4</td>
<td>2.9</td>
<td></td>
</tr>
<tr>
<td>GA</td>
<td>5.3</td>
<td>1.3</td>
<td>2.8</td>
<td></td>
</tr>
</tbody>
</table>

Table 6.5: Fuel usage [cm/s] for Case 3 reconfiguration maneuver.

6.2.4 Case 4 - Varying Transfer Duration

In order to assess the effect of extending the total thrust time on overall fuel consumption, Case 2 was repeated with various total transfer times and the total number of thrusts remaining constant. That is, in Eq. (6.1) only $t_f$ was modified (and $t_0 = 0$). Tab. 6.6 gives the final errors and fuel use for increasing total thrust times. Only the robust reconfiguration method is used with the AY STM.

Interestingly, the overall fuel use first begins to decrease before seemingly reaching a minimum and then increasing again. The increase in errors for the four orbit maneuver is thought

<table>
<thead>
<tr>
<th>$t_f$ (Chief periods)</th>
<th>$\delta R$ (m)</th>
<th>$\delta V$ (mm/s)</th>
<th>$\Delta V$ (cm/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4.5</td>
<td>2.7</td>
<td>36.8</td>
</tr>
<tr>
<td>2</td>
<td>2.3</td>
<td>5.0</td>
<td>21.2</td>
</tr>
<tr>
<td>3</td>
<td>1.8</td>
<td>4.3</td>
<td>14.9</td>
</tr>
<tr>
<td>4</td>
<td>10.9</td>
<td>10.3</td>
<td>14.9</td>
</tr>
<tr>
<td>6</td>
<td>1.3</td>
<td>3.3</td>
<td>7.2</td>
</tr>
<tr>
<td>8</td>
<td>6.3</td>
<td>4.3</td>
<td>8.0</td>
</tr>
<tr>
<td>10</td>
<td>21.5</td>
<td>0.5</td>
<td>8.7</td>
</tr>
</tbody>
</table>

Table 6.6: Final errors and fuel usages for Case 2 reconfiguration with varying transfer times.
to be related to poorly selected thrust times, since the errors for the six orbit maneuver are even lower than that for the nominal two orbit maneuver. These results seem to indicate that total $\Delta V$ can be initially reduced by increasing the transfer time, but that after a point the fuel use again begins to increase. Further, it is likely that with longer transfer times the maneuvers will be less accurate due to the limited predictive power of the STM.

Unfortunately, a method of selecting the optimal total transfer time which minimizes final errors and fuel use is not known; however, the linear primer vector theory may be of some use here. Further, it may be possible to improve the final errors over long transfer periods by selecting more thrusts in the maneuver, or by placing them more intelligently (i.e., optimally). Currently, the best solution is to simply select the transfer times based on mission simulations. These investigations are left as future work.

### 6.2.5 Reconfiguration Test Conclusions

Based on the test cases presented above, some important conclusions about reconfiguration maneuvers can be drawn.

First, in general multiple-thrust reconfiguration maneuvers do not offer any particular advantage over two-thrust maneuvers in terms of final state errors or fuel use. The situations must be analyzed on a case by case basis, since there does not seem to be any obvious trend. As previously mentioned, primer vector theory may be able to answer the question of “when is a multiple-thrust maneuver more fuel efficient than a two-thrust maneuver?”, however the final state errors will always be highly dependent on actuator performance, thrust application timings, thrust impulsivity, and relative dynamics modeling.

Second, it is apparent that employing an STM with a more accurate dynamic model leads to better reconfiguration results. In all cases, the AY and GA STMs outperformed the HCW STM. Using the HCW STM typically resulted in larger (sometimes unacceptable) final relative position and velocity errors, accompanied by a greater fuel consumption for the robust maneuvers. The relative position error for the HCW STM seems to be largest in Case 2, where the transfer distance is large. In the three cases presented, the GA STM leads to the smallest relative velocity errors, while the GA and AY STMs give very similar final position errors. These results make sense, given that both AY and GA account for the eccentricity of the Chief orbit that affects the evolution of the relative position over time, but only the GA STM accounts for the differential drift due to $J_2$, thus its relative velocity estimate is most accurate. That said, the difference in relative velocity error for the two STMs is small, and both errors are well within the acceptable limits. Given these results, only the AY or GA STMs should be implemented onboard.

Third, in some cases a decrease in fuel use is achievable by setting the transfer periods to be
longer, however there may also be a corresponding decrease in accuracy if the transfer interval is too long, since the relative motion models cease to be as accurate. More intelligent methods of selecting transfer time periods have not been investigated to date.

Lastly, these simulations demonstrate that the robust reconfiguration method presented in Sec. 5.2.5 performs as expected, consuming minimal fuel and leading to very small final state errors owing to the “closed-loop” nature of the algorithm which corrects for the errors incurred due to actuator saturation. As will be shown in Sec. 6.5, the algorithm is robust to further errors stemming from estimation errors, actuator biases, missed thrusts, and pointing errors, though the accuracy is somewhat reduced when all sources of error are considered.

6.3 Flight Hardware Implementation

A series of tests in which FIONA and its associated functions were performed on an OBC to answer two main questions. First, what is the general speed of code execution that can be expected on the flight hardware, and how does this change with certain key parameters? Second, what are the parameter sets that will lead to acceptable execution speeds, i.e., around one second?

6.3.1 Function Timing

In order to gain a better idea of how quickly FIONA will execute on the OBCs, timing tests on the flight hardware were performed. Since the operating environment is nominally multi-threaded (i.e., many tasks can be running at the same time) and applications can be interrupted, all other CANOE threads are disabled for these tests. The timing functions use the internal processor clock, and are accurate to the nearest millisecond with a repeatability of about 3 milliseconds. The code was compiled without any optimization settings. Another important point is that full double precision is used for many of the variables of the interest, which will increase the processing time. In the future, the use of float covariances matrices should be investigated to reduce the time associated with this computations. It should be kept in mind that the timings reported here will improve later in the development cycle, as there are various tricks that can be performed to improve code speed. However, since the tests performed here were run in CANOE, which has a nominal clock frequency of approximately 40 MHz, the timings are fairly representative of flight.

Since it was found that the major bottlenecks are the EKF and the reconfiguration algorithm, only these functions are analyzed in detail. The auxiliary functions such as orbital element determination and coordinate frame rotation do not contribute significantly to computational time. Lastly, entire loops of FIONA are analyzed for different modes of operation.
6.3.1.1 Coarse Mode EKF

Tab. 6.7 shows the run time for the EKF correction step. The measurement vector has dimension six for position states and dimension twelve for position and velocity states. The sequential linear update method presented in Sec. 4.1.1 is compared to the traditional update method, while the Joseph form update is included for reference, since this was the original update method. Since the sequential update developed here does not use the Joseph form update, any comparison between it and the traditional Joseph form update is unfair. The averaged results from five trials are presented.

As expected, the linear sequential update is much faster than the traditional update, yielding a 30% and 40% improvement for the six and twelve measurement scenarios, respectively. Further, it is seen that the Joseph form update adds another 30% to the execution time. As a result, the Joseph form will only be used if numerical stability becomes an issue. It should be noted that the actual execution times may vary depending on how the update is programmed: if one is not careful with the sequence of matrix multiplications the complexity of the algorithm increases as the cube of the number of states.

<table>
<thead>
<tr>
<th>Update Type</th>
<th>Measurements</th>
<th>Run Time (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sequential</td>
<td>Pos.</td>
<td>112</td>
</tr>
<tr>
<td></td>
<td>Pos. &amp; Vel.</td>
<td>217</td>
</tr>
<tr>
<td>All-At-Once</td>
<td>Pos.</td>
<td>166</td>
</tr>
<tr>
<td></td>
<td>Pos. &amp; Vel.</td>
<td>367</td>
</tr>
<tr>
<td>All-At-Once Joseph Form</td>
<td>Pos.</td>
<td>265</td>
</tr>
<tr>
<td></td>
<td>Pos. &amp; Vel.</td>
<td>530</td>
</tr>
</tbody>
</table>

Table 6.7: EKF update step timing for various methods - five run average.

The timing results for the EKF covariance prediction step are given in Tab. 6.8. It can be seen that the execution time rises linearly with the number of terms in the matrix exponential expansion. As mentioned in Sec. 4.2, currently the expansion order is set to two so as to balance the accuracy and computational complexity. This test further justifies the exclusion of higher order gravity terms or control forces in the STM.

<table>
<thead>
<tr>
<th>STM Expansion Order</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Run Time (ms)</td>
<td>126</td>
<td>193</td>
<td>256</td>
<td>319</td>
<td>381</td>
</tr>
</tbody>
</table>

Table 6.8: EKF covariance prediction step timing for various STM expansion orders - three run average.

Tab. 6.9 gives execution times for the RK4 method for an integration period of five seconds with various time steps and gravity models. This test is a direct measure of the EKF state prediction step execution time. During actual operation, it is desired that this time remains
as small as possible, since FIONA must share the payload computer with the more resource heavy GPS navigation algorithm. Based on these results, a five second time step coupled with a $6 \times 6$ or $10 \times 10$ gravity model should be selected for actual operation in order to balance speed and accuracy. A time step below one second, as was originally planned, becomes prohibitively expensive irrespective of gravity model. As an aside, it is noted that a numerical calculation of the perturbative forces outperforms the analytical calculation.

<table>
<thead>
<tr>
<th>Gravity Model</th>
<th>RK4 Time Step</th>
</tr>
</thead>
<tbody>
<tr>
<td>Degree</td>
<td>Order</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>20</td>
<td>20</td>
</tr>
</tbody>
</table>

Table 6.9: EKF state prediction step timing for various gravity models and RK4 time steps - three run average for a 5 second integration period.

6.3.1.2 Reconfiguration Algorithm

Fig. 6.6 shows the execution time of the reconfiguration algorithm presented in Sec. 5.2.3 for various numbers of thrusts in the solution and for different STMs. The time to recalculate the solution is also shown, and is much lower than the original calculation time owing to the storage and reuse of important matrices. For this reason, the recalculation time is independent of STM used. The HCW STM has the lowest computational cost associated with it; however, it was shown in Sec. 6.2 that its accuracy is insufficient. The GA STM is, at best, three times slower than the AY STM. Of course, this is immaterial following the initial calculation. These results show that a reconfiguration maneuver can be computed in a reasonable amount of time even for a large number of thrusts, thus meeting the requirement of computational efficiency.

6.3.1.3 FIONA Execution Times

The FIONA run time was obtained for various modes of operation. It is desired that the majority of calls to FIONA can be executed within one second. This gives one second for data transmission across the ISL and three seconds for the fine mode EKF.

For these tests the EKF state prediction employs a five second time step. The EKF covariance prediction employs a second order STM expansion. Twelve measurements are used during the EKF state update, with the sequential measurement update. The reconfiguration calculations are performed for a seven thrust maneuver using the GA STM. For attitude calculation
method 2, a 60 second prediction period is used. The variables in the simulation are the gravity model fidelity, and the ACS target prediction time step used during formation keeping and reconfiguration recalculations. The results from five separate trials are averaged and presented in Tab. 6.10.

The execution time for attitude target method 1, nominal formation keeping, nominal reconfiguration, and EHM all fall well within the desired one second execution time. This is primarily a result of the EKF update state improvements. The small increases across gravity models are a direct result of the higher fidelity.

The reconfiguration solution calculation is constant over all runs, with only slight increases in execution times resulting from the better gravity model. Two seconds is considered an acceptable benchmark for a call to FIONA where a reconfiguration is calculated, since these are relatively infrequent.

