COLLISION AVOIDANCE MANEUVER PLANNING WITH ROBUST OPTIMIZATION

Joseph B. Mueller[1] and Robin Larsson[2]
[1]*Princeton Satellite Systems, 33 Witherspoon St., Princeton NJ, U.S.A.*
[2]*Swedish Space Corporation, Stockholm, Sweden*

Abstract

In recent years, a growing number of space missions have emerged which are utilizing distributed systems of satellites. This is accompanied by a rising level of interest in both the scientific and defense communities to develop mature systems and software for autonomous rendezvous and formation flying. An underlying requirement for these types of missions is the need to ensure safe, collision-free operations.

The PRISMA mission, to be launched in 2009, will demonstrate Guidance, Navigation, and Control strategies for advanced autonomous formation flying between two spacecraft. The Swedish Space Corporation (SSC) is the prime contractor for the project which is funded by the Swedish National Space Board (SNSB). The safe guidance mode, based upon algorithms developed by Princeton Satellite Systems (PSS) under a CRADA, will be an integral part of the PRISMA mission. Its role is to plan a collision avoidance maneuver if the relative distance is too small, and to continually plan relative station-keeping maneuvers to keep the spacecraft on a safe relative trajectory.

One important requirement of the safe guidance algorithms for PRISMA is that they guarantee reliable plans immediately, which led to the choice of completely deterministic methods. In this paper, we consider alternative numerical methods that are appropriate for the online computation of collision avoidance maneuvers. In particular, we discuss how to pose the original non-convex problem as a linear programming (LP) problem, using a combination of well-defined convex constraints. We go on to show how navigation uncertainty can be included in the LP formulation, enabling the efficient solution of robust, fuel optimal maneuvers.

Key words: formation flying; collision avoidance; robust; optimization; maneuver planning; PRISMA.

1. INTRODUCTION

There has been an increasing level of interest recently in the satellite industry to develop reliable systems and software for autonomous rendezvous and formation flying. Examples include the TICS, F6 and Orbital Express programs at DARPA, the DART, MMS, SIRA, MAXIM and TPF missions at NASA, the Proba-3, Darwin and Cluster missions at ESA and commercial missions like OLEV. The applications range from automated rendezvous for equipment and fuel delivery, to long-duration precise formation flying of distributed sensors, which could enable the detection of distant Earth-like planets. In general, each mission utilizes a distributed system of space vehicles to accomplish its tasks. A common thread for all of these missions is the importance of ensuring safe, collision-free operations.

The PRISMA mission, to be launched in 2009, will demonstrate Guidance, Navigation, and Control strategies for advanced autonomous formation flying [1]. The Swedish Space Corporation (SSC) is the prime contractor for the project which is funded by the Swedish National Space Board (SNSB). The mission consists of two spacecraft: MAIN and TARGET, shown in Fig. 1. The MAIN satellite has full orbit control capability while TARGET is attitude controlled only. The safe guidance mode, based upon algorithms developed by Princeton Satellite Systems (PSS) under a CRADA, will be an integral part of the PRISMA mission. Its role is to plan a collision avoidance maneuver if the relative distance is too small, and to continually plan relative station-keeping maneuvers to keep the spacecraft on a safe relative trajectory.

One of the top requirements of the safe guidance algorithms for PRISMA is that they guarantee reliable plans immediately, which led to the choice of completely deterministic methods. Because the mission involves a circular orbit with only two spacecraft, it was possible to derive closed-form analytic solutions for avoidance maneuvers based on the Clohessy-Wiltshire equations of relative motion. However, for general orbits where eccentricity cannot be ignored, the geometry of the relative motion becomes more complicated. While the general strategy remains practical, the derivation and use of closed-form solutions may become impractical. In addition, the extension to multiple spacecraft adds a level of complexity that cannot easily be accommodated with purely analytic solutions. It is therefore meaningful to consider alternative numerical methods for planning collision avoidance maneuvers.

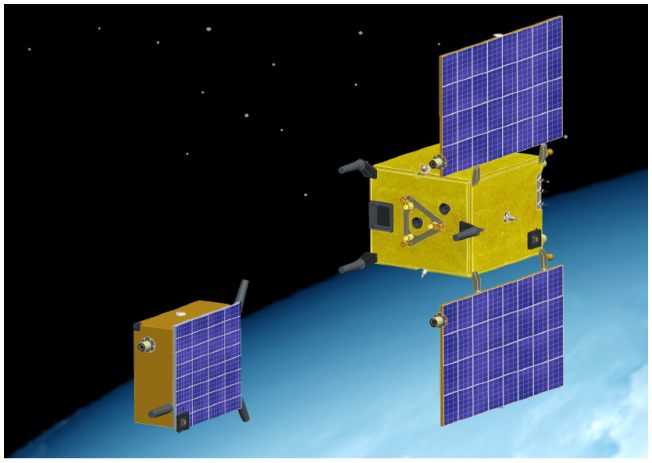

Figure 1. PRISMA Satellites. TARGET (left) and MAIN.

The topics of optimal maneuver planning and collision avoidance for close-orbiting spacecraft have been studied by numerous researchers in recent years [2, 3, 4, 5, 6, 7, 8, 5, 9, 10, 11, 12, 13, 14, 15, 16]. Recent papers by Princeton Satellite Systems [17, 18, 19] have discussed the architectural considerations for large scale systems of formation flying satellites. Included in these papers are discussions of relative orbit dynamics in circular and eccentric orbits, the use of convenient parameter sets to describe desired periodic relative motion, and methods for planning reconfiguration maneuvers by formulating the problem as a linear program (LP).

Reference [5] outlines a method for optimal spacecraft maneuver planning with collision avoidance constraints, where the optimal control problem is posed as a linear program, and logical constraints for collision avoidance and thruster plume avoidance are included with binary variables. The subsequent problem becomes a mixed integer linear program (MILP), which can be solved relatively efficiently. In reference [12], the work is extended by considering bounded disturbances, and proving that feasible solutions that are robust to the disturbances can be achieved in finite time.

In this paper, we consider the collision avoidance strategy for the PRISMA mission. We first outline the closed-form solutions derived for the safe orbit guidance software. This approach involves two modes: 1) separation guidance to maneuver out of an ellipsoidal avoidance region, and 2) nominal guidance to achieve and maintain a passively safe relative trajectory. We then develop an alternative numerical method that follows the original avoidance strategy. This method poses the optimization problem as a linear program (LP), approximating the original non-convex constraints in a novel fashion by using a combination of well-defined, time-varying convex constraints. The LP solution method is then extended by accounting for navigation uncertainty in the state estimate used for planning. This leads to a pure LP formulation for robust collision avoidance maneuver planning, without the need for integer constraints. The new LP-based method is a practical option for the online computation of collision avoidance maneuvers and is easily extensible to include eccentric dynamics, relative disturbance models, and time-varying thruster constraints.

2. PRISMA SAFE GUIDANCE APPROACH

The objective of PRISMA is to develop and qualify new technology necessary for future space missions. This applies to both hardware qualification as well as several sets of GNC experiments for formation flying. The mission consists of two spacecraft: MAIN and TARGET. The MAIN satellite is 3-axis stabilized and is equipped for full 3D delta-V manoeuvrability independent of its attitude. The TARGET satellite has a simplified, still 3-axis stabilizing, magnetic attitude control system [20] and no orbit maneuver capability. An illustration of the two spacecraft in orbit is shown in Fig. 1.

In this section we provide an overview of the safe guidance method. The approach described is based on algorithms developed jointly by PSS and SSC. The actual technique used by PRISMA is similar but is beyond the scope of this paper. For more information about the overall PRISMA mission and spacecraft design, see [21, 20]. In addition, reference [1] offers further material about the safe orbit guidance mode.

The purpose of the safe guidance mode is to provide a robust method of achieving safe relative motion between the MAIN and TARGET. The mission consists of a variety of experiments, and the GNC software design provides a separate mode for each experiment. The safe mode can be entered from every other mode, either automatically or by command. In

addition, safe mode is automatically entered upon starting up the GNC system. Automatic transitions from other other modes can occur as a result of requests from the previous mode, or from anomaly detection, such as the detection of a potential collision.

The safe guidance mode consists of two separate techniques: 1) *Separation Guidance*, and 2) *Nominal Guidance*. If the estimated relative position of the MAIN spacecraft is within an ellipsoidal avoidance region centered around the TARGET, then a single-burn separation maneuver is performed. This is referred to as *separation guidance*. The separation maneuver is guaranteed to monotonically increase the separation distance and exit the avoidance region within a prescribed time. Otherwise, if the sensed relative position is outside of the avoidance region, then a maneuver is planned to achieve a desired safe relative trajectory. This is referred to as nominal guidance. The safe relative trajectory is one that cannot intersect the ellipsoidal avoidance region, even in the presence of uncorrected along-track drift.

An illustration of the avoidance region is shown in Fig. 2. The region is a $2 \times 1 \times 1$ ellipsoid in the x, y, z axes of the relative coordinate frame, where x is oriented in the along-track direction, y in the cross-track direction, and z in the radial or nadir direction. This is a local-vertical, local-horizontal (LVLH) frame, sometimes also referred to as the SLO frame. The semi-major axis d is a design parameter that may be changed during the course of the mission. For this paper we assume $d = 60$ meters. An additional margin of $m = 30$ meters beyond the avoidance region defines the nominal boundary. This provides a buffer around the avoidance region. The nominal guidance algorithms attempt to keep the MAIN outside of the nominal boundary, so that small errors do not result in the unwanted reentry of the avoidance region.

