GEORGIA INSTITUTE OF TECHNOLOGY

Automated Proximity Operations Using Image-Based Relative Navigation

Special Problems Report

Luke Walker

Spring 2012

Automated Proximity Operations Using Image-Based Relative Navigation

Luke Walker

Center for Space Systems, Georgia Institute of Technology, Atlanta, GA 30332

This paper will describe a system for relative navigation and automated proximity operations for a small spacecraft about another spacecraft using continuous thrust propulsion and low cost imagers. Novel image processing algorithms provide range estimates in addition to traditional spherical angle estimates using knowledge of the target spacecraft's geometry. A differential correction batch filter is used to provide relative navigation and state estimation. These state estimates are used to provide input for the automated control of the chaser spacecraft via a Linear Quadratic Regulator. Propulsive maneuvers are accomplished using several low-thrust, non-throttleable thrusters using pulse-width modulation and thrust vectoring. A waypoint logic controller is used to define intermediate goals to reach the final goal in order to limit operational risk from an error in estimation of the spacecraft's relative state. The system is described and then initial simulation test results are shown.

I. Introduction

The Space Age approaches its 60th anniversary, and the space environment has become a crowded place. Scientific, military, and commercial interests all vie for safe and reliable access to space. As of January 2012, NASA estimated that there was approximately 16,000 cataloged objects on-orbit, including 9,000 pieces of fragmentation debris.[1] As the number of on-orbit objects continues to increase, the importance of Space Situational Awareness grows. Space Situational Awareness (SSA) may be defined as the knowledge of "the location of objects in Earth orbit to specified accuracies as well as operational status, size, shape, and mission." [2] This knowledge of space objects is acquired through measurements by ground-based assets (e.g. radar range-finding) and increasingly, on-orbit sensing and inspection by spacecraft.

The on-orbit sensing of a spacecraft by another spacecraft requires several pieces of technology related to the field of Guidance, Navigation and Control (GN&C). These technology capabilities must be developed and tested on-orbit if SSA capabilities are to be demonstrated. Of particular interest are rendezvous, relative navigation, and proximity operations. Additionally, the role of automation in future on-orbit inspection capabilities is very important. Past efforts for the rendezvous and proximity operations of two spacecraft have involved humans in the loop onboard the spacecraft (e.g. Gemini VI-A and Gemini VII) or ground control in the loop during operations. It is highly desirable to increase the level of automation, such that future missions will be able to autonomously maneuver and inspect objects; this will increase the operational efficiency of missions and provide abilities for formation-type missions.

Past Missions

Relative navigation, autonomous rendezvous and proximity operations have been examined both within the academic environment and in designed-and-built missions by several space agencies. These missions were mostly intended as technology demonstrators, designed to demonstrate critical portions of the larger problem of autonomous behavior. These missions include DART, XSS-10 and XSS-11, and Orbital Express; their characteristics are examined below.

The NASA Demonstration of Autonomous Rendezvous Technology (DART) Mission was designed to demonstrate the use of certain autonomous rendezvous capabilities, including the use of the Advanced Video Guidance Sensor (AVGS). DART was to perform an autonomous rendezvous and proximity operations with another spacecraft, the Multiple Paths, Beyond-Line-of-Site Communications

(MUBLCOM) satellite. MUBLCOM had been outfitted with optical reflectors to allow for imaging-based navigation. [3] AVGS targeted lasers on the spacecraft and tracked the reflection of the light by the corner-cube reflectors on MUBLCOM; by utilizing two different wavelengths of lasers (800 nm and 850 nm) and selective absorbance by the reflectors, the sensor was able to "subtract" one picture from a second, allowing for range and orientation estimates. Additionally, the two spacecraft used a GPS cross-link to allow for rendezvous before AVGS could be used. [4] Unfortunately, the mission was only a partial success, as DART automatically shut off when it came within 100m of MUBLCOM, and moved to a different orbit. [5]

Two Air Force Research Laboratory (AFRL) spacecraft have also demonstrated autonomous proximity operations capabilities. The XSS-10 microsat was launched on January 29, 2003 to demonstrate semi-autonomous operations about a Delta II second stage. XSS-10 used a visible camera system (VCS), rate gyros and accelerometers, and an internal propagator to estimate the relative state of XSS-10 with respect to the Delta II. By integrating the motion of XSS-10 as it was ejected from the launch vehicle and adding in visible images of the Delta II, the relative states of the two spacecraft were determined. [6] The XSS-11 mission also served to demonstrate autonomous rendezvous and inspection capabilities when it was launched in April 2005. During its mission, it rendezvoused with several resident space objects (RSO), including the fourth stage of the Minotaur I that carried XSS-11. Chief among the technologies used on XSS-11 was the Rendezvous Lidar System (RLS) that was used to estimate the range of the target RSO for rendezvous; by "painting" the RSO with laser pulses, the system created a 3-D image of the RSO for state estimation. [7]-[10]

Finally, the DARPA Orbital Express mission exhibited several advanced technologies and capabilities. Orbital Express was designed to demonstrate autonomous proximity operations and on-orbit refueling. Two spacecraft were used for this mission: ASTRO, a servicing demonstrator, and NEXTSat, a prototype for future serviceable satellites. The Autonomous Rendezvous and Capture Sensor System (ARCSS) on ASTRO was used as the primary technology for relative navigation and rendezvous at medium ranges. The ARCSS included a wide field-of-view (FOV) visible camera, an infrared camera, and a precision laser rangefinder. Additionally, the AVGS developed for DART was also used for close-range navigation and capture. [11] Included in the operations of the two spacecraft was the successful docking of the two craft and transfer of hydrazine propellant from ASTRO to NEXTSat. [12]

These missions have demonstrated key technologies and capabilities related to relative navigation and autonomous rendezvous and proximity operations. However, there are several key capabilities that

have not been addressed by current missions. To perform relative navigation with another body, all missions used either an active sensor such as a laser range finder (DART, XSS-11, Orbital Express) or a precisely known state of the other spacecraft, either through GPS cross-link (DART) or an ejection point of the launch vehicle (XSS-10). A key capability for future missions involving orbital debris or non-cooperative spacecraft will be relative navigation without previous high-fidelity knowledge of the other object. Additionally, passive techniques are desirable for certain hostile spacecraft maneuvers. Second, all previous missions have used impulsive thrust maneuvers to perform trajectory control. All missions have used some sort of hydrazine propulsion system, which allowed them to generate enough thrust for impulsive maneuvers. The use of electric propulsion and other low-thrust options is a desirable feature for future missions, given their ability to increase mission lifetimes. Finally, most of these missions exhibited some form of "ground in the loop" behavior during critical maneuvers, such as Orbital Express' docking and fluid transfer. Increasing the level of autonomy for operations is also desirable for future missions. A summary of the characteristics of the previous missions can be seen in Table 1.

Table 1. Current Proximity Operations

Spacecraft	Agency	Mass	Cost	Propulsion	Key Technology
DART	NASA	363 kg	$95 Million	Hydrazine	AVGS (Laser Reflectance) GPS Cross-Link
XSS-10	AFRL	27 kg	$100 Million [7]	MMH / N204	VCS Relative Propagation Semi-Autonomous Behavior
XSS-11	AFRL	138 kg	$82 Million [10]	Hydrazine	RLS Autonomous Planner
Orbital Express	DARPA	952 kg 226 kg	$300 Million [13]	Hydrazine	ARCSS (Cameras, Laser Ranging) AVGS Autonomous Docking

Relative Navigation

A critical problem within the area of automated rendezvous and proximity operations is the problem of relative navigation, also referred to as relative orbit determination. Any maneuver strategy for a spacecraft must first determine where the two or more spacecraft are in relation to each other. A variety of strategies may be employed to determine these relative states. First, ground-based state estimates for spacecraft provide an inertial solution for positions; these estimates are often in the form of a Two Line Element (TLE) or Vector Covariance Message (VCM) from the US Joint Operations Center.

However, TLEs and VCMs both offer positional accuracies of only kilometers, far too large for most proximity operations. [14] Alternatively, if all spacecraft are operating GPS receivers and are cooperative, the relative position of the spacecraft can be known to less than a meter. However, if one or more spacecraft are inoperable, uncooperative, or do not possess GPS receivers, this method is not valid. A combination of an active range finder (e.g. Lidar) and imagers offers solutions for both range and angle estimates of the relative position; this is the approach utilized by most of the previous proximity operations missions. As previously mentioned, there is an interest in removing the active sensing portion from this sensing process. However, the current usage of imagers has been limited to angle estimation. It can be shown that for given line-of-sight vectors provided by angle estimates, the solution for the relative orbit is non-unique; only families of relative orbits can be determined, as can be seen in Figure 1. [15] These non-unique solutions will not suffice for automated proximity operations; therefore a range estimate must be incorporated.

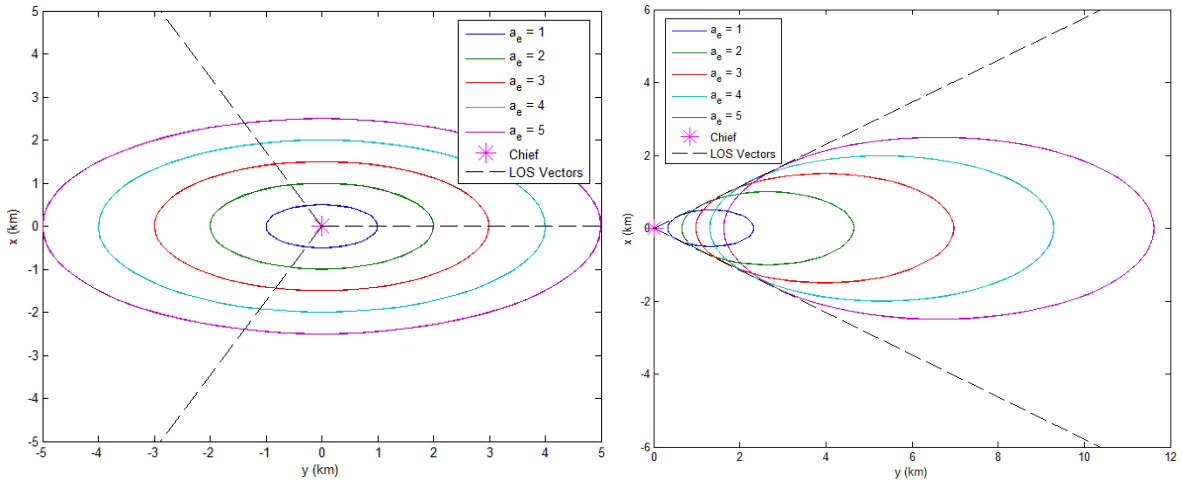

Figure 1. Families of Orbits for Angles-Only Estimates [15]

Automated Maneuver Planning

A second critical aspect of automated proximity operations is the maneuver planning required for the operations. The maneuver planning must take the spacecraft from its estimated relative state and move it to a desired position, given the designed mission profile, by defining a series of propulsive maneuvers. Within automated proximity operations missions, these propulsive maneuvers have been traditionally defined as impulsive maneuvers, applying a change in velocity (ΔV) instantly. This method allows for simpler maneuver definition and orbit propagation, especially in the linear Clohessy-Wiltshire

framework. Low-thrust, non-impulsive automated maneuver planning has not been used on-orbit, although these systems present several advantages over traditional propulsion systems.

Summary

It is clear that the field of automated proximity operations, including relative navigation and automated maneuver planning, is very important to future SSA capabilities and offers several important areas for development. The rest of this paper will describe a proposed system to advance these capabilities by utilizing imagers to perform range and angle estimate for relative navigation and a simple automated maneuver planning strategy utilizing continuous thrust propulsion. Section II will describe the system components and interactions, Section III will provide an overview of the Simulation, Analysis, and Testing required, and Section IV will describe the future work for the system.

II. System Overview

The Auto-Navigation System (AutoNav) was designed based upon a survey of the existing proximity operations capabilities and the requirements for future systems. As highlighted previously, the areas of passive imaging-based relative navigation and automated low-thrust propulsion maneuver planning were identified as key areas of interest. The basic design goals of the system were developed from these areas:

1. Rendezvous and proximity operations with target spacecraft using continuous thrust propulsion.
2. Automated relative navigation and control on-board spacecraft.
3. Closed-loop attitude control based upon automated image processing.
4. Relative orbit determination using angle and range estimates from low-cost imagers and image processing.

These design goals influenced the design of the system and helped generate the requirements flowdown.