The main constraints on the attitude prediction time step and gravity model come from the two modes where a prediction is required: ACS target mode 2, and reconfiguration solution recalculation. For these two modes, the current state is integrated forward in time to obtain estimates of the relative state at a future time, which enables attitude maneuvers. Integration of the orbit equations with an RK4 method has been found to be the heaviest computational burden in the timing tests. Further, these prediction modes will be used fairly frequently, so it is important that they execute quickly while maintaining an adequate level of accuracy. Qualita-
Table 6.10: FIONA execution times for various operational modes.

tively, the required accuracy is one which allows the sub-meter formation tracking requirement to be met.

A 2 \times 0 \text{ gravity model would meet the timing needs, however it is unknown whether or not its accuracy is sufficient. A 6 \times 0 \text{ gravity model meets the timing needs with a 10 or 20 seconds prediction time step, and is borderline unacceptable with a five second time step. For the 6 \times 6 \text{ gravity model, only a 20 second prediction time step would produce acceptable timing results. While one would expect a 20 second time step to provide much poorer predictions than a 1 or 5 second step, this is not the case. Fig. 6.7 shows the error of a 1, 5, and 20 second time step as}
compared to a 0.1 second time step. The nonlinear equations of motion including $J_2 - J_6$ were integrated with fixed time steps, and the absolute position and velocity error was calculated at each 20 second epoch. For a prediction time span of 60 seconds, the position and velocity errors are roughly 1 mm and 1 $\mu$m/s, which is almost negligible. This shows that larger time steps provide adequate accuracy for the orbital equations of motion, which makes sense since the dynamics change very slowly. However, one would expect to see this difference increase slightly with the fidelity of the gravity model.

### 6.4 Simulation Environment

The simulation environment, i.e., the truth model used to test the formation flying algorithm, is similar to that in [39], with several extra features added. It is based in MATLAB/Simulink and runs in discrete time steps. Signal time delay is not accounted for; it is assumed all calculations occur instantaneously. It should also be noted that the simulation environment does not currently support “role switching”, i.e., having CanX-4 and CanX-5 reverse their Chief and Deputy roles. The environment described herein is used in Sections 6.5 and 6.6 to assess FIONA’s performance.

![Figure 6.8: Simulation environment block diagram.](image)

The simulation block diagram is shown in Fig. 6.8. At each time step, the CanX-4/-5 truth states are propagated forward in time using a fourth order Runge-Kutta method with a fixed time step. The orbital model can include a gravity model up to degree and order 29, atmospheric drag using an exponential density model, and third body perturbations due to the Sun and Moon. A full implementation of the FK5 reduction presented in Sec. 3.2.5.1 is included. Gaussian noise with a specified variance is added to the truth state prior to input to FIONA. It is also possible to specify a noise correlation, to mimic the effect of the GPS receivers on-orbit, whose solutions will likely be correlated due to similar sources of error. At the same time, the truth relative state is calculated and corrupted with additive Gaussian noise with another specified variance. These noisy relative states act to simulate the “fine” positioning
algorithm, so that the control algorithms can be tested with relative states of known accuracy.

GPS blackouts and other emergency situations are simulated by setting their occurrence to preset times. The control flight code is used directly in the simulation by compiling the C code into an S-function. Lastly, the effect of thrust error is simulated by adding random thrust magnitude biases and a random angular offset due to the attitude control system. These thrust errors are fed back into the truth model, whereas the formation control algorithm assumes perfect knowledge of the thrust.

### 6.4.1 Blackout Simulation

On orbit, the Deputy spacecraft is likely to frequently lose lock on GPS satellites. In the event that less than four satellites are currently being tracked, a temporary loss of measurement data, termed blackout, will be experienced. In order make the simulation environment as realistic as possible, a representative simulation of blackouts is included. In what follows, the chosen blackout simulation is briefly described.

First, define $T_{\text{sim}}$ as the total simulation time, $\Delta t_{\text{bo}}$ as the period of time for which a blackout lasts, and $p_{\text{bo}}$ as the percentage of $T_{\text{sim}}$ for which the Deputy experiences a blackout. These three quantities determine how often and for how long a blackout condition is present, which is an obvious over-simplification of a realistic scenario. In practice, the percentage of blackout will depend the required control thrusts and hence the attitude targets and slew rates. However, the blackout will affect the frequency of control thrusts, so a true percentage is difficult to estimate. Second, the blackout periods are not fixed, since there are many factors affecting the re-acquisition of GPS lock including whether or not a satellite’s ephemeris data is available and whether the receiver is performing a cold, warm, or hot start. The idea then, is to select a worst case $p_{\text{bo}}$ and a representative $\Delta t_{\text{bo}}$ to simulate a scenario with realistic blackouts. Currently $p_{\text{bo}}$ is obtained from STK scenarios that calculate GPS satellite line of sight based on the true attitude of the Deputy, while $\Delta t_{\text{bo}}$ is selected based on observed performance of the OEM4-G2L receiver onboard CanX-2, and combines the minimum warm start time with the time delay before the receiver data is usable.

Based on $\Delta t_{\text{bo}}$ and $p_{\text{bo}}$, the simulation time scale is divided up into $n$ intervals using $n \cdot \Delta t_{\text{bo}} = p_{\text{bo}} \cdot T_{\text{sim}}$, as shown in Fig. [6.9]. This determines the width of the “blackout interval”, $\Delta t$ with the relation $n \cdot \Delta t = T_{\text{sim}}$. A blackout of length $\Delta t_{\text{bo}}$ must occur within each of these intervals. In order to avoid a case with overlapping blackouts, the blackout start time in the $i^{th}$ interval is limited to $t_{i-1} \leq t_{s,i} < t_{i} - \Delta t_{\text{bo}}$. Finally, at the start of a simulation, the $n$ blackout start times are randomly chosen by drawing a number from a uniform probability distribution. In this way a representative blackout scenario can be simulated.
Chapter 6. Simulation and Hardware Test Results

6.5 CanX-4/-5 Reconfiguration Simulations

In this section the reconfiguration algorithm from Sec. 5.2.3 is integrated with FIONA (Sec. 6.1) and tested within the simulation environment described in Sec. 6.4. The results for transfers between the four formations are presented and compared to those from [47].

The truth model parameters are given in Tab. 6.11, where all errors are normally distributed with zero mean. The absolute state errors are chosen to be representative of the actual GPS state estimation errors on-orbit, based on the CanX-2 satellite data. The selected absolute state errors represent the GPS receiver estimates of the absolute position and velocity determination error with at least 9 satellites in view. The relative state errors are chosen to provide roughly 5 cm and 1 mm/s RMS error for relative position and velocity, respectively. The thrust errors are also chosen to be representative of predicted OASYS and CNAPS performance. The initial Chief orbital elements are again given in Tab. 6.1.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gravity Model</td>
<td>Degree and Order 20</td>
</tr>
<tr>
<td>RK4 Time Step</td>
<td>0.25 s</td>
</tr>
<tr>
<td>Absolute Position Std. Dev.</td>
<td>5 m</td>
</tr>
<tr>
<td>Relative Position Std. Dev.</td>
<td>0.026 m</td>
</tr>
<tr>
<td>Absolute Velocity Std. Dev.</td>
<td>0.5 m/s</td>
</tr>
<tr>
<td>Relative Velocity Std. Dev.</td>
<td>0.001 m/s</td>
</tr>
<tr>
<td>Absolute State Noise Correlation Coefficient</td>
<td>0.5</td>
</tr>
<tr>
<td>Thruster Bias Std. Dev.</td>
<td>10% of max.</td>
</tr>
<tr>
<td>Thruster Pointing Error Std. Dev.</td>
<td>1.7°</td>
</tr>
</tbody>
</table>

Table 6.11: FIONA reconfiguration simulation - truth model parameters.

FIONA directly uses the relative states with additive errors in the control calculations, whereas the absolute states are filtered and used to calculate orbital elements and rotation matrices. For these simulations it is also assumed that measurements are always available, i.e., there are no GPS blackouts. While this is a poor assumption, it is required so that the results can be compared to the original reconfiguration algorithm. A summary of the pertinent
FIONA parameters is given in Tab. 6.12. These parameters essentially dictate the performance of the EKF and the thrust system. All reconfiguration maneuvers are assumed to occur over one Chief orbital period with a total of seven thrusts, where the transfers all begin at Chief periapsis. Five of the thrusts are equally spaced throughout the transfer domain, while the two others occur 2.5% and 97.5% of the transfer time.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Reconfig. Thrusts</td>
<td>7</td>
</tr>
<tr>
<td>Transfer Period</td>
<td>1 Chief Orbit</td>
</tr>
<tr>
<td>Reconfig. STM</td>
<td>Gim-Alfriend</td>
</tr>
<tr>
<td>Est. Mode</td>
<td>Fine</td>
</tr>
<tr>
<td>EKF Time Step</td>
<td>5 s</td>
</tr>
<tr>
<td>EKF STM Order</td>
<td>2</td>
</tr>
<tr>
<td>EKF Process Noise Variances</td>
<td>$5 \times 10^{-6} \text{ m}^2$, $8 \times 10^{-6} \text{ m}^2/\text{s}^2$</td>
</tr>
<tr>
<td>EKF Meas. Noise Variances</td>
<td>$300 \text{ m}^2$, $0.5 \text{ m}^2/\text{s}^2$</td>
</tr>
<tr>
<td>EKF Grav. Model</td>
<td>Degree 2, Order 0</td>
</tr>
<tr>
<td>Spacecraft Mass</td>
<td>7 Kg</td>
</tr>
<tr>
<td>Maximum Thrust</td>
<td>5 mN</td>
</tr>
</tbody>
</table>

Table 6.12: FIONA reconfiguration simulation - FIONA parameters.

A summary of the reconfiguration test results are presented in Tab. 6.13, where the “old” method denotes the numerical optimization based method from [47]. The corresponding plots of final position and velocity error as defined in Eq. (6.2) are given in Fig. 6.10.

The proposed reconfiguration method performs slightly better than the baseline method, yielding lower final position errors with slightly less fuel. However, it should be kept in mind that due to the stochastic nature of the simulations, the final position error for the simulations may change by about one meter from one trial to the next, thus we conservatively say that the two methods perform roughly the same. While there were no final velocity errors reported in [47], it is assumed that the present method yields far lower errors since the optimization constraint was velocity error less than 10 cm/s. The similarity of the two methods’ fuel consumption highlights the fact that the present method is indeed fuel optimal.

The robust method holds two main advantages over the baseline method. First, it is independent of orbit truth model; since the method is closed loop, it is reasonable to assume that similar error and fuel consumption results are attained regardless of disturbance accelerations. Instead, state estimation and thruster errors will be primary contributors to maneuver inaccuracy. Second, the robust solution is calculated in-situ, using the current relative position and velocity states. As a result, the reconfiguration would still be successful if commanded at another point in the orbit, though this would probably result in higher fuel use. With the baseline method, the final accuracy is closely linked with the thrust times, and hence how closely the initial conditions match those of the optimizer. In practice this may yield inaccurate maneu-
vers due to clock errors, or poor formation keeping prior to the maneuver. Furthermore, the optimization would have to be re-run for a different thruster configuration (e.g. three nozzles
instead of one), whereas the proposed method is not sensitive to such changes.

<table>
<thead>
<tr>
<th>Maneuver</th>
<th>Robust (New) Method</th>
<th>Old Method</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\delta R$ (m)</td>
<td>$\delta V$ (m/s)</td>
</tr>
<tr>
<td>1000 m ATO to 500 m ATO</td>
<td>1.9</td>
<td>0.003</td>
</tr>
<tr>
<td>500 m ATO to 50 m PCO</td>
<td>0.8</td>
<td>0.002</td>
</tr>
<tr>
<td>50 m PCO to 100 m PCO</td>
<td>3.0</td>
<td>0.007</td>
</tr>
</tbody>
</table>

Table 6.13: Results of reconfiguration simulations with FIONA.