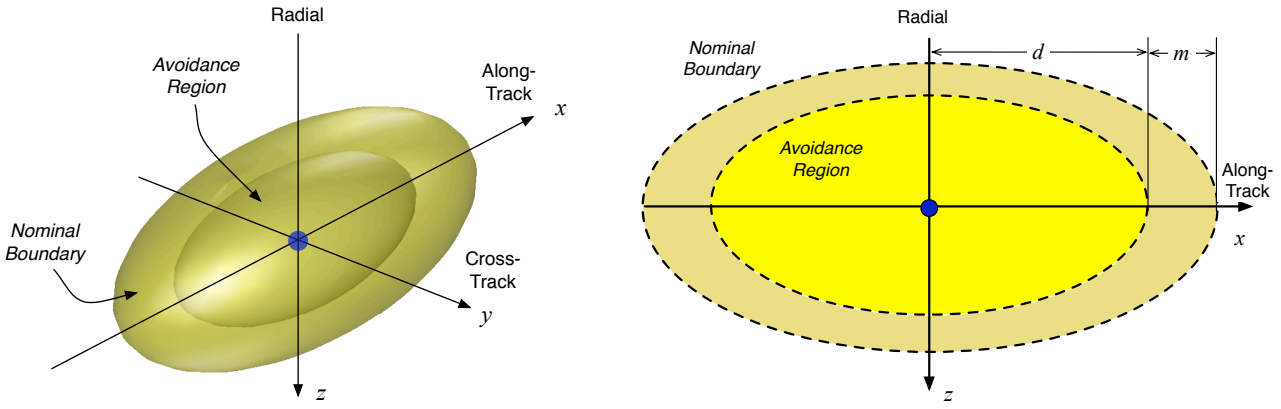

Figure 2. 3D perspective of the ellipsoidal avoidance region (left). 2D orbital-plane view of the region (right).

Separation Guidance

The separation guidance law is designed to compute a single immediate delta-v that achieves the following objectives:

1. Exit the avoidance region within a specified time, T_{exit}
2. Ensure increasing separation distance while inside the region
3. Achieve a trajectory that is "passively safe".

The resulting trajectory is considered passively safe in that it will not re-enter the avoidance region. The time period T_{exit} is a tunable parameter of the algorithm. Clearly, the choice of smaller exit times leads to larger separation delta-v's. In addition to meeting the above objectives, the algorithm was designed to be robust to expected levels of uncertainty in the initial relative position and velocity estimate used to plan the maneuver. Its performance has been verified in several Monte Carlo simulations with random initial conditions inside the avoidance region.

Let $\boldsymbol{r_i}$ and $\boldsymbol{v_i}$ be the initial relative position and velocity vectors in the orbital plane, respectively. The algorithm computes a desired velocity in the orbital plane, $\boldsymbol{v_i^*} = [v_x^*, v_z^*]^\top$. The delta-v is computed at the end of the algorithm as the difference between the desired and current in-plane velocities, with no delta-v applied in the cross-track direction. The desired in-plane velocity is first computed with a nominal magnitude V and direction $\hat{\boldsymbol{u}}$, based solely on the position vector. The

direction is aligned with the position vector, while the magnitude is proportional to the distance from the nominal boundary and inversely proportional to the separation time. Let $\boldsymbol{r_i} = [x, z]^\top$ and $\boldsymbol{v_i} = [v_x, v_z]^\top$ be the in-plane components of the relative position and velocity output from Hill's equations, and let T_{exit} be the separation time. The desired separation velocity is initially computed as follows:

$$\hat{\boldsymbol{u}} = \frac{\boldsymbol{r_i}}{|\boldsymbol{r_i}|} \tag{1}$$

$$V = \frac{d + m - \sqrt{x^2 + 4z^2}}{T_{\text{exit}}} \tag{2}$$

$$\boldsymbol{v_i^*} = V\hat{\boldsymbol{u}} \tag{3}$$

$$\tag{4}$$

The delta-v is defined as:

$$\boldsymbol{\Delta V} = \left\{ \begin{array}{cc} \boldsymbol{v_i^*} - \boldsymbol{v_i} & \boldsymbol{r_i} \cdot (\boldsymbol{v_i^*} - \boldsymbol{v_i}) > 0 \\ 0 & \text{else} \end{array} \right\} \tag{5}$$

The dot product $\boldsymbol{r_i} \cdot \boldsymbol{\Delta V}$ is negative if the MAIN is already flying away from the TARGET at a higher velocity than $\boldsymbol{v_i^*}$. In this case, no delta-v is necessary.

We next compute the properties of the along-track motion that will result from the new initial state, $[\boldsymbol{r_i}, \boldsymbol{v_i^*}]^\top$. This includes the initial center of motion x_c, the drift per orbit D, and the amplitude of oscillation A. These parameters are derived from the Clohessy-Wiltshire (CW) equations, shown below for ease of reference.

$$x(t) = (4v_x/n - 6z)\sin(nt) - 2v_z/n\cos(nt) - (3v_x - 6zn)\,t + x + 2v_z/n \tag{6}$$

$$y(t) = y\cos(nt) + v_y/n\sin(nt) \tag{7}$$

$$z(t) = v_z/n\sin(nt) + (2v_x/n - 3z)\cos(nt) - (2v_x/n - 4z) \tag{8}$$

$$v_x(t) = 4v_x - 6zn\cos(nt) + 2 * v_z\sin(nt) - (3v_x - 6zn) \tag{9}$$

$$v_y(t) = -yn\sin(nt) + v_y\cos(nt) \tag{10}$$

$$v_z(t) = v_z\cos(nt) - (2v_x/n - 3z)n\sin(nt) \tag{11}$$

where n is the mean orbit rate, and $[x, y, z]^T$ and $[\dot{x}, \dot{y}, \dot{z}]^T$ are the initial relative position and velocity vectors, respectively.

The along-track motion is described in the $x(t)$ equation. The sine and cosine terms give the amplitude, the coefficient on time gives the drift, and the constant term is the center of motion. The parameters are defined as follows:

$$x_c = x + 2v_z/n \tag{12}$$

$$D = -(2\pi/n) \times (3v_x - 6zn) \tag{13}$$

$$A = 2\sqrt{(2v_x/n - 3z)^2 + (v_z/n)^2} \tag{14}$$

These parameters are illustrated in Fig. 3. The diagram on the left shows an example where the drift is positive, but the initial center of motion is negative. The result is that the trajectory must pass through $x = 0$ at least once. In this example, it passes through 3 different times. Each time the along-track position approaches zero, it could re-enter the avoidance region. We therefore want to choose a relative velocity that produces an along-track drift per orbit, D, which has the same sign as the center of motion, x_c.

The diagram on the right of Fig. 3 illustrates an example where the drift and along-track center are both negative. This diagram also shows the x boundary of the avoidance region in red ($\pm d$). The trajectory starts inside the avoidance region, as will be the case when the separation guidance algorithm is used. In this case, even though D and x_c have the same sign, the trajectory is observed to re-enter the avoidance region after 1 orbit. It can be easily seen that further increasing the size of the drift (making it more negative) will prevent the re-entry from happening.

The algorithm checks for two different cases which can result in re-entry of the avoidance region (details of the derivation for these cases are omitted for brevity). One case is when the magnitude of the drift per orbit is too small, when $D < 2d$. The other case occurs when D and x_c are of opposite sign, and the ratio of the amplitude to drift larger than 0.5. To summarize, it is possible that the initial trajectory could re-enter the avoidance region if either of the following conditions are true:

$$|D| < 2d \tag{15}$$

or

$$\frac{D}{|D|} \neq \frac{x_c}{|x_c|}, \qquad \left|\frac{A}{D}\right| > \frac{1}{2} \tag{16}$$

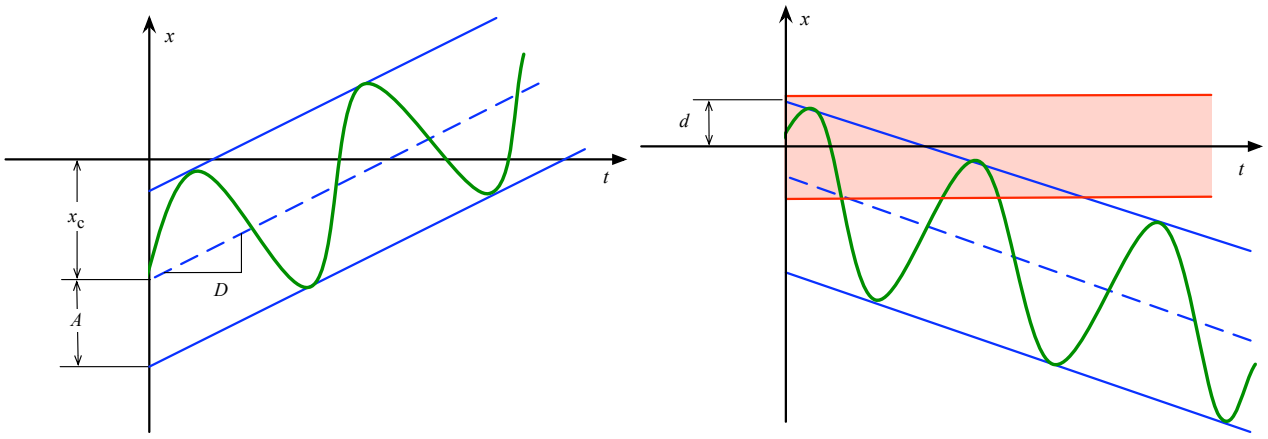

Figure 3. Illustration of along track motion parameters. On the left, the drift rate and initial center of motion have opposite sign. On the right, they have the same sign.