Subsystem Interaction

There are several spacecraft subsystems and software components that must interact for the system to perform effectively. These subsystems and components include the imagers, Attitude Determination and Control Subsystem (ADCS), Image Processing Algorithms, Relative Orbit Determination, Maneuver

Planning, and Propulsion subsystem. A basic block diagram of the system interactions can be seen in Figure 2. The critical subsystems will be described in more detail in the following sections.

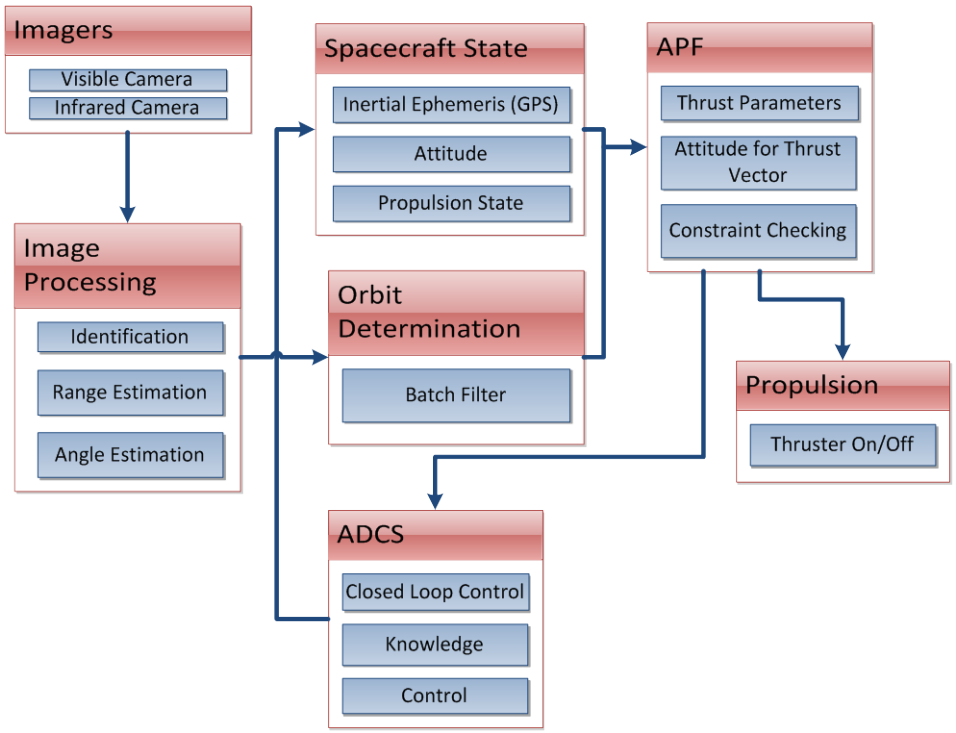

Figure 2. Block Diagram of AutoNav System

Image Processing Algorithms

Image Processing Algorithms (IPA) are used to process the images taken by the imaging instruments in order to provide positional information about the target spacecraft. There are three main steps involved in the IPAs that lead to positional information: 1) Identification of the Target Spacecraft, 2) Unit Vector Determination, 3) Range Estimation. This section summarizes the operations undertaken within each step; for a more detailed explanation, see [16].

Identification of Target Spacecraft

When an imager, in this case visible or infrared, takes an image, the target spacecraft must first be identified within the image. The Blobber Algorithm is the name of the algorithm developed to serve this purpose; it is named after its method of finding “blobs” of pixels within an image. First, it detects the pixels in an image which are in a certain range of intensity; for a thermal infrared imager, this intensity is an analog to temperature, and for a visible camera it is a measure of visible radiance. Next it detects contiguous groups of pixels that share similar intensities; a group of pixels forms a blob (Binary Large

OBject). It will often occur that this process will detect more than one blob, especially when the image background is the Earth. However the target spacecraft should be one of these blobs, if the closed-loop attitude control is effective. Therefore the remaining blobs can be screened for size parameters to determine the correct blob. Given estimated ranges of the target spacecraft and the estimated size of the spacecraft, minimum and maximum expected pixel areas can be computed that will serve to filter incorrect blobs. If only one blob meets these requirements, it is assumed that this is the target spacecraft. If no blobs or more than one remain, the images will be discarded and the imagers will provide a new set.

The successful results of the Blobber algorithm are a calculated area for the identified blob and the location of the Center of Brightness (COB) for the blob. The COB is similar to an area centroid of or a center of mass and is used as the central location of the CubeSat. The COB coordinates are given in a (X_{COB},Y_{COB}) orientation with respect to the imager's focal plane; therefore these coordinates will have to be transformed to find the unit vector in body-fixed frame (BFF) coordinates for the chaser spacecraft. An example of the resulting images from the Blobber algorithm can be seen in Figure 3. The image on the left was taken with a thermal infrared camera of a heated 3U CubeSat at 50m. The image on the right is the resulting image after processing.

Figure 3. Blobber Algorithm Results from Infrared Camera

Unit Vector Determination

Once the target has been identified, its relative location with respect to the chaser may be determined. This location may be expressed as a line-of-sight unit vector or as spherical coordinates. Using the COB coordinates, this location can be determined using principles from an imager's optics. Since the target

spacecraft can at all times be considered to be focused at infinity in relation to the focal length of the camera lenses (Range of Target >> Focal Length), the calculation of the unit vector or rotation angles can be determined. The unit vector$\vec{u}$, as seen in Figure 4, can be determined using the focal length of the lens and the coordinates of the COB. Alternatively, the position vector of the target can be expressed in spherical coordinates using the radial distance of the COB and the rotation angles θ and ϕ.

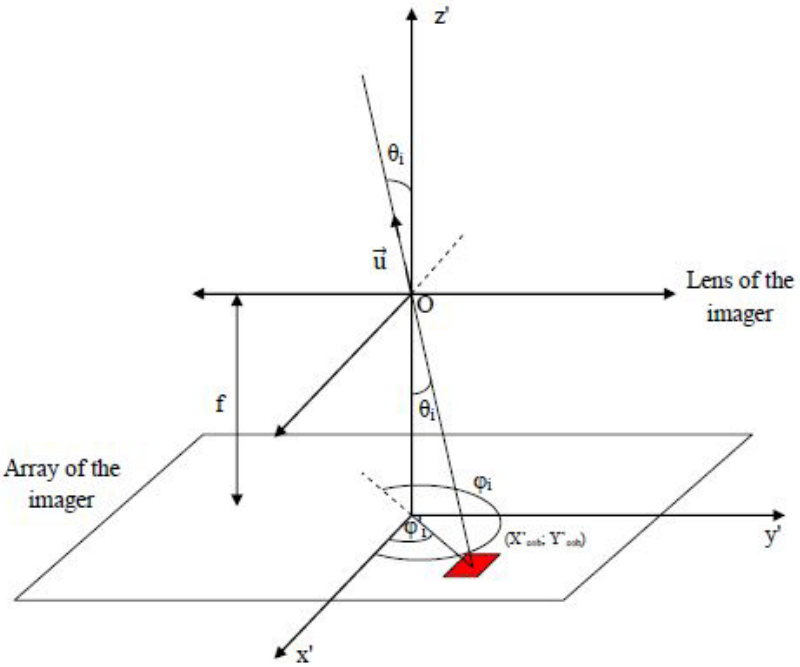

Figure 4. Unit Vector Determination (Credit: Bellet)

(1) and (2) defines the spherical angles from the COB coordinates, and (3) shows how the unit vector is calculated from the spherical angles. These angles and unit vector are expressed in the focal plane frame (FPF).

$$\theta = \text{atan}\left(\frac{\sqrt{p^2(x_{COB}^2 + y_{COB}^2)}}{f}\right) \quad (1)$$

$$\phi = \text{atan2}(\text{y}_{\text{COB}}, \text{x}_{\text{COB}}) \quad (2)$$

p = Pixel Pitch of Imager

f = Focal Length of Imager

$$\underline{u} = \begin{bmatrix} \hat{x}_{FPF} \\ \hat{y}_{FPF} \\ \hat{z}_{FPF} \end{bmatrix} = \begin{bmatrix} \sin(\theta_i)\cos(\phi_i) \\ \sin(\theta_i)\sin(\phi_i) \\ \cos(\theta_i) \end{bmatrix} \quad (3)$$

The unit vector must then be rotated to match the BFF. The rotation from the FPF to the BFF is spacecraft geometry-defined; a rotation matrix $\mathbf{R_i}$ defines this transformation and can be seen in (4).

$$\begin{pmatrix} \hat{x}_{BFF} \\ \hat{y}_{BFF} \\ \hat{z}_{BFF} \end{pmatrix} = \boldsymbol{R_i} \begin{pmatrix} \hat{x}_{FPF} \\ \hat{y}_{FPF} \\ \hat{z}_{FPF} \end{pmatrix} \quad (4)$$

Range Estimation

Once the target has been identified and its unit vector determined, the range from chaser to target is determined. Range estimation is based upon the ratio between the sensed area of the blob by the imager and the actual area of the target spacecraft. However, the actual projected area of the spacecraft is unknown, unless both the exact dimensions and the current orientation of the target are known. If either of these is not known, alternative methods must be employed. For the purpose of this study, the dimensions of the spacecraft are known but the orientation is not. For this purpose, range estimation can be broken down into four main steps: 1) determine the major and minor axes of the blob, 2) calculate the ratio between the lengths along these two axes, 3) estimate the minimum and maximum apparent areas of the target, 4) determine the range and uncertainty.

1. **Determine the major and minor axes of the blob**

   Because the orientation of the target spacecraft is unknown, and therefore the projected area is unknown, the algorithm must first seek to determine the orientation of the spacecraft. A 2-D proxy for spacecraft orientation is the combination of major and minor axes of the blob; these axes are determined in conventional 2-D methods using the pixel locations and the COB coordinates. The results of this step can be seen in Figure 5.

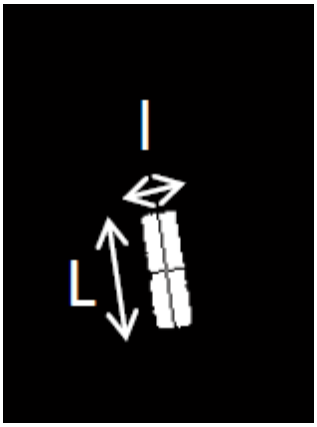

Figure 5. Major and Minor Axes of Blob

2. **Calculate the ratio of axis lengths**

   This step provides a crude understanding of the orientation of the CubeSat. Given that the image provides only two dimensions of information, it is impossible to determine accurately the orientation in all three axes, but this ratio provides a parameter useful for estimating projected area.

   Estimate minimum and maximum projected areas

   Using the ratio of the axis lengths, a range of projected areas can be determined. In order to accomplish this, numerical approximations for the projected area as a function of axis ratio must be derived. For this case, a 3U (10cm x 10cm x 30cm) CubeSat was used. The CubeSat orientation was randomly generated and its projected area and axis ratio were determined. Using thousands of samples, numerical approximations for minimum, maximum, and mean projected areas were derived. Figure 6 shows these samples and numerical functions. In the figure the blue lines surrounding the samples represent a numerical approximation of the maximum and minimum projected areas, as well as the mean projected area for any given axes

ratio. Note that the area is described in relation to A_0, which is the area of the smallest face of the CubeSat, 100 cm^2. Using the derived numeric functions, the minimum and maximum expected areas are determined.