### 6.6 Mission Simulation

In this section mission simulation results using fine and coarse estimation mode and attitude targeting method 2 (Sec. 5.3.2) are presented. Each simulation has a total of 204 orbits – 50 orbits per formation and 4 reconfiguration orbits. Regardless of estimation mode, the coarse estimation mode filter always runs in the background to provide more accurate estimates of the Chief absolute position and velocity to calculate rotation matrices and orbital elements needed for the control algorithms. Coarse estimation mode performance is briefly reviewed in Sec. 6.6.1, as the results are common to all simulations. The attitude targeting method and FIONA parameters have been adjusted in order to meet two main goals. First, we wish the FIONA run times to be under or near one second for each execution. As such, the parameters reflect those determined to reduce the run times in the FIONA timing tests of Sec. 6.3.1.3. Second, and most importantly, we wish to demonstrate sub-meter formation control throughout the mission. To meet this goal, a set of higher control gains for fine estimation mode have been added to achieve better steady state tracking control when accurate relative state estimates are available.

It was found through simulation that while attitude targeting method 1 works to stabilize the formation, it results in non-zero centered formation tracking errors. This is because a phase delay in the formation tracking is induced by waiting to slew the satellite. The net result is tracking errors that do not meet the sub-meter requirement, even in the presence of fine measurements. Explicit examples of this are provided in the analysis of the mission simulations below.

In the first simulation presented, no formation flying control occurs during GPS blackouts. In the second, coarse simple-difference relative state estimates are used during blackouts to maintain a small tracking error. In the third simulation, coarse relative measurements are used throughout, even during blackouts. The longer the blackout period, the more the relative state accuracy degrades as a result of two sources: Chief absolute state errors that build up from the coarse EKF operating openloop with a low fidelity dynamic model, and Deputy absolute state
errors that accumulate in the absence of thruster performance feedback.

The truth model parameters for these simulations are given in Tab. 6.11. In addition, the blackout simulation parameters $\Delta t_{bo}$ and $p_{bo}$ (described in Sec. 6.4.1) are chosen to be 300 s and 0.25, respectively. Some of the FIONA parameters are given in Tab. 6.12 while the rest are given in Tab. 6.14. Note that the maximum thrust has been increased by a factor of three to represent three active thrust nozzles – the current momentum management scheme dictates that only three thrusters should be active during any given maneuver.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum Thrust</td>
<td>15 mN</td>
</tr>
<tr>
<td>Short PWM Period</td>
<td>75 s</td>
</tr>
<tr>
<td>Long PWM Period</td>
<td>215 s</td>
</tr>
<tr>
<td>Attitude Target Prediction Time</td>
<td>60 s</td>
</tr>
<tr>
<td>Attitude Target Prediction Time Step</td>
<td>20 s</td>
</tr>
<tr>
<td>Min. Formation Keeping Thrust Duration</td>
<td>0.1 s</td>
</tr>
<tr>
<td>Max. Formation Keeping Thrust Duration</td>
<td>15.0 s</td>
</tr>
<tr>
<td>Fine Control Tracking Error Limit</td>
<td>5 m</td>
</tr>
<tr>
<td>Max. Tracking/Reconfig. Pos. Error</td>
<td>50 m</td>
</tr>
<tr>
<td>Max. Tracking/Reconfig. Vel. Error</td>
<td>5 cm/s</td>
</tr>
<tr>
<td>Discrete LQR, Fine - ATO - $r_{u,ato}$</td>
<td>$1.2 \times 10^4$</td>
</tr>
<tr>
<td>Discrete LQR, Fine - ATO - $q_{r,ato}$, $q_{v,ato}$</td>
<td>$2.8 \times 10^{-5}$, $7.8 \times 10^{-1}$</td>
</tr>
<tr>
<td>Discrete LQR, Fine - PCO - $r_{u,pco}$</td>
<td>$1.2 \times 10^4$</td>
</tr>
<tr>
<td>Discrete LQR, Fine - PCO - $q_{r,pco}$, $q_{v,pco}$</td>
<td>$1.2 \times 10^{-6}$, $7.8 \times 10^{-1}$</td>
</tr>
<tr>
<td>Discrete LQR, Coarse - ATO - $r_{u,ato}$</td>
<td>$2.5 \times 10^6$</td>
</tr>
<tr>
<td>Discrete LQR, Coarse - ATO - $q_{r,ato}$, $q_{v,ato}$</td>
<td>$2.8 \times 10^{-6}$, $7.8 \times 10^{-4}$</td>
</tr>
<tr>
<td>Discrete LQR, Coarse - PCO - $r_{u,pco}$</td>
<td>$2 \times 10^6$</td>
</tr>
<tr>
<td>Discrete LQR, Coarse - PCO - $q_{r,pco}$, $q_{v,pco}$</td>
<td>$2.8 \times 10^{-6}$, $7.8 \times 10^{-3}$</td>
</tr>
<tr>
<td>Discretization Time Step</td>
<td>300 s</td>
</tr>
<tr>
<td>Continuous LQR ($n =$ Chief’s orbital rate) - $r_{u,c}$</td>
<td>$n^{-2} \times 10^{-1}$</td>
</tr>
<tr>
<td>Continuous LQR - $q_{r,c}$, $q_{v,c}$</td>
<td>$n^2$, $1 \times 10^{-2}$</td>
</tr>
</tbody>
</table>

Table 6.14: FIONA mission simulation - FIONA parameters.

If the tracking error ever exceeds the fine control tracking error limit, the nominal control gain is replaced with a higher gain to improve tracking performance. If at any point during formation keeping the maximum position or velocity tracking errors are exceeded, EHM is automatically activated. If these errors are exceeded at the end of a reconfiguration maneuver, the maneuver is automatically attempted again, using the original parameters. The maximum allowable reconfiguration errors may seem high, at 50 m and 5 cm/s, however this is required due to the expected reduction in accuracy due to GPS blackouts. Furthermore, it was found that these final errors can still be tolerated by the formation keeping control without causing instability. Due to the importance of these parameters to mission success, it is likely that the exact values for will be changed in the future. The LQR matrix weights shown in Tab. 6.14 are...
used in conjunction with the form of the weighting matrices in Sec. 5.1.3, the plant dynamics
given by Eq. (3.25), and the matrix $B = [0_{3 \times 3} \ I_{3 \times 3}]^T$ to obtain the FIONA formation keeping
control gains. These matrices are given in App. E.

### 6.6.1 Coarse EKF Performance

The coarse mode EKF performed well for the duration of the simulations. An example of the
Deputy absolute state estimation is given in Fig. 6.11. The frequent spikes in the error bounds
correspond to the blackout periods, where the state estimate is propagated forward without
any measurements. During the blackouts, the accuracy of the measurement decreases quickly;
however, it is encouraging to see that the covariance provides a reasonable estimate of the true
error during the entire mission. The Chief states are not shown, since they follow an identical
trend.

A summary of the coarse mode EKF absolute and relative position estimates is given in
Tab. 6.15. The error levels are mostly driven by the blackout frequency and the level of noise
added to the measurements. The absolute state estimation errors for the Chief and Deputy are
large in comparison to the relative estimation error. This discrepancy is mostly due to the high
level of correlation on the random noise levels; fully uncorrelated additive noise on the Chief
and Deputy would yield much more inaccurate estimates. The correlation coefficient of 0.5 is
justified based on the proximity of the two satellites. A large portion of this error is removed
by taking the difference of the two states.

It is interesting to note that there is very little difference between the relative state estimates
in the Hill and Inertial frames, thus indicating that the error in the Chief absolute state estimate
does not contribute significantly to relative errors through the inertial to Hill rotation matrix.
This agrees with the analytical predictions presented in App. C. A sample of the relative state
estimation error in the inertial frame is shown in Fig. 6.12.

<table>
<thead>
<tr>
<th>Case</th>
<th>Pos. Error (m)</th>
<th>Vel. Error (m/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chief, Inertial</td>
<td>6.45</td>
<td>0.038</td>
</tr>
<tr>
<td>Deputy, Inertial</td>
<td>6.43</td>
<td>0.038</td>
</tr>
<tr>
<td>Relative, Inertial</td>
<td>1.93</td>
<td>0.0060</td>
</tr>
<tr>
<td>Relative, Hill</td>
<td>1.93</td>
<td>0.0062</td>
</tr>
</tbody>
</table>

Table 6.15: Coarse mode estimation errors (RMS).

The on-orbit performance of this EKF is expected to decrease due to several factors. First,
there will be additional errors introduced due to the body to inertial rotations needed to trans-
form the GPS measurements. These errors will add more noise, and potentially bias. Second,
in these simulations the level of additive noise was given as fixed, whereas the accuracy of the
Figure 6.11: Sample of coarse EKF performance for Deputy satellite. The dashed lines represent estimated $\pm 3\sigma$ error bounds.

GPS receiver state estimates vary with the number of satellites in view. A four satellite solution will have a much higher error than a nine satellite solution. In addition, it is likely that
Figure 6.12: Sample of coarse relative state estimation performance. The dashed lines represent estimated $\pm 3\sigma$ error bounds.

more bias will be introduced due to these measurements. Lastly, the “truth” model will be substantially different than the onboard orbit model, thus the measurement innovations will be
higher, potentially leading to spikes in the estimate greater overall error. With these additional sources of error, the assumptions of Gaussian white noise will be violated, and may lead to filter inconsistency (i.e., poor error estimates) and possibly divergence. Careful tuning of the coarse filter will be required before on-orbit operations. Since the exact separation between the two GPS antennas will be known, the filter parameters should be modified until the coarse filter yields exactly the known separation.

6.6.2 Case 1 - No Control During Blackouts

In this simulation there are no control actions performed during GPS blackouts. During these periods of inactivity, the formation drifts apart, only to be re-established once measurements again become available. The overall tracking errors for this simulation are 1.56 m and 3.7 mm/s in relative position and velocity, respectively. The total fuel use is 17.9 m/s, which exceeds the allowed fuel use by close to 30%. This indicates that for the current level of control authority, the number of orbits in each formation should not exceed 25 or so.

6.6.2.1 Reconfiguration Performance

One notable result from this simulation is the increase in reconfiguration fuel use and final errors as shown in Tab. 6.16 as compared to those in Sec. 6.5. This is due to a variety of reasons, to be discussed shortly. In the case of the ATO to PCO transition, the fuel use actually decreased as compared to the baseline. This is due to the fact that one or more reconfiguration thrusts were missed because of GPS blackout. However, the tradeoff is that the resulting final errors are an order of magnitude higher than in the baseline case. As mentioned previously, the allowable error for these maneuvers was increased to 50 m and 5 cm/s for relative position and velocity, respectively, because of the decrease in accuracy due to various sources of error. These final errors still lie within the stability bounds of the LQR.

<table>
<thead>
<tr>
<th>Maneuver</th>
<th>(\delta R) (m)</th>
<th>(\delta V) (m/s)</th>
<th>(\Delta V) (m/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Separation</td>
<td>6.08</td>
<td>0.029</td>
<td>0.30</td>
</tr>
<tr>
<td>1000 m ATO to 500 m ATO</td>
<td>11.48</td>
<td>0.011</td>
<td>0.10</td>
</tr>
<tr>
<td>500 m ATO to 50 m PCO</td>
<td>7.00</td>
<td>0.042</td>
<td>0.07</td>
</tr>
<tr>
<td>50 m PCO to 100 m PCO</td>
<td>4.38</td>
<td>0.0084</td>
<td>0.10</td>
</tr>
</tbody>
</table>

Table 6.16: Case 1 mission simulation reconfiguration final errors.

The primary reason for the increase in fuel use and final errors is the loss of control during GPS blackouts. Since there are no control actions performed during blackout, roughly two out of seven reconfiguration thrusts are missed for each planned maneuver, which has a large effect on the final accuracy, though the robust scheme works to complete the maneuver. The
ATO to ATO maneuver is a good example of the reduced accuracy due to missed thrusts. An equivalent source of error is the relative state estimation error, which induces thrust magnitude and direction errors when they are used in the reconfiguration control solution. This refers to errors inherent to the relative state estimate (additive noise), and those coming from the coarse EKF Chief state estimate, which is used to calculate the attitude targets and orbital elements. Integration errors due to large time steps and a low fidelity absolute dynamic model further contribute to these Chief state estimation errors. Lastly, reconfiguration errors due to ACS accuracy, thruster bias, and poor relative dynamic modeling are ever-present, though these are secondary as compared to estimation errors and blackouts.