If either condition holds, then the x-component of velocity is recomputed such that $D = \pm 2df_s$, where the sign is selected to match the sign of x_c, and $f_s > 1$ is a safety factor used to account for state uncertainty. The solution for v_x is:

$$v_x = -\frac{n}{3}\left(\frac{df_s x_c}{\pi|x_c|} - 6z\right) \tag{17}$$

A Monte Carlo simulation illustrates the effectiveness of the approach. We consider 2000 random initial conditions inside the avoidance region with $d = 60$ meters, and include random errors in the initial position and velocity (10 cm and 10 mm/s, respectively, 1-sigma). The separation time is set to 1500 seconds, approximately a quarter of an orbit. Only one delta-v is computed and applied immediately, with no subsequent control, and the CW equations are used to propagate the relative motion. Repeated evaluations of this Monte Carlo simulation were conducted with increasing safety factors, until a value of $f_s = 6$ was found to produce zero re-entries of the avoidance region. A plot of a sample trajectory is shown in Fig. 4, along with a scatter plot of the drift per orbit achieved for all 2000 cases. The average delta-v was about 0.1 m/s, with a peak of 0.23 m/s.

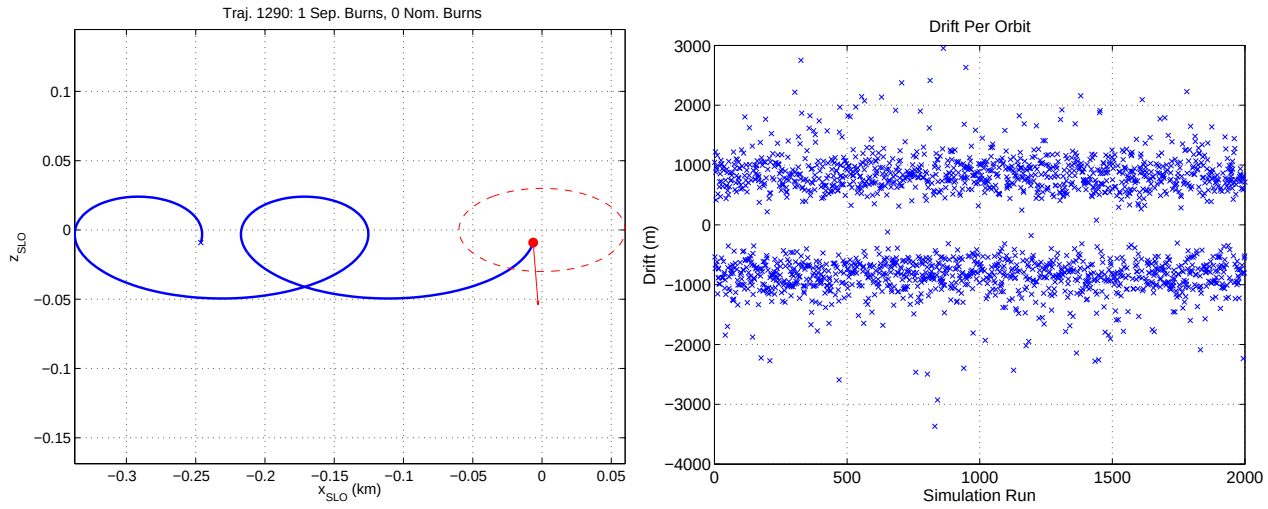

Figure 4. Example in-plane trajectory of a separation maneuver (right), and a scatter plot of the resulting drift per orbit.

In general, larger uncertainty in the initial state requires a larger safety factor. This has the effect of increasing the drift per orbit, which will require more delta-v to correct after the avoidance region is exited. In addition, if a subsequent delta-v cannot be performed for some reason, then the consequence of the higher safety factor is that the MAIN will drift away from the TARGET faster. This presents an interesting tradeoff within the larger context of mission safety and operations.

Nominal Guidance

Once the MAIN is outside of the avoidance region, the nominal guidance algorithm takes over. This algorithm is designed to keep the MAIN spacecraft outside of the avoidance region, and to also keep the two spacecraft within communication range. The approach is to find a passive relative trajectory that will remain out of the avoidance region, even in the presence of along-track drift. Some small amount of along-track drift will always be present after a maneuver due to unavoidable thruster inaccuracies and navigation error. Therefore, it is important to find safe relative trajectories that are independent of the along-track position, as this enables the safe guidance law to consume much less fuel over time.

The solution is a trajectory known as a "safe ellipse". An example of a drifting safe ellipse is shown in Fig. 5. The oscillation in the radial (z) and cross-track (y) directions are large enough and phased properly so that the motion in the YZ plane surrounds the avoidance region. In this case, the y and z motion is aligned to form a circular path, but in general the YZ plane path follows an ellipse. In addition, the trajectory is drifting in the along-track direction. If there were zero along-track drift, then the orbital-plane (XZ plane) motion would follow a 2×1 ellipse.

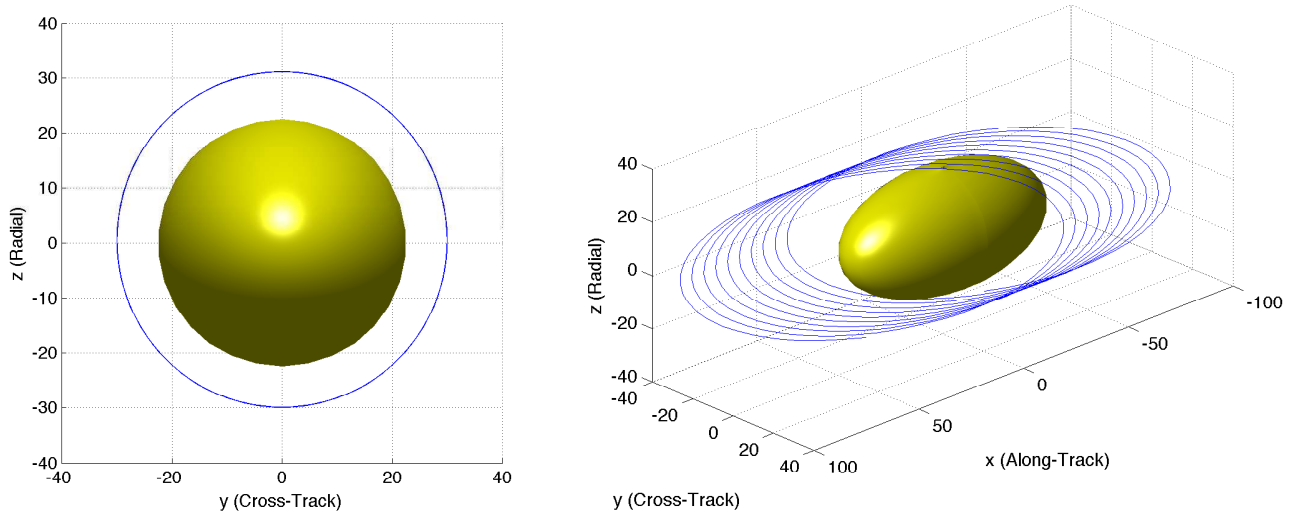

Figure 5. Example of a drifting safe ellipse. Projection on YZ plane (left) and 3D perspective (right).

The task of the nominal guidance algorithm is to plan a maneuver to achieve a safe ellipse. Under normal conditions, the objective is to plan this maneuver so that the total delta-v is minimized. In general, this requires the application of 1-2 delta-v's at specific times over the next half orbit period. Given the current relative state the optimal maneuver time and maneuver is calculated as in [1], here given in brief for clarity. We start by rewriting (7) and (8) using (13) as follows:

$$\tilde{z}(\theta) = z(\theta) - \frac{D}{3\pi} \tag{18}$$

$$\tilde{z}(\theta) = \tilde{z}\cos(\theta) + \frac{v_z}{n}\sin(\theta) = r_z\cos(\theta - \Phi_z) \tag{19}$$

$$y(\theta) = y\cos(\theta) + \frac{v_y}{n}\sin(\theta) = r_y\cos(\theta - \Phi_y) \tag{20}$$

where $\Delta\Phi = \Phi_y - \Phi_z$. The optimal maneuver time is given by the location of the min and max of the distance to the center of motion; these extreme points are given by:

$$\hat{\theta} = \frac{1}{2}\tan^{-1}\left(\frac{r_y^2\sin(2\Delta\Phi)}{r_y^2\cos(2\Delta\Phi) + r_z^2}\right) + \Phi_z \tag{21}$$

The next extreme point that the spacecraft will pass through ($\boldsymbol{p_1}$) is given by

$$\theta_c = \hat{\theta} \pm i\pi/2, \quad i = 1, 2, \ldots \quad \text{such that} \quad 0 <= \theta_c < \pi/2$$