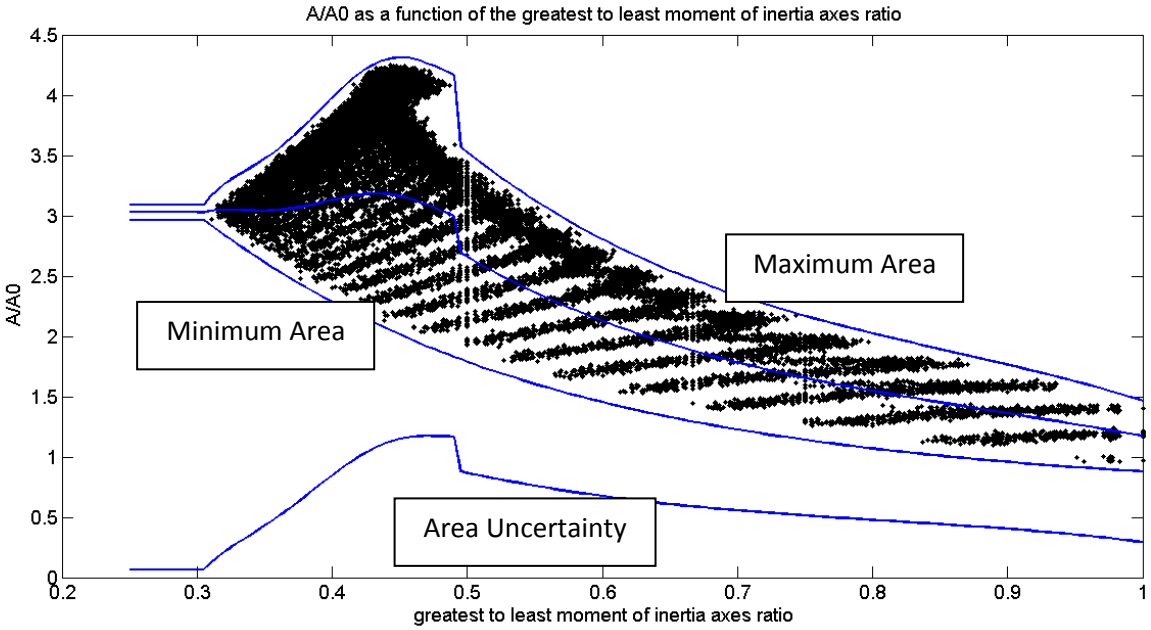

Figure 6. Projected Area as a Function of Axis Length Ratio

3. Estimate Range and Uncertainty

Using the numerical approximation for projected area of the target, an estimate of the range can be determined, using (5)-(7).

$$\rho_{Min} = f\sqrt{\frac{A_0 * \left(\frac{A}{A_0}\right)_{min}}{N_{blob}p^2}} \tag{5}$$

$$\rho_{Max} = f\sqrt{\frac{A_0 * \left(\frac{A}{A_0}\right)_{max}}{N_{blob}p^2}} \tag{6}$$

$$\rho_{Average} = \frac{\rho_{Min} + \rho_{Max}}{2} \tag{7}$$

The uncertainty of the estimate can also be evaluated; [16] shows that it is mainly dependent upon the uncertainty in the area ratio function calculated from the body axis ratio. A more detailed analysis can be seen in the paper; however, it is estimated that the mean relative uncertainty in range for the 3U CubeSat will be 16%.

The end result of the IPA function is a range estimate, a unit vector or rotation angles in BFF, and the uncertainty estimate for the range. This information will then be fed into the orbit determination filter that will help determine the relative motion of the chaser spacecraft.

Relative Navigation

The estimates from the IPAs provide the basis for relative navigation and orbit determination. Proximity operations will be accomplished using the Clohessy-Wiltshire (CW) equations in the Local-Vertical Local Horizontal (LVLH) frame, also known as the RSW frame. Therefore, unlike traditional orbit determination, this process will be focused not on inertial state but on relative state and relative orbital elements (ROEs).

Orbital Dynamics

The relative motion of two bodies in orbit about a third body in close proximity to each other has been studied and characterized. In particular, Hill [17] and Clohessy-Wiltshire [18] described a linear, time invariant system where a "chaser" spacecraft's relative motion about a "chief" spacecraft in circular orbit is described. The basic dynamics are described by (8) as modified by Vallado [19].

$$\begin{aligned} \ddot{x} - 2n\dot{y} - 3\mathrm{n}^2 x &= f_x \\ \ddot{y} + 2n\dot{x} &= f_y \\ \ddot{z} + n^2 z &= f_z \end{aligned} \tag{8}$$

These dynamics are expressed using the RSW coordinate system:

- $\hat{R}$: Radial component, collinear with the position vector.
- $\hat{S}$: In-track component, in the direction of the Chief's velocity vector for a circular chief. Formally, it is$\widehat{W}x\hat{R}$.
- $\widehat{W}$: Cross-track component, normal to the orbital plane, or$\hat{R}x\hat{V}$, where $\hat{V}$ is the unit vector for the velocity of the chief.

Within RSW, the relative position can be defined as $(x\hat{R}, y\hat{S}, z\hat{W})$ and the relative velocity as$(\dot{x}\hat{R}, \dot{y}\hat{S}, \dot{z}\hat{W})$.

For an unforced system, these differential equations can be solved for a closed-form solution, the LTI system previously mentioned. This allows the state of the chaser to be directly computed at any time, without need for numerical integration.

The motion of the chaser may also be defined by Relative Orbital Elements, analogues to orbital elements, as defined by Lovell [20]. These ROEs allow for the motion to be described in an intuitive way that allows for relative orbit design similar to the way a mission planner would design an inertial orbit using orbital elements. The transformations from relative position and velocity to ROEs can be seen in (9)-(14).

$$a_e = 2\sqrt{\left(\frac{\dot{x}}{n}\right)^2 + \left(3x + 2\frac{\dot{y}}{n}\right)^2} \tag{9}$$

$$x_d = 4x + \frac{2\,\dot{y}}{n} \tag{10}$$

$$y_d = y - 2\frac{\dot{x}}{n} \tag{11}$$

$$\beta = atan2(\dot{x}, 3nx + 2\dot{y}) \tag{12}$$

$$z_{max} = \sqrt{\left(\frac{\dot{z}}{n}\right)^2 + z^2} \tag{13}$$

$$\gamma = atan2(nz, \dot{z}) - \beta \tag{14}$$

Differential Correction Batch Filter

Given the dynamics of the system described above, a filter can be used to estimate the relative state of the chaser spacecraft, given the range and angle estimates provided by the IPAs. For the purposes of this study, a differential correction batch filter using non-linear least squares estimation was used. Future iterations of this process may incorporate an Extended Kalman Filter (EKF) to better account for the covariances of the estimates. The basics of the batch filter are described below; the formulation follows [19] with some adaptations.

Mathematical Model

The mathematical model that forms the backbone of the filter is based upon the orbital dynamics described previously. The model uses differential corrections to change the estimate for the state of the spacecraft at a particular time. There are two main components in this model, the state transition matrix (STM) and the mapping matrix.

For typical orbit determination batch filters, the model state of the spacecraft and the STM must be propagated using numerical integration between sample times. However, since this system has a closed form solution for the state, the STM and the state may be calculated directly. The equations for the STM and the state are shown in (15)-(16).

$$\boldsymbol{\Phi}(t) = \begin{bmatrix} 4-3\cos(nt) & 0 & 0 & \frac{\sin(nt)}{n} & \frac{2-2\cos(nt)}{n} & 0 \\ 6(\sin(nt)-nt) & 1 & 0 & \frac{2\cos(nt)-2}{n} & \frac{4\sin(nt)-3nt}{n} & 0 \\ 0 & 0 & \cos(nt) & 0 & 0 & \frac{\sin(nt)}{n} \\ 3n\sin(nt) & 0 & 0 & \cos(nt) & 2\sin(nt) & 0 \\ 6n(\cos(nt)-1) & 0 & 0 & -2\sin(nt) & -3+4\cos(nt) & 0 \\ 0 & 0 & -n\sin(nt) & 0 & 0 & \cos(nt) \end{bmatrix} \tag{15}$$

$$\underline{X}(t-t_0) = \boldsymbol{\Phi}(t-t_0)\underline{X}(t_0) \tag{16}$$

These equations hold for the unforced, or homogeneous, solution. For the thrusting case, a closed-form to the differential equations exists if the thrust is constant in magnitude and direction. For this case, a particular solution is added to the homogeneous solution, as seen in (17)-(18).

$$\underline{X}(t-t_0) = \underline{X}_{\boldsymbol{h}}(t-t_0) + \underline{X}_{\boldsymbol{p}}(t-t_0) = \boldsymbol{\Phi}(t-t_0)\underline{X}(t_0) + \boldsymbol{\Psi}(t-t_0)\underline{f}(t) \tag{17}$$

$$\boldsymbol{\Psi}(t) = \begin{bmatrix} \frac{1}{n^2} - \frac{\cos(nt)}{n^2} & \frac{2t}{n} - \frac{2\sin(n*t)}{n^2} & 0 \\ -\frac{2t}{n} + \frac{2\sin(nt)}{n^2} & \frac{4}{n^2} - \frac{3t^2}{2} - \frac{4\cos(nt)}{n^2} & 0 \\ 0 & 0 & \frac{1}{n^2} - \frac{\cos(nt)}{n^2} \\ \frac{\sin(nt)}{n} & \frac{2}{n} - \frac{2\cos(nt)}{n} & 0 \\ -\frac{2}{n} + \frac{2\cos(nt)}{n} & -3t + \frac{4\sin(nt)}{n} & 0 \\ 0 & 0 & \frac{\sin(nt)}{n} \end{bmatrix} \tag{18}$$

Next, the mapping matrix must be defined. The mapping matrix is used within the Normal Equations of least squares estimation. To define this matrix, the measurement model G(X) must be defined first, as seen in (19).

$$\begin{bmatrix} \rho \\ x_{FPF} \\ y_{FPF} \end{bmatrix} = \boldsymbol{G}(\underline{X}) = \boldsymbol{G}\left(\begin{bmatrix} x_{RSW} \\ y_{RSW} \\ z_{RSW} \\ \dot{x}_{RSW} \\ \dot{y}_{RSW} \\ \dot{z}_{RSW} \end{bmatrix} \right) \tag{19}$$

The model state of the chaser, expressed in the RSW frame, must be transformed into the FPF frame for the measurement model. Given the inertial ephemeris from the spacecraft's onboard state estimator and the spacecraft attitude from the ADCS, the RSW frame may be calculated and used to transform the BFF into RSW, as shown in (20).

$$\underline{r}_{RSW} = \left[\hat{R}\ \hat{S}\ \hat{W}\right] * \underline{r}_{BFF} \tag{20}$$

The mapping matrix is now defined as the partial derivative of **G**(X) with respect to the state, as shown in (21)-(22).

$$\widetilde{\boldsymbol{H}}_{\boldsymbol{i}} = \frac{\partial \boldsymbol{G}(\underline{X})}{\partial \underline{X}} = \begin{bmatrix} \frac{\partial \rho}{\partial x_{rsw}} & \frac{\partial \rho}{\partial y_{rsw}} & \frac{\partial \rho}{\partial z_{rsw}} & \frac{\partial \rho}{\partial \dot{x}_{rsw}} & \frac{\partial \rho}{\partial \dot{y}_{rsw}} & \frac{\partial \rho}{\partial \dot{z}_{rsw}} \\ \frac{\partial \theta}{\partial x_{rsw}} & \frac{\partial \theta}{\partial y_{rsw}} & \frac{\partial \theta}{\partial z_{rsw}} & \frac{\partial \theta}{\partial \dot{x}_{rsw}} & \frac{\partial \theta}{\partial \dot{y}_{rsw}} & \frac{\partial \theta}{\partial \dot{z}_{rsw}} \\ \frac{\partial \phi}{\partial x_{rsw}} & \frac{\partial \phi}{\partial y_{rsw}} & \frac{\partial \phi}{\partial z_{rsw}} & \frac{\partial \phi}{\partial \dot{x}_{rsw}} & \frac{\partial \phi}{\partial \dot{y}_{rsw}} & \frac{\partial \phi}{\partial \dot{z}_{rsw}} \end{bmatrix} \tag{21}$$

$$\boldsymbol{H}_i = \boldsymbol{\Phi}(t)\widetilde{\boldsymbol{H}}_i \tag{22}$$

Filter Algorithm

The batch filter algorithm uses differential correction to modify an initial guess for the state of the spacecraft. The initial guess for the state of the spacecraft will discussed at a later point in this discussion. For a given guess, the model predictions for measurements are computed and the residual error for each measurement is computed.

