JET PROPULSION LABORATORY **INTERNAL DOCUMENT**
JPL D-32948
September 2, 2005

TO: David S. Bayard (Fiscal Year 2005 R&TD on Small Body Guidance, Navigation, and Control)[1]

FROM: John M. Carson III and A. Behçet Açıkmeşe

SUBJECT: Small Body GN&C Research Report: A Guidance and Control Technique for Small-Body Proximity Operations with Guaranteed Guidance Resolvability and Required Thruster Silent Time

ABSTRACT

The guidance and control (G&C) algorithms for enabling small-body proximity operations are developed by using a model predictive control approach along with a convexification of the governing dynamics, control constraints, and trajectory/state constraints. The open-loop guidance is solved ahead of time or in a resolvable, real-time manner through the use of PWG (Pseudo Way-point Generation), a technique developed in this research. The PWG scheme ensures required thruster silent times during trajectory maneuvers. The feedback control is implemented to track the PWG trajectories in a manner that guarantees the resolvability for the open-loop problem, enabling the ability to update the G&C in a model-predictive manner. The schemes incorporate gravity models and thruster firing times into discrete dynamics that are solved as a optimal control problem to minimize fuel consumption or thruster energy expenditure. The optimal control problem is cast as an LMI (Linear Matrix Inequality) and then solved through Semi-Definite Programming techniques in a computationally efficient manner that provides convergence and constraint guarantees.

[1]Cleared for U.S. and foreign release, CL#05-2804.

1 Introduction

Determination of an optimal trajectory for landing, hopping, proximity maneuvering, or take-off on a small-body can be accomplished through classical way-point techniques, state-of-the-art real-time trajectory generation methods, or a blending of the two. This blending, herein termed PWG (pseudo way-point generation), utilizes onboard knowledge of the small-body properties to autonomously generate state-space models at discrete time intervals during a trajectory maneuver. The time intervals are chosen such that required thruster silence times, as well as set thruster firing times are respected. The PWG can be utilized either as a reference trajectory for an outer-loop tracking controller or as the basis of a receding horizon implementation. The main intention of this work is to utilize PWG in the former capacity, with the possibility of updating the reference trajectory as better estimates of the small-body properties become available.

The control inputs developed for PWG are determined through the solution of LMI's (Linear Matrix Inequalities) that optimize either fuel usage or total thrust energy. The trajectory determination and control inputs are iterated with a gravity model to converge upon a model-accurate, open-loop trajectory and thus a set of admissible way-points. Figure 1 illustrates the iteration process behind PWG.

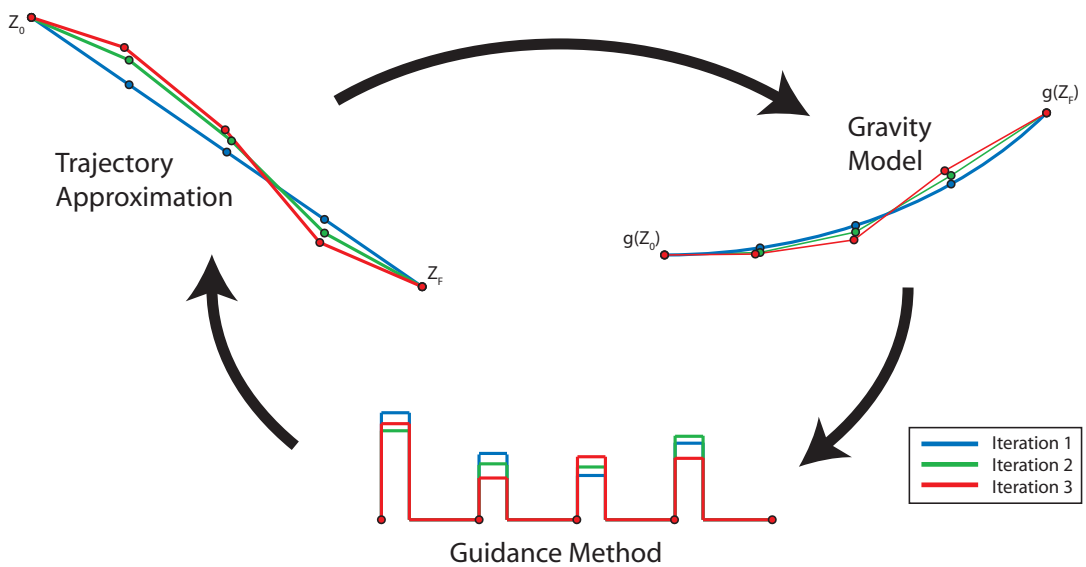

Figure 1: Iterative Process for Pseudo Way-point Generation

2 Development of Pseudo Way-Point Guidance Scheme

2.1 Governing Dynamics

The state dynamics of a spacecraft orbiting, landing, or conducting proximity operations at a small-body are governed by the following equations of motion, expressed in a rotating frame (i.e. a frame fixed to the small-body):

$$\ddot{\boldsymbol{r}} + \dot{\boldsymbol{\omega}} \times \boldsymbol{r} + 2\boldsymbol{\omega} \times \dot{\boldsymbol{r}} + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \boldsymbol{r}) = u + \boldsymbol{f} + \boldsymbol{d} + \boldsymbol{g}(\boldsymbol{r}) \tag{1}$$

where $\mathbf{r} \in \mathbb{R}^3$ is the radius vector from the small-body center of mass to the spacecraft, $\boldsymbol{\omega} \in \mathbb{R}^3$ is the rotation rate of the body, $\mathbf{u} \in \mathbb{R}^3$ are applied specific forces (i.e. thruster firings or other control inputs - force per unit mass), $\mathbf{f}$ and $\mathbf{d}$ are disturbance force per unit mass (due to external gravity disturbances, solar forces, or comet dust ejection), and $\mathbf{g}(\mathbf{r})$ is the gravity.

The small-body gravity field can be modeled through several methods, two of which are gravity harmonics and polyhedral gravity models. Regardless of the chosen method, linearized models of gravity can be used in the development of a guidance scheme:

$$\boldsymbol{g}(\boldsymbol{r}) \approx \boldsymbol{g}(\boldsymbol{r}_k) + \left.\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{r}}\right|_{\boldsymbol{r}_k} (\boldsymbol{r} - \boldsymbol{r}_k) + \boldsymbol{\delta}(\boldsymbol{r}, \boldsymbol{r}_k) = G_{r_k}\boldsymbol{r} + g_k + \boldsymbol{\delta}_k \tag{2}$$

where $\boldsymbol{r}_k$ is a reference radius, $\boldsymbol{\delta}_k$ is a norm-bounded, higher-order gravity perturbation, $g_k = \boldsymbol{g}(\boldsymbol{r}_k) - \left.\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{r}}\right|_{\boldsymbol{r}_k} \boldsymbol{r}_k$, and $G_{r_k} = \left.\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{r}}\right|_{\boldsymbol{r}_k}$.

The class of small-bodies to be studied will have stable and constant rotation rates ($\dot{\boldsymbol{\omega}} = 0$), which is a necessary assumption in order to rewrite the dynamics of equation 1 in state space form:

$$\begin{aligned} x &= \begin{pmatrix} \boldsymbol{r} \\ \dot{\boldsymbol{r}} \end{pmatrix} \\ \dot{x} &= A_0 x + Bu + Bg(C_q x) && (3) \\ &\approx A_k x + Bu + Bg_k && (4) \end{aligned}$$

where

$$\begin{aligned} A_0 &= \begin{bmatrix} 0 & I \\ -\widehat{\boldsymbol{\omega}}^2 & -2\widehat{\boldsymbol{\omega}} \end{bmatrix}, \qquad B = \begin{bmatrix} 0 \\ I \end{bmatrix}, \qquad C_q = \begin{bmatrix} I & 0 \end{bmatrix} \\ A_k &= A_0 + BG_{r_k}E, \qquad E = \begin{bmatrix} I & 0 \end{bmatrix}. \end{aligned}$$

and $\widehat{\boldsymbol{\omega}}$ is the 3×3 matrix representation of the vector cross product $\boldsymbol{\omega} \times (\cdot)$. Note, the subscript k on A_k and g_k indicates a dependence on chosen reference radius $\boldsymbol{r}_k$.

2.2 Discrete Dynamics Model

A discrete model of the dynamics is necessary for development of the PWG scheme. An assumption is made that finite burns are used in the guidance and that they are constant during each burn interval (though not necessarily the same constant), this facilitates the use of zero-order hold in the discretization. The time interval Δt used in the discrete dynamics is based on finite burns of duration δ_f and required thruster silent times of duration δ_s such that $\Delta t \geq \delta_f + \delta_s$.

The solution to the equations of motion in equation 4 is

$$x(t) = e^{A_k(t-t_0)}x(t_0) + \int_{t_0}^{t} e^{A_k(t-\tau)}B(u(\tau) + g_k)d\tau$$

which can be used to develop the discrete models over fixed time interval $\Delta t = t_{k+1} - t_k$.