It is important to keep in mind that the separation sequence is not treated exactly as it will be during the mission, but rather as another reconfiguration maneuver. The actual mission plan has the satellites drifting apart without control for approximately seven orbits while final sensor, actuator, and relative navigation algorithm commissioning occurs. The first reconfiguration maneuver will then be initiated by the ground station. For simplicity, here the first maneuver commences upon separation, and the 1000 m ATO formation is achieved after one orbit. Future simulations will have to account for the extended drift at the start, and as such the first maneuver will have to be considered more carefully in terms of total duration and number of thrusts.

A final point to note is that the separation maneuver was originally planned to occur over one orbit, though it ended up taking two orbits. The reason is that the final thrust at the end of the planned maneuver was missed due to a blackout. Consequently, the final error targets were not achieved, which caused FIONA to automatically retry the maneuver. In this case the second attempt was successful, and a reasonable amount of fuel was consumed. However, in several other simulations this automatic second attempt has resulted in very high $\Delta V$, on order of 1 m/s. This leads to the conclusion that in general, a missed reconfiguration maneuver should not be automatically retried. Instead, the satellites should enter a mode which stops the relative drift between them and the ground station should command the maneuver. This strategy will avoid the autonomous depletion of limited fuel reserves.

An alternative strategy is to continue the reconfiguration maneuver through GPS blackout, using coarse state estimates in order to recalculate the maneuvers and provide attitude targets. This will cause higher fuel use and final errors, as a direct result of the poorer measurements. The benefit is that the maneuver will be completed autonomously in the desired period of time, simplifying the ground station’s involvement considerably. This strategy is explored further in Sec. 6.6.3.
6.6.2.2 Formation Keeping Performance

Breakdowns of the tracking error and fuel consumption by orbit are provided in Tabs. 6.17 and 6.19 respectively. The tracking errors represent RMS values calculated over the 50 orbits in each formation. The errors for all formations are fairly uniform, with the tighter formations exhibiting the smallest errors. This is because the reference trajectories are most similar to the natural motion at small relative distances. The fuel consumption per formation is roughly doubled for the ATO formations as compared to the PCO formations. This is directly linked to the magnitude of the control gains, which have been adjusted to provide precise control. This adversely affects fuel consumption. The magnitudes of the tracking errors over the course of the simulation are visualized in Fig. 6.13. The large spikes denote the tracking error immediately after the reconfiguration maneuver. The LQR control law does a good job in reducing these large errors.

![Mission Tracking Error](image)

Figure 6.13: Case 1 mission simulation tracking errors.

The primary reasons the RMS tracking errors exceed the sub-meter requirement are the loss of control during periods of GPS blackout, and the frequency of these blackouts. The blackouts result in periods of formation drift, after which sub-meter formation tracking is re-established until the next blackout. Given that 25% of the time was spent in blackout, and another fraction of time is spent recovering from these drifts, the results are quite good. From a similar simulation with no blackouts and 10 orbits per formation, whose results are given in
Tab. 6.18 it is concluded that consistently small tracking errors are achievable. A secondary reason is the decrease in accuracy of the attitude targets resulting from the degradation in Chief state estimate during and following blackouts.

One strategy to reduce the overall tracking error is to increase the control gains used to re-establish the formations following a temporary loss of control. This would reduce the length of time spent in the transient region, but would result in potentially large fuel penalties, depending on the quality of the relative measurements and the tracking error.

<table>
<thead>
<tr>
<th>Formation</th>
<th>1000 m ATO</th>
<th>500 m ATO</th>
<th>50 m PCO</th>
<th>100 m PCO</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pos. Track. Error (m)</td>
<td>1.75</td>
<td>1.36</td>
<td>1.54</td>
<td>1.66</td>
<td>1.56</td>
</tr>
<tr>
<td>Vel. Track. Error (m/s)</td>
<td>0.005</td>
<td>0.004</td>
<td>0.003</td>
<td>0.003</td>
<td>0.004</td>
</tr>
</tbody>
</table>

Table 6.17: Case 1 mission simulation RMS tracking error summary.

<table>
<thead>
<tr>
<th>Formation</th>
<th>1000 m ATO</th>
<th>500 m ATO</th>
<th>50 m PCO</th>
<th>100 m PCO</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pos. Track. Error (m)</td>
<td>0.90</td>
<td>0.54</td>
<td>0.64</td>
<td>0.76</td>
<td>0.69</td>
</tr>
<tr>
<td>Vel. Track. Error (m/s)</td>
<td>0.003</td>
<td>0.002</td>
<td>0.001</td>
<td>0.002</td>
<td>0.002</td>
</tr>
</tbody>
</table>

Table 6.18: Blackout-free tracking error summary; 10 orbits per formation.

<table>
<thead>
<tr>
<th>Orbit</th>
<th>ΔV (m/s)</th>
<th>Number of Orbits</th>
<th>ΔV/orbit (m/s/orbit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Separation</td>
<td>0.30</td>
<td>2</td>
<td>0.15</td>
</tr>
<tr>
<td>1000 m ATO</td>
<td>6.76</td>
<td>50</td>
<td>0.135</td>
</tr>
<tr>
<td>1000 m ATO to 500 m ATO</td>
<td>0.10</td>
<td>1</td>
<td>0.10</td>
</tr>
<tr>
<td>500 m ATO</td>
<td>5.16</td>
<td>50</td>
<td>0.103</td>
</tr>
<tr>
<td>500 m ATO to 50 m PCO</td>
<td>0.07</td>
<td>1</td>
<td>0.07</td>
</tr>
<tr>
<td>50 m PCO</td>
<td>2.53</td>
<td>50</td>
<td>0.051</td>
</tr>
<tr>
<td>50 m PCO to 100 m PCO</td>
<td>0.10</td>
<td>1</td>
<td>0.10</td>
</tr>
<tr>
<td>100 m PCO</td>
<td>2.88</td>
<td>50</td>
<td>0.059</td>
</tr>
</tbody>
</table>

Table 6.19: Case 1 mission simulation fuel use breakdown.

The fuel consumption per formation is a product of several different factors. The greatest contributor is the level of control authority, where higher gains lead to longer thrusts and greater fuel consumption. The control gains must be carefully selected to provide adequate tracking coupled with an acceptable ΔV. Another major factor is the relative state estimation error: though it is not shown here, due to the large control gains on the velocity terms (see App. E) errors in velocity estimation cause the majority of thrust calculation errors which lead to unnecessary fuel consumption and larger formation tracking errors. In turn, formation tracking errors further contribute to fuel consumption through the control gain matrix. GPS blackouts are another large source of formation tracking errors which contribute to fuel use because less formation keeping maneuvers allow tracking errors to accumulate, and greater control effort is exerted in the future to compensate. An example of the thrust profile under
constant measurement blackouts is shown in Fig. 6.14. The long gaps represent periods with blackouts, where no control is applied. Oftentimes, the blackouts are followed by periods of frequent thrusting to reduce the tracking error which has increased beyond the preset limit. The intermediate-sized gaps represent the long PWM period, which is used once the formation is stable. A third source of formation tracking error are thrust system biases or mis-alignments, and attitude pointing errors.

Figure 6.14: Sample thrust profile with GPS blackouts.

A secondary factor contributing to the required control effort may be the reference trajectory itself. Any errors in the reference trajectory calculations (true anomaly, orbital rate, eccentricity, semi-major axis, etc.) will mean additional effort must be exerted to track these unnatural relative motions. Similarly, these trajectories are designed using relative motion theories that do not account for perturbations. Thus tracking a formation that is periodic with the unperturbed motion of the Chief will result in greater fuel consumption due to the departure from the natural relative motion.

It is noted here that attitude targeting method 2 outperforms targeting method 1 (see Sec. 5.3), but not by as wide a margin as one would expect. Tab. 6.20 shows the difference in tracking error between the two methods, which were determined in blackout-free simulations with 10 orbits in each formation. While there is virtually no difference between the velocity tracking error using the two methods, targeting method 2 shows better relative position tracking errors in all formations. The reason is that with attitude targeting method 1, one ends up tracking
a delayed formation, i.e., a phase lag is induced due to the slewing period. The 1000 m ATO and 100 m PCO formations show the biggest difference in tracking errors. This is due to the fact that formation drift is more of a concern in these configurations, so that the control is less effective at the application times because the Deputy is no longer where it was when the control was calculated. It is interesting that the fuel consumption for method 1 in the 500 m ATO is about 2 cm/s lower than for method 2. The reason for this is unknown, though it may be due to thrust errors incurred from targeting method 2 combined with the large prediction time step (20 s), though in this case one would expect to see a similar increase in fuel use during the 1000 m ATO. While it may not seem like one method is much better than the other, it should be noted that the differences become more pronounced in the presence of GPS blackouts and larger relative state estimation errors. For this reason attitude targeting method 2 is selected.

<table>
<thead>
<tr>
<th>Formation</th>
<th>1000 m ATO</th>
<th>500 m ATO</th>
<th>50 m PCO</th>
<th>100 m PCO</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pos. Track. Error (m)</td>
<td>+0.22</td>
<td>+0.13</td>
<td>+0.04</td>
<td>+0.17</td>
<td>+0.14</td>
</tr>
<tr>
<td>Vel. Track. Error (m/s)</td>
<td>0</td>
<td>0</td>
<td>+0.001</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Fuel Use (m/s/orbit)</td>
<td>+0.003</td>
<td>-0.017</td>
<td>-0.0005</td>
<td>-0.004</td>
<td>N/A</td>
</tr>
</tbody>
</table>

Table 6.20: Improvement in tracking errors and fuel use with attitude targeting method 2. All fields record the difference Method 1-Method 2.

### 6.6.3 Case 2 - Coarse Control During Blackouts

This simulation is identical to Case 1 except that coarse measurements are used for control during blackouts. The overall tracking errors for this simulation are 1.26 m and 3.5 mm/s in relative position and velocity, respectively. This is a marginal improvement over the results from the Case 1 simulation, where no control was applied during blackouts. The overall ∆V was roughly 3.3 m/s greater than in Case 1, which is a direct result of having used coarse measurements for formation flying 25% of the time. Performance during reconfigurations is also improved in that the maneuvers are completed in the desired period of time and some errors are reduced compared to Case 1, where thrusts are missed. However, the use of coarse measurements during reconfiguration also increases the fuel use.

#### 6.6.3.1 Reconfiguration Performance

The final reconfiguration errors and fuel usages are provided in Tab. 6.21. As compared to results of Case 1, the separation maneuver was more accurate, used less fuel and did not require additional orbits. This is a direct result of the use of coarse relative measurements to calculate reconfiguration thrusts during GPS blackout. The ATO to ATO transfer also shows substantial improvement, especially in the final position error, though the reduced fuel use
during this maneuver is a surprisingly result. The only explanation is more control effort was required in Case 1 to make up for missed thrusts.

The increased fuel use during the transfers to PCO formations is expected, since relative state estimation errors are the primary cause of reconfiguration thrust errors. The final position errors for these two maneuvers are greater than in Case 1, again due to estimation error. The increase in final velocity error for the PCO to PCO transfer is not concerning, since the error is still within the stability bounds of the LQR. Likewise, the increased fuel use is tolerable given that the reconfiguration maneuvers can be performed with greater confidence when coarse measurements are used in the absence of fine measurements during GPS blackout.