The filter works by accumulating portions of the Normal Equations for each measurement sample (for n samples) and then solving for the differential change in the estimated state. (23)-(24) show the accumulation of the Normal Equations for the samples, and (25)-(26) show the differential change.

$$\boldsymbol{L} = \sum_{i=1}^{n} \boldsymbol{H_i}^{\boldsymbol{T}} \boldsymbol{W_i} \boldsymbol{H_i} \tag{23}$$

$$\underline{M} = \sum_{i=1}^{n} \boldsymbol{H_i}^{\boldsymbol{T}} \boldsymbol{W_i} \delta \underline{y_i} \tag{24}$$

$$\delta \underline{X} = \boldsymbol{L}^{-\mathbf{1}} \underline{M} \tag{25}$$

$$\underline{X_0^*} = \underline{X_0} + \delta \underline{X} \tag{26}$$

This newly estimated state then provides the next initial guess for the filter. A convergence criterion for the RMS of the residual values for each sample can be set, such that estimation is complete when the percent change in residual RMS is less than the criterion. When convergence is met, the best estimate for the state of the spacecraft at the specified time is given. This estimate for the state can be propagated forward to any point in time; alternatively a state may be found using the previous state and the STM between the two times. Additionally, the covariance matrix for the estimate may be calculated by inverting the final **L** matrix. [19]

$$\boldsymbol{P} = \boldsymbol{L}^{-1} = \begin{bmatrix} \sigma_x^2 & \sigma_{xy} & \sigma_{xz} & \sigma_{x\dot{x}} & \sigma_{x\dot{y}} & \sigma_{x\dot{z}} \\ \sigma_{xy} & \sigma_y^2 & \sigma_{yz} & \sigma_{y\dot{x}} & \sigma_{y\dot{y}} & \sigma_{y\dot{z}} \\ \sigma_{xz} & \sigma_{yz} & \sigma_z^2 & \sigma_{z\dot{x}} & \sigma_{z\dot{y}} & \sigma_{z\dot{z}} \\ \sigma_{x\dot{x}} & \sigma_{y\dot{x}} & \sigma_{z\dot{x}} & \sigma_{\dot{x}}^2 & \sigma_{\dot{x}\dot{y}} & \sigma_{\dot{x}\dot{z}} \\ \sigma_{x\dot{y}} & \sigma_{y\dot{y}} & \sigma_{z\dot{y}} & \sigma_{\dot{x}\dot{y}} & \sigma_{\dot{y}}^2 & \sigma_{\dot{y}\dot{z}} \\ \sigma_{x\dot{z}} & \sigma_{y\dot{z}} & \sigma_{z\dot{z}} & \sigma_{\dot{x}\dot{z}} & \sigma_{\dot{y}\dot{z}} & \sigma_{\dot{z}}^2 \end{bmatrix} \tag{27}$$

This covariance matrix can be used to define the confidence intervals for the state estimate. This covariance matrix can also be propagated forward with the equations of motion. However, since the STM is closed-form and can be computed directly for any time, it is easier to use (28) to determine the covariance matrix at any time after the estimate. [21] Note that this equation only works for the unforced solution, if the covariance matrix is from unforced motion.

$$\boldsymbol{P_1} = \boldsymbol{\Phi}(t_1, t_0)\, \boldsymbol{P_0}\, \boldsymbol{\Phi}^{\mathbf{T}}(t_1, t_0) \tag{28}$$

Maneuver Planning

Once the current state of the spacecraft is estimated, the automated maneuver planning determines the necessary maneuvers to take the spacecraft from this state to the desired state from the mission profile. This section describes the basic maneuver planning strategy.

LQR Introduction

Linear Quadratic Regulators (LQR) provide a method to define the feedback control necessary to minimize a cost function for a linear system. Since the CW formulation of relative motion is a linear time-invariant (LTI) system, LQR control is well-suited. For general LQR theory, see [22]-[23]. The specific formulation will follow [24]-[25]. First, the linear dynamics of the system must be defined in a state-space model, as seen in (29)-(33).

$$\begin{aligned} \dot{\underline{X}}(t) &= \boldsymbol{A}\underline{X}(t) + \boldsymbol{B}\underline{u}(t) \\ \underline{Y(t)} &= \boldsymbol{C}\underline{X}(t) + \boldsymbol{D}\underline{u}(t) \end{aligned} \tag{29}$$

$$\boldsymbol{A} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 3n^2 & 0 & 0 & 0 & 2n & 0 \\ 0 & 0 & 0 & -2n & 0 & 0 \\ 0 & 0 & -n^2 & 0 & 0 & 0 \end{bmatrix} \tag{30}$$

$$\boldsymbol{B} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{31}$$

$$\boldsymbol{C} = I_{6x6} \tag{32}$$

$$\boldsymbol{D} = 0 \tag{33}$$

A quadratic cost function that accounts for positional error and control effort is defined in (34).

$$J = \frac{1}{2}\int_0^\infty \left(\underline{X_e}^T \boldsymbol{Q} \underline{X_e} + \underline{u}^T \boldsymbol{R} \underline{u} + 2\underline{u}^T \boldsymbol{N} \underline{X_e}\right) dt \tag{34}$$

$\boldsymbol{Q}, \boldsymbol{R}, \boldsymbol{N}: Weighting\ Matrices \qquad \underline{X_e}: Tracking\ Error \qquad \underline{u}: Control\ Effort$

The tracking error $\underline{X_e}$ is defined as the difference in current state and desired state. The weighting matrices **Q** and **R** are now defined; the **N** matrix is set to zero.

$$\boldsymbol{Q} = \begin{bmatrix} \frac{\alpha_{Q_1}}{x_{max}^2} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{\alpha_{Q_2}}{y_{max}^2} & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{\alpha_{Q_1}}{z_{max}^2} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{\alpha_{Q_1}}{\dot{x}_{max}^2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{\alpha_{Q_1}}{\dot{y}_{max}^2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{\alpha_{Q_1}}{\dot{z}_{max}^2} \end{bmatrix} \tag{35}$$

$$\boldsymbol{R} = \begin{bmatrix} \frac{\beta_{R_1}}{u_{x_{max}}^2} & 0 & 0 \\ 0 & \frac{\beta_{R_2}}{u_{y_{max}}^2} & 0 \\ 0 & 0 & \frac{\beta_{R_3}}{u_{z_{max}}^2} \end{bmatrix} \tag{36}$$

$$x_{max} = y_{max} = z_{max} = r_g = Distance\ to\ Goal \tag{37}$$

$$\dot{x}_{max} = \dot{y}_{max} = \dot{z}_{max} = \frac{r_{init}}{r_m} v_m \tag{38}$$

$$\alpha_{Q_1} = \alpha_{Q_2} = \alpha_{Q_3} = \alpha_{Q_4} = \alpha_{Q_5} = \alpha_{Q_6} = r_g \tag{39}$$

$$u_{x_{max}} = u_{y_{max}} = u_{z_{max}} = u_m = \frac{F_t}{m_s} \tag{40}$$

Using the state-space model and Q and R, the algebraic Riccati equation may be solved and used to find the gain matrix K that defines the control. For techniques on solving the algebraic Riccati equation, see [23]; the MATLAB function lqr has been used for this study. The definition of Q and R using the current range of the target necessitates an iterative solution for the gain matrix K. The control vector is then defined by (41).

$$\underline{u}(t) = -\boldsymbol{K}_{LQR} \cdot \underline{X}(t) \tag{41}$$

This control will be the desired acceleration vector for the spacecraft during the time before the next decision point. Thus, given an acceleration vector and a burn time between maneuver decisions, the proper orientation and thrust control may be determined. For the purposes of this study, the spacecraft only contains thrusters in 3 body axes (+X, -X, +Y in BFF), and these thrusters do not possess throttle ability. Therefore, a combination of attitude control and Pulse-Width-Modulation (PWM) will be used for thruster control; see section "Closed-Loop Attitude Control" for more details.

LQR Operations

The operational characteristics for the LQR control must now be defined in relation to the relative guidance capabilities. There are several characteristics of the relative navigation process that will limit the functionality of the LQR control. First, the batch filter is only updated when the target appears within the FOV of the chaser imagers. During active maneuvering, it is likely that thrust maneuvers will require chaser orientations that do not allow for imaging; therefore, it must be assumed that the state estimate will not be corrected during maneuvering. The state estimate will be propagated using the thrust profile of the system, but no new filtered estimates will be given. This may be modified if LOS may be maintained, as the filter is capable of updating estimates during thrusting if images are available. The lack of state update forces the maneuver plan to operate in a quasi-open loop fashion: the spacecraft will thrust for some period of time before it is able to re-estimate its state and correct the maneuver plan.

The quasi-open loop nature of the control strategy introduces an area of risk: as the LQR control time grows, the uncertainty in state grows and the chance of re-contact or other problems grows. Because

the maneuver time required grows as the distance grows between the initial and desired states, it is necessary to define a waypoint strategy, where the LQR control guides the spacecraft to smaller distance waypoints on the way to the target point. At each waypoint the spacecraft can re-acquire LOS with the target and update the estimated state. A basic diagram of this maneuvering can be seen in Figure 7.

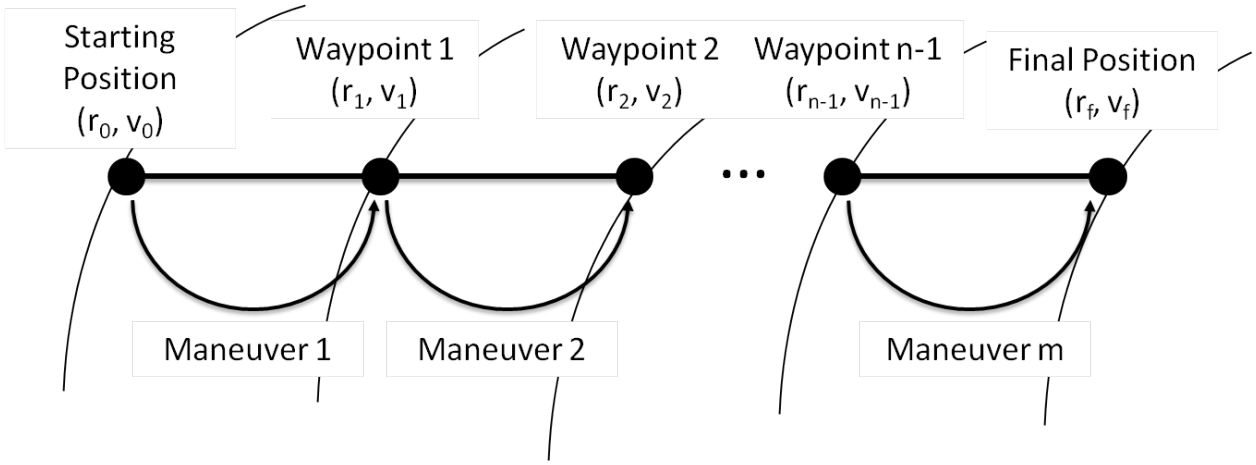

Figure 7. Waypoint Navigation

As the spacecraft enters into an automated maneuvering mission phase, its initial relative state (r_0, v_0) are estimated. The final state (r_f, v_f) are defined by the mission profile, and then the first waypoint state (r_1, v_1) is determined. LQR open-loop control defines Maneuver 1; when the propagated state has reached Waypoint 1, thrusting is stopped and the target spacecraft is re-acquired in the FOV of the imagers. The estimated state is then updated using the new relative navigation solution, and Waypoint 2 is defined. Maneuver 2 then occurs. This process occurs iteratively for *n* waypoints and *m* maneuvers until the final position is within the bounds of waypoint definition and this state is maneuvered to; at this point the spacecraft will continue to re-estimate its state periodically and perform station-keeping to maintain the desired state.

The relative distance and frequency of Waypoints come from a sensitivity analysis of the propagation of the estimation errors. The tradeoff in design is between a time-efficient maneuver and a safe maneuver: the more frequent waypoints occur, the longer the maneuver plan will take, but the more accurate the state estimate will be and the lower the risk. Currently the waypoints are defined very simply: the spacecraft shall not move more than 15m in-track during one control cycle. However, more complex waypoint strategies may be defined; see [27] and [28] for an automated “glideslope” algorithm.

The operations that occur at these states must also be determined. At the estimated arrival at a waypoint, the chaser spacecraft will proceed through a series of ADCS maneuvers to re-acquire the target spacecraft within the imager FOV, if it not already. These search patterns are informed by the estimated LOS vector from the propagated state and any images with the target spacecraft visible. At this point the spacecraft will image the spacecraft until the state can be updated; the duration of this "stop and stare" period is to be defined by the sensitivity of the batch filter to sample duration. Once the state has been updated the next target point can be defined and then LQR control may be resumed.

Propulsion Control

The LQR controller provides a thrust vector for the propulsion system, but this thrust vector must be translated into propulsion system performance. In large high-performance spacecraft with a 6-DOF propulsion system, this thrust vector may be produced directly using the correct throttleable thrusters or by slewing to the desired orientation. However, in performance and volume limited systems, several of these capabilities may not be available, including throttling, correct orientation of the thrusters, or rapid ADCS slewing. For the purpose of this study, a small 50 kg spacecraft with three small thrusters has been designed. The orientation of the spacecraft can be seen in Figure 8.