During the firing portion (i.e. $t \in [t_k, t_k + \delta_f]$ where $t_k + \delta f < t_{k+1}$ and $u = u(t_k)$ is a constant, finite burn), the discrete solution becomes

$$\begin{aligned} x(t_k + \delta_f) &= e^{A_k((t_k+\delta_f)-t_k)}x(t_k) + \int_{t_k}^{t_k+\delta_f} e^{A_k((t_k+\delta_f)-\tau)}B(u(\tau) + g_k)d\tau \\ &= e^{A_k\delta_f}x(t_k) + \int_0^{\delta_f} e^{A_k(\delta_f-\tau)}Bd\tau \cdot u(t_k) + \int_0^{\delta_f} e^{A_k(\delta_f-\tau)}Bd\tau \cdot g_k \end{aligned}$$

and during the following silent portion (i.e. $t \in (t_k + \delta_f, t_{k+1})$ with $u = 0$) the discrete solution becomes

$$\begin{aligned} x(t_{k+1}) &= e^{A_k(t_{k+1}-(t_k+\delta_f))}x(t_k+\delta_f) + \int_{t_k+\delta_f}^{t_{k+1}} e^{A_k(t_{k+1}-\tau)}Bd\tau \cdot g_k \\ &= e^{A_k(\Delta t-\delta_f)}x(t_k+\delta_f) + \int_{\delta_f}^{\Delta t} e^{A_k(\Delta t-\tau)}Bd\tau \cdot g_k \end{aligned}$$

Combining these two equations provides the discrete equations over each time interval Δt:

$$\begin{aligned} &x(t_{k+1}) = e^{A_k\Delta t}x(t_k) + e^{A_k(\Delta t-\delta_f)}\int_0^{\delta_f} e^{A_k(\delta_f-\tau)}Bd\tau \cdot u(t_k) + \int_0^{\Delta t} e^{A_k(\Delta t-\tau)}Bd\tau \cdot g_k \\ &\Rightarrow x_{k+1} = A_{d,k}x_k + B_{d,k}u_k + E_{d,k}g_k \end{aligned} \tag{5}$$

where $A_{d,k} = e^{A_k\Delta t}$, $B_{d,k} = e^{A_k(\Delta t-\delta_f)}\int_0^{\delta_f} e^{A_k(\delta_f-\tau)}Bd\tau$, $E_{d,k} = \int_0^{\Delta t} e^{A_k(\Delta t-\tau)}Bd\tau$, g_k fixed over each Δt,

$$u(t_k) = \begin{cases} u_k, & t \in [t_k, t_k+\delta_f] \\ 0, & t \in (t_k+\delta_f, t_{k+1}) \end{cases}$$

with u_k constant, and $k = 0, ..., N-1$. Note, g_k can be chosen based on the state $x(t_k)$ or $x(t_{k+1})$ to provide discrete updates to gravity, thereby increasing the accuracy of the discrete solution. Additionally, the dependence on the reference radius $\boldsymbol{r}_k$ used to linearize gravity translates into dependencies on k for all three matrices A_d, B_d, and E_d.

2.3 Solution of the Open-Loop Guidance Problem

The open loop trajectory is designed based on a discrete linear time varying system

$$x_{k+1} = A_{d,k}x_k + B_{d,k}u_k + E_{d,k}g_k\,. \tag{6}$$

As explained in the previous section, (6) is obtained by discretizing the continuous time dynamics

$$\dot{x} = A_0x + Bu + Bg(C_qx)\,, \tag{7}$$

along a given reference state trajectory. Once a trajectory for the open-loop problem is obtained, it is used as the reference trajectory to re-linearize and discretize the dynamics in (7), to resolve the open-loop problem, and so on. This is the iterative procedure to solve the open-loop optimal control problem for the nonlinear system given in (7). Now, we will introduce additional notation to describe the this process. First we introduce an index for trajectory iterations, $j = 0, \ldots, m$ where m is the index for the last trajectory in a given computation cycle, i.e. the variable names are changed as follows

$$\begin{gathered} x_k \to x_{j,k}\,, \quad u_k \to u_{j,k}\,, \quad g_k \to g_{j,k}\,, \\ A_{d,k} \to A_{j,k}\,, \quad B_{d,k} \to B_{j,k}\,, \quad E_{d,k} \to E_{j,k}\,. \end{gathered}$$

Then, we can rewrite (6) as,

$$x_{j+1,k+1} = A_{j,k}x_{j+1,k} + B_{j,k}u_{j+1,k} + E_{j,k}g_{j,k}, \quad j = 0, \ldots, m-1, \quad k = 0, \ldots, N-1. \tag{8}$$

Now, we can describe the solution algorithm:

PWG Algorithm

Given an *initial reference trajectory* $\{x_{0,0}, \ldots, x_{0,N-1}\}$, perform the following steps for $j = 0, \ldots, m-1$

1. Compute the discrete time model parameters $A_{j,k}$, $B_{j,k}$, $E_{j,k}$, $k = 0, \ldots, N-1$, by using $\{x_{j,0}, \ldots, x_{j,N-1}\}$ as explained sections 2.1 and 2.2.

2. Solve the following *second-order cone programming* problem to generate $\{x_{j+1,0}, \ldots, x_{j+1,N-1}\}$ and $\{u_{j+1,0}, \ldots, u_{j+1,N-1}\}$:

$$\left.\begin{array}{l} \text{Minimize} \quad \sum_{k=0}^{N-1} \left(\alpha\|u_{j+1,k}\| + \beta\|u_{j+1,k}\|^2\right) + \epsilon\|E_c(x - x_F)\| \\ \quad \text{subject to} \\ x_{j+1,k+1} = A_{j,k}x_{j+1,k} + B_{j,k}u_{j+1,k} + E_{j,k}g_{j,k}, \quad k = 0, \ldots, N-1 \\ x_{j+1,k} \in \mathcal{X}_o\,, \quad k = 1, \ldots, N-1 \\ u_{j+1,k} \in \mathcal{U}\,, \quad k = 0, \ldots, N-1 \\ x_{j+1,0} = x_S \quad \text{or} \quad (x_{j+1,0} - x_S)^T P(x_{j+1,0} - x_S) \leq 1 \\ E_e(x_{j+1,N} - x_F) = 0 \end{array}\right\} \quad (9)$$

where $\mathcal{X}_o$ and $\mathcal{U}$ are convex sets that can be described by using second-order cone constraints (including linear and quadratic inequalities), $\alpha = 0$, $\beta = 1$ are used for *minimum energy* and $\alpha = 1$, $\beta = 0$ for *minimum fuel* problems, x_F is the desired final state to be reached, x_S is the current state (measured), E_e is for the equality constraint at the end state, E_c is for the contribution of the end state to the cost, $\epsilon \geq 0$, and $P = P^T > 0$ (the ellipsoid constraint on the initial state will be useful if the trajectories are recomputed at future times to guarantee resolvability, see Section 6).

Remark 1. *In PWG algorithm, there is no guarantee that the algorithm generates a convergent set of trajectories. One approach to assure such convergence is adding the following constraint to the optimization problem (9) after the first computation in the trajectory iteration,*

$$\|x_{j+1,k} - x_{j,k}\| \leq \kappa\|x_{j,k} - x_{j-1,k}\|\,, \quad k = 1, \ldots, N-1, \quad j = 1, \ldots, m-1\,,$$

where $0 < \kappa < 1$. This additional constraint guarantees that a Cauchy sequence is generated for each k, which imposes a convergence.

Remark 2. *We relax some of the end state constraints and make it a part of the cost. This is done to handle the some of the typical position state constraints. For example let the time sample horizon $N = 2$, and $x_0 = x_F$ that has zero velocity, and suppose that there is constant small body gravity and zero rotation rate for simplicity. Also suppose that motion in the direction of the gravity is limited to several meters (close proximity to the ground) not to collide with the small body. To have $x_N = x_F$ with two constant thrust firings, the spacecraft must be closer to the ground at time indices $k = 1$ than at $k = 0$ and $k = 2$ to accommodate zero velocity error at the end. However, depending on the firing and silent times, this proximity at the interior point must be such that*

collision is unavoidable, i.e. there exists to feasible trajectory for this case. One solution to this problem is not to match the velocity at the end exactly, and allow limit cycling type motion about the desired final position that can be accomplished in PWG by choosing $\epsilon > 0$, and

$$E_e = \begin{bmatrix} I & 0 \end{bmatrix}, \quad E_c = \begin{bmatrix} 0 & I \end{bmatrix}.$$

3 Implementation of Open-Loop Guidance Law

The open-loop PWG technique from equation 5 was implemented and tested for landing scenarios on an Eros-like asteroid [5]. The following sections detail verification of the implementation and evaluation of it for different gravity models.

In this report, we use the following example case in our simulations: Mass of the spacecraft is 400 kg, maximum available open loop thrust magnitude is 125 N, maximum feedback thrust is 20 N, the specific impulse for thrusters is $I_{sp} = 300$ sec, the maneuver initial state (position, in meters, and velocity, in meters/second,), x_0, and desired final state, x_F, are,

$$x_0 = \begin{bmatrix} 8950 \\ 100 \\ 0 \\ 1.5 \\ 2 \\ 0 \end{bmatrix}, \qquad x_F = \begin{bmatrix} 8450 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix},$$

and a state constrained must be imposed to ensure that the spacecraft does not fly subsurface, i.e.

$$c^T x(t) \geq 1, \quad \text{where} \quad C = \begin{bmatrix} 1/8445 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}.$$

3.1 Verification of Implementation

The PWG implementation was verified by zeroing the nonlinear portion of the dynamics (i.e. the gravity). This simplification means that when the PWG method is implemented discretely it will match the results of integration of the continuous equations of motion since the system becomes linear. The results were as expected, as shown in figure 2. Note that the pseudo-waypoints are shown as circles (labeled as X_{dis}, Y_{dis}, and Z_{dis}), and the integrated equations of motion provide the solid trajectories that intersect the pseudo-waypoints.

