

MULTI-OBJECTIVE ONLINE INITIALIZATION OF SPACECRAFT FORMATIONS

Matthew Jeffrey*, Louis Breger† and Jonathan P. How‡

Massachusetts Institute of Technology

This paper extends a previously developed method for finding spacecraft initial conditions (ICs) that minimize the drift resulting from J_2 disturbances while also minimizing the fuel required to attain those ICs. It generalizes the single spacecraft optimization to a formation-wide optimization valid for an arbitrary number of vehicles. Additionally, the desired locations of the spacecraft, separate from the starting location, can be specified, either with respect to a reference orbit, or relative to the other spacecraft in the formation. The three objectives (minimize drift, minimize fuel, and maintain a geometric template) are expressed as competing costs in a linear optimization, and are traded against one another through the use of scalar weights. By carefully selecting these weights and re-initializing the formation at regular intervals, a closed-loop, formation-wide control system is created. This control system can be used to reconfigure the formations on the fly, and creates fuel-efficient plans by placing the spacecraft in semi-invariant orbits. The overall approach is demonstrated through nonlinear simulations for two formations – a GEO orbit, and an elliptical orbit.

I. INTRODUCTION

Spacecraft formation flying is expected to be an enabling technology for many future missions [1]. In a spacecraft formation, it is critically important both to conserve fuel when maneuvering and to maneuver to a state that will, over time, conserve fuel. The latter is an initial condition (IC) problem, the specifications of which will depend upon the unique requirements of a particular mission. However, in any spacecraft formation, a primary goal will be to prevent the vehicles from drifting apart, since that will typically end the mission. If the spacecraft in a formation tend to drift apart, then periodic maintenance maneuvers will be required to restore the formation. Initial conditions are called *invariant* if they eliminate drift, thereby allowing spacecraft to maintain their relative orbits without expending fuel. Two expected sources of relative drift in a formation are mismatched semimajor axes and J_2 disturbances. This paper presents an online method for finding spacecraft formation initial conditions that not only achieve mission objectives and minimize the drift resulting from natural dynamics, but also minimize the fuel required to attain those ICs.

References [2] and [3] present approaches for achieving J_2 invariance based on solving necessary conditions for partial invariance of the mean orbital elements. These methods minimize secular drift in the mean orbital elements, but contain unspecified degrees of freedom. Previous work [4, 5] introduced an optimization-based approach for finding invariant orbits that can minimize more general definitions of drift (*e.g.*, Cartesian separation, osculating states) over arbitrary time frames while minimizing fuel and considering performance objectives. The optimizations used linear programs [6], which are fast and reliable, in combination with the J_2 -modified state transition matrices presented in Ref. [7]. This approach yields results that can be used online to optimize and re-optimize initial conditions for Earth-orbiting formation flying missions (*e.g.*, LEO, HEO) that require drift to be minimized, but that also have particular geometry requirements on separation distance or shape (*e.g.*, MMS [8]). This paper extends that method by separating the desired geometry from the starting geometry, and by including multiple spacecraft in a single optimization. This enables a fully formation-wide approach to initialization and control.

*Aerospace Controls Lab, MIT, Cambridge, MA 02139, mjeffre4@mit.edu

†Aerospace Controls Lab, MIT, Cambridge, MA 02139, lbreger@mit.edu

‡Professor, Dept. of Aeronautics and Astronautics, MIT, Cambridge, MA 02139, jhow@mit.edu

II. FORMATION OPTIMIZATION

Reference [4] derived the single spacecraft IC optimization

$$C^* = \min_U \{Q_d \|W_d(\bar{\Phi}_{D1}^* - I)(\bar{\Phi}_{D0}^* \delta \mathbf{e}(t_0) + \hat{\Gamma}U)\| + Q_x \|W_x \hat{\Gamma}U\| + Q_u \|U\|\} \quad (1)$$

where C^* is the optimal cost, U is a column vector of control inputs, $\delta \mathbf{e}$ is a differential column vector of osculating orbital elements, $\hat{\Gamma}$ is a discrete convolution matrix, and the weighting matrices W_d and W_x specify the form of the penalties for drift and geometry, respectively. The scalar weights Q_d , Q_x , and Q_u determine the penalties on drifting, variation from a geometric template (*i.e.*, performance), and fuel use. The state transition matrix, $\bar{\Phi}_{Dn}^*$, propagates the differential state vector from time t_n to time t_{n+1} . Additionally, t_0 is the time at the start of the optimization, t_1 is the time at the end of the initialization maneuver, and $[t_1, t_2]$ is the period over which drift is minimized (in this paper, one orbit). The control sequence $U(\tau)$, $\tau \in [t_0, t_1]$ is of the form