<table>
<thead>
<tr>
<th>Maneuver</th>
<th>δR (m)</th>
<th>δV (m/s)</th>
<th>ΔV (m/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Separation</td>
<td>4.93</td>
<td>0.010</td>
<td>0.23</td>
</tr>
<tr>
<td>1000 m ATO to 500 m ATO</td>
<td>1.82</td>
<td>0.005</td>
<td>0.09</td>
</tr>
<tr>
<td>500 m ATO to 50 m PCO</td>
<td>7.31</td>
<td>0.015</td>
<td>0.11</td>
</tr>
<tr>
<td>50 m PCO to 100 m PCO</td>
<td>7.27</td>
<td>0.013</td>
<td>0.13</td>
</tr>
</tbody>
</table>

Table 6.21: Case 2 mission simulation reconfiguration final errors.

### 6.6.3.2 Formation Keeping Performance

The RMS tracking error per formation for this simulation is provided in Tab. 6.22. As compared to Case 1, the overall position tracking error has been reduced by 30 cm, while there is no significant effect on velocity tracking error. The greatest improvement is seen in the ATO formations, whose errors are reduced by over 40 cm. This is likely due to the fact that greater formation drift is incurred at larger relative distances, so that the overall error is significantly affected by constantly applying control thrusts through blackouts, even if the thrusts are less accurate due to measurement error. The 50 m and 100 m PCO formations show 25 cm and 7 cm improvements, respectively. A more significant performance gain in the 100 m PCO was expected. A possible explanation is that the improvement in tracking error from more regular control actions is effectively cancelled out by the reduction in thrust accuracy due to estimation errors.

<table>
<thead>
<tr>
<th>Formation</th>
<th>1000 m ATO</th>
<th>500 m ATO</th>
<th>50 m PCO</th>
<th>100 m PCO</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pos. Track. Error (m)</td>
<td>1.12</td>
<td>0.93</td>
<td>1.29</td>
<td>1.59</td>
<td>1.26</td>
</tr>
<tr>
<td>Vel. Track. Error (m/s)</td>
<td>0.004</td>
<td>0.004</td>
<td>0.003</td>
<td>0.003</td>
<td>0.003</td>
</tr>
</tbody>
</table>

Table 6.22: Case 2 mission simulation RMS tracking error summary.

As always, the improved tracking performance comes at the cost of greater fuel use, as shown in Tab. 6.23. With the exception of the 1000 m ATO, all formations use roughly 1 m/s more ΔV than in Case 1. As in the case of the reconfiguration maneuvers, this is due to the use
of coarse relative state estimates, which increase thrust direction and magnitude errors, thus causing degraded tracking performance and greater fuel use in order to continuously compensate for the tracking errors. Operating with coarse relative state estimates during GPS blackout will reduce the total number of orbits possible since the fuel is consumed faster, though the mission requirements of at least 10 orbits per formation can still be easily met. In this view, a gain in tracking performance is more valuable than a reduction in the number of total formation keeping orbits.

<table>
<thead>
<tr>
<th>Orbit</th>
<th>(\Delta V) (m/s)</th>
<th>Number of Orbits</th>
<th>(\Delta V/\text{orbit}) (m/s/orbit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Separation</td>
<td>0.23</td>
<td>1</td>
<td>0.23</td>
</tr>
<tr>
<td>1000 m ATO</td>
<td>6.96</td>
<td>50</td>
<td>0.139</td>
</tr>
<tr>
<td>1000 m ATO to 500 m ATO</td>
<td>0.09</td>
<td>1</td>
<td>0.09</td>
</tr>
<tr>
<td>500 m ATO</td>
<td>6.03</td>
<td>50</td>
<td>0.121</td>
</tr>
<tr>
<td>500 m ATO to 50 m PCO</td>
<td>0.11</td>
<td>1</td>
<td>0.11</td>
</tr>
<tr>
<td>50 m PCO</td>
<td>3.73</td>
<td>50</td>
<td>0.075</td>
</tr>
<tr>
<td>50 m PCO to 100 m PCO</td>
<td>0.13</td>
<td>1</td>
<td>0.13</td>
</tr>
<tr>
<td>100 m PCO</td>
<td>3.96</td>
<td>50</td>
<td>0.079</td>
</tr>
</tbody>
</table>

Table 6.23: Case 2 mission simulation fuel use breakdown.

The PCO fuel usages have increased by about 2 cm/s/orbit as compared to Case 1, while the tracking error has not been substantially reduced. This indicates that it is not beneficial to continue formation flying through blackouts in these PCO orbits, since the fuel usage increases drastically due to the thrust calculation errors.

6.6.4 Case 3 - Coarse Control Throughout

As mentioned previously, in this simulation coarse mode relative measurements were used for all formation keeping and reconfiguration activities. Coupled with a reduced control authority, this resulted in an overall formation tracking error of 4.9 m, roughly four times that of the Case 2 simulation. Similarly, final reconfiguration errors are higher than those seen using fine relative estimates. Even with low quality relative state estimates (see Sec. 6.6.1) the formations remained stable, demonstrating that coarse formation keeping is a viable option should the fine relative navigation algorithm fail.

6.6.4.1 Reconfiguration Results

The final reconfiguration errors for this simulation are given in Tab. 6.24. As in Case 2, all reconfigurations were completed in the desired period of time owing to the fact that control thrusts were applied during GPS blackout. As expected, the fuel consumption and final errors for three of the four maneuvers are greater than those in Case 2, however the final errors were
kept within reason owing to the robust reconfiguration method. For the separation maneuver, the final errors and $\Delta V$ are very similar to those in Case 2, though slightly larger. The likely cause is that several coarse measurements were used in the Case 2 separation (due to blackouts), so that the fuel use and error characteristics are similar. This marked degradation in performance is principally a result of the coarse mode estimates, which emphasizes the need for accurate relative states.

<table>
<thead>
<tr>
<th>Maneuver</th>
<th>$\delta R$ (m)</th>
<th>$\delta V$ (m/s)</th>
<th>$\Delta V$ (m/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Separation</td>
<td>8.78</td>
<td>0.016</td>
<td>0.23</td>
</tr>
<tr>
<td>1000 m ATO to 500 m ATO</td>
<td>5.69</td>
<td>0.011</td>
<td>0.20</td>
</tr>
<tr>
<td>500 m ATO to 50 m PCO</td>
<td>13.03</td>
<td>0.016</td>
<td>0.14</td>
</tr>
<tr>
<td>50 m PCO to 100 m PCO</td>
<td>9.66</td>
<td>0.018</td>
<td>0.16</td>
</tr>
</tbody>
</table>

Table 6.24: Case 3 mission simulation reconfiguration final errors.

### 6.6.4.2 Formation Keeping Results

The tracking error as a function of formation is given in Tab. 6.25, while the corresponding graphs are given in Fig. 6.15. The RMS errors are roughly four times that observed during Case 1 and 2. This is due to both the reduced control authority and the less accurate relative states: the tracking errors cannot possibly be reduced to sub-meter levels because the gains are not large enough, and the attitude targeting and thrust calculation errors resulting from the state estimates cause further degraded tracking. The position tracking errors are roughly double the relative position error, and the velocity tracking errors are the same as the velocity estimation error. The position tracking error is heavily influenced by the velocity estimation error, through the control gain matrix. It is certain that with further degraded relative state estimates the controller will go unstable since the thrusts will be too much in error. This exact point of instability is unknown, as it depends on several factors including actuator errors, control authority, and PWM period.

<table>
<thead>
<tr>
<th>Formation</th>
<th>1000 m ATO</th>
<th>500 m ATO</th>
<th>50 m PCO</th>
<th>100 m PCO</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pos. Track. Error (m)</td>
<td>4.84</td>
<td>4.85</td>
<td>4.77</td>
<td>5.11</td>
<td>4.85</td>
</tr>
<tr>
<td>Vel. Track. Error (m/s)</td>
<td>0.007</td>
<td>0.007</td>
<td>0.007</td>
<td>0.007</td>
<td>0.007</td>
</tr>
</tbody>
</table>

Table 6.25: Case 3 mission simulation tracking error summary.

The fuel use breakdown, shown in Tab. 6.26, emphasizes the lower control authority employed in this simulation, especially for the ATO formations, whose fuel usages have been reduced by nearly 50% as compared to Case 2. It should be noted that with these control gains and fine estimates, the fuel use would be reduced even further, since state errors contribute significantly to fuel usage through the mechanisms discussed in Sec. 6.6.2.2. It is interesting to
Figure 6.15: Case 3 mission simulation tracking errors.

see that the fuel usage per orbit for the PCO formations are the same as in Case 2, and higher than in Case 1. This means that the reduction in fuel from smaller gains equals the increase due to estimation errors. Though the gains could be reduced even further to conserve more fuel in these formations, one should be careful since this may lead to instability.

As in Case 2, a portion of this increase in fuel use is caused by the constant process of correcting for tracking errors incurred due to thrust errors. Additional control effort is exerted since the tracking errors are quite poor, but the control is never effective in reducing these errors due to the small gains and estimation errors.

<table>
<thead>
<tr>
<th>Orbit</th>
<th>$\Delta V$ (m/s)</th>
<th>Number of Orbits</th>
<th>$\Delta V$/orbit (m/s/orbit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Separation</td>
<td>0.23</td>
<td>1</td>
<td>0.23</td>
</tr>
<tr>
<td>1000 m ATO</td>
<td>4.06</td>
<td>50</td>
<td>0.081</td>
</tr>
<tr>
<td>1000 m ATO to 500 m ATO</td>
<td>0.20</td>
<td>1</td>
<td>0.20</td>
</tr>
<tr>
<td>500 m ATO</td>
<td>3.78</td>
<td>50</td>
<td>0.076</td>
</tr>
<tr>
<td>500 m ATO to 50 m PCO</td>
<td>0.14</td>
<td>1</td>
<td>0.14</td>
</tr>
<tr>
<td>50 m PCO</td>
<td>3.86</td>
<td>50</td>
<td>0.077</td>
</tr>
<tr>
<td>50 m PCO to 100 m PCO</td>
<td>0.16</td>
<td>1</td>
<td>0.16</td>
</tr>
<tr>
<td>100 m PCO</td>
<td>3.95</td>
<td>50</td>
<td>0.079</td>
</tr>
</tbody>
</table>

Table 6.26: Case 3 mission simulation fuel use breakdown.
6.6.5 Mission Simulation Conclusions

Through the three full mission simulations, several important results were obtained. First, formation keeping stability was demonstrated with both fine and coarse estimation modes, even in Case 1 where no control actions were performed during GPS blackout. In fact, similar simulations indicate stability in coarse mode with no control during blackout, though further degraded tracking performance is experienced. Both attitude targeting method 1 and 2 can be used, though it was found that attitude targeting method 2 leads to smaller tracking errors, since it induces no phase lag. Further, the FIONA settings used were quite conservative employing large integration time steps and poor dynamic modeling. These correspond to the settings in Sec. 6.3.1 that were determined to produce fast execution times on the flight hardware. This is a positive result, indicating that very high fidelity models are not necessary to meet requirements, thereby allowing more time for data transmission and fine mode relative navigation. It should be noted, however, that these results are highly dependent on the level of estimation errors, and do not hold for relative velocity errors on the order of cm/s.

Second, it was shown in both Case 1 and 2 that sub-meter formation tracking errors are achievable if fine relative state estimates are available for a long enough period of time. This was investigated in another simulation, where the control mode was changed from coarse to fine in the third orbit in each formation and back to coarse mode in the eighth orbit, with a total of ten orbits per formation. The tracking error throughout the mission, given in Fig. 6.16, show that with the current level of control authority, sub-meter tracking error is reliably acheived after approximately 0.2 orbits, or about 20 minutes. For a faster settling time, the PWM period during the transition from coarse to fine mode could be reduced from its current value of 215 s. Alternatively, the control gains could be increased.