3.2 Implementation with Harmonic Gravity Model based on J_n Coefficients

The PWG technique was implemented for a gravity model based on J_n harmonic coefficients [1]. This gravity model comes from a potential of the form

$$V(r) = -\frac{\mu}{r}\left[1 - \sum_{n=2}^{\infty} J_n \left(\frac{r_e}{r}\right)^n P_n(\sin L)\right] \tag{10}$$

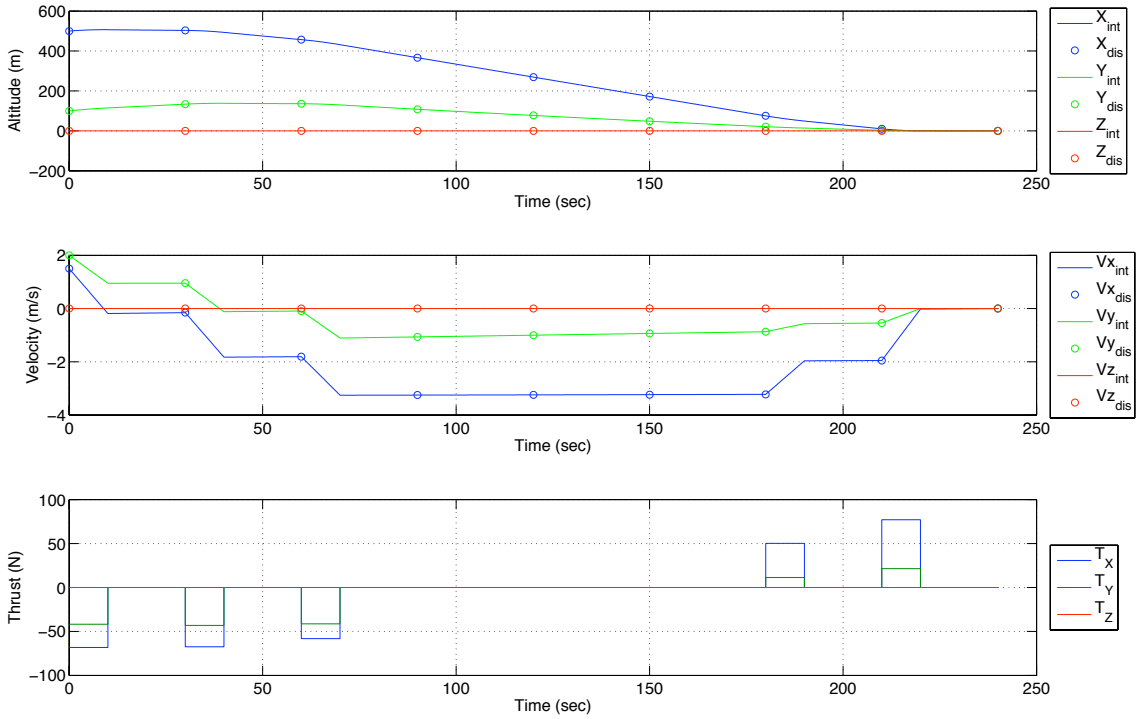

Figure 2: Linear Dynamics Controlled Exactly by Discrete Method

where $r = ||\boldsymbol{r}||$, P_n are Legendre polynomials of order n, L is the latitude $\frac{z}{r}$, μ is the gravitational parameter for the small-body, r_e is a reference radius, and J_n are coefficients for the gravity harmonics. This gravity model has a known linearization that is implemented as discussed in equation 2.

Results from the open-loop PWG method indicate that guidance without uncertainty can place the spacecraft to within 3 centimeters of the desired landing location, and final velocity is very close to the desired 0 mark. Figure 3 shows the results for the desired 240 second landing maneuver, and figure 4 zooms in on the final 5 seconds of the maneuver to highlight the precision of the open-loop PWG method. Note the built-in silent times seen in the thrust profiles, which are respected through the PWG technique.

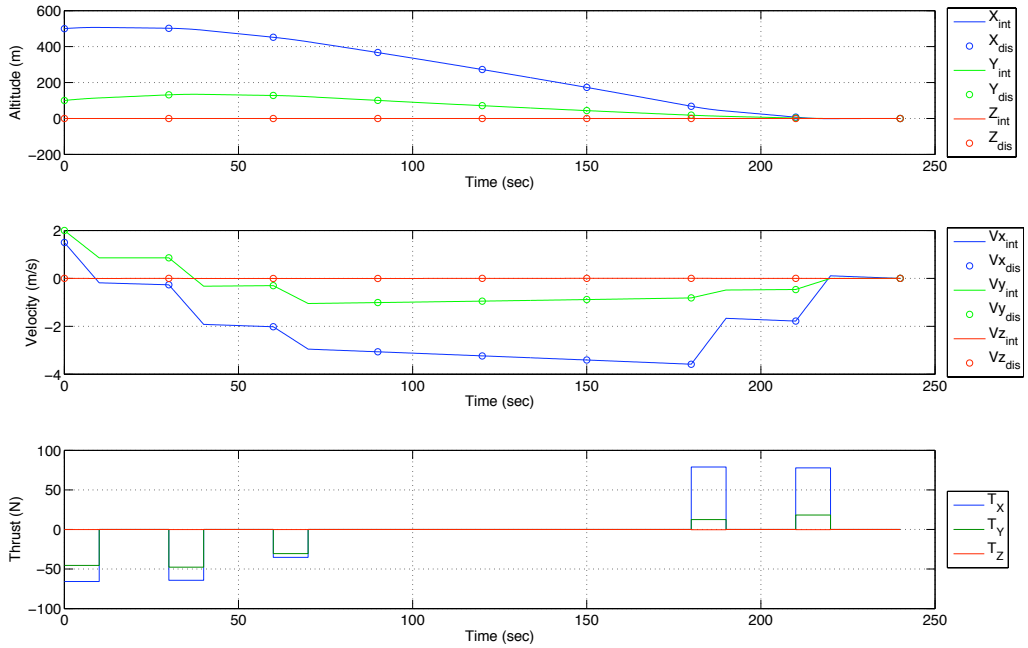

Figure 3: Guidance with PWG-based Control for Dynamics with J_n Harmonic Gravity

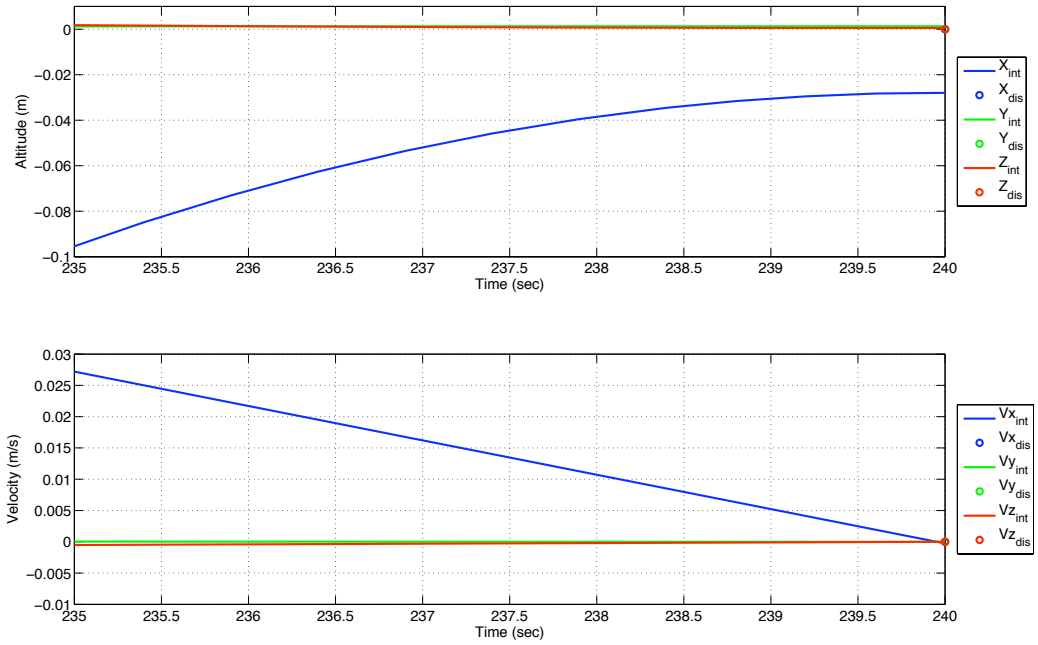

Figure 4: Zoom of Results in Figure 3 Depicting Landing Precision

3.3 Implementation with a Numerical-Gradient-Based Gravity Model

The gravity data for Eros contains up through fourth-order harmonics; however, the published coefficients are based on a more complicated spherical harmonic model for which linearizations are not currently available. This model has the form

$$V(r,\theta,\phi) = \left(\frac{\mu}{r}\right)\left[1 + \sum_{n=1}^{\infty}\sum_{m=0}^{n}\left(\frac{r_e}{r}\right)^n P_n^m(\cos\theta)\left\{C_{nm}\cos m\phi + S_{nm}\sin m\phi\right\}\right] \tag{11}$$

The prior data based on the J_n coefficients was computed through a fit of the published Eros data. However, since the published data provides a much more detailed gravity model, the fit for J_n coefficients actually introduces significant error in the gravity model. When the PWG method is utilized to develop guidance based on the linearized J_n gravity and the resultant control is applied to the dynamics integrated with the second gravity model given in equation 11, the results are very poor (See figure 5). As seen in the figure, the pseudo-waypoints are not properly tracked due to an improper gravity model.