Inserting θ_c in (19) and (20) gives $\boldsymbol{p_1} = [y(\theta_c), \tilde{z}(\theta_c)]^T$ and $\boldsymbol{p_2} = [y(\theta_c + \pi/2), \tilde{z}(\theta_c + \pi/2)]^T$. If the distance to $\boldsymbol{p_2}$ is smaller than d a maneuver, $\boldsymbol{\Delta V_1}$, is executed at $\boldsymbol{p_1}$ such that:

$$\hat{\boldsymbol{p}}_{\boldsymbol{2}} = (d+m)\frac{\boldsymbol{p_2}}{||\boldsymbol{p_2}||} \tag{22}$$

where $\hat{\boldsymbol{p}}_{\boldsymbol{2}}$ is the new location for the extreme point $\boldsymbol{p_2}$. In (22) the avoidance region is also extended in the z-direction with the same length as the current center of radial motion. The delta-v is given directly from the required position change at $\boldsymbol{p_2}$:

$$\boldsymbol{\Delta V_1} = n(\hat{\boldsymbol{p}}_{\boldsymbol{2}} - \boldsymbol{p_2}). \tag{23}$$

Note that $\boldsymbol{\Delta V_1}$ has components only in y and z direction, the x component is used to control the drift in along-track to stay within a specified bound. An example of a maneuver is visualized in the left part of Fig. 6. The method described is used to extend both the semi-minor and the semi-major axis of the YZ movement if necessary.

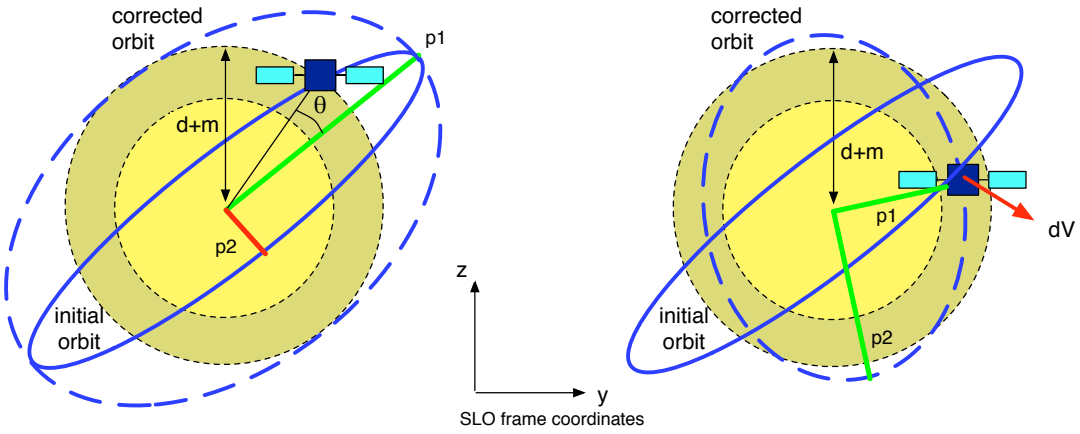

Figure 6. Example of a relative trajectory in the YZ plane that intersects the avoidance region. On the left, the spacecraft is allowed to wait until the optimal time to apply the delta-v. On the right, the spacecraft must maneuver immediately. The corrected orbits are shown in dashed lines.

However, there are a number of possible scenarios where the need to maneuver quickly outweighs the desire to minimize fuel. An example of such a case is if the navigation solution accuracy is degrading rapidly due to a sensor failure. In such cases, the algorithm computes a single delta-v that is applied immediately. In order to achieve a safe ellipse with a single delta-v, the current position of the MAIN must be outside the YZ projection of the avoidance region. If this is the case the method described above is used but with $\theta_c = 0$, so that the current location becomes an extreme point. This is illustrated on the right side of Fig. 6, where the spacecraft must maneuver immediately. Here we compute a safe ellipse such that the current position becomes the semi-minor axis. In other words, the trajectory will get no closer to the avoidance region than the current distance. For more details about the computation of maneuvers for the safe ellipse, see [1].

If the current position of the MAIN is inside the YZ projection of the avoidance region, the only option is to plan a maneuver in the orbital plane (XZ plane). The objective in this scenario is to achieve a non-drifting 2×1 relative ellipse that does not intersect the nominal boundary. In addition, we enforce the new trajectory to have a sufficiently high radial (z) oscillation so that a subsequent maneuver to a desired safe ellipse can be performed if possible. An illustration is shown in Fig. 7. The distance from the along-track axis is too small to enable a single-burn transition to a desired safe ellipse that surrounds the YZ projection of the avoidance region.

The XZ plane projection of the safe ellipse passes through the current position (x, z), has an along-track center of motion x_0 and a semi-major axis, a_E. The semi-major axis can be expressed in terms of the position and along-track center as follows:

$$a_E = \sqrt{(x - x_0)^2 + 4z^2} \tag{24}$$

The relative velocity required to follow this ellipse is:

$$v_x^* = 2zn \tag{25}$$

$$v_z^* = -\frac{n}{2}(x - x_0) \tag{26}$$

The first geometric constraint requires the radial oscillation to exceed the radial size of the nominal boundary. This is equivalent to:

$$a_E > d + m \tag{27}$$

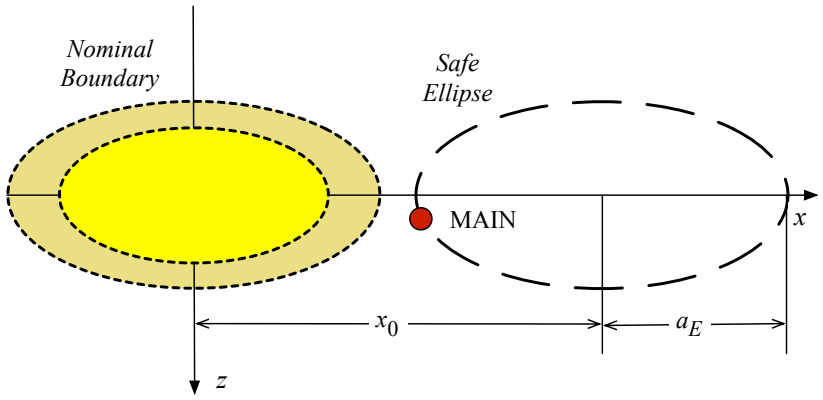

Figure 7. Illustration of the XZ-plane projection of a safe ellipse.

The second geometric constraint is that the ellipse not intersect the nominal boundary, which is equivalent to:

$$a_E < |x_0| - (d+m), \quad \text{for } |x_0| > (d+m) \tag{28}$$
$$a_E > |x_0| + (d+m), \quad \text{for } |x_0| < (d+m) \tag{29}$$

The procedure is to choose x_0, and thereby a_E via (24), in order to minimize the required change in the radial velocity, while meeting the above geometric constraints on the ellipse. If the current position satisfies these constraints, then no delta-v is required. Otherwise, two candidate solutions are obtained:

$$x_0^* = \frac{4z^2 + x^2 - (d+m)^2}{x \pm (d+m)} \tag{30}$$

where the two possible signs in the denominator represent the two choices. These solutions correspond to two feasible ellipses that touch the border of the nominal boundary. The trajectory requiring the smaller delta-v is chosen.

3. COLLISION AVOIDANCE USING LINEAR PROGRAMMING METHODS

In this section we describe a method for formulating the collision avoidance maneuver planning problem as a sequence of linear programs. The objective in the design of this approach is to closely approximate the original PRISMA collision avoidance strategy, described in Section 2. The main advantage of this approach over the closed-form solutions is its flexibility. For example, it can account for changing constraints on the time-window during which the thrusters may be used. It can also be easily extended to include time-varying dynamics of eccentric orbits and relative disturbance models.

Similar to the original method, the overall strategy for the LP-based method consists of two separate components: 1) Separation guidance, and 2) Nominal guidance. Each method is described in the subsections below. First, however, we discuss how to express the relative orbit dynamics in a form suitable for linear programming.

Discretized Relative Dynamics

The linear time-varying (LTV) dynamic equations of a satellite's relative motion in the LVLH frame are given as follows:

$$\frac{d}{dt}\begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{z}\end{bmatrix} = -2\begin{bmatrix}0 & 0 & -\dot{\nu}\\ 0 & 0 & 0\\ \dot{\nu} & 0 & 0\end{bmatrix}\begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{z}\end{bmatrix} - \begin{bmatrix}-\dot{\nu}^2 & 0 & 0\\ 0 & 0 & 0\\ 0 & 0 & -\dot{\nu}^2\end{bmatrix}\begin{bmatrix}x\\ y\\ z\end{bmatrix} - \begin{bmatrix}0 & -\ddot{\nu} & 0\\ \ddot{\nu} & 0 & 0\\ 0 & 0 & 0\end{bmatrix}\begin{bmatrix}x\\ y\\ z\end{bmatrix}$$
$$+ n^2\left(\frac{1+e\cos\nu}{1-e^2}\right)\times\begin{bmatrix}-x\\ -y\\ 2z\end{bmatrix} + \begin{bmatrix}a_x\\ a_y\\ a_z\end{bmatrix} \tag{31}$$

where ν is the true anomaly, e is the eccentricity, n is the mean orbit rate, and a is the applied acceleration. Let the state vector of relative position and velocity be $\boldsymbol{x} = [x, y, z, \dot{x}, \dot{y}, \dot{z}]^T$, and let the control vector of accelerations be

$\boldsymbol{u} = [a_x, a_y, a_z]^T$. The system system is then written in state-space form as:

$$\dot{\boldsymbol{x}} = A(\nu, \dot{\nu})\boldsymbol{x} + B(\nu, \dot{\nu})\boldsymbol{u} \tag{32}$$

where $A \in \mathbb{R}^{6\times 6}$ and $B \in \mathbb{R}^{6\times 3}$. In the case of zero eccentricity, the $\dot{\nu}$ terms reduce to the mean orbit rate n, and the $\ddot{\nu}$ terms vanish. The dynamics simplify to an undamped, linear time-invariant (LTI) system, with a natural frequency equal to the mean orbit rate. The unforced response to an initial state is given by the familiar Clohessy-Wiltshire or Hill's equations (see Section 2, (6) through (11)).