Third, using fine mode estimates and the current level of control authority, the fuel use per orbit in order to achieve sub-meter tracking error was higher in the ATOs than in the PCOs, with the 50 m PCO showing the lowest fuel use per orbit. This indicates that larger control thrusts are needed to achieve fine formation flying at large relative separations, which makes sense considering the formations diverge from the natural relative motion at larger separations, thus increasing the control effort required for formation keeping. At current fuel use levels (from Case 1 and 2), roughly 20 orbits in each formation can be safely performed, assuming a total $\Delta V$ of 13.0 m/s is available, which meets the requirement of at least 10 orbits per formation. Alternatively, if the sub-meter tracking error requirement is relaxed to less than 5 or 10 m, then a reduction in the control authority coupled with fine estimates can result in a fuel consumption of 3 cm/s in the two ATOs and the 50 m PCO. In this way, the number of orbits per formation can be doubled.
Fourth, it was demonstrated in Case 2 and 3 that so long as the blackout periods remain relatively short, formation keeping and reconfiguration can continue, with a degradation in performance due to the gradually worsening coarse state estimates. That being said, there is a significant increase in fuel expenditure between fine and coarse mode formation keeping, and while the overall tracking error can be reduced by thrusting through blackout, the large fuel penalty is unnecessary given that there are no requirements for sub-meter tracking during blackout. Further, if for some reason an extended blackout occurs (e.g. the GPS receiver was reset and must be cold started again), formation flying open-loop would not be prudent due to the degradation in state estimate accuracy, especially since there is no foreknowledge of a blackout’s duration. For these reasons, a strategy where no formation keeping thrusts occur during blackout is recommended. However, there is a case to be made for using coarse estimates when fine measurements become unavailable during reconfiguration. The major benefit is that the maneuvers are completed in the desired time period, rather than requiring second or third attempts because the critical final thrust was missed. It is also possible to reduce final errors, since no thrusts in the planned maneuver are excluded. There is a small fuel penalty associated with this strategy, but the additional fuel used in coarse reconfiguration is much less than that for an entirely new maneuver. In order to prevent the continuation of a maneuver through an extended blackout period, a “blackout timer” could be implemented, where control actions are halted after a preset maximum tolerable blackout period has been exceeded. This
reconfiguration strategy should be further investigated in the future.

Fifth, the robust reconfiguration method presented in Sec. 5.2.5 was validated in a more realistic dynamic formation flying simulation than in Secs. 6.2 or 6.5. The final desired state was achieved with sufficient accuracy (i.e., such that the LQR controller did not go unstable), even in Case 3, where only coarse relative state estimates were available. Furthermore, the additional errors due to incorrect Chief orbital elements stemming from Chief state errors, attitude pointing errors, thrust magnitude errors, and dynamic modeling errors were also successfully mitigated. The reconfiguration maneuvers consume much less fuel when fine measurements are used, however even with coarse measurements the largest $\Delta V$ is 24 cm/s, followed by 17 and 14 cm/s. These fuel usages are acceptable for the CanX-4/-5 mission, especially given the level of accuracy and reliability achieved in these maneuvers.

Lastly, it should be noted that the contribution of relative state estimation error to formation tracking error and fuel consumption is more significant than those contributions due to absolute state estimation error (through attitude targets, rotation matrices, reference trajectories, etc.), attitude pointing errors, thrust magnitude errors and thrust timing offsets. For this reason, it is absolutely critical to attain a high degree of accuracy with the fine mode navigation filter. However, since fine mode navigation will not be available until after the separation maneuver, a rigorous calibration of the coarse mode navigation filter is required. This will be possible on-orbit prior to separation, since the exact distance between the two GPS antennas and the precise attitude and angular rate will be known. The fuel consumption and accuracy of the controlled maneuver into the 1000 m ATO, and coarse formation keeping at 1000 m will depend highly on the quality of these estimated relative states. In this view, future work should focus on establishing the first formation and performing coarse formation keeping using even worse relative state estimates than those presented here, especially in relative velocity.

6.7 GPS Navigation

Two basic tests were performed to ensure the C implementation of the fine mode navigation algorithm functions properly. In the first, raw measurement data on a single receiver is logged and processed in real time in a sequential kinematic least squares filter to obtain absolute position and velocity for one antenna. In the second, raw measurement data from two receivers are collected simultaneously and processed in a real-time EKF to obtain a relative position and velocity between the two antennas. These tests were performed on a single laptop computer, thus it still needs to be verified that the data is being properly collected and processed on an OBC, to which the GPS receiver will eventually be linked. For these simple tests, no communication lag or dropouts were simulated or induced. Both tests employ NovAtel OEM4-
G2L receivers, which are very similar to the OEMV-1G planned to be used on-orbit, except that it receives dual-frequency information.

6.7.1 Static Absolute Navigation Test

In order to verify that the GPS logs are being correctly parsed and processed, a test is performed to calculate the absolute position of a single GPS antenna on the roof of the building. For this test, pseudorange and Doppler measurements from the L1 carrier frequency are used in a single point kinematic mode to estimate the position and velocity of the stationary receiver. No tropospheric or ionospheric corrections are applied to the measurements, since these corrections will not be applied on-orbit. As a result, the estimated states are not expected to be accurate to the same level as the GPS receiver solution.

At each five second interval the RANGECMPB and BESTXYZB GPS logs are received and processed, while the RAWEPHEMB log is received asynchronously and processed as required. The RANGECMP log contains the raw measurement data, the BESTXYZ log contains the receiver’s best estimate of the antenna position, and the RAWEPHEM log contains satellite ephemeris data for those satellites in view. The “B” at the end of the log names denote that the data is in binary format. The solution is then obtained sequentially, by first estimating the absolute position using an iterative least-squares approach, and then estimating the absolute velocity by using a single least-squares update. The states are position and velocity in the ECEF frame, and GPS receiver clock bias and drift.

The initial guesses of the position and clock bias are zero. For each satellite in view whose ephemeris data exists, approximate signal transmission time, satellite position and clock correction are calculated. Then the design matrix \( H_k \in \mathbb{R}^{m \times 4} \) and the pseudorange measurement residuals are formed. A weighted least squares (with the inverse of the pseudorange measurement variances as weights) solution is computed, which gives the first update to the state estimate. The procedure is repeated with a new guess of the position and clock bias until the least squares update to the position states is smaller than a specified tolerance. Then, using the estimated position with an initial guess of zero velocity and clock drift, a single non-weighted least squares update (since Doppler measurements have no estimated variance) is performed to estimate the absolute velocity and clock drift, where the satellite vehicle clock drift correction is accounted for. The output of the first estimation is used as the input to the second, and so on.

At each step, the least squares position solution is compared to the receiver estimate of the position, as obtained from the BESTXYZB log. The velocity is compared to zero, since it is known that the antenna is stationary. The position and velocity error for this test over roughly a one hour span are shown in Fig. 6.17.
It is interesting to note that the accuracy of the solution seems to depend on the time of day. This may either be due to adverse weather conditions (increased atmospheric delay), and/or less favourable geometry during certain parts of the day. At times, a maximum error of 40 m was recorded. The exact reason for this was not investigated further. It should also be noted that, as expected, the weighted least squares position solution is approximately 5 m more accurate than the non-weighted least squares.

The results indicate that the data is indeed being processed and stored correctly. The latent errors present are due to neglecting tropospheric, ionospheric and general relativistic effects, in addition to only using pseudorange data. Further, only C/A L1 measurements are being used here while the receiver processes dual frequency data. Additional estimation errors are caused by satellite orbit error, receiver noise, multipath, etc. The level of errors present in the solution are reasonable considering the simplifications made.

### 6.7.2 Static Baseline Relative Navigation Test

The purpose of this test is to ensure that the relative processing techniques used in fine mode navigation are functioning as expected, i.e., the relative navigation data collection and processing, filter initialization, state addition and removal, measurement update, EKF prediction and correction, and double difference ambiguity resolution steps are programmed correctly such that a reasonable solution is obtained. Two GPS antennas connected to two different receivers are
affixed to the roof of the building with a known separation of 232.0 \pm 0.5\,\text{inches}. The measurement data is collected and processed in real time on a single laptop computer to determine the relative position and velocity between the two antennas. Single difference pseudorange, carrier phase, and Doppler measurements on the L1 carrier frequency are processed in an EKF where the prediction step will employ a constant velocity dynamics model. Despite the fact that the test is static in that the antennas never actually move, the filter is operated in a 'kinematic' mode whereby the Chief receiver's estimated position and velocity (from the BESTXYZ log) is used as the base state. This base state tends to fluctuate due to inherent inaccuracies in the absolute positioning system. This processing method is selected since it mimics the procedure to be used on-orbit. The rest of the EKF parameters for the test are given in Tab. 6.27.

Prior to each test, several additional commands are supplied to the receivers to achieve better performance. First, the CSMOOTH command is used to apply carrier phase smoothing to the pseudorange measurements. A 10 second smoothing window is used. Second, the ECUTOFF command is used to apply a 20 degree elevation mask, so that only satellites at least 20 degrees above the antenna plane are tracked. Lastly, the receivers are configured to have foot dynamics, which configures the internal receiver parameters based on the knowledge that the absolute velocity is no more than 3\,\text{m/s}.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Rate</td>
<td>0.2 ,\text{Hz}</td>
</tr>
<tr>
<td><strong>Process Noise Variances</strong></td>
<td></td>
</tr>
<tr>
<td>Rel. position: $\sigma_r^2$</td>
<td>0.09 ,\text{m}^2</td>
</tr>
<tr>
<td>Rel. velocity: $\sigma_v^2$</td>
<td>$1 \times 10^{-6}$ ,\text{m}^2/\text{s}^2</td>
</tr>
<tr>
<td>Diff. clock error: $\sigma_b^2$</td>
<td>200 ,\text{m}^2</td>
</tr>
<tr>
<td>Diff. clock error drift: $\sigma_{\dot{b}}^2$</td>
<td>0.025 ,\text{m}^2/\text{s}^2</td>
</tr>
<tr>
<td><strong>Measurement Noise Variances</strong></td>
<td></td>
</tr>
<tr>
<td>Pseudorange: $\sigma_P^2$</td>
<td>1.0 ,\text{m}^2</td>
</tr>
<tr>
<td>Carrier phase: $\sigma_\phi^2$</td>
<td>$2.25 \times 10^{-4}$ ,\text{m}^2</td>
</tr>
<tr>
<td>Doppler: $\sigma_{\dot{\phi}}^2$</td>
<td>0.35 ,\text{m}^2/\text{s}^2</td>
</tr>
</tbody>
</table>

Table 6.27: Static baseline differential navigation test EKF parameters.

To obtain the truth relative state, the absolute range between the receivers is measured and is given by the variable $s$. Referring to Fig. 6.18, the east and north components of the Deputy position with respect to the Chief can be easily determined given the angle $\theta$, which is approximately measured using a map. Once expressed in this local (Chief-centered) East-North-Up (ENU) frame, the truth position in the ECEF frame can be estimated by performing a simple coordinate rotation.

The ENU frame is depicted in Fig. 6.19. The rotation matrix from the ECEF frame to the
Figure 6.18: Differential positioning test - antenna setup.

ENU frame is given by

\[ C_{ENU, ECEF} = C_1(90^\circ - \phi)C_3(\lambda + 90^\circ), \]  

(6.3)

where \( C_i \) is a type \( i \) rotation matrix, and \( \phi \) and \( \lambda \) are the Chief receiver’s geodetic latitude and longitude, respectively.