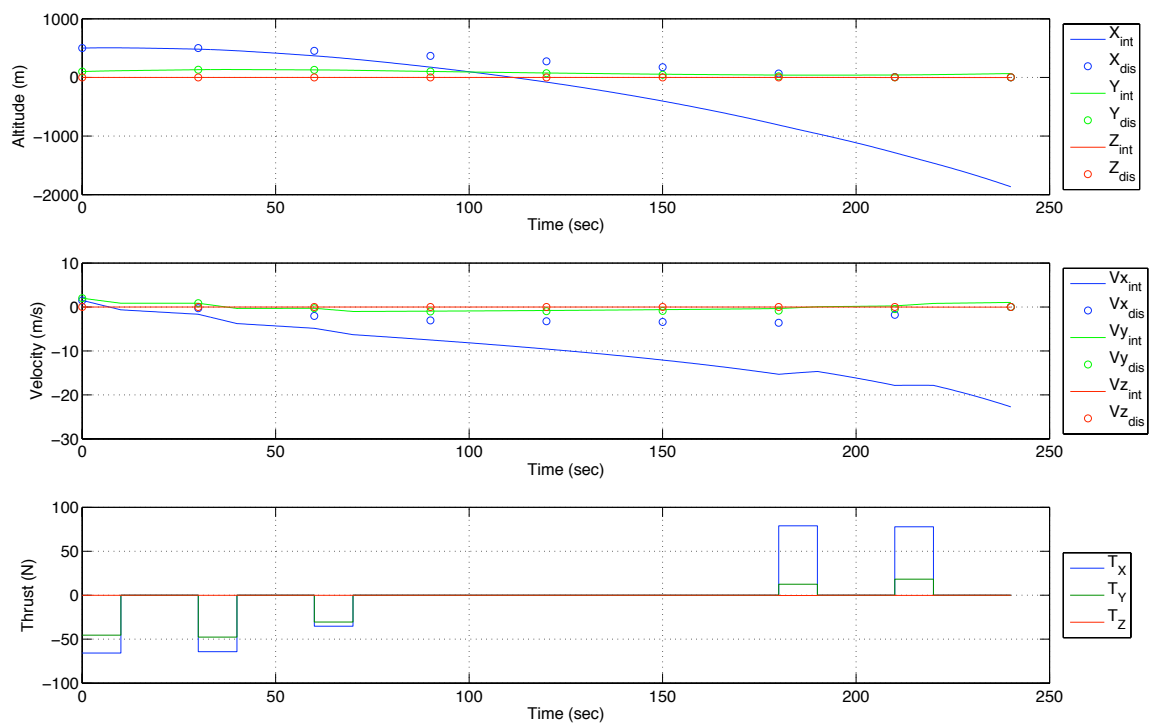

Figure 5: Guidance from J_n-Model has Poor Performance with $S_{n,m}$- and $C_{n,m}$-based Gravity

Fortunately, rather than using an improper gravity linearization for developing the discrete control, numerical gravity gradients instead provide a vast improvement, as shown in figure 6 where the implementation of the discrete controller shows that the system response now properly

tracks the pseudo-waypoints. This is a very promising method when a linearization is not available for a particular gravity model.

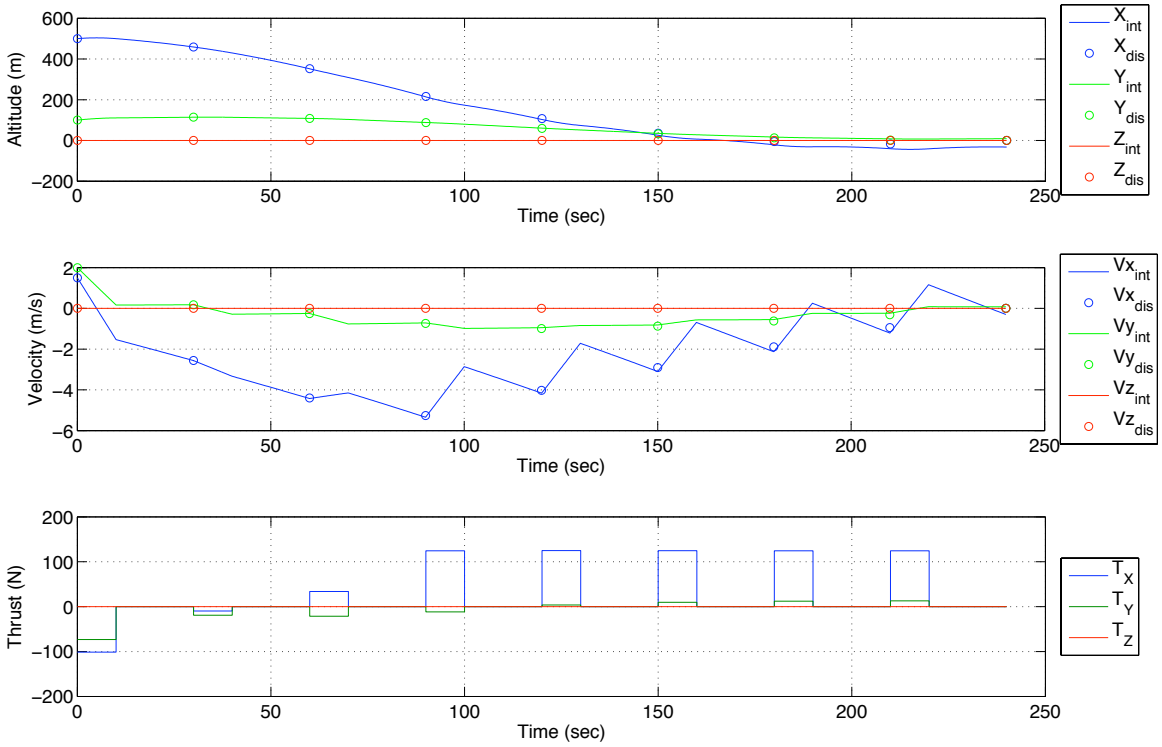

Figure 6: Improved PWG-Guidance through Numerical Gravity Gradients

4 Development of Feedback Control Scheme for Trajectory Tracking

In this section, we describe a feedback control method to track the trajectory generated by the guidance scheme. First we develop an approximate discrete-time dynamics of the tracking error. The discrete model is used for feedback design in order to accommodate the required thruster silent times, i.e. the feedback control input (such as the feed-forward control) is assumed to be active on prescribed time intervals.

Suppose the dynamics of the real system is given by,

$$\dot{\xi} = A_0\xi + B(u+v) + B\phi(C_q\xi) \tag{12}$$

where ξ is the actual state containing the position and the velocity, ϕ is the real gravitational acceleration, v is the feedback control force, and

$$C_q = \begin{bmatrix} I & 0 \end{bmatrix} .$$

Note that the model of the real dynamics used for the guidance is,

$$\dot{x} = A_0x + Bu + Bg(C_qx) ,$$

where g is the model of the gravitational acceleration. The error dynamics is given by,

$$\dot{\eta} = A_0\eta + Bv + B[\phi(C_q\xi) - g(C_qx)] , \tag{13}$$

where

$$\eta \triangleq \xi - x . \tag{14}$$

Then we have,

$$\dot{\eta} = A_0\eta + Bv + B(\psi + w) , \tag{15}$$

where

$$\begin{aligned} \psi &= \phi(C_q\xi) - \phi(C_qx) , \\ w &= \phi(C_qx) - g(C_qx) . \end{aligned}$$

Now, if we discretize (15) by assuming that the feedback control is nonzero on $(t_k,\, t_k + \delta_f]$ and zero on $(t_k + \delta_f,\, t_k + \Delta t]$, and ψ and w are constant on $(t_k,\, t_k + \Delta t)$, we obtain the following approximation to the discrete dynamics (see Section 2.2 for details),

$$\eta_{k+1} \approx A_d\eta_k + B_dv_k + E_d(\psi_k + w_k) . \tag{16}$$

We will use (16) as the model to design the feedback component of the control. This is clearly an approximate model of the actual discrete dynamics. Since the feedback design will be done by characterizing the terms ψ_k and w_k in a conservative manner, the design results are expected to be valid for the actual system.

The feedback control is given by

$$v(t) = \begin{cases} K(\xi(t_k) - x(t_k)), & t \in (t_k, t_k + t_f]; \\ 0, & t \in (t_k + t_f, t_{k+1}]. \end{cases} \qquad k = 0, 1, 2, \dots , \tag{17}$$

where K is a constant feedback gain matrix. We introduce the following assumptions for the design of the feedback gain matrix.