For the LTI system (circular orbits), the A matrix is constant, whereas for the more general LTV system, it varies with the true anomaly of the orbit. In this paper, we consider only circular orbits, but the discretization approach is presented here for the general case. The continuous time system can be discretized with a zero-order hold at an appropriate sampling period Δt_k over N samples to become:

$$\boldsymbol{x}_{k+1} = A_k \boldsymbol{x}_{\boldsymbol{k}} + B_k \boldsymbol{u}_{\boldsymbol{k}} \tag{33}$$

where k is an index increasing with time, A_k varies with k for non-circular orbits, and both matrices vary with Δt_k. Applying the equation recursively from $k = 0$ to $k = N - 1$, we have the following expression for the final state:

$$\boldsymbol{x}_N = \left(A_{N-1}\cdots A_1 A_0\right)\boldsymbol{x}_0 + \left[A_{N-1}\cdots A_1 B_0,\ \ A_{N-1}\cdots A_2 B_1,\ \ \ldots,\ \ A_{N-1}B_{N-2},\ \ B_{N-1}\right]\begin{bmatrix}\boldsymbol{u}_0\\ \boldsymbol{u}_1\\ \vdots\\ \boldsymbol{u}_{N-2}\\ \boldsymbol{u}_{N-1}\end{bmatrix} \tag{34}$$

More generally, the state at time $t(j)$, $j \in (1, N)$, is expressed as:

$$\boldsymbol{x}_j = \left(A_{j-1}\cdots A_1 A_0\right)\boldsymbol{x}_0 + \left[A_{j-1}\cdots A_1 B_0,\ \ A_{j-1}\cdots A_2 B_1 \mid \ldots,\ \ A_{j-1}B_{j-2},\ \ B_{j-1},\ \ \mathbf{0}\right]\begin{bmatrix}\boldsymbol{u}_0\\ \boldsymbol{u}_1\\ \vdots\\ \boldsymbol{u}_{j-2}\\ \boldsymbol{u}_{j-1}\\ \vdots\\ \boldsymbol{u}_{N-1}\end{bmatrix} \tag{35}$$

Let us now write this more compactly as:

$$\boldsymbol{x}_j = H_j \boldsymbol{x}_0 + G_j \tilde{\boldsymbol{u}} \tag{36}$$

where $\tilde{\boldsymbol{u}} \in \mathbb{R}^{3N\times 1}$ is the stacked vector of controls, and $H_j \in \mathbb{R}^{6\times 6}$ and $G_j \in \mathbb{R}^{6\times 3N}$ are given from (35). This formulation allows us to express the relative position and velocity at any future time as a linear function of the original state $\boldsymbol{x}_0$ and the control history $\tilde{\boldsymbol{u}}$.

LP-Based Separation Guidance

We now wish to identify linear constraints on the state vector that capture the objectives of the separation guidance strategy. Recall that the objectives are to: 1) exit the avoidance region within a specified time, T_{exit} 2) ensure increasing separation distance while inside the region, and 3) not re-enter the region.

We first consider the constraint to exit the avoidance region by time T_{exit}. Define a time vector $t = [0, t_1, t_2, \ldots, t_N]$, where $t_N = T_{\text{exit}}$. The goal is to define a linear constraint on the state $\boldsymbol{x}_N$ that ensures the position is outside of the avoidance region. This requires the definition of a tangent line on the ellipse. The most conservative choice is to place the tangent line at the semimajor axis of the ellipsoid, constraining only the along-track position. The constraint may be written as:

$$-\boldsymbol{q_1}^T \boldsymbol{x}_N + (d + m) \quad \leq \quad 0 \tag{37}$$

where $\boldsymbol{q}_1^T = [s_x\ 0\ 0\ 0\ 0\ 0]$ and $s_x = \pm 1$ is the sign of x_0, the initial along-track position. The sign convention ensures that the trajectory does not have to pass through $x = 0$ for safety.

The second requirement is to ensure increasing separation distance until the region is exited. This can be achieved a number of different ways. For example, we could enforce a strictly positive (or strictly negative) along-track velocity over

the entire trajectory. Instead we choose to enforce a lower bound on the absolute value of the along-track position. This can be written compactly as:

$$-\boldsymbol{q_2}^T \boldsymbol{x}_j + s_x x_0 \quad \leq \quad 0 \tag{38}$$

where $\boldsymbol{q}_2^T = [s_x \ 0 \ 0 \ 0 \ 0 \ 0]$ and $j \in [1, N]$. This is somewhat relaxed from the original statement of growing separation distance, but it suffices to ensure collision avoidance. Furthermore, when T_{exit} is less than half of an orbit period, which it is in practice, the optimal solution to meet these requirements naturally causes a growing separation distance anyway.

Finally, in order to ensure the region is not reentered, we can enforce the terminal state to have a small along-track drift away from the region. In addition, the along-track oscillation must be bounded to ensure that the subsequent trajectory does not temporarily re-enter the region.

Let D represent the desired along-track drift per orbit. Refer to (6) to find the coefficient on time, which is the drift in units of meters per second. Dividing D by the orbit period of $T = 2\pi/n$, the relationship becomes:

$$6nz - 3\dot{x} = \frac{Dn}{2\pi} \tag{39}$$

where $z, \dot{x}$ are respectively the radial position and along-track velocity at the final time $t(N)$. Letting $\boldsymbol{q_3}^T = [-6n, 0, 0, -3]$, this is equivalent to:

$$\boldsymbol{q_3}^T \boldsymbol{x}_N = \frac{Dn}{2\pi} \tag{40}$$

Rather than requiring the desired drift rate to be achieve exactly, we can enforce the absolute value of the error to be less than a prescribed tolerance, ϵ_D, as follows:

$$\left|\boldsymbol{q_3}^T \boldsymbol{x}_N - D\right| \leq \epsilon_D \tag{41}$$

This is equivalent to:

$$\begin{aligned} \boldsymbol{q_3}^T \boldsymbol{x}_N - D - \epsilon_D &\leq 0 &(42)\\ -\boldsymbol{q_3}^T \boldsymbol{x}_N + D - \epsilon_D &\leq 0 &(43) \end{aligned}$$

In a similar fashion, the amplitude of the along-track oscillation of the relative trajectory after the maneuver can be bounded. Referring to (6), the coefficients on the sine and cosine terms give the maximum amplitude as:

$$\begin{aligned} \bar{x} &= \sqrt{(4\dot{z}/n)^2 + (4\dot{x}/n - 6z)^2} &(44)\\ &\leq 2|\dot{z}|/n + 2|2\dot{x}/n - 3z| &(45) \end{aligned}$$

We wish to constrain the final states so that the along-track oscillation is bounded from above by some allowable amount, $\bar{x}_{\max}$. Because the final along-track position will be outside the nominal boundary ($|x| > d + m$), it makes sense to choose $\bar{x}_{\max} = m$. This ensures the maximum possible oscillation will still result in $|x| > d$, preventing re-entry of the avoidance region. Separating the absolute values, we get the conservative bounds:

$$\begin{aligned} 2|\dot{z}|/n &\leq m &(46)\\ 2|2\dot{x}/n - 3z| &\leq m &(47) \end{aligned}$$

Letting $\boldsymbol{q_4}^T = [0, 0, 0, 0, 0, 4/n]$ and $\boldsymbol{q_5}^T = [0, 0, -6, 4/n, 0, 0]$, we can write the above constraints as:

$$\begin{aligned} \boldsymbol{q_4}^T \boldsymbol{x}_N - m &\leq 0 &(48)\\ -\boldsymbol{q_4}^T \boldsymbol{x}_N - m &\leq 0 &(49)\\ \boldsymbol{q_5}^T \boldsymbol{x}_N - m &\leq 0 &(50)\\ -\boldsymbol{q_5}^T \boldsymbol{x}_N - m &\leq 0 &(51) \end{aligned}$$

The decision variables in the optimization are the applied accelerations, $\tilde{\boldsymbol{u}}$. In order to conform to the standard form of a linear program, we must constrain the decision variables to lie in the non-negative orthant. This is done by partitioning the control into positive and negative components. Define the new control vector $\tilde{\tilde{\boldsymbol{u}}}$ as follows:

$$\tilde{\tilde{\boldsymbol{u}}} = \begin{bmatrix} \tilde{\boldsymbol{u}}^- \\ \tilde{\boldsymbol{u}}^+ \end{bmatrix} \geq 0 \tag{52}$$

Define $\tilde{G}_j = [-G_j,\ G_j]$ for convenient notation. The relationship between the j^{th} state and the the control vector, given in (35), is now written as:

$$\begin{aligned} \boldsymbol{x}_j &= H_j \boldsymbol{x}_0 - G_j \tilde{\boldsymbol{u}}^- + G_j \tilde{\boldsymbol{u}}^+ \\ &= H_j \boldsymbol{x}_0 + \tilde{G}_j \tilde{\tilde{\boldsymbol{u}}} \end{aligned} \tag{53}$$

where the original control vector is now broken down into the positive component $\tilde{\boldsymbol{u}}^+$ and the negative component $\tilde{\boldsymbol{u}}^-$. This doubles the dimension of the control vector, but allows us to use it in the standard form required by LP solvers.