The error in the truth position in the ECEF frame will be a function of the errors in the Chief’s absolute latitude and longitude, the baseline measurement error, and the error in \( \theta \). To first order, these errors are given by

\[
\begin{align*}
\delta a &= |c_\theta| \delta s + |s_\theta| \delta \theta \\
\delta b &= |s_\theta| \delta s + |c_\theta| \delta \theta \\
\delta x &= |bc_\lambda \cos \phi| \delta \phi + (|a c_\lambda| + |b s_\lambda s_\phi|) \delta \lambda + |s_\lambda| \delta a + |c_\lambda s_\phi| \delta b \\
\delta y &= |b s_\lambda c_\phi| \delta \phi + (|a s_\lambda| + |b c_\lambda s_\phi|) \delta \lambda + |c_\lambda| \delta a + |s_\lambda s_\phi| \delta b \\
\delta z &= |b s_\phi| \delta \phi + |s_\phi| \delta b,
\end{align*}
\]  

(6.4)

where \( a = s \cos \theta, b = s \sin \theta, c_\alpha = \cos \alpha, \) and \( s_\alpha = \sin \alpha \). The absolute values of each term have been taken to provide an upper bound on the error. The accuracy of the relative positioning algorithm can only be verified to be within these error bounds. In what follows, the variables and their errors are taken to be: \( \theta = 14^\circ \pm 2^\circ, \delta s = 5.893 \pm 0.013 \) m, \( \phi = 43.7823^\circ \pm 0.0001^\circ, \) and \( \lambda = -79.4654^\circ \pm 0.0001^\circ. \)

The fine mode filter’s performance (float solution) as compared to the coarse solution, obtained from subtracting each receiver’s BESTXYZ position and velocity estimate, is given in Fig. 6.20. This graph gives the error of the estimated position and velocity magnitudes, which are known with good accuracy. Clearly, the relative navigation solution is much more accurate than the coarse solution. Following initialization, the RMS accuracies of the estimated fine and coarse position magnitudes are 10 cm and 90 cm, respectively. The coarse position solution
magnitude fluctuates wildly from one epoch to the next, sometimes by a meter or more, while the fine position solution magnitude stays relatively constant (though showing some fluctuations at the millimetre level). With an RMS error of 0.003 m/s, the fine relative velocity solution is an order of magnitude better than the coarse solution, which is accurate only to 0.06 m/s. This clearly demonstrates the benefit and necessity of a relative navigation filter.

Figure 6.19: East-North-Up frame.

The position and velocity errors in the ECEF frame, by coordinate, are given in Figs. 6.21 and 6.22, respectively. In these plots, the blue lines denote the filter’s float solution, the magenta

Figure 6.20: Static baseline fine mode test - comparison with coarse solution.
lines denote the filter’s fixed solution following ambiguity resolution, the red and green lines denote the three-sigma standard deviations as calculated from the filter’s covariance matrix, and the black dotted lines denote the error in the truth model relative states. From initializing with no knowledge of the relative state, the position states converge after about 800 s. One possible reason for this slow convergence is the large pseudorange measurement noise employed, which is selected to mitigate the pseudorange multipath errors. After convergence, the float solution stays within the truth model error bounds, thus the filter is judged to be functioning correctly.

![Fine Relative Position Estimation Errors in ECEF frame](image.png)

Figure 6.21: Static baseline fine mode test - relative position errors in ECEF frame.

For the most part, the true errors stay within the estimated error bounds, except at around 2500 s, where the x-axis error slightly exceeds the three-sigma bound. Given the small magnitude of this departure, this is not a serious cause for concern. In theory the steady state error bounds can be easily adjusted by increasing the process noise variance. The slight jumps in estimated error are thought to be correlated with changes in the number of commonly viewed satellites. The relative velocity error remains small, and well within its estimated error bounds for the duration of the test.

Following convergence, the first fixed solutions become available. Of these solutions, some are reasonable, while many others are off by more than 30 cm. This is likely due to improper fixing of the integer ambiguities, given that the float solution shows no such departures from the solution. In the future, the fixed solution should be verified to be within a certain tolerance
of the float solution, otherwise it should be rejected as erroneous. Alternatively, fixed solutions
should not be used for autonomous formation control on-orbit, since an improperly fixed solution
may lead to formation divergence. When the ambiguities \( \text{are} \) fixed correctly, the fixed relative
position solution is marginally more accurate than the float solution. In relative velocity, the
fixed solutions are almost identical to the float solutions.

A final important point is that the converged position solutions exhibit a slight bias of about
4 to 5 cm. Despite the fact that this bias is also observed in the magnitude of the relative range,
it may be caused by errors in the truth model. The range between the two geometric centers of
the antennas was measured; however, the geometric centers may not correspond to the phase
centers. If both centers are shifted by 2 cm in opposite directions, this could account for the
final bias error. Both float and fixed solutions show similar biases.

There are several potential sources for the final errors observed. First, biases could exist
in the measurements, thus violating the assumption of zero-mean measurement noise. Second,
it is possible that because a fixed measurement noise covariance was employed, measurements
with larger errors were used with false confidence. Third, small timing errors could be induced
since the cables attaching the antennas and receivers are not the same length. Fourth, the rel-
ative position is difficult to estimate exactly, since the base station (Chief) receiver’s estimated
absolute state changes from one epoch to the next. If this was taken to be a fixed quantity, one
would expect the relative state estimation error to approach zero. Lastly, the absolute range

---

Figure 6.22: Static fine mode test - relative velocity errors in ECEF frame.
between the two antennas was measured from their geometric centres, which may not coincide with the phase centres. If both phase centres are offset in opposite directions, this would cause a bias in the final errors.

Given the aforementioned results, it is concluded that the relative navigation filter has been correctly implemented.
Chapter 7

Conclusions

In this chapter the work presented in this thesis is summarized. In addition, the future improvements and investigations required before the control and navigation system can be deemed “flight ready” are outlined.

7.1 Summary

In this work the design and implementation of a control and navigation system for the CanX-4/-5 formation flying mission are described. While this work builds on existing work, significant steps were taken towards a functional real-time implementation of the system. The existing FIONA C flight code was improved in terms of speed by eliminating unnecessary calculations, and in terms of functionality by correcting several major errors. The orbital element determination algorithm was updated to preclude singularities. The ability to propagate orbits in the ECEF frame was added, in addition to the framework for a high fidelity gravitational force model, and the coordinate transformations from the GCI to ECEF frame and vice-versa. The coarse mode EKF was improved so that it can accept any combination of Chief/Deputy measurements, and a sequential method for the EKF correction state was implemented to significantly reduce execution time. A computationally efficient, accurate, and robust method for formation reconfiguration maneuvers was designed, implemented, and tested. The robust method was shown to outperform the previous method in all areas. Finally, the FIONA execution time was tested on representative flight hardware, and a parameter set yielding execution times near 1 s was found and used in all subsequent formation control simulations.

The updated FIONA was tested and validated in MATLAB simulations with an improved truth model, which employed a high-fidelity force model, fixed time step integration, representative GPS blackout simulation, and correlated Gaussian white noise corrupting the relative state estimates. The algorithm was shown to produce stable formation flying with both fine and
coarse relative state estimates. The coarse navigation filter was shown to produce acceptable absolute and relative state estimates in addition to reasonable error estimates despite the frequent occurrence blackouts. It was demonstrated that the requirement of sub-meter formation tracking can be met, on the condition that “fine” relative estimates are available for a long enough period of time. The reconfiguration algorithm also worked to provide accurate formation transfers for both coarse and fine relative state estimates, such that the formation keeping LQR controller was stable when it took over after the transfer. The greatest impediment to the formation flying control accuracy and fuel consumption was found to be errors in the relative velocity estimate.

Drawing heavily upon the existing literature, a real-time “fine mode” relative navigation filter was designed and implemented. The processing algorithms were tested in static measurement tests in both absolute and relative navigation mode. The results of these tests indicate that the filter has been programmed correctly, and is ready to undergo further testing and integration with the flight hardware.

7.2 Future Work

Despite the work that has been performed to date, there are still some improvements that should be made to FIONA. The biggest change is the timing of the EKF prediction step: it should be performed at the start of time step “k+1” rather than at the end of step “k” (the current step), where it is now performed. This change is required so that feedback from CNAPS and the ACS can be incorporated into the state prediction, and that an assumed control force is not erroneously assimilated into the state estimate. The second required change is the addition of a “station-keeping” mode, where the relative separation between the Chief and Deputy is maintained in the absence of inter-satellite communication. This mode is required to prevent excessive drift during the initial phases of the mission, and during any emergency situations. A basic design would be to have the Deputy follow a pre-programmed (or uploaded) relative trajectory; however, such trajectories have not yet been designed, and the mode is yet to be added.

A third possible change is to continuously update the satellite mass variable to reflect the changes in mass due to loss of fuel. This would lead to more accurate control thrusts, since the discrepancy between the assumed and actual impulse would be much lower. A fourth possible change is to add a data structure which fully parameterizes each reconfiguration maneuver. In the current implementation, the satellite operator would need to manually input each reconfiguration maneuver parameter set by hand. This change is not critical, since a reconfiguration needs to be initiated by the ground in any case. Another possible improvement would be to
program additional reference trajectories or additional formation keeping modes (e.g. impulsive thrusting using orbital element differences), which could be performed after the main mission objectives have been fulfilled.

In the future, the FIONA simulation environment should be enhanced to simulate role switching, i.e., CanX-4 and CanX-5 exchanging the roles of Chief and Deputy. In theory, this switch should be quite simple, since identical software is running on both satellites. However, to date this has not been simulated.

Formation flying simulations should continue to be performed, and the effect of non-Gaussian noise on both coarse and fine navigation modes should be investigated. In addition, the noise models should be updated to reflect a worst-case scenario for absolute state estimation, based on CanX-2 data.

Aside from the aforementioned code and simulation changes, there remains much future work in integrating the control and navigation systems. The first step is to perform the differential GPS testing on two laptop computers, each connected to a GPS receiver, with ethernet communication between the two. In this way, the communication protocol (e.g., data packet formation, initiation of transmission and reception) for the ISL can be developed. Further, delay in the navigation system can be simulated in software. Next, the GPS receivers should be connected to the flight computers, and verification of the data parsing and processing algorithms should be performed. At first, the two flight computers can communicate their information using a cable, and then the ISL radio hardware can be incorporated. Static relative navigation algorithm testing should be performed to ensure that the results agree with that produced on the laptops. In addition, improvements to the algorithm should be made so that the execution time of the algorithm is no more than 3 seconds.

The final tests will need to be performed using a GPS signal simulator. In these tests, all hardware should be inter-connected in as close to a flight configuration as possible, and closed loop navigation and control demonstrated.
Bibliography


Appendix A

EKF Matrices

Given the motion model for the EKF (both fine and coarse) given in Eq. (4.15), with both \( u = 0 \) and \( f_p(R) = 0 \),

\[
A(R) = \frac{\partial f}{\partial x} = \begin{bmatrix}
0_{3 \times 3} & I_{3 \times 3} \\
-\omega_\oplus & \frac{\mu}{\|R\|^3} (R^T R - 3 R R^T) & -2 \omega_\oplus
\end{bmatrix},
\]  

(A.1)

where \( R \) is the position vector expressed in the ECEF frame, and \( \omega_\oplus \) is the angular velocity of the Earth, which is assumed to be constant with only a Z component.