Assumption 1. *The derivative of the real gravity ϕ is in a convex and close set $\mathcal{G}$, i.e.*

$$\frac{\partial \phi}{\partial q}(q) \in \mathcal{G}\,, \qquad \forall\, q\,. \tag{18}$$

Assumption 2. *There is a known bound on the accuracy of the real vector, i.e. there exists $\sigma > 0$*

$$\|\phi(q) - g(q)\| \leq \sigma\,, \qquad \forall\, q\,. \tag{19}$$

Note that both vector q in above assumptions represent the position vector relative to a rotating frame of reference. In our model, the origin of this frame is taken as the center-of-mass of the small body.

In reality, it is not possible to satisfy the conditions given in Assumption 2 and 1 everywhere in position-space. This can easily be seen by considering the gravity field in the closed neighborhood of the center-of-mass of the small body. The justification of these assumptions comes from the fact that the proximity operations around a small body take place in a restricted region in position-space. The feed-forward control design includes position constraints that will guarantee that nominal trajectory lives in a bounded region away from the center-of-mass, call it $\mathcal{X}_o$ (not to collide with the small body). Then we will guarantee that the actual trajectories live in an invariant ellipsoid, ε_P, around the nominal trajectories by assuming that all the actual trajectories are in some bounded region $\mathcal{X}$ in the position-space. Here set $\mathcal{X}$ satisfies that $\mathcal{X}_o + \varepsilon_P \subseteq \mathcal{X}$, where $+$ between sets forms a new set that is obtained by the vectorial sum of the elements in both sets. Then, we find the bounds σ and $\mathcal{G}$ in Assumption 2 and 1 for set $\mathcal{X}$. Then, the analysis presented later will be valid for all trajectories starting in $\mathcal{X}$. The following describes a general form of the geometrical constraints on the invariant ellipsoid ε_P in order to guarantee the satisfaction of $\mathcal{X}_o + \varepsilon_P \subseteq \mathcal{X}$,

$$\varepsilon_P \subseteq (\Gamma_1 \cap \ldots \cap \Gamma_{n_1}) \cap (\Lambda_1 \cap \ldots \cap \Lambda_{n_2}) \tag{20}$$

where

$$\Gamma_n = \{\eta \,:\, a_n^T \eta \leq 1\}, \quad n = 1, \ldots, n_1, \tag{21}$$

$$\Lambda_n = \{\eta \,:\, \eta^T Y_n \eta \leq 1\}, \quad n = 1, \ldots, n_2. \tag{22}$$

The following theorem establishes the basis of an LMI based feedback control design procedure to generate an invariant ellipsoid with desired properties.

Theorem 1. *Consider the discrete time system (16) satisfying Assumption 1 and Assumption 2. Furthermore assume that $\mathcal{G}$ in Assumption 1 is given by,*

$$\mathcal{G} = \{\Theta \,:\, \Theta^T \Theta \leq \gamma^2 I\}, \tag{23}$$

where $\gamma > 0$. Suppose that there exist matrices $Q = Q^T > 0$ and L, and scalars $\alpha > 0$ and $\lambda > 0$ such that the following set of matrix inequalities are satisfied,

$$\begin{bmatrix} (\lambda - 1)Q & 0 & 0 & QA_d^T + L^T B_d^T & QC_q^T \\ 0 & -\alpha I & 0 & \alpha\gamma E_d^T & 0 \\ 0 & 0 & -\lambda I & \sigma E_d & 0 \\ A_d Q + B_d L & \alpha\gamma E_d & \sigma E_d & -Q & 0 \\ C_q Q & 0 & 0 & 0 & -\alpha I \end{bmatrix} \leq 0\,, \tag{24}$$

$$\begin{bmatrix} Q & L^T \\ L & V_{max}^2 I \end{bmatrix} \geq 0\,, \tag{25}$$

$$a_n^T Q a_n \leq 1\,, \quad n = 1, \ldots, n_1, \tag{26}$$

$$\begin{bmatrix} Q & Q \\ Q & Y_n^{-1} \end{bmatrix} \geq 0\,, \quad n = 1, \ldots, n_2\,, \tag{27}$$

where V_{max} is a bound on the norm of feedback control. Then, the control law given by

$$v_k = K\eta_k\,, \qquad \textit{where} \quad K = LQ^{-1}\,, \tag{28}$$

renders $\varepsilon_P = \{\eta \,:\, \eta^T Q^{-1}\eta \leq 1\}$ an invariant set for (16), i.e. $\eta_k \in \varepsilon_P,\ k = 1, 2, 3, \ldots,$ for any solution of (16) with $\eta_0 \in \varepsilon$. Additionally, ε_P satisfies 20, and

$$\|v_k\| \leq V_{max}\,, \qquad \forall\, \eta_k \in \varepsilon_P\,.$$

The matrix inequalities (25), (26), and (27) are LMIs, but (24) is a BMI (bilinear matrix inequality). However note that (24) is an LMI for a given λ. Also note that $\lambda \in [0, 1)$ in order to have a feasible solution for matrix inequality (24). Consequently an LMI based design procedure can easily be constructed by using Theorem 1 via a line search on λ as follows: Apply a line search on $\lambda \in [0\,,\ 1)$ to maximize the volume of ε_P, where for a given λ the volume of ε_P can be maximized via the following SDP [2],

$$\begin{gathered} \text{Minimize } \log(\det Q^{-1}) \\ \text{subject to } Q = Q^T > 0,\ \alpha > 0\,, \text{ and equations (24), (26), (27)}\,. \end{gathered} \tag{29}$$

4.1 Rationale for Accurate Gravity Models in Trajectory Generation

In this section, we discuss the rationale behind using accurate gravity models to achieve better performance in small body proximity operations. The main operational constraint in the operations is the required thruster silent time. In these time periods, there is no control authority to keep the spacecraft in a desired region, and the tracking error can become larger. To analyze the error, η, dynamics that is given by,

$$\dot{\eta} = A_0\eta + Bv + B[\psi + w]\,,$$

where w is the disturbance due to mismatch of the real gravity and the the model of the gravity, i.e.

$$\psi(t) = \phi(C_q x(t)) - g(C_q x(t))\,,$$

where $x(\cdot)$ defines the nominal trajectory. The assumption on this disturbance is

$$\|w\| \leq \gamma\,,$$

where $\gamma > 0$ defines the bound on the modelling error. Our current assumption is that the gravity modelling errors are in the order $1 - 5\%$ of the actual gravity. In an extreme case, we can neglect gravity for trajectory design, i.e. $g \equiv 0$. In this case, the disturbance bound γ is at least 20 times of the corresponding bound with $1 - 5\%$ modelling error. This implies significantly reduced tracking performance due to thruster silent periods. Indeed, this is an intuitive result. When an accurate gravity model is used, the trajectory is optimized such that the state of the spacecraft is favorable at

the end of a firing period, which leads to a motion ending at a desired state after the thruster silent period. However, when the trajectory is designed based on an inaccurate model, the spacecraft is forced to follow an unnatural path. Then, when the thruster activity stops, the spacecraft is not at a favorable state to track the nominal trajectory, and it can deviate significantly from it before the thruster activity is resumed. This puts an inherent limit in the tracking performance imposed by the length of thruster silent time independent of the feedback control authority.

In Section 7, a simulation is depicted in Figure 10 where a trajectory is generated by using a cubic spline for given initial and desired final state. This implies that a double-integrator nominal state dynamics is assumed (i.e. gravity as well as asteroid rotation rate are both zero). The simulation is performed with the same gain matrices used in the other simulations. The results clearly shows an unacceptable tracking performance.

5 Implementation of Feedback to Track Nominal PWG Trajectories

The feedback technique developed in Section 4 was incorporated into simulations of small-body landings that utilize the PWG scheme to generate open-loop guidance trajectories. The addition of feedback made significant improvements to landing precision when disturbance uncertainties in the gravity model were included. Gravity disturbances were generated from a random-number generator with a maximum disturbance of 5% from nominal gravity models. For the simulations, the numerical-gradient-based gravity model from subsection 3.3 was utilized as well. Further, the feedback was implemented so that feedback thruster firings occur simultaneously with open-loop firings, thrust preserving the structure of built-in thruster silence times.

Figure 7 shows the landing position is within 1 meter of desired (and velocity is nullified) when feedback is enabled to counter an unknown 5% error in the gravity model. The figure also shows the simultaneous open-loop and feedback firings, as computed from the PWG guidance and feedback schemes, respectively. Notice also that the feedback states line up with the desired waypoints, as determined through the PWG scheme. No final proximity controller is implemented in this example in order to demonstrate the precision enabled by the PWG with feedback scheme alone.