The maneuver planning problem is to find a control history that meets the above constraints while minimizing the total delta-v. The linear program is formulated as follows:

$$\begin{aligned} \text{Minimize} \quad & ||\tilde{\tilde{\boldsymbol{u}}}||_1 \\ \text{Subject to} \quad & -\boldsymbol{q_1}^T \tilde{G}_N \tilde{\tilde{\boldsymbol{u}}} - \boldsymbol{q_1}^T H \boldsymbol{x}_0 + (d+m) \le 0 \\ & -\boldsymbol{q_2}^T \tilde{G}_j \tilde{\tilde{\boldsymbol{u}}} - \boldsymbol{q_2}^T H \boldsymbol{x}_0 + s_x x_0 \le 0 \quad j = 1, \ldots, N \\ & \boldsymbol{q_3}^T \tilde{G}_N \tilde{\tilde{\boldsymbol{u}}} + \boldsymbol{q_3}^T H \boldsymbol{x}_0 - D - \epsilon_D \le 0 \\ & -\boldsymbol{q_3}^T \tilde{G}_N \tilde{\tilde{\boldsymbol{u}}} - \boldsymbol{q_3}^T H \boldsymbol{x}_0 + D - \epsilon_D \le 0 \\ & \boldsymbol{q_4}^T \tilde{G}_N \tilde{\tilde{\boldsymbol{u}}} + \boldsymbol{q_4}^T H \boldsymbol{x}_0 - m \le 0 \\ & -\boldsymbol{q_4}^T \tilde{G}_N \tilde{\tilde{\boldsymbol{u}}} - \boldsymbol{q_4}^T H \boldsymbol{x}_0 - m \le 0 \\ & \boldsymbol{q_5}^T \tilde{G}_N \tilde{\tilde{\boldsymbol{u}}} + \boldsymbol{q_4}^T H \boldsymbol{x}_0 - m \le 0 \\ & -\boldsymbol{q_5}^T \tilde{G}_N \tilde{\tilde{\boldsymbol{u}}} - \boldsymbol{q_4}^T H \boldsymbol{x}_0 - m \le 0 \\ & \tilde{\tilde{\boldsymbol{u}}}(j) - \bar{u}_{\max} \le 0 \quad j = 1, \ldots, 4N \end{aligned} \tag{54}$$

There are a total of $M = 5N + 7$ inequality constraints. To write the problem in standard form, the constraints must be posed as equalities. To do this we include slack variables $s_i \ge 0$. The compact formulation is:

$$\begin{aligned} \text{Minimize} \quad & ||\tilde{\tilde{\boldsymbol{u}}}||_1 \\ \text{Subject to} \quad & a_i^T \tilde{\tilde{\boldsymbol{u}}} - b_i + s_i = 0 \quad i = 1, \ldots, M \\ & \tilde{\tilde{\boldsymbol{u}}} \ge 0 \end{aligned} \tag{55}$$

$$\boldsymbol{s} \ge 0 \tag{56}$$

Note that using the 1-norm to define the cost provides an equal penalty on every control. In general the cost may be defined as $\boldsymbol{c}^T \tilde{\tilde{\boldsymbol{u}}}$, where the penalty vector $\boldsymbol{c}$ can be chosen to provide variable weights across time.

An example is shown in Fig. 8. The initial relative state is a $+5$ meter along-track offset with zero velocity. The desired along-track drift is 10 m per orbit, with a tolerance of ± 1 m. The separation maneuver is planned for $T_{\text{exit}} = 600$ seconds, $d = 30$ m, and $m = 15$ m. The trajectory, which is plotted in the XZ plane, illustrates that the desired final separation of 45 m is achieved. In addition, the along-track drift is 11 m per orbit, within the desired range, and the along-track oscillation of 11.4 m is less than the maximum allowance of 15 m. This maneuver requires a delta-v of 0.13 m/s.

LP-Based Nominal Guidance

Recall that the main objectives of the nominal guidance mode are to bound the along-track drift rate and to achieve a safe ellipse. The constraints for bounding the along-track drift were given in the previous section, in (42) and (43). Recall from Section 2 that the definition of a safe ellipse is a trajectory in the radial / cross-track (YZ) plane that surrounds the circular avoidance region. The constraint is therefore:

$$y^2 + z^2 \ge (d+m)^2$$

This, however, is a non-linear and non-convex constraint. In order to model this constraint in the context of a linear program, a novel approach is required. Consider the illustration in Fig. 9. The true feasible space for the trajectory is everything outside of the yellow avoidance region. At any given time, however, the position must lie in a halfspace defined by a tangent line on the circle. The general equation for the tangent line is:

$$\cos\theta y + \sin\theta z = d + m \tag{57}$$

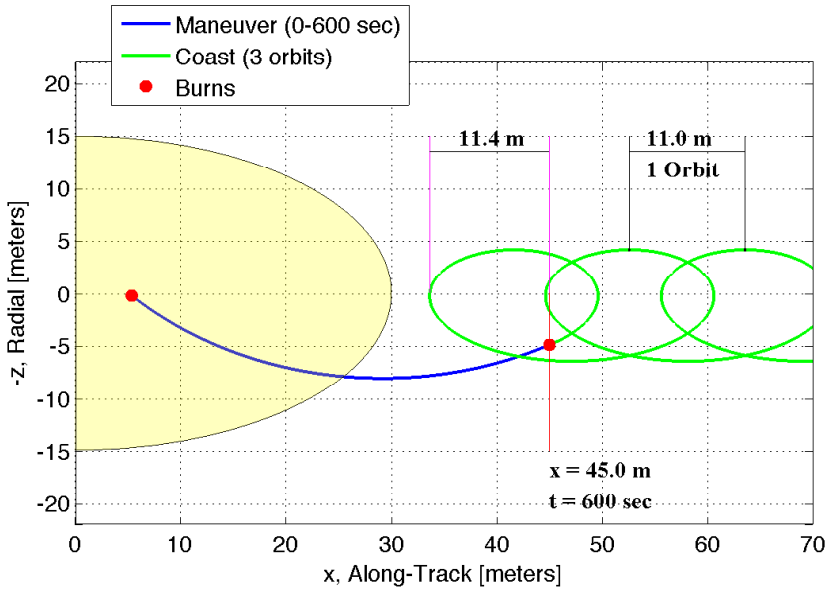

Figure 8. Separation maneuver planned to reach $x = 45$ m at $T_{exit} = 600$ sec, with $9 - 11$ m/orbit drift.

The relative dynamics dictate that the motion will rotate about the avoidance region. Therefore, we can define a rotating tangent line, where θ_j varies linearly with time. The line should make one full rotation each orbit period, so the equation becomes:

$$\theta = \theta_0 + nt \tag{58}$$

where n is again the mean orbit rate. The resulting inequality constraint on the state vector $\boldsymbol{x}_j$ is:

$$-\boldsymbol{q_6}^T \boldsymbol{x}_j - (d+m) \leq 0 \tag{59}$$

where $\boldsymbol{q_6}^T = [0, \cos\theta_j, \sin\theta_j, 0, 0, 0]$. The LP formulation becomes:

$$\begin{aligned}
&\text{Minimize} && ||\tilde{\tilde{\boldsymbol{u}}}||_1 \\
&\text{Subject to} && \boldsymbol{q_3}^T \tilde{G}_N \tilde{\tilde{\boldsymbol{u}}} + \boldsymbol{q_3}^T H \boldsymbol{x}_0 - d - \epsilon_D \leq 0 \\
& && -\boldsymbol{q_3}^T \tilde{G}_N \tilde{\tilde{\boldsymbol{u}}} - \boldsymbol{q_3}^T H \boldsymbol{x}_0 + d - \epsilon_D \leq 0 \\
& && -\boldsymbol{q_6}^T \tilde{G}_j \tilde{\tilde{\boldsymbol{u}}} - \boldsymbol{q_6}^T H \boldsymbol{x}_0 - (d+m) \leq 0 \quad j = K, \ldots, N \\
& && \tilde{\tilde{\boldsymbol{u}}}(j) - \bar{u}_{\max} \leq 0 \quad j = 1, \ldots, 4N
\end{aligned} \tag{60}$$

The first two constraints are for bounding the drift rate, just as in the separation problem. The next constraint enforces the YZ plane trajectory to surround the projected avoidance region, for times $t(K)$ through $t(N)$, with $K > 1$. In general, it is not possible to enforce this constraint from $t = 0$ since the initial position may be inside the projected avoidance region. This approach affords the spacecraft $t(K)$ seconds to maneuver out of the YZ plane circle.