$$U = \begin{bmatrix} \mathbf{u}_0^T & \mathbf{u}_1^T & \cdots & \mathbf{u}_n^T \end{bmatrix}^T \quad (2)$$

where n is the number of steps in the plan. Moreover, each entry \mathbf{u}_k is a three element vector

$$\mathbf{u}_k = \begin{bmatrix} u_{x_k} & u_{y_k} & u_{z_k} \end{bmatrix}^T \quad (3)$$

where the subscripts x , y , and z denote the radial, in-track, and cross-track directions in the LVLH frame. Since control is not allowed during the drift period, a good solution to the optimization problem must balance the desire to achieve low-drift initial conditions at t_1 with the desire to preserve formation geometry and conserve fuel. Writing the three costs separately as C_d , C_x and C_u , the overall cost for a given control sequence U becomes

$$C = Q_d C_d + Q_x C_x + Q_u C_u \quad (4)$$

Separating the Desired State

The optimization in Eq. 1 can be improved by separating the initial state, $\delta \mathbf{e}(t_0)$, from the desired state. This is done by introducing the vector $\delta \mathbf{e}_d(t_1)$ for the desired geometry at the end of the initialization maneuver. The difference between the desired position and the actual position at t_1 is penalized

$$C_x = \|W_x(\delta \mathbf{e}_d(t_1) - \delta \mathbf{e}(t_1))\| \quad (5)$$

Substituting $\delta \mathbf{e}(t_1) = \bar{\Phi}_{D0}^* \delta \mathbf{e}(t_0) + \hat{\Gamma}U$ gives a new geometry cost

$$C_x = \|W_x(\delta \mathbf{e}_d(t_1) - \bar{\Phi}_{D0}^* \delta \mathbf{e}(t_0) - \hat{\Gamma}U)\| \quad (6)$$

making the altered problem statement

$$C^* = \min_U \{Q_d \|W_d(\bar{\Phi}_{D1}^* - I)(\bar{\Phi}_{D0}^* \delta \mathbf{e}(t_0) + \hat{\Gamma}U)\| + Q_x \|W_x(\delta \mathbf{e}_d(t_1) - \bar{\Phi}_{D0}^* \delta \mathbf{e}(t_0) - \hat{\Gamma}U)\| + Q_u \|U\|\} \quad (7)$$

The independence of the starting position, $\delta \mathbf{e}(t_0)$, from the desired position, $\delta \mathbf{e}_d(t_1)$, provides the capability to design trajectories that deploy the spacecraft to some new orbit, while still minimizing drift and fuel use. This is more consistent with the concept of an initialization, where the start of a new operations phase might require the relocation of a spacecraft. It is also useful for implementing the optimization as a closed-loop controller, because in practice, the spacecraft will never exactly achieve the desired position.

Extension to the Multi-Vehicle Case

Previously, formation flight consisted of combining separate solutions to the single spacecraft optimization in Eq. 1 [4]. While this has the advantage of keeping the problem size small (less decision variables and constraints), it is unlikely that implementing independent solutions for each spacecraft will result in an optimal overall plan. For example, sometimes it may be advantageous for one spacecraft to sacrifice individual

optimality (*e.g.*, one spacecraft uses more fuel so that all of the others can use less). To extend the formulation to the multi-vehicle case, introduce the formation state $\delta\tilde{\mathbf{e}}$, a desired formation state $\delta\tilde{\mathbf{e}}_d$, and a formation plan \tilde{U} as

$$\begin{aligned}\delta\tilde{\mathbf{e}} &= \begin{bmatrix} \delta\mathbf{e}_1^T & \delta\mathbf{e}_2^T & \cdots & \delta\mathbf{e}_N^T \end{bmatrix}^T, & \delta\tilde{\mathbf{e}}_d &= \begin{bmatrix} \delta\mathbf{e}_{d1}^T & \delta\mathbf{e}_{d2}^T & \cdots & \delta\mathbf{e}_{dN}^T \end{bmatrix}^T, \\ \tilde{U} &= \begin{bmatrix} U_1^T & U_2^T & \cdots & U_N^T \end{bmatrix}^T\end{aligned}\quad (8)$$

where N is the number of spacecraft in the formation, $\delta\mathbf{e}_i$ is the state of the i^{th} spacecraft in the formation, $\delta\mathbf{e}_{di}$ is the desired state of the i^{th} spacecraft, and U_i is a vector of control inputs for the i^{th} spacecraft. Similarly, define the formation-wide propagation matrix from t_i to t_{i+1} as $\tilde{\Phi}_i$ and a formation-wide discrete convolution matrix $\tilde{\Gamma}$ as

$$\tilde{\Phi}_i = \begin{bmatrix} \tilde{\Phi}_{Di}^* & 0 & \cdots & 0 \\ 0 & \tilde{\Phi}_{Di}^* & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \cdots & 0 & \tilde{\Phi}_{Di}^* \end{bmatrix} \quad \tilde{\Gamma} = \begin{bmatrix} \hat{\Gamma} & 0 & \cdots & 0 \\ 0 & \hat{\Gamma} & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \cdots & 0 & \hat{\Gamma} \end{bmatrix}\quad (9)$$