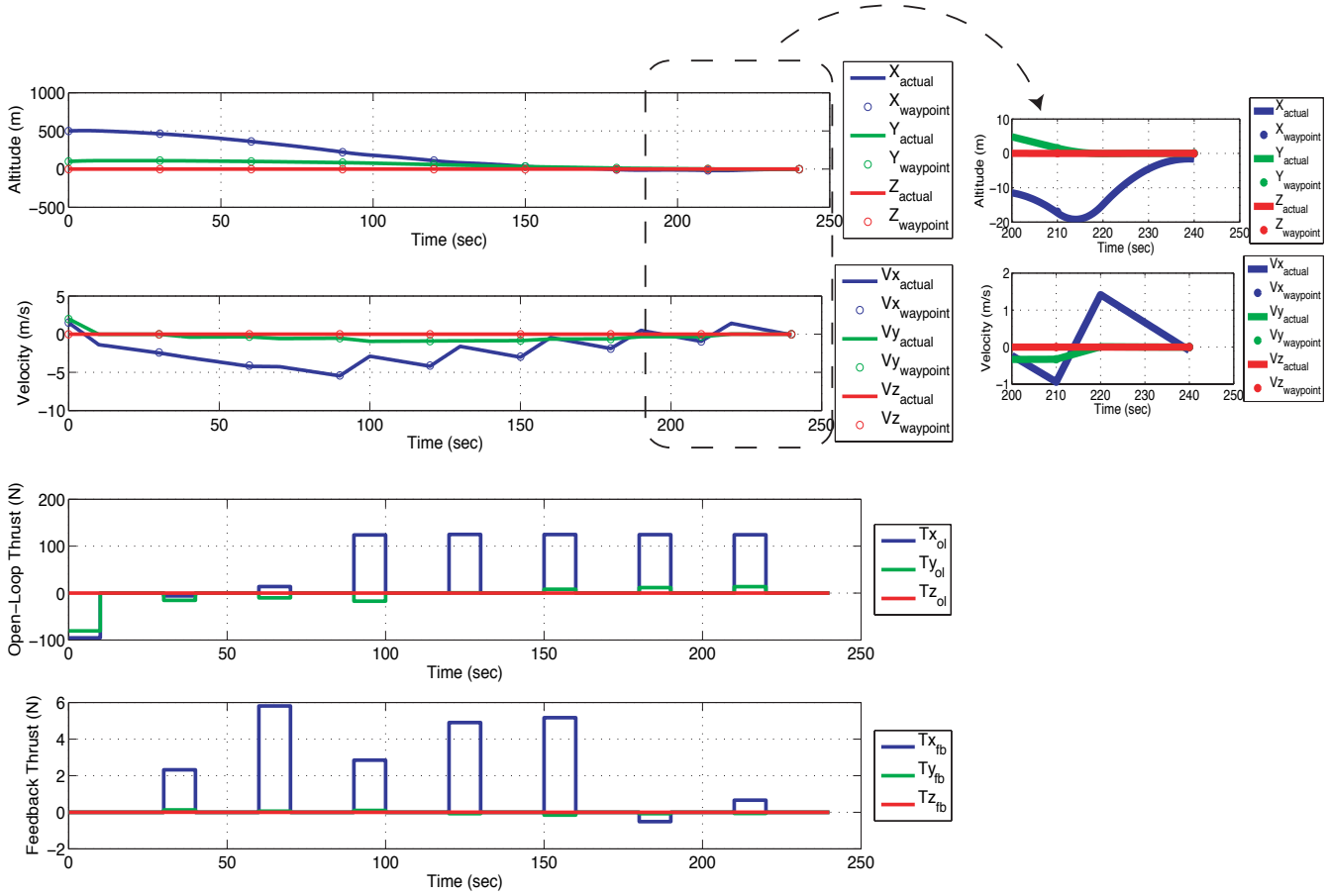

Figure 7: Feedback Designed to Handle Errors in Gravity Model

Without closing the loop through feedback, the 5% uncertainty in gravity proves detrimental in the ability to land with open-loop-only methods. As shown in figure 8, the spacecraft lands off mark by more than 75 meters and one component of velocity is off by about 0.75 meters/second; further, in the zoom of the end time, note the large error between actual X-component states and the desired waypoints.

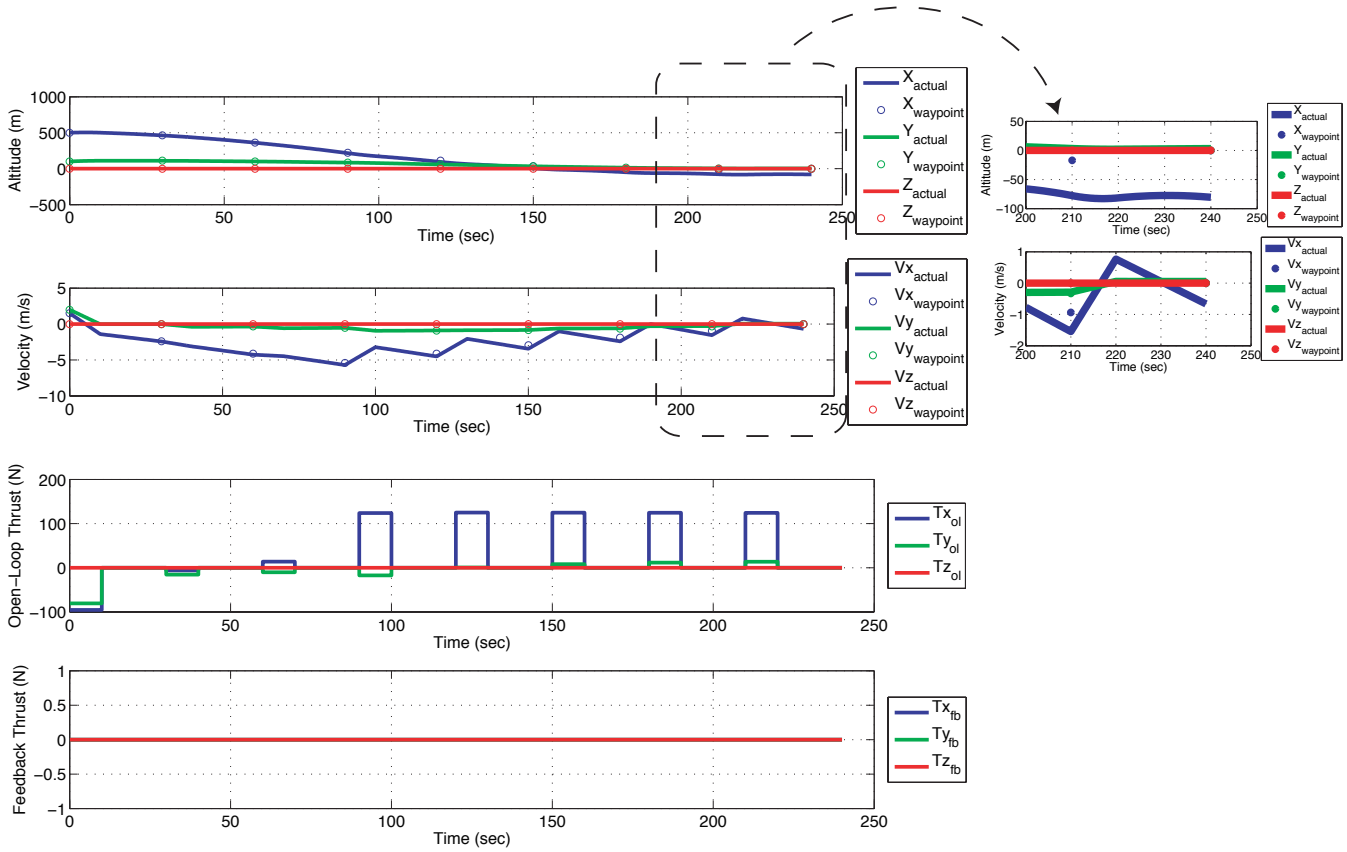

Figure 8: Open-Loop-Only Control is Incapable of Handling Errors in the Gravity Model

6 Resolvability of the Open-Loop Guidance Problem and Convergence to the Target

In this section we introduce a technique that guarantees the resolvability of the open-loop guidance problem for any time horizon. Since it is assumed that there will be finite number of open-loop trajectory computations, resolvability implies the convergence to the target, which will later be described precisely. The following assumption is needed for resolvability:

Assumption 3. *Consider A_F, B_F, E_F and g_F are parameters for (6) obtained by discretizing (7) as explained in sections 2.1 and 2.2. Assume that there exists some $u_F \in \mathcal{U}$ (see (9) for a description of $\mathcal{U}$) such that*

$$x_F = A_F x_F + B_F u_F + E_F g_F \,. \tag{30}$$

Now, suppose that a trajectory is computed via PWG algorithm, and $\{x_{m,0}, \ldots, x_{m,N-1}\}$ is generated. Furthermore, the feedback control law (28) with gain K computed as explained in Theorem 1 is applied to track $\{x_{m,0}, \ldots, x_{m,N-1}\}$. Consider re-execution of the PWG algorithm at a time corresponding to time index n, $0 < n < N-1$, of the previous solution, with an initial reference trajectory $\{x_{m,n}, \ldots, x_{m,N-1}, x_F, \ldots, x_F\}$.
In this case, $\{x_{m,n}, \ldots, x_{m,N-1}, x_F, \ldots, x_F\}$ also defines a feasible solution to the optimization prob-

lem (9) given Assumption 3. As a result, one can guarantee resolvability by using $\{x_{m,n}, \ldots, x_{m,N-1}, x_F, \ldots, x_F\}$ as the initial reference trajectory, and replacing the initial condition constraint $x_{1,0} = x_S$, wherex_S is the measured state at the time of re-computation of the trajectory, with

$$(x_{1,0} - x_S)^T P(x_{1,0} - x_S) \leq 1\,, \tag{31}$$

with $P = Q^{-1}$ where $Q = Q^T > 0$ is the matrix defined in Theorem 1. This follows from the fact that $(x_{m,n} - x_S)^T P(x_{m,n} - x_S) \leq 1$, where $x_{m,n}$ is the first state in the initial reference trajectory for re-computations (which is a result of Theorem 1). In later iterations in PWG algorithm, we use the following constraint on the initial condition,

$$x_{j,0} = x_{1,0}\,, \qquad j = 2, \ldots, m. \tag{32}$$

Consequently, we obtain resolvability of the open-loop problem once an initial trajectory is generated by PWG algorithm. Since we resolve the open-loop problem for finite number of times, there exists a last trajectory generated, that will be followed till reaching x_F. Since feedback action guarantees a tracking of this trajectory with an *error ellipsoid* described by $\varepsilon_P = \{\eta \,:\, \eta^T P \eta \leq 1\}$, where η is the vector for tracking error, the actual trajectory will converge to ε_P neighborhood of x_F, i.e. there exists some time T_f such that

$$\left(\xi(k\Delta t) - x_F\right)^T P\left(\xi(k\Delta t) - x_F\right) \leq 1\,, \qquad \forall\; k\Delta t \geq T_f\,,$$

where ξ is the actual (measured) system state.