It is also necessary to choose an appropriate value for θ_0. In fact, because this is a parameter of the optimization problem, the problem may be solved iteratively over a range of values of θ_0, so that the best solution may be selected. Consider the example shown in Fig. 10. The original trajectory (red) intersects the projected avoidance region. The LP method described above is used to plan a maneuver (blue) that reaches a safe ellipse (green).The total delta-v in this case is 0.018 m/s. In this approach, multiple burns are allowed, so that a maneuver may begin while the MAIN is inside the projected avoidance region.

4. ROBUSTNESS TO NAVIGATION UNCERTAINTY

Referring back to the optimization problems (54) and (60), the i^{th} inequality constraint has the form:

$$\boldsymbol{a}_i^T \tilde{\tilde{\boldsymbol{u}}} - b_i \leq 0 \tag{61}$$

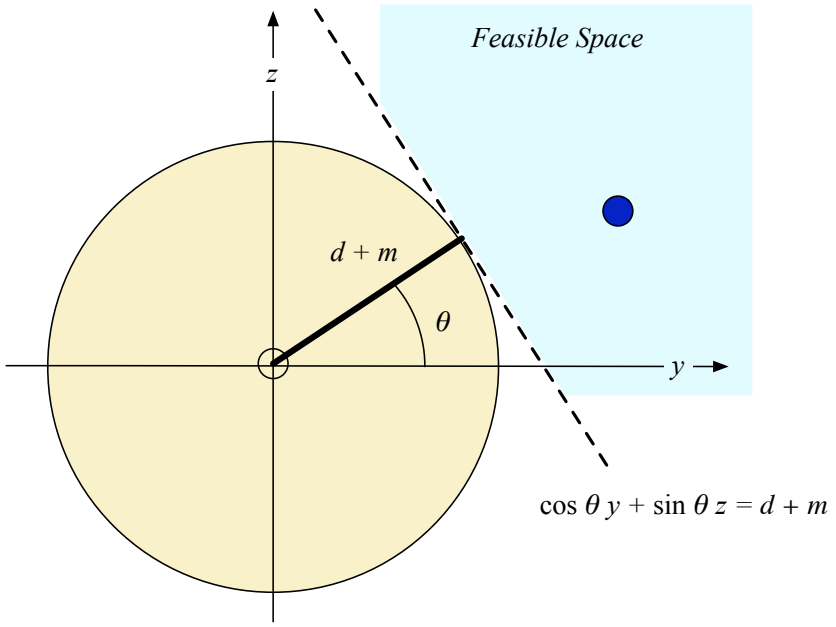

Figure 9. Illustration of a convex halfspace used to approximate the non-convex avoidance region. This halfspace represents the feasible set for the YZ plane position at one instant in time. The rotation of the tangent line around the circle enables the trajectory to encircle the avoidance region.

where the general formulas for $\boldsymbol{a}_i$ and b_i are: $\boldsymbol{a}_i^T = \boldsymbol{q}^T \tilde{G}$ and $b_i = \boldsymbol{q}^T H \boldsymbol{x}_0 + p_i$. The p_i term depends on parameters such as d, m, and ϵ_D,

Uncertainty in the initial state $\boldsymbol{x}_0$ may be modeled by defining $\boldsymbol{x}_0$ as:

$$\boldsymbol{x}_0 = \hat{\boldsymbol{x}}_0 + B_{COV}\delta, \quad \delta \in \mathbb{R}^{n_x}, ||\delta|| = 1 \tag{62}$$

where $\hat{\boldsymbol{x}}_0$ is the estimated initial state, n_x is the number of states, and B_{COV} is the diagonal matrix of the state covariance [22, 23]. The true initial state lies in an ellipsoid defined by this covariance matrix. The objective for maneuver planning is to satisfy the inequality of (61) for all possible values of $\boldsymbol{x}_0$. Therefore, the new constraint becomes:

$$\boldsymbol{a}_i^T \tilde{\tilde{\boldsymbol{u}}} - b_i \leq 0 \tag{63}$$

$$\boldsymbol{a}_i^T \tilde{\tilde{\boldsymbol{u}}} - \boldsymbol{q}^T H \hat{\boldsymbol{x}}_0 - p_i + ||\boldsymbol{q}^T H^{(i)} B_{COV}|| \leq 0 \tag{64}$$

This constraint remains linear in $\tilde{\tilde{\boldsymbol{u}}}$, and so the problem remains an LP. The norm term depends only on known problem data. If uncertainty in the dynamics were considered, this would introduce second order terms in the decision vector, and would become a second order cone programming (SOCP) problem.

The addition of this term to each constraint effectively models the worst case perturbation in $\boldsymbol{x_0}$ for that constraint. The along-track motion is particularly sensitive to navigation errors in the relative velocity. It is therefore important to choose the along-track drift rate bounds appropriately, given the size of the covariance matrix. For example, with any initial state uncertainty, it is impossible to realize an exact drift rate, so the tolerance must be greater than zero. In general, the size of the tolerance must grow with the size of the covariance matrix. In fact, we can compute the minimum feasible tolerance as:

$$\epsilon_D = ||[6n, 0, 0, 0, 0, 3]A^N B_{COV}|| \left(\frac{2\pi}{n}\right) \tag{65}$$

This relationship is illustrated in the next section.

Numerical Simulations

To illustrate the effectiveness of the robust optimization method, we consider two separate Monte Carlo simulations, shown in Fig. 11. In each case, random state errors are applied. In the plot on the left, we have the same scenario as

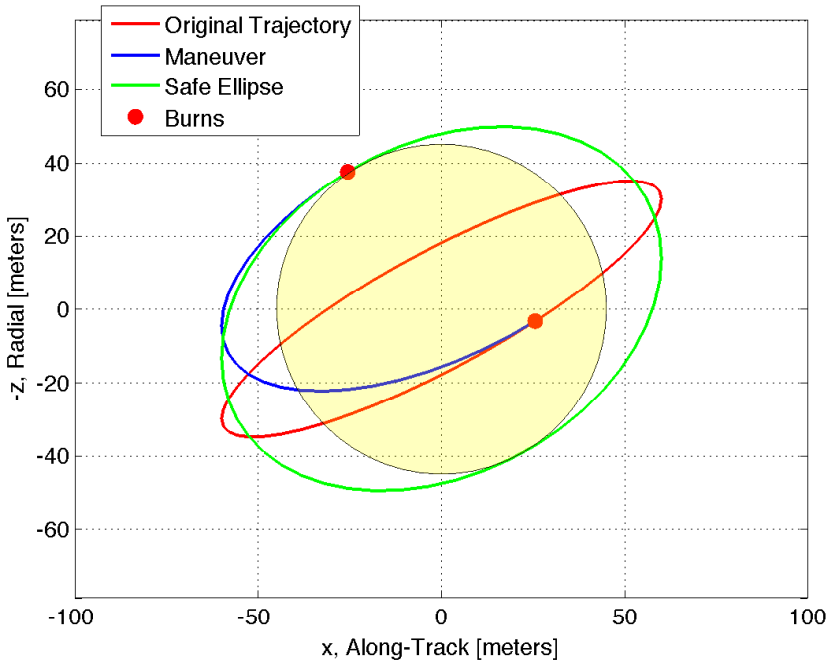

Figure 10. Nominal maneuver planned to reach a safe ellipse of minimum distance 45 *m.*

in Fig. 8. The uncertainty is 1.0 cm standard deviation in position, and 1.0 mm/s in velocity (1-sigma). The original non-robust solution applied a 1 m/orbit tolerance on the along-track drift, which is too small for this level of uncertainty. The minimum tolerance, found from (65), is 18 m/orbit. The robust optimization method is used with this tolerance to compute a solution, which is evaluated against 1,000 random samples of the initial state. The original non-robust solution (bottom plot) is seen to re-enter the avoidance region in 133 cases, with a worst case distance of 10.6 m. The robust solution re-enters only 27 times with a worst case of 28.3 m. A small number of re-entries occur even in the robust case because the constraints allow it. Specifically, the drift rate tolerance is large enough that it is possible for the associated along-track oscillation to cause re-entry after the maneuver has been completed. This could be avoided by reducing the allowed along-track oscillation, or increasing the final along-track position. This approach is similar to the separation guidance law outlined in Section 2 where the safety factor is used, but provides a more rigorous proof of robustness.

In the right-hand-side plot of Fig. 11, the MAIN starts out inside the avoidance region, only 1.2 m from the TARGET. The noise in this case is $10\times$ that of the previous simulation. It first plans and executes a separation maneuver, then plans and executes a nominal maneuver to reach a safe ellipse in the YZ plane. The minimum tolerance on the drift per orbit in this case is about 182 m. The center of radial motion can therefore be significantly high; according to (12) it is equivalent to

$$x_c = D/(3\pi)$$

which in this case is 19.27 m. We must increase the size of the of the avoidance region to account for this offset. The simulation results illustrate that trajectories planned with the robust optimization approach successfully avoid the region.

5. CONCLUSIONS

This paper has summarized the collision avoidance strategy for the PRISMA mission, which employs fully closed-form solutions, and has outlined an alternative numerical method using robust optimization. The PRISMA safe guidance strategy involves maneuvering to a "safe ellipse" – a passively safe relative trajectory, which will encircle an ellipsoidal avoidance region centered at the TARGET. The closed-form solutions for planning separation maneuvers and achieving the safe ellipse are derived from the Clohessy-Wiltshire equations, and are shown to meet the mission requirements.