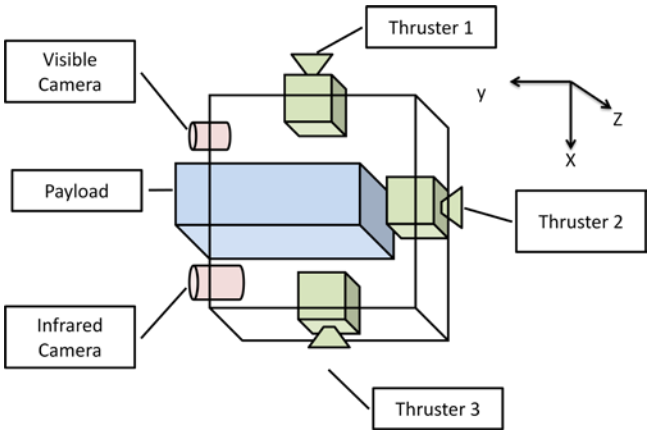

Figure 8. Spacecraft Orientation

To adequately use the proposed spacecraft propulsion system, a propulsion controller must be designed using Pulse-Width-Modulation (PWM). To account for the lack of throttle-ability in the thrusters, the thrusters will be rapidly turned on and off to generate a time-averaged desired thrust magnitude. Two thrusters may be used in tandem to provide thrust vectoring in order to limit slewing. An example thrust control for the three thrusters can be seen in Figure 9. In the case of the current spacecraft orientation, a thrust in the -Y BFF direction is not possible using thrust vectoring. Therefore, the spacecraft must slew to an

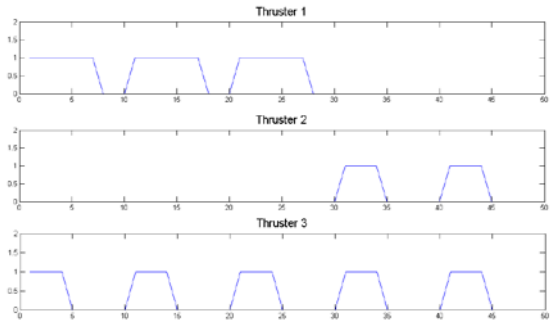

Figure 9. Example PWM Control

orientation where one of the thrusters is capable of thrusting in the desired RSW direction. The propulsion controller is responsible for determining the minimum slew required to maintain control.

Closed-Loop Attitude Control

Closed-loop attitude control is used to maintain the target spacecraft within the FOV to ensure continued state estimation. Given the results of the IPAs, the closed-loop control will determine the necessary steps required to keep the object in view. At a basic level, this requires the control to determine if the target's position violates a designated "Keep Out" zone; if the zone is violated, the control must determine the required re-orientation to align the LOS vector with the imager direction. Figure 10 shows a basic diagram of the Keep Out Zone.

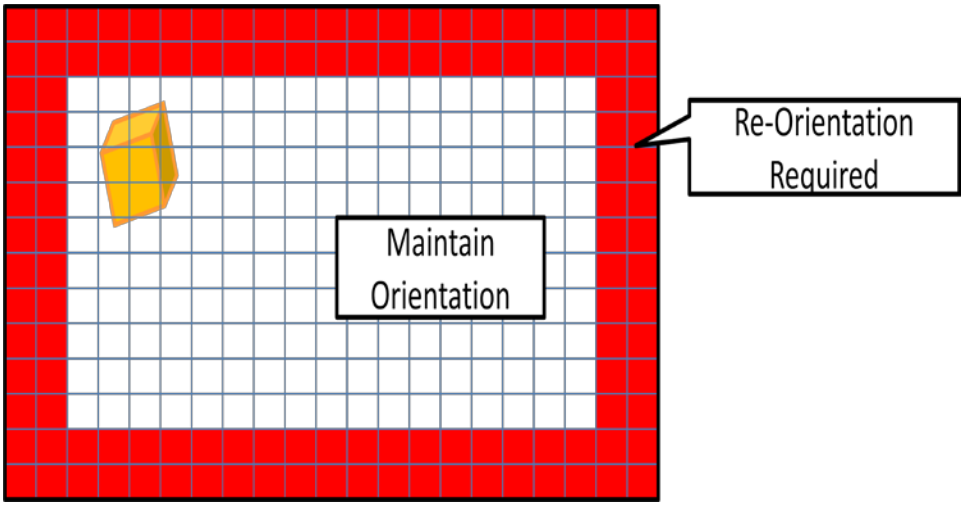

Figure 10. Keep Out Zone for Closed-Loop Control

In the case where the target appears in the image but violates the zone, a re-orientation maneuver is planned. The IPAs have estimated the LOS vector, and a simple attitude slew is designed to re-align the spacecraft with this vector. This maneuver may be specifically designed to minimize the deviation from the LOS vector during the slew. In the case where the target does not appear, the LOS vector must be calculated from the propagated state estimate and the maneuvers planned from this. However, if the state estimate presents too large an uncertainty in the LOS vector, a "Lost in Space" search method must be designed to systemize a search procedure.

Mission Profile

Although this system is generalized for use with any mission profile, a mission profile may be designed for use in simulation and analysis. For the purposes of this study, it is assumed that a ground-in-the-loop rendezvous process has been used to bring the two spacecraft to stable orbits approximately 100m from each other in the along-track (y) direction. Next, a Formation Flying Phase has been designed to move the chaser spacecraft to a stable, trailing orbit 50 m behind the target spacecraft using automated maneuver planning and image processing. This location provides a stable position for the chaser to image the target, while still maintaining a sufficient collision avoidance zone; the resolution of the imagers also helps to inform an effective imaging distance. Once in this location the maneuver planning will continue to maintain the desired position until it exits the mission phase. Future work will examine an additional phase, in which a Natural Motion Circumnavigation (NMC) of the target spacecraft is performed using automated maneuver planning.

III. Simulation, Analysis and Testing

The system must now be tested to evaluate its ability to perform the necessary automated proximity operations required for a proposed mission. Similar to hardware testing, the system will be tested on a component-level system first and then in a piece-wise integrated fashion. The following sections will describe several of these tests performed.

Open Loop Testing

Each component of the AutoNav system must be tested to analyze its accuracy and efficacy, and to analyze the impact of uncertainty on its performance. The three key components tested in this section are the IPAs, the Rel-Nav orbit determination filter, and the LQR system.

Image Processing Algorithms Testing

The primary testing of the Image Processing Algorithms was performed by Bellet [16]. The IPAs were tested for ability to locate the target spacecraft against a varied background as well as the uncertainty in range estimation. Figure 11 shows the relative range uncertainty as a function of axes ratio for the 3x1x1 cuboid geometry; it can be seen that there is an average of 16% range uncertainty for the geometry.

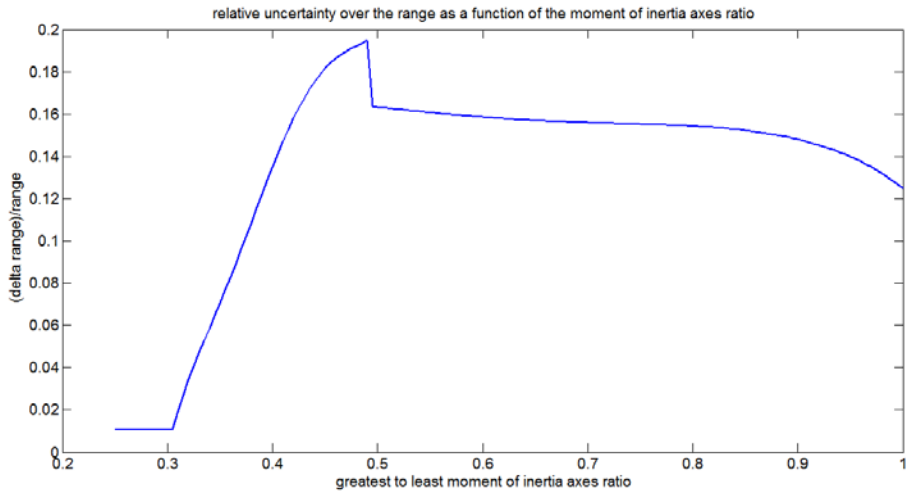

Figure 11. Relative Uncertainty of Range Estimation

The current testing has primarily used simulated images for analysis. Future testing will utilize images from a real thermal imager for better understanding of real performance. These tests will include testing with varied backgrounds and target spacecraft radiance profiles.

Relative Navigation Testing

The relative orbit determination batch filter can be tested using simulated data to evaluate its ability to estimate the state of the chaser spacecraft. In this case a reference relative orbit for the chaser spacecraft is defined, and then the state is taken at discrete periods of time. Using the measurement model previously described, the state is transformed into range and angle estimates. Gaussian noise is then added for each sample to evaluate the filter's ability to handle error and uncertainty. Once the OD filter provides an estimate for the state, the performance of the filter can be examined. In particular, the reduction in RMS residuals from the initial guess to the final solution, and the reduction in error of the state estimate are evaluated. Specifically, the norm of the state estimate error is measured, as shown in (42).

$$X_{norm\ error} = \left\| \underline{X}_{estimate} - \underline{X}_{state} \right\| \tag{42}$$

Table 2 shows a comparison of six different cases tested with noisy data. In this case, errors were assumed to be 20% for range estimates and 0.05 rad for the two angle estimates; these errors were distributed normally about the true measurement. Three different orbits were tested: a simple trailing

orbit at 50m, a static ellipse (a_e = 20, y_d = -75m), and a drifting ellipse (a_e = 20m, x_d = 2m, y_d = -55m). Each orbit was sample for two different durations, an entire orbital period and one-third of a period. The initial and final RMS values and X_{norm} errors are compared.

Table 2. Orbit Determination Filter Testing

Relative Orbit	**Time**	**RMS (Initial)**	**X_{norm} Error (Initial)**	**RMS (Final)**	**X_{norm} Error (Final)**
Trailing @ 50m	P	25.77	13.70	10.21	5.11
Static Ellipse	P	18.56	7.87	16.10	1.41
Drifting Ellipse	P	16.11	7.58	13.27	3.91
Trailing @ 50m	P/3	12.19	8.28	9.41	16.61
Static Ellipse	P/3	27.23	16.99	25.57	4.33
Drifting Ellipse	P/3	12.45	12.86	9.61	12.21

Several things may be gleaned from these tests. First, the initial orbit determination provides a reasonable estimate to begin the differential correction; on average, the initial X_{norm} error is 11.21m. Second, it is clear that a shorter duration for sampling has negative effects on final solution accuracy. It is clear that increased frequency of sampling is not sufficient; increased accuracy in sampling comes from sampling duration. Third, the batch filter is designed to minimize RMS, not necessarily final state error; therefore, in some cases the initial solution closes to a better state than the filter.

LQR Testing

The LQR guidance algorithm is tested as a controller for accuracy, rise and settle times, and control efficiency. The gains of the LQR system are set based upon these tests. For the purpose of these tests, the state knowledge is assumed to be continuous and perfect.

Additionally, the effects of an inaccurate original state estimate are tested by injecting noise into the initial state estimate. The resulting error in the final state of the spacecraft is examined, in order to

determine the sensitivity to error. This analysis is useful in determining parameters for the way-point navigation.

The gains of the LQR were set by examining the rise time of the system and the fuel expended to reach a desired location. Rise time is defined to be the time required to reach 90% of the distance following a step input in the in-track direction. For testing, the LQR received a step input to move the spacecraft from (0,0,0) to (0,100,0). The results of the testing can be seen in Table 3; the gains selected are the ones boxed in. The resulting motion and thrust profile from these gains can be seen in Figure 12. It is clear that the LQR controller effectively moves the spacecraft to the desired final state in under one orbital period.