7 Example Simulations

Example simulations are presented in this section. In these simulations, we use E_e and E_c in PWG as defined in Remark 2. The only state constraints is that the x_1 coordinate of the position must remain larger than 8445 meters, $x = 8445$ meters is 5 meters below the target point which also defines a point on the surface of Eros. To satisfy this constraint, a position constraint $x_1(t) \geq 8455$ is imposed at all time nodes except at the end point. First simulation is given in Figure 9, where a single trajectory is obtained by solving PWG and followed by the feedback control. The results show that a feasible solution is obtained with a 2.4 kg fuel use. The same simulation is repeated in Figure 10 where a cubic position time profile is used, and the feedback control loop is closed around it. The results clearly show that the feedback action is not sufficient in the cubic case to generate a feasible solution, and the spacecraft collides with Eros (see Section 4.1 for a discussion). The third simulation results are given by Figure 11, where a shrinking horizon solution approach is used, and the open-loop trajectory is updated on the way to the target at every two samples in time; this example demonstrates resolvability, as discussed in Section 6. Again, we obtain a feasible solution to the problem with a fuel use of 2.3 kg. The fuel gain relative to the single trajectory solution is not significant. The actual trajectory stays longer at a higher altitude than the final position when compared with the single trajectory solution.

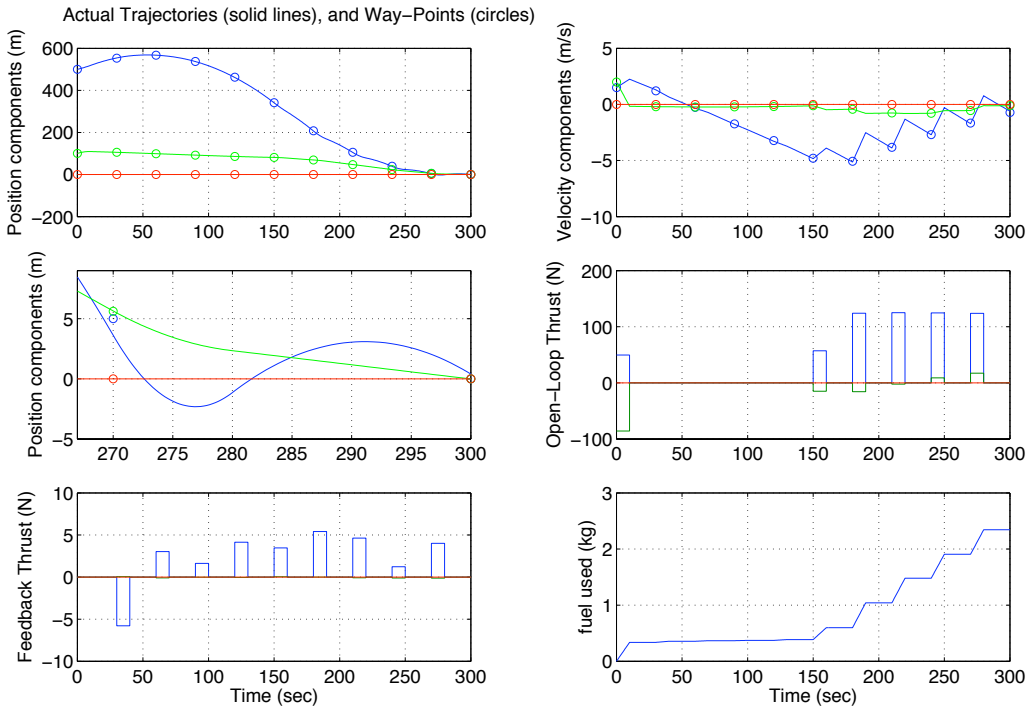

Figure 9: A Single Trajectory Solution

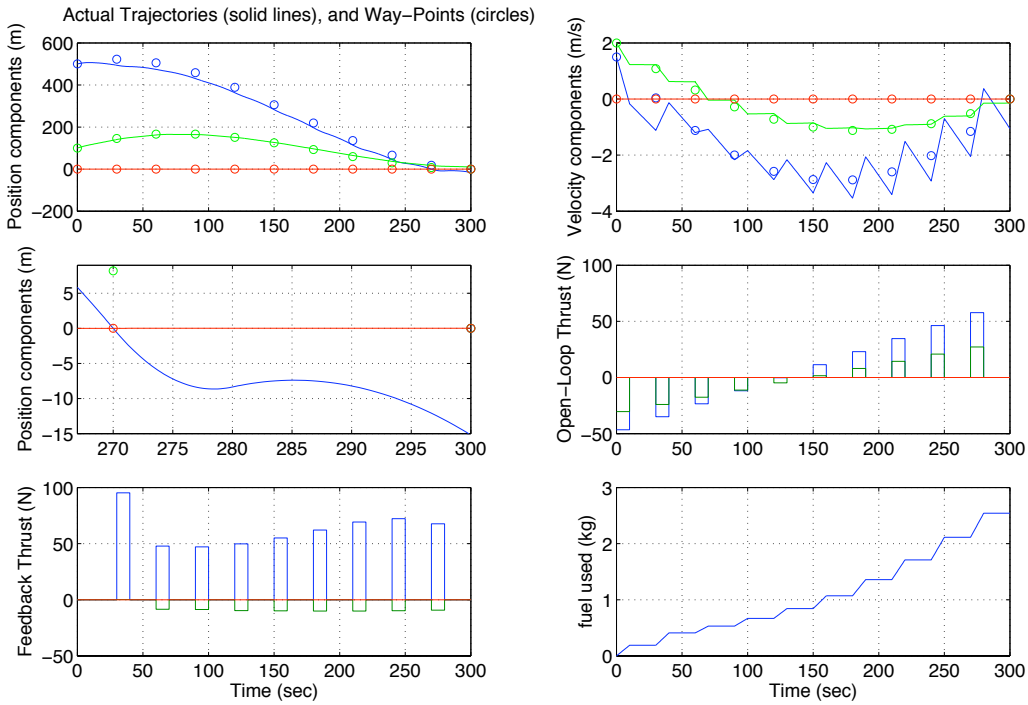

Figure 10: Cubic Trajectory Solution

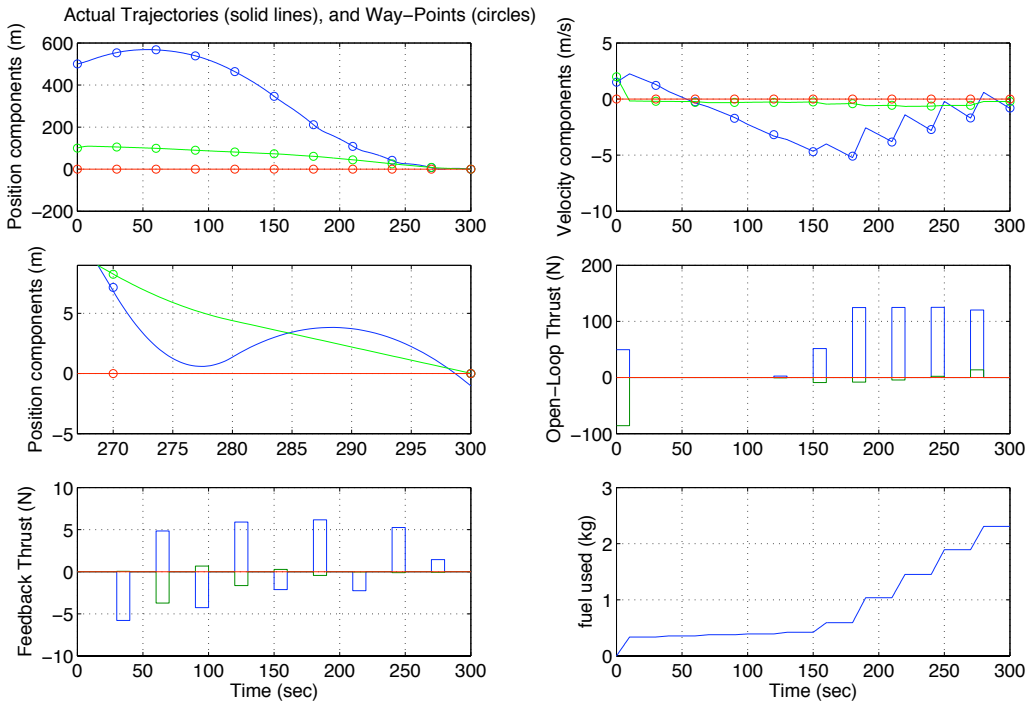

Figure 11: Shrinking Horizon Solution

8 Conclusions

The PWG method provides a suitable open-loop guidance method that can be augmented with an additional feedback controller or be implemented in receding horizon control with guarantees on resolvability. When linearizations are available for gravity models, the PWG technique provides very accurate open-loop guidance when disturbances are not present. In addition, when the gravity linearizations are not available, numerical gradients still provide a very solid means of computing open-loop guidance with the PWG method. This technique is widely applicable for generating the open-loop guidance portion of maneuvers for any body (planet, moon, small-body) and is valid for any form of maneuver (trajectory transfer, landing, hoping, and take-off). Wrapping feedback around the PWG technique allows for significant robustness to uncertainty, and the feedback technique can be designed to respect the built-in thruster silent times of the PWG scheme.