The alternative numerical method follows the original avoidance strategy, but formulates the maneuver planning problem as an optimization problem. The requirements for exiting the avoidance region and achieving a safe ellipse are expressed

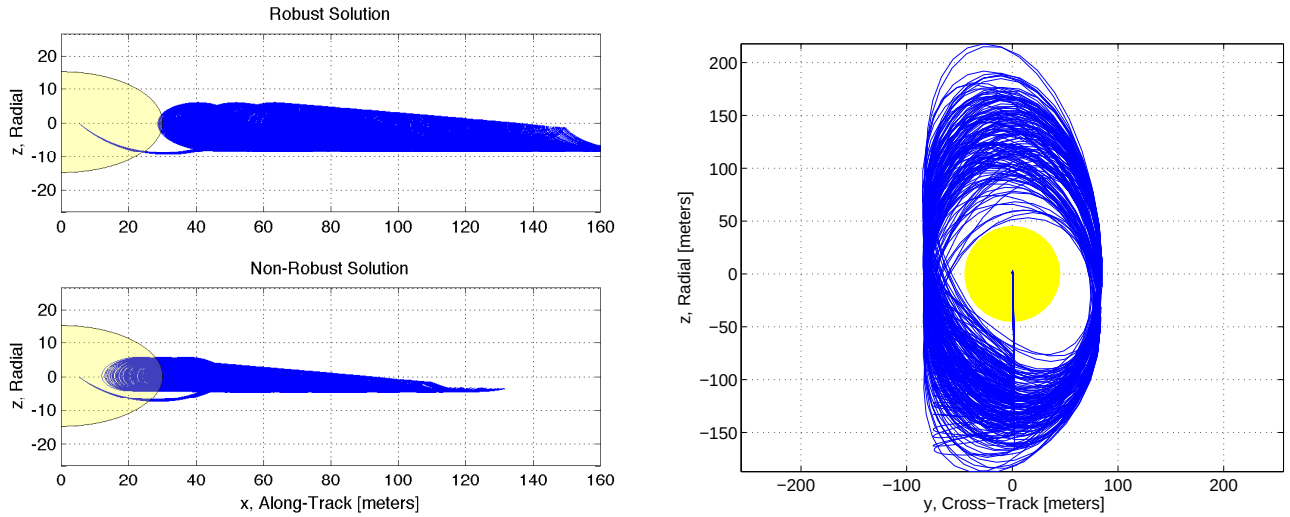

Figure 11. Monte Carlo simulations using random navigation errors. (Left) Separation maneuver to exit avoidance region, starting at $x_0 = 5$ m. 1,000 Simulations with random initial state errors, 1 cm position, 1 mm/s velocity (1-sigma). (Right) 100 simulations of a combined separation and nominal maneuver to reach a safe ellipse, starting inside the avoidance region with random noise, 10 cm position, 10 mm/s velocity (1-sigma).

as linear constraints on the state vector, which is itself a linear function of the decision variables. The problem is therefore well-posed as a linear program (LP). The novelty in this approach is that we approximate the original non-convex constraints by using a combination of well-defined, time-varying convex constraints.

Finally, we extend the LP solution method by accounting for navigation uncertainty in the state estimate used for planning. This leads to a pure LP formulation for robust collision avoidance maneuver planning, without the need for integer constraints. This new numerical method is a practical option for the online computation of collision avoidance maneuvers and is easily extensible to include eccentric dynamics, relative disturbance models, and time-varying thruster constraints. In order to extend to highly eccentric orbits, additional work is needed to derive expressions for the bounds on relative motion, so that they may be used to plan robust maneuvers.

ACKNOWLEDGMENTS

The work described in this paper was funded in part by a Cooperative Research and Development Agreement with the Swedish Space Corporation, in support of the PRISMA program.

REFERENCES

[1] R. Larsson, J. Mueller, S. Thomas, B. Jakobsson, and P. Bodin, "Orbit Constellation Safety on the PRISMA In-Orbit Formation Flying Testbed," in *3rd International Symposium on Formation Flying, Missions and Technologies*, ESA, April 2008.

[2] M. Tillerson and J. How, "Formation Flying Control in Eccentric Orbits," in *AIAA Guidance, Navigation and Control Conference and Exhibit*, (Montreal, Canada), 2001.

[3] N. Hamilton, D. Folta, and R. Carpenter, "Formation Flying Satellite Control Around the Sun-Earth L2 Libration Point," in *AIAA/AAS Astrodynamics Specialist Conference and Exhibit*, no. 2002-4528, AIAA, 2002.

[4] G. Inalhan, M. Tillerson, and J. How, "Relative Dynamics and Control of Spacecraft Formations in Eccentric Orbits," *Journal of Guidance, Control and Dynamics*, vol. 25, no. 1, 2002.

[5] A. Richards, T. Schouwenaars, J. How, and E. Feron, "Spacecraft trajectory planning with avoidance constraints using mixed-integer linear programming," *Journal of Guidance, Control, and Dynamics*, vol. 25, no. 4, pp. 755–764, 2002.

[6] R. Smith and F. Hadaegh, "Control Topologies for Deep Space Formation Flying Spacecraft," in *Proceedings of the American Control Conference*, pp. 2836–2841, 2002.

[7] M. Tillerson, G. Inalhan, and J. How, "Co-ordination and Control of Distributed Spacecraft Systems Using Convex Optimization Techniques," *International Journal of Robust Nonlinear Control*, vol. 12, no. 1, pp. 207–242, 2002.

[8] K. Yamanaka and F. Ankersen, "New State Transition Matrix for Relative Motion on an Arbitrary Elliptical Orbit," *Journal of Guidance, Control, and Dynamics*, vol. 25, no. 1, pp. 60–66, 2002.

[9] R. A. Broucke, "Solution of the Elliptic Rendezvoud Problem with Time as the Independent Variable," *Journal of Guidance, Control and Dynamics*, vol. 26, no. 4, pp. 615–621, 2003.

[10] J. R. Carpenter, J. Leitner, D. Folta, and R. Burns, "Benchmark Problems for Spacecraft Formation Flying Missions," in *Proceedings of the AIAA Guidance, Navigation and Control Conference*, no. AIAA 2003-5364, (Austin, TX), AIAA, 2003.

[11] D. Gim and K. Alfriend, "State transition matrix of relative motion for the perturbed noncircular reference orbit," *Journal of Guidance, Control, and Dynamics*, vol. 26, no. 6, pp. 956–971, 2003.

[12] A. Richards and J. How, "Model Predictive Control of Vehicle Maneuvers with Guaranteed Completion Time and Robust Feasibility," in *American Control Conference*, vol. 5, 2003.

[13] D. Scharf, F. Hadaegh, and S. Ploen, "A survey of spacecraft formation flying guidance and control (Part I): guidance," in *American Control Conference*, vol. 2, 2003.

[14] L. Breger and J. How, "GVE-Based Dynamics and Control for Formation Flying Spacecraft," in *2nd International Symposium on Formation Flying Missions and Technologies*, (Washington, D.C.), NASA Goddard Space Flight Center, 2004.

[15] D. Scharf, F. Hadaegh, and S. Ploen, "A survey of spacecraft formation flying guidance and control (Part II): control," *Pasadena, CA: Jet Propulsion Laboratory, National Aeronautics and Space Administration*, 2004.

[16] Y. Kim, M. Mesbahi, and F. Hadaegh, "Multiple-Spacecraft Reconfiguration Through Collision Avoidance, Bouncing, and Stalemate," *Journal of Optimization Theory and Applications*, vol. 122, no. 2, pp. 323–343, 2004.

[17] J. B. Mueller, "A Multiple-Team Organization for Decentralized Guidance and Control of Formation Flying Spacecraft," in *Proceedings of the 1st AIAA Intelligent Systems Conference*, no. AIAA-2004-6249, (Chicago, IL), AIAA, Sep 2004.

[18] J. B. Mueller and S. J. Thomas, "Decentralized Formation Flying Control in a Multiple-Team Hierarchy," *Annals of the New York Academy of Sciences*, vol. 1065, pp. 112–138, 2005.

[19] J. Mueller and S. Thomas, "A Scalable System for Formation Flying Guidance and Control," in *ESA Guidance Navigation and Control Conference*, (Loutaki, Greece), October 2005.

[20] C. Chasset, S. Berge, P. Bodin, and B. Jakobsson, "3-axis Magnetic Control with Multiple Attitude Profile Capabilities in the PRISMA Mission," *Space Technology*, vol. 26, no. 3-4, pp. 137–154, 2007.

[21] S. Persson, B. Jacobsson, and E. Gill, "PRISMA Demonstration Mission for Advanced Rendezvous and Formation Flying Technologies and Sensors," in *56th International Astronautical Congress*, no. IAF-05-B5.6.B.07, (Fukuoka, Japan), 2005.

[22] F. Alizadeh and D. Goldfarb, "Second-Order Cone Programming," *Mathematical Programming*, vol. 95, no. 1, pp. 3–51, 2003.

[23] A. Ben-Tal and A. Nemirovski, "Robust Convex Optimization," *Mathematics of Operations Research*, vol. 23, no. 4, pp. 769–805, 1998.