Table 3. LQR Test Results

Inputs				Outputs			
r_{init}	r_m	v_m	Beta	Rise Time (s)	Rise Time (Orbits)	Fuel Expended (kg)	Max Thrust Exceeded
100	75	0.1	10,000	4.85E+03	0.85	7.90E-04	No
100	75	0.1	1,000	3.54E+03	0.62	1.50E-03	No
100	75	0.1	100	2.57E+03	0.45	4.00E-03	No
100	75	0.1	10	1.70E+03	0.30	1.02E-02	Yes
100	75	0.01	100	7.79E+03	1.37	5.70E-03	No
100	75	0.5	100	2.29E+03	0.40	3.90E-03	No
100	75	1	100	2.28E+03	0.40	3.90E-03	No
100	150	0.5	100	2.33E+03	0.41	3.90E-03	No
100	50	0.5	100	2.28E+03	0.40	3.90E-03	No
100	25	0.5	100	2.28E+03	0.40	3.90E-03	No
1	50	0.5	100	9.87E+03	1.74	5.80E-03	No
10	50	0.5	100	2.76E+03	0.49	4.20E-03	No
50	50	0.5	100	2.30E+03	0.41	3.90E-03	No
250	50	0.5	100	2.28E+03	0.40	3.90E-03	No

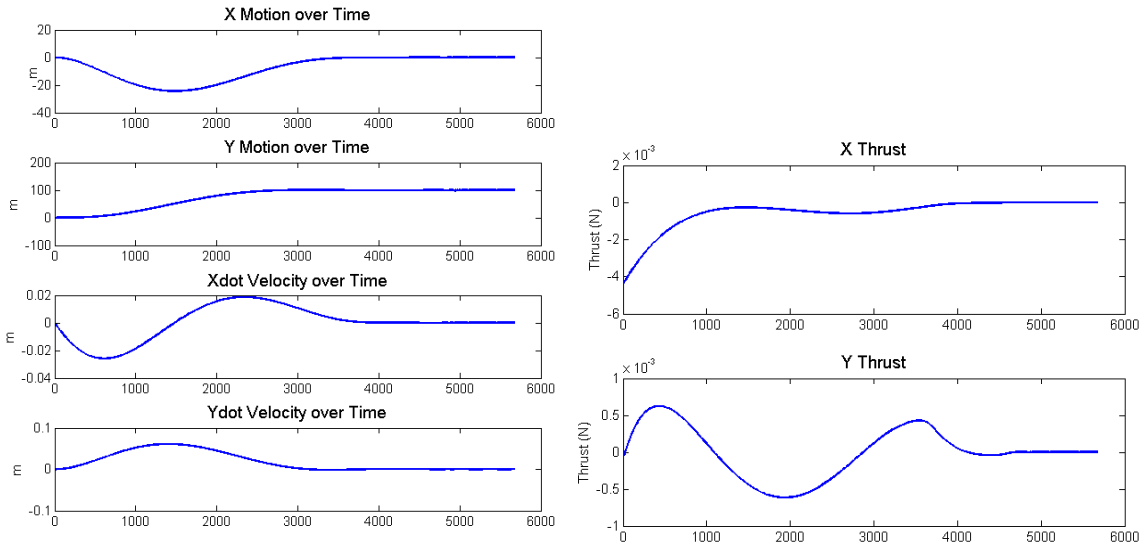

Figure 12. LQR Control Results

Next, the LQR controller must be evaluated for the influence of input estimate errors on final resulting state. Given an initial state estimate with an injected error, the LQR controller was used to control the estimated state to a desired state 100m in-track. Errors in X, Y, $\dot{X}$, and $\dot{Y}$ were examined individually, and then randomized errors were examined. The resulting errors were recorded after 3 orbits. These can be seen in Table 4.

Table 4. Results from Input Estimate Error

Input Estimate Error				Outputs			
X	Y	Xdot	Ydot	Mean X Error	Mean Y Error	Final Position Error (m)	Final Velocity Error (m/s)
10	0	0	0	39.99	188.50	377.12	0
-10	0	0	0	39.99	188.50	377.12	0
0	10	0	0	0.00	10.00	10.00	0
0	0	0.001	0	0.57	1.81	0.00	0.001
0	0	-0.001	0	0.57	1.81	0.00	0.001
0	0	0	0.001	1.81	8.63	17.03	0.001
0	0	0	-0.001	1.81	8.63	17.03	0.001
4.59	8.97	-0.0013	-0.0022	14.37	59.21	126.61	0.0025
-3.98	-1.63	-0.0009	0.0007	14.75	69.77	137.48	0.0011
-7.34	1.62	0.001	-0.0002	29.64	139.60	281.20	0.001
2.78	0.64	0	0.0003	11.60	54.06	108.73	0.0003
1.38	-0.63	0.0002	-0.0002	5.21	25.58	49.75	0.0003
-1.02	-30.73	0.0003	-0.0001	4.34	16.07	10.24	0.0003
-0.20	4.06	-0.0007	-0.0004	1.45	12.16	17.71	0.0008

There are several important results. First, estimate errors in Y and $\dot{X}$ do not significantly influence the final state. In particular, an error in Y estimates will simply shift the estimated trajectory by the error, without changing the dynamics estimated. This is very beneficial, as the largest errors in estimation will be in the in-track direction for the stated mission profile. On the other hand, X and $\dot{Y}$ errors significantly change the motion of the spacecraft. An estimate error in the relative radial state (X) of the spacecraft will cause significant control issues if not corrected; this is because a nonzero x_d causes the chaser to drift away from the target. The large final position errors are largely due to this unchecked drift; although the controller thinks the spacecraft is in a stable state, it is actually drifting quickly down-range. This reinforces the need for waypoints and state estimate updates. If the state can be updated quickly before the chaser drifts too far, the effects of the estimation error are minimized. The 100m meter desired move and 3 orbit propagation period are far too long.

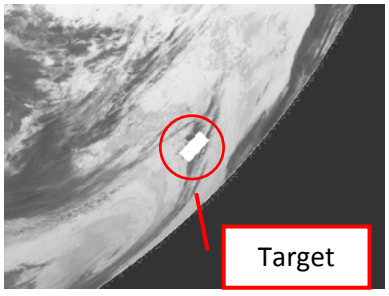

Figure 13 Simulated Image of Target

Closed Loop Simulation

Next, the components are tested when connected with each other to evaluate to propagation of errors and uncertainty throughout the process. In order to provide closed-loop simulation, an image generator was created to simulate images

of the target spacecraft at desired range and spherical coordinates. An example image can be seen in Figure 13 with the background of Earth's horizon. Using the image generator and an orbit propagator, successive steps in the AutoNav sequence were added and tested together. The specifications and performance of the chaser spacecraft must be defined for the simulation; these parameters can be seen in Table 5.

Table 5. Spacecraft Parameters

Parameter	Value
Wet Mass	50 kg
Maximum Thrust (per thruster)	4.4 mN
Thrusters	3 (+X,-X,+Y)
ADCS	3-Axis Stabilized
Imager Array Size (pixels)	640 x 480
Pixel Pitch	25 µm
Focal Length	0.1 m
Imager FOV	9.1° x 6.8°

Orbital Simulation and Relative Navigation

The first step in closed loop testing was to test the batch filter's efficacy using measurement data from the IPAs given generated images. In this process a set of ROEs were generated for the simulated orbit, and the chaser's relative motion propagated forward for one orbital period. At discrete times in the orbit images of the target spacecraft were generated at desired range and spherical coordinates. These images were processed by the IPAs and the measurement estimates were given to the batch filter for state estimation. Figure 14 shows the range estimates from a simulation, along with the true values and a 16% uncertainty band; Figure 15 shows the estimated and real trajectories of the chaser, as well as the individual measurement estimates.

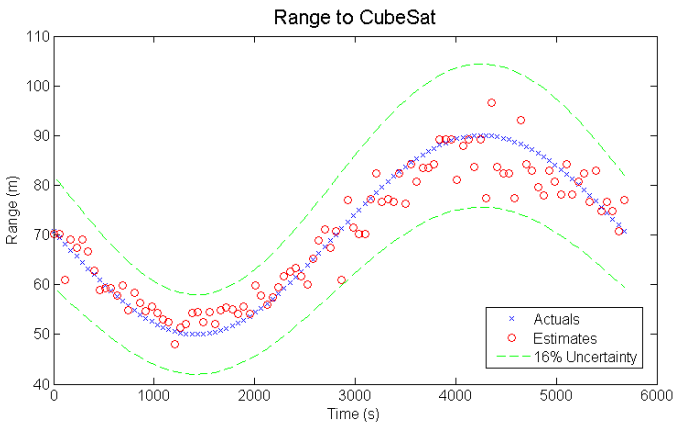

Figure 14. Range Estimates from IPAs for Simulated Orbit

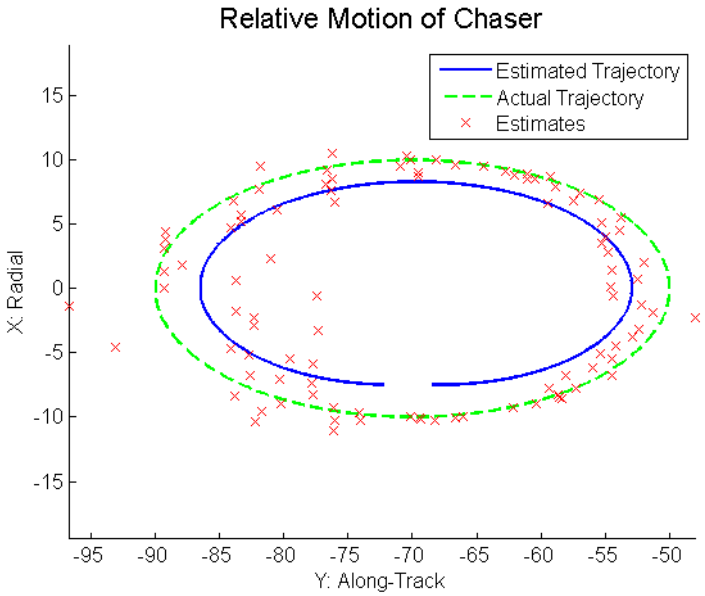

Figure 15. Estimated Relative Trajectory of Chaser

Similar to the open loop orbit determination filter tests, a series of test cases were examined to determined filter performance. The ROE estimates and the range errors are shown in Table 6.

Table 6. Relative OD Results with IPAs

Test Case	a_e (m)		x_d (m)		y_d(m)		Beta (rad)		Mean Range Error (m)
	Act.	Est.	Act.	Est.	Act.	Est.	Act.	Est.	
Trailing @ 50m	0	0.14	0	0.07	-50	-46.22	n/a	1.61	3.30
Trailing @ 125m	0	3.03	0	-0.68	-100	-107.43	n/a	4.49	20.15
Static Ellipse	20	15.56	0	-0.18	-75	-66.27	0	-0.01	9.60
Drifting Ellipse	20	21.74	2	2.60	-55	-59.61	0	-0.05	7.47

The orbit determination filter shows a good ability to determine the relevant ROEs after an orbit of imaging, especially for the cases when the target is within 100m.

LQR Control with Relative Navigation

The next step in the simulation process is to control the spacecraft using state estimates provided by the relative navigation. A block diagram for the simulation is shown in Figure 16. An initial orbit determination is used to provide an initial state estimate for the control. The waypoint logic is used to define the intermediate goals for the LQR control along the way to the ultimate goal, as defined by the mission profile. LQR control is used for one half period to move to the desired waypoint. At this time the spacecraft stops maneuvering and observes the target for one orbital period to update the state estimate. The new state estimate is given to the waypoint decision logic, and the process is repeated.

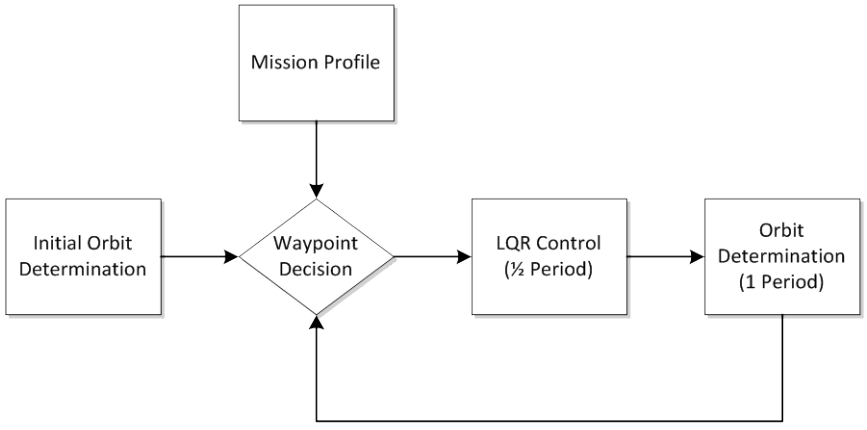

Figure 16. Simulation Block Diagram

For the purposes of testing, the chaser spacecraft was given an initial position of 110 meters behind the target and a final goal of 60 meters behind the target. The simulation was run for 15 waypoints, giving the chaser ample time to close to the goal and maintain its position. The results from one simulation can be seen in Figure 17 - Figure 19.