Appendix

A Proof of Theorem 1

We need the following lemma in order to prove Theorem 1.

Lemma 1. *Consider the discrete time system (16) satisfying Assumption 1 and Assumption 2. Furthermore assume that $\mathcal{G}$ in Assumption 1 is given by (23). Suppose that there exists some*

$P = P^T > 0$ and associated Lyapunov function $V_k = \eta_k^T P \eta_k$ such that

$$V_{k+1} - V_k + \lambda \left(V_k - \frac{w_k^T w_k}{\sigma^2} \right) + \beta(\gamma^2 \eta_k C_q^T C_q \eta_k - \psi_k^T \psi_k) \leq 0 \,, \quad \forall\, \eta_k,\ \forall\, w_k,\ \forall\, \psi_k, \tag{33}$$

with some $\lambda \in [0\,,\ 1)$ and $\beta > 0$. Then $\varepsilon_P = \{\eta\ :\ \eta^T P \eta \leq 1\}$ is an invariant set for (16), i.e. if the initial state is in ε_P, so is the rest of the trajectory.

Proof. First, since $\mathcal{G}$ is a closed and convex set [3], for any ψ there exists η and $G \in \mathcal{G}$ such that

$$\psi_k = G \eta_k \,.$$

This implies that

$$\psi_k^T \psi_k = \eta_k^T G^T G \eta_k \leq \gamma^2 \eta_k^T \eta_k \,.$$

Note that ε_P is an invariant set for (16) if

$$V_{k+1} \leq 1 \,, \quad \forall\, V_k \leq 1, \ \ \forall\, w_k^T w_k \leq \sigma^2, \ \text{ and } \ \forall\, \gamma^2 \eta_k^T C_q^T C_q \eta_k \geq \psi_k^T \psi_k \,. \tag{34}$$

By applying the S-procedure [2], the existence of positive scalars c_1, c_2, and c_3 satisfying the following inequality is sufficient for the satisfaction of the inequalities in (34),

$$V_{k+1} - 1 + c_1(1 - V_k) + c_2(\sigma^2 - w_k^T w_k) + c_3(\gamma^2 \eta_k^T C_q^T C_q \eta_k - \psi_k^T \psi_k) \leq 0 \,, \quad \forall\, \eta_k,\ \forall\, w_k,\ \forall\, \psi_k \,.$$

This inequality is equivalent to

$$V_{k+1} - c_1 V_k - c_2 w_k^T w_k - (1 - c_1 - c_2 \sigma^2) + c_3(\gamma^2 \eta_k^T C_q^T C_q \eta_k - \psi_k^T \psi_k) \leq 0 \,, \quad \forall\, \eta_k,\ \forall\, w_k,\ \forall\, \psi_k \,. \tag{35}$$

Note that inequality (33) can equivalently be written as

$$V_{k+1} - (1 - \sigma^2 \lambda) V_k - \lambda w_k^T w_k + \beta(\gamma^2 \eta_k C_q^T C_q \eta_k - \psi_k^T \psi_k) \leq 0 \,, \quad \forall\, \eta_k,\ \forall\, w_k,\ \forall\, \psi_k \,.$$

Letting $c_1 = 1 - \lambda$ and $c_2 = \frac{\lambda}{\sigma^2}$, the above inequality implies the inequality (35), which completes the proof. □

Proof of Theorem 1. Inequality (33) gives a sufficient condition to have ε_P an invariant set for system in (16), and it is equivalent to the following matrix inequality for the closed loop system with $v_k = K \eta_k$,

$$\begin{bmatrix} A_c^T P A_c - (1-\lambda) P + \beta C_q^T C_q & \gamma A_c^T P E_d & \sigma A_c^T P E_d \\ \gamma E_d^T P A_c & -\beta I + \gamma^2 E_d^T P E_d & \gamma \sigma E_d^T P E_d \\ \sigma E_d^T P A_c & \gamma \sigma E_d^T P E_d & -\lambda I + \sigma^2 E_d^T P E_d \end{bmatrix} \leq 0 \,,$$

where $A_c = A_d + B_d K$. This inequality is equivalent to,

$$\begin{bmatrix} A_c^T P A_c - (1-\lambda) P + \beta C_q^T C_q & 0 & 0 \\ 0 & -\beta I & 0 \\ 0 & 0 & -\lambda I \end{bmatrix} + \begin{bmatrix} A_c^T \\ \gamma E_d^T \\ \sigma E_d^T \end{bmatrix} P \begin{pmatrix} A_c & \gamma E_d & \sigma E_d \end{pmatrix} \leq 0 \,.$$

By using Schur complements [4], the inequality above is equivalent to,

$$\begin{bmatrix} -(1-\lambda)P + \beta C_q^T C_q & 0 & 0 & A_c^T \\ 0 & -\beta I & 0 & \gamma E_d^T \\ 0 & 0 & -\lambda I & \sigma E_d^T \\ A_c & \gamma E_d & \sigma E_d 1 & -P^{-1} \end{bmatrix} \leq 0\,. \tag{36}$$

Pre and post-multiplying inequality (36) by

$$diag([P^{-1},\, I,\, I,\, I]),$$

and letting $Q = P^{-1}$ and $L = KP^{-1}$, we obtain the following equivalent characterization of inequality (36) as,

$$\begin{bmatrix} -(1-\lambda)Q + \beta A C_q^T C_q Q & 0 & 0 & QA_d^T + L^T B_d^T \\ 0 & -\beta I & 0 & \gamma E_d^T \\ 0 & 0 & -\lambda I & \sigma F_d^T \\ A_d Q + B_d L & \gamma E_d & \sigma F_d & -Q \end{bmatrix} \leq 0\,.$$

Applying Schur complements one more time, letting $\alpha = \frac{1}{\beta}$, we obtain

$$\begin{bmatrix} -(1-\lambda)Q & 0 & 0 & QA_d^T + L^T B_d^T & QC_q^T \\ 0 & -\frac{1}{\alpha} I & 0 & \gamma E_d^T & 0 \\ 0 & 0 & -\lambda I & \sigma F_d^T & 0 \\ A_d Q + B_d L & \gamma E_d & \sigma F_d & -Q & 0 \\ C_q Q & 0 & 0 & 0 & -\alpha I \end{bmatrix} \leq 0\,.$$

Now, by pre and post-multiplying the inequality above with

$$diag([I,\, \alpha I,\, I,\, I,\, I])$$

it can be shown to be equivalent to inequality (24). Consequently satisfaction of (24) guarantees that ε_P is invariant set for (16). Once invariance of ε_P is established, inequality (25) guarantees that $\|K\eta_k\| \leq V_{max}\,, \quad \forall\, \eta_k \in \varepsilon_P$ [2].

The polytope described by (21) contains ε_P if and only if (26) are satisfied. Also, the region described by (22) contains ε_P if and only if

$$Q^{-1} - Y_n \geq 0\,, \qquad n = 1, \ldots, n_2\,.$$

By pre and post-multiplying this inequality by Q, it is shown to be equivalent to

$$Q - QY_nQ \geq 0\,,$$

which is equivalent to (27) by applying Schur complements. As a result, the satisfaction of inequalities (26) and (27) guarantee the satisfaction of (20). This completes the proof of the theorem. □

Acknowledgments

This research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration, and funded through the internal Research and Technology Development program.

References

[1] R.R. Bate, D.D. Mueller, and J.E. White. *Fundamentals of Astrodynamics*. Dover Publications, 1971.

[2] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory*. SIAM, 1994.

[3] L.P. D'Alto. Incremental quadratic stability. Master's thesis, Purdue University, 2004.

[4] R.A. Horn and C.R. Johnson. *Matrix Analysis*. Cambridge University Press, 1985.

[5] D.K. Yeomans, P.G. Antreasian, J.-P. Barriot, S.R. Chesley, D.W. Dunham, R.W. Farquhar, J.D. Giorgini, C.E. Helfrich, A.S. Konopliv, J.V. McAdams, J.K. Miller, W.M. Owen Jr., D.J. Scheeres, P.C. Thomas, J. Veverka, and B.G. Williams. Radio science results during the NEAR-Shoemaker spacecraft rendezvous with Eros. *Science*, 289:2085–2088, September 2000.