One can obtain a rough estimate of the orbit state process noise covariance matrix by keeping the linear term in the Taylor series expansion of the equation of motion:

\[
f(x + \Delta x) \approx f(x) + \frac{\partial f}{\partial x} \big|_{x} \Delta x + h.o.t.
\]  

(A.2)

Thus to first order the process noise covariance matrix is given by

\[
Q_{w,x} = E \left[ \left( \frac{\partial f}{\partial x} \big|_{x} \Delta x \right) \left( \frac{\partial f}{\partial x} \big|_{x} \Delta x \right)^T \right] = \frac{\partial f}{\partial x} \big|_{x} Q_{\Delta x} \left( \frac{\partial f}{\partial x} \big|_{x} \right)^T,
\]  

(A.3)

where \( Q_{\Delta x} \) is the covariance matrix of the state error. However, as discussed in Sec. 4.2, this process noise is not actually used to reduce the computational cost. Instead, a constant process noise covariance matrix given by

\[
Q_{w,x} = \begin{bmatrix}
I_{3 \times 3} \sigma_{\Delta r}^2 & 0_{3 \times 3} \\
0_{3 \times 3} & I_{3 \times 3} \sigma_{\Delta v}^2
\end{bmatrix}
\]  

(A.4)

is used, where \( \sigma_{\Delta r}^2 \) and \( \sigma_{\Delta v}^2 \) are the position and velocity error variances, respectively.
The clock process noise matrix $Q_b$ is given explicitly as

$$Q_b = \begin{bmatrix} \Delta t^2 \sigma_{\Delta b}^2 + \Delta t^4 \sigma_{\Delta b}^2 & \Delta t^3 \sigma_{\Delta b}^2 & \Delta t^3 \sigma_{\Delta b}^2 & \Delta t^2 \sigma_{\Delta b}^2 \\ \Delta t^3 \sigma_{\Delta b}^2 & \Delta t^4 \sigma_{\Delta b}^2 & \Delta t^3 \sigma_{\Delta b}^2 & \Delta t^2 \sigma_{\Delta b}^2 \\ \Delta t^3 \sigma_{\Delta b}^2 & \Delta t^4 \sigma_{\Delta b}^2 & \Delta t^3 \sigma_{\Delta b}^2 & \Delta t^2 \sigma_{\Delta b}^2 \\ \Delta t^2 \sigma_{\Delta b}^2 & \Delta t^3 \sigma_{\Delta b}^2 & \Delta t^4 \sigma_{\Delta b}^2 & \Delta t^3 \sigma_{\Delta b}^2 \end{bmatrix}, \quad (A.5)$$

where $\sigma_{\Delta b}^2$ and $\sigma_{\Delta \dot{b}}^2$ are the variances of the terms $w_{\Delta b}$ and $w_{\Delta \dot{b}}$ in Eq. (4.31), respectively. This expression is obtained directly from the definition of the covariance matrix and Eq. (4.31).

The partial derivative of the geometric range between a receiver and a GPS satellite and the receiver satellite position w.r.t. receiver satellite position:

$$\frac{\partial \rho_s^r(t_r, t_e)}{\partial R_r} = \partial \frac{\partial}{\partial R_r} \left( (R^s - R_r)^T (R^s - R_r) \right)^{\frac{1}{2}} = -\left( \frac{(R^s - R_r)^T}{\|R^s - R_r\|} \right) = -e_r^s T, \quad (A.6)$$

where $e_r^s$ is defined as the unit vector drawn from GPS satellite ‘s’ to receiver satellite ‘r’.

The time derivative of the geometric range between a receiver and a GPS satellite and the satellite position:

$$\dot{\rho}_s^r(t_r, t_e) = \left( \dot{R}^s - \dot{R}_r \right)^T e_r^s, \quad (A.7)$$

where $\dot{R}$ is the time derivative of the position vector in the rotating frame.

The derivative of Eq. (A.7) w.r.t. receiver position:

$$\frac{\partial \dot{\rho}_s^r(t_r, t_e)}{\partial R_r} = -\left( \frac{\dot{R}^s - \dot{R}_r}{\|R^s - R_r\|} \right) \times \left( e_r^s \times \ell \right) = e_r^s \ell^T \quad (A.8)$$

Given ‘n’ common GPS satellites in view, the partial derivatives of the nonlinear GPS measurement equations with respect to the state is given by $H_k \in \mathbb{R}^{3n \times 8+n}$:

$$H_k = \frac{\partial h(x_k)}{\partial x_k} = \begin{bmatrix} E & 0_{n \times 3} & \ell & 0_{n \times 1} & 0_{n \times n} \\ E & 0_{n \times 3} & \ell & 0_{n \times 1} & I_{n \times n} \\ E' & E & 0_{n \times 1} & \ell & 0_{n \times n} \end{bmatrix}, \quad (A.9)$$

$\ell \in \mathbb{R}^{n \times 1}$ is a column vector of ones. The matrix $E \in \mathbb{R}^{n \times 3}$ is given by

$$E = \begin{bmatrix} e_1^d & \vdots & e_n^d \end{bmatrix}^T, \quad (A.10)$$

where $e_d^i$ is the unit vector from the Deputy satellite to GPS satellite ‘i’. The matrix $E' \in \mathbb{R}^{n \times 3}$
is given by
\[
E' = \begin{bmatrix}
e_{d1}^{1/T} \\
\vdots \\
e_{dn}^{n/T}
\end{bmatrix},
\] (A.11)
where \( e_{d}^{i/T} \) is defined in Eq. (A.8).
Appendix B

Flow Charts

Figure B.1: Flowchart depicting the relative navigation algorithm.

Figure B.2: Flowchart depicting the double difference ambiguity resolution algorithm.

Figure B.3: Flowchart depicting the parsing of the RANGECMP GPS log.
Figure B.4: Flowchart depicting the data quality check.
Appendix C

First Order Rotation Matrix Error

From Sec. 3.2.5.2, the rotation matrix from ECEF frame to the local frame can be written as

\[
C_{LE} (R, V) = \begin{bmatrix} \frac{R}{\|R\|} & \frac{O}{\|O\|} & \frac{H}{\|H\|} \end{bmatrix}^T, \tag{C.1}
\]

where \(H = R \times V\), and \(O = H \times R\). Now, consider \(C_{LE} (R, V)\) calculated with errors in position and velocity due to orbit determination errors: \(C_{LE} (R + \delta R, V + \delta V)\).

The question we ask ourselves is: what is the magnitude of the error induced for a given \(\delta R\) and \(\delta V\)? We seek a first order solution by making small error approximations.

Define

\[
\begin{align*}
\hat{R} &= R + \delta R, \\
\hat{V} &= V + \delta V, \\
\hat{H} &= H + \delta H, \\
\hat{O} &= O + \delta O,
\end{align*} \tag{C.2}
\]

where it is easy to show that

\[
\begin{align*}
\delta H &= R \times \delta V + \delta R \times V + \delta R \times \delta V, \\
\delta O &= \delta H \times R + H \times \delta R + \delta H \times \delta R. \tag{C.3}
\end{align*}
\]

Then define

\[
\begin{align*}
\hat{C} &= C (\hat{R}, \hat{V}), \\
\delta C &= C (R, V) \left( \hat{C} \right)^T \approx I_{3 \times 3} - \delta \theta^\times, \tag{C.4}
\end{align*}
\]

where a small angle approximation has been used. This expression can then be expanded as
\[ \delta C = \begin{bmatrix}
R^T \dot{R} & R^T \dot{O} & R^T \dot{H} \\
O^T \dot{R} & O^T \dot{O} & O^T \dot{H} \\
H^T \dot{R} & H^T \dot{O} & H^T \dot{H}
\end{bmatrix} \]  

(C.5)

Linearizing each matrix element in Eq. (C.5) about small \( \delta R \) and \( \delta V \), one obtains

\[
\delta C = \begin{bmatrix}
1 & \delta c_{12} & \delta c_{13} \\
\delta c_{21} & 1 & \delta c_{23} \\
\delta c_{31} & \delta c_{32} & 1
\end{bmatrix},
\]  

(C.6)

where

\[
\begin{align*}
\delta c_{12} &= \frac{1}{\| R \| \| O \|} R^T O^T O^T (- (H^\times + R^\times V^\times) \delta R + R^\times R^\times \delta V) \\
\delta c_{13} &= \frac{1}{\| R \| \| H \|} R^T (H^\times H^\times V^\times \delta R - H H^T R^\times \delta V) \\
\delta c_{21} &= \frac{-1}{\| O \| \| R \|} O^T R^\times R^\times \delta R \\
\delta c_{23} &= \frac{1}{\| O \| \| H \|} O^T H^\times H^\times (V^\times \delta R - R^\times \delta V) \\
\delta c_{31} &= \frac{-1}{\| H \| \| R \|} H^T R^\times R^\times \delta R \\
\delta c_{32} &= \frac{1}{\| H \| \| O \|} H^T O^T O^T (- (H^\times + R^\times V^\times) \delta R + R^\times R^\times \delta V)
\end{align*}
\]  

(C.7)
Appendix D

Simulation Constants

<table>
<thead>
<tr>
<th>Constant</th>
<th>Description</th>
<th>Value</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mu_\oplus$</td>
<td>Earth’s grav. param.</td>
<td>$3.986004415 \times 10^{15} \text{ m}^3/\text{s}^2$</td>
<td>[55]</td>
</tr>
<tr>
<td>$R_\oplus$</td>
<td>Earth’s radius</td>
<td>$6.378137 \times 10^6 \text{ m}$</td>
<td>[55]</td>
</tr>
<tr>
<td>$\omega_\oplus$</td>
<td>Earth’s angular velocity</td>
<td>$7.292115 \times 10^{-5} \text{ rad/s}$</td>
<td>[55]</td>
</tr>
<tr>
<td>$J_2$</td>
<td>Second zonal harmonic</td>
<td>$1.082626 \times 10^{-3}$</td>
<td>[73]</td>
</tr>
<tr>
<td>$J_3$</td>
<td>Third zonal harmonic</td>
<td>$-2.532547 \times 10^{-6}$</td>
<td>[73]</td>
</tr>
<tr>
<td>$J_4$</td>
<td>Fourth zonal harmonic</td>
<td>$-1.619964 \times 10^{-6}$</td>
<td>[73]</td>
</tr>
<tr>
<td>$J_5$</td>
<td>Fifth zonal harmonic</td>
<td>$-2.277928 \times 10^{-6}$</td>
<td>[73]</td>
</tr>
<tr>
<td>$J_6$</td>
<td>Sixth zonal harmonic</td>
<td>$5.406654 \times 10^{-7}$</td>
<td>[73]</td>
</tr>
<tr>
<td>$U_{\text{max}}$</td>
<td>CanX-4/-5 constant thrust level (per nozzle)</td>
<td>5 mN</td>
<td>[74]</td>
</tr>
<tr>
<td>$m_{\text{sat}}$</td>
<td>CanX-4/-5 wet mass</td>
<td>7 kg</td>
<td>[1]</td>
</tr>
</tbody>
</table>

Table D.1: Simulation constants.
Appendix E

FIONA Control Gains

\[ K_{\text{cont}} = \begin{bmatrix} 6 & -3 & 0 & 3312 & 455 & 0 \\ 4 & 2 & 0 & 455 & 2530 & 0 \\ 0 & 0 & 3 & 0 & 0 & 2293 \end{bmatrix} \times 10^{-6} \] \hspace{1cm} (E.1)

\[ K_{d,\text{ato, coarse}} = \begin{bmatrix} 3 & -1 & 0 & 1621 & 800 & 0 \\ 3 & 0 & 0 & 411 & 1461 & 0 \\ 0 & 0 & 0 & 0 & 0 & 849 \end{bmatrix} \times 10^{-6} \] \hspace{1cm} (E.2)

\[ K_{d,\text{pco, coarse}} = \begin{bmatrix} 3 & -1 & 0 & 1715 & 795 & 0 \\ 3 & 0 & 0 & 383 & 1490 & 0 \\ 0 & 0 & 0 & 0 & 0 & 928 \end{bmatrix} \times 10^{-6} \] \hspace{1cm} (E.3)

\[ K_{d,\text{ato, fine}} = \begin{bmatrix} 13 & -3 & 0 & 4977 & 746 & 0 \\ 3 & 10 & 0 & -713 & 4750 & 0 \\ 0 & 0 & 9 & 0 & 0 & 4771 \end{bmatrix} \times 10^{-6} \] \hspace{1cm} (E.4)

\[ K_{d,\text{pco, fine}} = \begin{bmatrix} 7 & -1 & 0 & 4050 & 833 & 0 \\ 1 & 3 & 0 & -768 & 3743 & 0 \\ 0 & 0 & 2 & 0 & 0 & 3742 \end{bmatrix} \times 10^{-6} \] \hspace{1cm} (E.5)