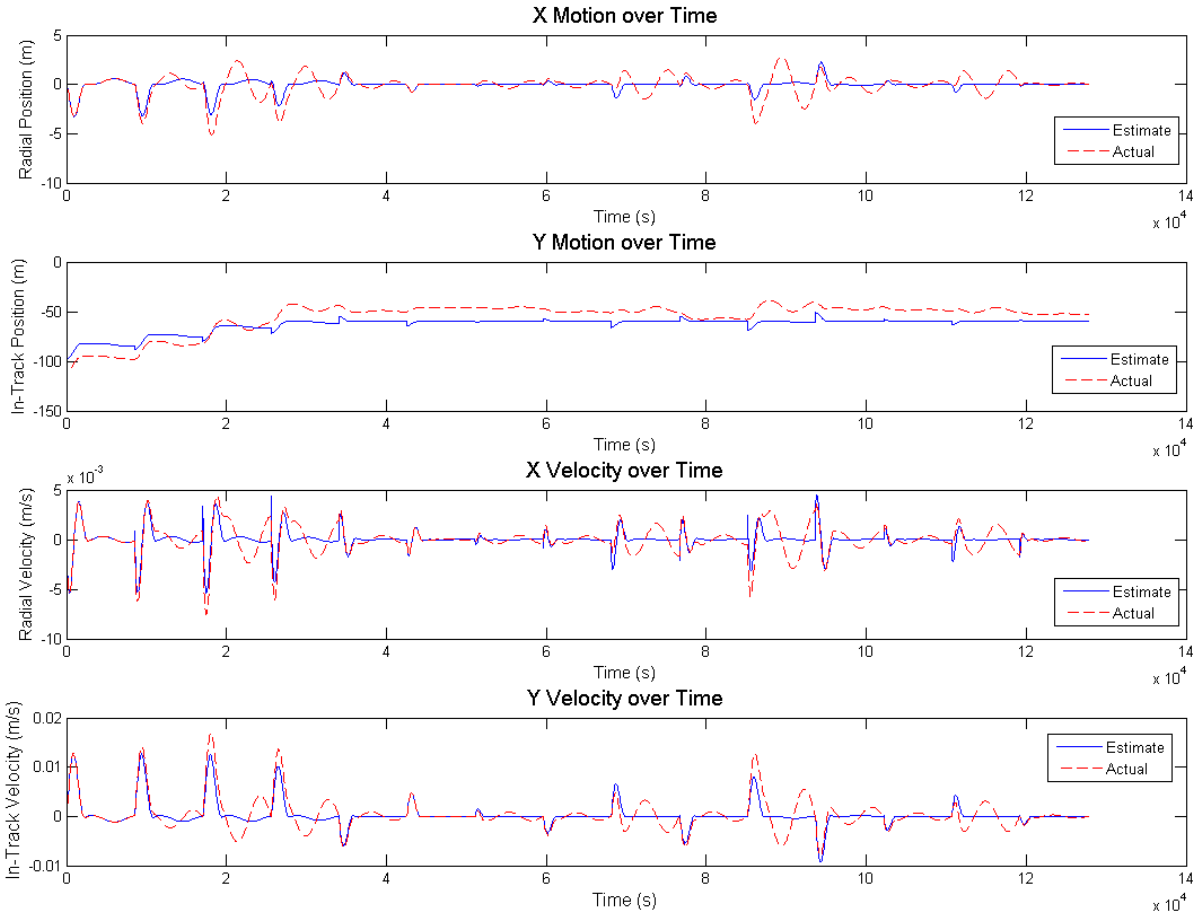

Figure 17. Chaser Spacecraft Motion over Time

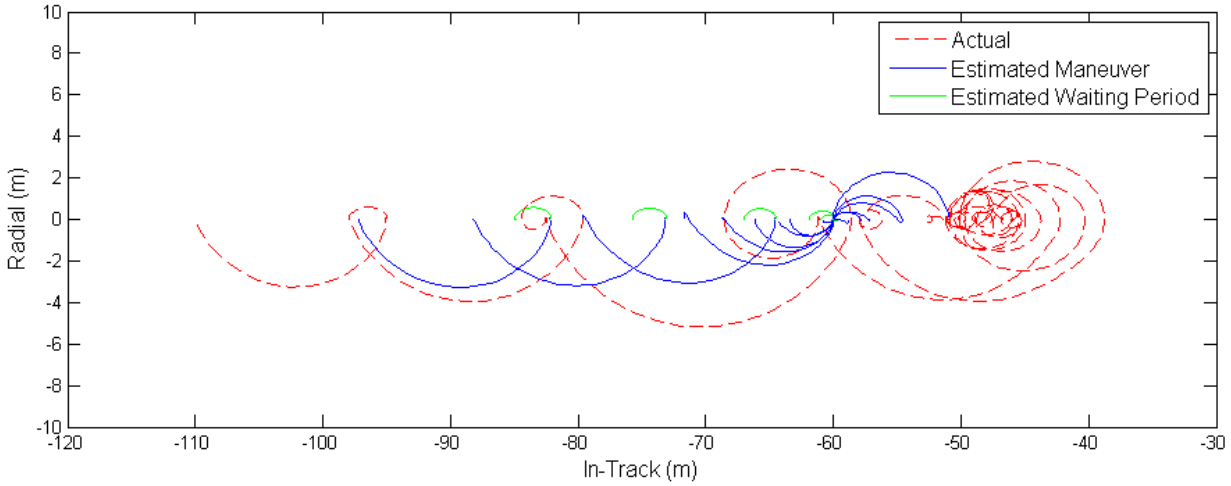

Figure 18. Trajectory of Chaser Spacecraft

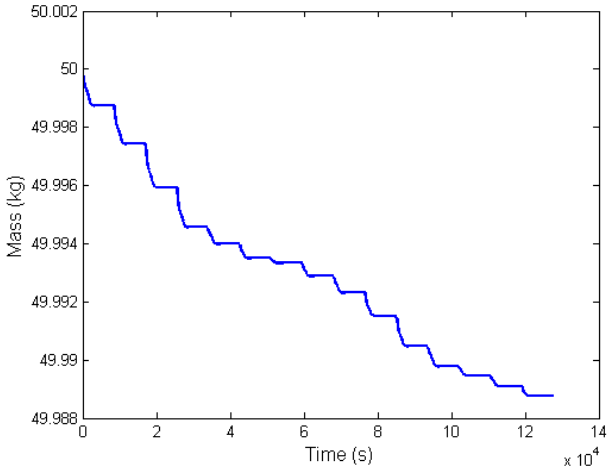

Figure 19. Mass of Chaser Spacecraft

There are several important aspects of the motion of the spacecraft in Figure 17 and Figure 18 to note. First, there is an error in the initial state estimate provided by the relative navigation filter before LQR control begins; this error is approximately 12m in the in-track direction. This error, and all other estimation errors throughout the simulation, contributes to the difference in the actual position of the spacecraft and estimated position of the spacecraft. At regular time intervals, 1.5 orbital periods in this case, the estimated state is updated and can be seen by the discontinuities in both figures for the estimated state. These state estimate updates usually bring the estimate closer to the actual position,

but this is not always the case, due to the uncertainty bands in the OD filter. The green trajectories in Figure 17 represent the periods of orbit determination where the spacecraft is not maneuvering; thus the end of these trajectories are where the discontinuities occur.

Most importantly, it can be seen that the spacecraft moves to a location close to the desired end goal, and a position well within the success criteria for the proposed mission. The final position of the spacecraft is ### away from the desired goal. Of some concern is the trajectory movement within 50m, the designated "obstacle avoidance" zone. This is largely due to the filter bias at close ranges: the filter shows a bias to over-estimate the range to the target spacecraft at 50m; however, this bias may actually be related to the image generator, not to the filter itself.

Finally, the propellant usage may be noted. Very little propellant was expended, as the LQR finds an optimal path to each waypoint, minimizing control usage by using natural dynamics for much of its movement. Thus the spacecraft only expends 0.01 kg of propellant over 22.5 orbits.

The simulation was run 20 times to examine average performance, accounting for variability in target spacecraft orientation which changes the uncertainty of estimates. The resulting statistics can be seen in Table 7. Overall the simulation shows that the LQR controller with the relative navigation filter provides a very successful and fuel-efficient solution for automated proximity operations. The controller closed to within 7.48m on average after 15 waypoints, and during navigation averaged a range estimation error of 9.17m. Finally, an average of only 0.01 kg of propellant was used; this will be useful in extended life missions or low mass missions with small propellant budgets.

Table 7. Mean Statistics for Simulations

Parameter	Value
Error in Final Position (m)	7.479
Mean Error in Position Estimate (m)	9.165
Max Error in Position Estimate (m)	18.49
Propellant Used (m)	0.0109

IV. Future Work

This work has helped to demonstrate the viability of utilizing image-based relative navigation and automated maneuver planning for proximity operations. However, there are several key areas that would enable flight-qualified capabilities by reducing operational risk and increasing the fidelity of the estimates or would enhance the value of such a mission. These areas include the development and use of an Extended Kalman Filter, the design of an NMC phase, the definition of operations rules for risk avoidance, a study of nonlinear forces, and the introduction of uncertainty and error from other subsystems.

Extended Kalman Filter

Currently a Differential Correction Batch Filter is used for the relative navigation. The batch filter shows efficacy in quickly determining a state estimate given a batch of measurements, but there are several disadvantages associated with its use. First, the filter is operated at discrete periods of time after a number of measurements are taken. Each time the filter is operated, each sample is compared against the model, residuals are calculated, and the state estimate is modified. This requires redundant processing each time a new estimate is generated, using valuable onboard computing resources. Second, the batch filter's handling of the covariance matrix does not appear to provide a realistic estimate of the uncertainty of the uncertainty. An alternative to the batch filter is a continuous Extended Kalman Filter, which would be better able to estimate the uncertainty of an estimate and provide continuous updates and operations when a measurement is added.

NMC Phase Definition

The relative navigation and automated maneuver planning have been demonstrated for a formation flying phase in which the chaser spacecraft approaches the target and maintains a stable, trailing position for imaging purposes. However, a complete circumnavigation of the target spacecraft would present a more ideal mission profile for inspection and spacecraft characterization. The harmonic motion of the CW framework presents a stable passive motion for this to occur, using an NMC.

The process of entering into an NMC using relative navigation and automated maneuver planning presents another level of complexity from the current mission profile. In this case the desired state of the spacecraft is continuously changing, rather than staying at a fixed point. Therefore the state estimate must be updated

Several potential options exist that may be explored. First, the LQR controller may be adapted to drive the spacecraft through the desired trajectory; each time a control decision is made, the state would represent the desired state at that point in the NMC. In essence, this would provide a Forced Motion Circumnavigation (FMC) that would approximate the NMC. One concern is the risk of collision when circumnavigating the target. It is possible to utilize Artificial Potential Functions (APF) in addition to LQR control to ensure collision avoidance, as seen in [24]-[25]. In this case the LQR would provide an attractive potential to drive the spacecraft to a desired trajectory, and the APF would provide a repulsive potential, centered at the target spacecraft, to prevent collision or close approach. A second option is to define a menu-driven approach that utilizes a series of pre-defined maneuvers to move into an NMC and fine-tune the NMC based upon updated state estimates. This option allows for some simplification in development work, but the robustness of the procedure is questionable.

Operations Parameters

To implement the AutoNav system for an operational mission, a number of operational procedures must be defined. During operations the target spacecraft will not be within the FOV at all times due to operational constraints, such as reaction wheel desaturation events, thrust maneuvers, and communications overpasses. This loss of sight adds risk to the operations and must be accounted for in the operations plan in several ways. First, rules concerning the reliance on the propagation of a state estimate must be determined. A state estimate provided by the orbit determination filter provides a state and a level of uncertainty characterized by the covariance matrix. This uncertainty grows over time as the state is propagated, especially if propulsive maneuvers occur. Flight rules must be defined to identify the “acceptable” level of uncertainty in the state estimates, in order to determine when operations must be ceased to re-acquire the spacecraft and re-estimate the state. Second, a search pattern to find the target spacecraft within the imager FOV must be defined. This search pattern should use information from the state propagator and the IPAs to prepare a series of ADCS slews to find the target spacecraft in an efficient manner. Efficiency in this search pattern is especially important during operations, as the target spacecraft must be re-acquired quickly before it drifts downrange or threatens to collide with the chaser. Third, an exit plan must be defined in case the uncertainty rules are violated and a safe orientation must be sought. The goal of this plan is to define a series of maneuvers that will safely take the chaser away from the target to a stable position to re-acquire line of sight.

Non-Linear Forces

The CW equations used for propagation of the spacecraft and the LQR are a linearized set with several assumptions. They assume a uniform gravity field, a perfectly circular orbit for the target spacecraft, small distances between target and chaser spacecraft, and no third-body or non-linear forces. For high-fidelity testing, a non-linear force model should be used to compare the simplified assumptions to an actual state. A non-uniform gravity field, containing at least the J_2 zonal harmonic and possibly higher order harmonics, can be used to propagate the inertial state of the two spacecraft. Non-linear forces such as drag, third body forces, and solar radiation pressure can also be modeled; see [26] for more details of such a simulation. Finally, an examination of a non-zero eccentricity for the target spacecraft should be undertaken. These simulation tools can help verify that the simplified assumptions are valid for the relevant design space.

Uncertainty and Errors

The work on the simulation to date has included uncertainty and errors only from the imager and AutoNav algorithms. Several other subsystems will generate uncertainties and errors that will affect the performance of the AutoNav system. Most importantly, the ADCS plays a large role in the ability of the system to achieve success, in three key ways. First, the inertial orbit determination provided by the GPS receiver and the inertial orbit determination filter, in conjunction with attitude knowledge, will be used to estimate the RSW frame orientation with respect to the spacecraft; this is vital for correctly orienting the spacecraft for imaging and maneuvers. Second, the attitude determination of the spacecraft, provided by a magnetometer, sun sensors, and rate gyros, will be required for this frame estimation and proper imaging. Finally, the attitude control of the spacecraft, provided by reaction wheels and torque rods, will be required for slewing for imaging and for proper thrust vector control. Errors in these three components (orbit determination, attitude knowledge, attitude control) should be estimated and then simulated with the AutoNav simulation. The other main system that affects the AutoNav ability is the Propulsion system. Errors in the thruster's timing of thrust pulses will affect the efficacy of PWM control for maneuvering. These errors should be simulated as well.

V. Conclusion

The Auto-Navigation System presents a viable solution for automated proximity operations about an uncooperative spacecraft using passive imagers and continuous thrust for small spacecraft. The use of image processing algorithms to estimate range, in addition to spherical angles, provides a valid solution

for relative orbit determination necessary for GN&C. The use of an LQR controller for continuous-thrust maneuvering was shown to effectively control the spacecraft to a desired location. Closed loop simulations utilizing the IPAs and LQR control demonstrated the ability to successfully maneuver to a desired location and stay within mission profile bounds. Future work will develop the operational capabilities of the AutoNav system and increase the fidelity of testing modules to improve the accuracy of simulation and tests.

Acknowledgements

The author would like to thank his advisor, David Spencer, for the time and dedication he has given in overseeing the research. Additionally, he would like to thank Dr. Frank Chavez, Dr. Alan Lovell, and Dr. Josue Muñoz from the Air Force Research Laboratory for serving as technical advisors for the project. Finally, he would like to thank the other members of the AutoNav team for their work and inputs.

References

[1] NASA Orbital Debris Program Office, "Orbital Debris Quarterly News." January 2012. <http://orbitaldebris.jsc.nasa.gov/newsletter/pdfs/ODQNv16i1.pdf> Accessed: April 2, 2012.

[2] Holzinger, M.J. and Scheeres, D.J. "Applied Reachability for Space Situational Awareness and Safety in Spacecraft Proximity Operations." *AIAA Guidance, Navigation, and Control Conference*. August 10-13, 2009. Chicago, IL.

[3] NASA Marshall Space Flight Center, "DART Demonstrator to Test Future Autonomous Rendezvous Technologies in Orbit." *NASA Facts*. September 2004. <http://www.nasa.gov/centers/marshall/pdf/172361main_dart.pdf> Accessed: April 2, 2012.

[4] Howard, R.T. and Bryan, T.C. "DART AVGS Performance." *NASA Marshall Space Flight Center Technical Report*. April 9, 2007. < http://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/20070031729_2007031265.pdf> Accessed: April 2, 2012.

[5] Lawler, Andrew. "NASA DART Misses Bull's-Eye." *Science.* April 22, 2005. P. 479.

[6] Davis, T.M. and Melanson, D. "XSS-10 Micro-Satellite Flight Demonstration Program Results." *Proceedings of SPIE Conference on Spacecraft Platforms and Infrastructure*, SPIE Vol. 5419.

[7] Covault, C. "USAF Technology Satellite Plays Tag with GPS Delta." *Aviation Week & Space Technology*, Vol. 158 Issue 5, p. 39. February 3, 2003.

[8] Dornheim, M.A. "Rendezvous Trials." *Aviation Week & Space Technology*. Vol. 162 Issue 16, pp. 35-36. April 18, 2005.

[9] Allen, A.C.M., Langley, C., *et al.* "Rendezvous Lidar Sensor System for Terminal Rendezvous, Capture, and Berthing to the International Space Station." *Proceedings of SPIE Conference on Sensors and Systems for Space Applications II*, SPIE Vol. 6958.

[10] Coppinger, Rob. "USAF Releases Images from Orbiting XSS-11." *Flight International*. Nov 15-Nov 21, 2005, p. 41.

[11] Weismuller, T. and Leinz, M. "GN&C Technology Demonstrated by the Orbital Express Autonomous Rendezvous and Capture Sensor System." *Proceedings from the 29th Annual AAS Guidance and Control Conference*. Breckenridge, CO. February 4-8, 2006.

[12] Malik, Tariq. "Prototype Satellites Demonstrate In-Orbit Refueling." *Space.com.* April 4, 2007. <http://www.space.com/3644-prototype-satellites-demonstrate-orbit-refueling.html> Accessed: April 2, 2012.

[13] Clark, Stephen. "Satellite In-Space Servicing Demo Mission a Success." *Spaceflight Now*. July 23, 2007. <http://spaceflightnow.com/news/n0707/23oe/> Accessed: April 2, 2012.

[14] Kelso, T.S. "Supplemental Two-Line Element Sets." *Today from the Center for Space Standards & Innovation.* December 16, 2000. Online: http://celestrak.com/NORAD/elements/supplemental/. Accessed: April 15, 2012.

[15] Patel, H., Lovell, T.A., *et al.* "Relative Navigation for Satellites in Close Proximity Using Angles-Only Observations." *AAS/AIAA Space Flight Mechanics Meeting*, January 29 – February 2, 2012. AAS 12-202.

[16] Bellet, Edouard. "Detection and Localization of a Target on a Thermal Image Using a 'Blobber' Algorithm."

[17] Hill, G.W. "Researches in the Lunar Theory." *American Journal of Mathematics*, Volume 1, 1878, pp. 5-26.

[18] Clohessy, W.H. and R.S. Wiltshire. "Terminal Guidance System for Satellite Rendezvous", *Journal of the Aerospace Sciences*, Vol. 27, No. 9, 1960, pp. 653-658.

[19] Vallado, D. A. *Fundamentals of Astrodynamics and Applications.* Hawthorne, CA: Microcosm Press. 2007.

[20] Lovell, T. A., and Tragesser, S. G., "Guidance for Relative Motion of Low Earth Orbit Spacecraft Based on Relative Orbit Elements," AIAA/AAS Astrodynamics Specialist Conference and Exhibit, Province, RI, AIAA Paper 2004-4988, 2004.

[21] Tapley, B.D., Schutz, B.E., Born, G.H. *Statistical Orbit Determination*. Burlington, MA. Elsevier Inc., 2004.

[22] Wie, B. *Space Vehicle Dynamics and Control.* Reston, VA: AIAA Education Series. 1998.

[23] Bryson, A.E., Jr., and Ho, Y.-C., *Applied Optimal Control*. Washington DC: Hemisphere. 1975.

[24] Bevilacqua, R., Lehmann, M., Romano, M., "Development and experimentation of LQR/APF guidance and control for autonomous proximity maneuvers of multiple spacecraft." *Acta Astronautica*, Vol. 68. 2011.

[25] McCamish, S.B., Romano, M., Yun, X., "Autonomous Distributed Control of Simultaneous Multiple Spacecraft Proximity Maneuvers." *IEEE Transactions on Automation Science and Engineering*, Vol. 7, No. 3. July 2010.

[26] McCamish, S. B. and Romano, M. "Simulations of Relative Multiple Spacecraft Dynamics and Control by MATLAB-Simulink and Satellite Tool Kit," in Proc. AIAA 2007 Modeling and Simulation Technol. Conf., Hilton Head, SC, Aug. 2007.

[27] Hablani, H.R., Tapper, M.L., and Dana-Bashian, D.J. "Guidance and Relative Navigation for Autonomous Rendezvous in a Circular Orbit," *Journal of Guidance, Control, and Dynamics*, Vol. 25, No. 3, May-June 2002.

[28] Pearson, D.J., "The Glideslope Approach," *Advances in the Astronautical Sciences*, Vol. 69, American Astronautical Society, San Diego, CA, 1989, pp. 109-123.

Nomenclature

Acronyms and Abbreviations

3U	3 Unit CubeSat
ADCS	Attitude Determination and Control Subsystem
AFRL	Air Force Research Laboratory
APF	Artificial Potential Function
AutoNav	Automated Navigation System
BFF	Body-Fixed Frame
Blob	Binary Large Object
COB	Center of Brightness
CW	Clohessy-Wiltshire
EKF	Extended Kalman Filter
FOV	Field of View
FMC	Forced Motion Circumnavigation
FPF	Focal Plane Frame
GN&C	Guidance, Navigation & Control
GPS	Global Positioning System
IPA	Image Processing Algorithm
LOS	Line of Sight
LQR	Linear Quadratic Regulator
LTI	Linear Time-Invariant
LVLH	Local Vertical, Local Horizontal
NMC	Natural Motion Circumnavigation
OD	Orbit Determination
RMS	Root Mean Square
SSA	Space Situational Awareness
STM	State Transition Matrix

Symbols

A,B,C,D	State Space Matrices
A	Projected area of target spacecraft
A_o	Reference area of target spacecraft
a_e	Semi-major axis of relative motion ellipse
F_t	Maximum thrust for thruster
f	Focal Length of Imager Lens
f_x , f_y , f_z	Force in x,y,z direction in RSW frame
$\boldsymbol{G}(\underline{X})$	Measurement model function
$\boldsymbol{H_i}$	Mapping matrix for sample i
$\widetilde{\boldsymbol{H}}_{\boldsymbol{i}}$	Partial derivative matrix for sample i
J	Quadratic cost function
$\boldsymbol{K_{LQR}}$	LQR Gain Matrix
L	Accumulation of normal equations matrix
$\underline{M}$	Accumulation of normal equations vector
m_s	Mass of spacecraft
$\boldsymbol{N}$	LQR state-control combination gain matrix
N_{blob}	Number of pixels in blob
n	Mean Orbital Motion of target spacecraft
$\boldsymbol{P}$	Covariance matrix of estimate
p	Pixel Pitch
$\boldsymbol{Q}$	LQR state gain matrix
$\boldsymbol{R}$	LQR control effort gain matrix
$\hat{R}, \hat{S}, \hat{W}$	Unit vectors for RSW/LVLH frame
$\boldsymbol{R_i}$	Rotation Matrix
r_g	Range to target
r_{init}	Initial range to target
r_m	Maximum allowable range
t	Time
$\underline{u}(t)$	Control effort
u_{max}	Maximum control effort
v_m	Maximum allowable relative velocity
$\boldsymbol{W}_i$	Measurement weighting matrix for sample i
$\underline{X}(t)$	Relative state (position, velocity) of chaser
$\underline{X}_0^*$	Updated state estimate
$\underline{X}_e$	Tracking error for desired trajectory
x, y, z	Positional coordinates in RSW frame
x_d	X Location of relative motion ellipse
y_d	Y Location of relative motion ellipse
z_{max}	Amplitude of harmonic z motion
α_{Q_i}	Position gain for state variable i
β	Angle between perigee and chaser in relative motion ellipse
β_{r_i}	Control gain for thruster i
γ	Angle between orbital planes of target and chaser spacecraft
ΔV	Change in Velocity
$\delta\underline{X}$	Differential correction to estimated state
$\delta\underline{y}_i$	Residual for sample i
θ	Spherical coordinate (elevation) to target
ρ	Range of target
σ_x	Standard deviation of x
σ_{xy}	Covariance of x,y
$\boldsymbol{\Phi(t)}$	State transition matrix at t
ϕ	Spherical coordinate (azimuth) to CubeSat
$\boldsymbol{\Psi(t)}$	Particular solution transition matrix