Finally, the multi-spacecraft weighting matrices are

$$\tilde{W}_d = \begin{bmatrix} W_d & 0 & \cdots & 0 \\ 0 & W_d & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \cdots & 0 & W_d \end{bmatrix} \quad \tilde{W}_x = \begin{bmatrix} W_x & 0 & \cdots & 0 \\ 0 & W_x & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \cdots & 0 & W_x \end{bmatrix}\quad (10)$$

Combining Eqs. 8–9 yields the state of the entire formation at time t_1 given the state at t_0

$$\delta\tilde{\mathbf{e}}(t_1) = \tilde{\Phi}_0\delta\tilde{\mathbf{e}}(t_0) + \tilde{\Gamma}\tilde{U}\quad (11)$$

The propagation in Eq. 11 and the formation weighting matrices in Eq. 10 can be combined to form a full formation initialization optimization based on Eq. 7

$$\tilde{C}^* = \min_{\tilde{U}} \{Q_d\|\tilde{W}_d(\tilde{\Phi}_1 - I)(\tilde{\Phi}_0\delta\tilde{\mathbf{e}}(t_0) + \tilde{\Gamma}\tilde{U})\| + Q_x\|\tilde{W}_x(\delta\tilde{\mathbf{e}}_d(t_1) - \tilde{\Phi}_0\delta\tilde{\mathbf{e}}(t_0) - \tilde{\Gamma}\tilde{U})\| + Q_u\|\tilde{U}\|\}\quad (12)$$

Solving Eq. 12 yields the optimal formation-wide initialization cost, \tilde{C}^* , and produces a set of optimized control inputs for each spacecraft in the formation. These control inputs, in turn, specify the trajectories each spacecraft should follow to achieve the initial conditions. In contrast to the analytic methods for finding invariant orbits that are available in the literature, this new approach to finding formation ICs can be used to minimize drift globally across an entire formation of spacecraft.

Relative Position Cost

The primary reason for converting to a formation-wide optimization is to provide coupling between the spacecraft. An example of coupling is when a mission requires the position of one spacecraft to be controlled relative to another. Equation 12 only penalizes absolute geometry deviations from a reference orbit; penalizing relative geometry requires the trajectories to be optimized jointly. This is because a variation in the trajectory of one spacecraft will require an adjustment in the trajectory of the other, and vice-versa. This is available only if the formation-wide optimization is used. Start by defining one spacecraft as the leader, with a differential state at t_0 of $\delta\mathbf{e}_L(t_0)$. The relative position of the i^{th} spacecraft is then given by

$$\Delta\delta\mathbf{e}_i(t) = \delta\mathbf{e}_i(t) - \delta\mathbf{e}_L(t)\quad (13)$$

Using Eq. 13 and the single-spacecraft equivalent of Eq. 11 yields the desired relative position at t_1

$$\Delta\delta\mathbf{e}_{di}(t_1) = \tilde{\Phi}_{Di}^*(\delta\mathbf{e}_i(t_0) - \delta\mathbf{e}_L(t_0)) + \Gamma(U_i - U_c)\quad (14)$$

where U_c is the control plan of the leader. To penalize the deviation from the desired relative position at t_1 , $\Delta\delta\tilde{\mathbf{e}}_d$ and $\Delta\delta\tilde{\mathbf{e}}(t_1)$ are constructed in the same manner as Eq. 8 and a new term, C_r , is added to the cost function:

$$\tilde{C}_r = Q_r \|W_r(\Delta\delta\tilde{\mathbf{e}}_d(t_1) - \Delta\delta\tilde{\mathbf{e}}(t_1))\| \quad (15)$$

In this cost, Q_r is a scalar weight specifying the penalty on relative position error, and W_r is a weighting matrix specifying the form of this error. The relative geometry can be penalized in a Cartesian sense through the transformation

$$M\mathbf{x} = \delta\mathbf{e}_{\text{osc}} \rightarrow \mathbf{x} = M^{-1}\delta\mathbf{e}_{\text{osc}} \quad (16)$$

where $M(\mathbf{e}(t))$ is defined in Ref. [9] and rotates

$$\mathbf{x} = [x \ y \ z \ \dot{x} \ \dot{y} \ \dot{z}]^T \quad (17)$$

from an LVLH frame to differential osculating elements. Although the form of \tilde{C}_r is similar to \tilde{C}_x in Eq. 12, the matrices in Eq. 9 must be slightly modified to incorporate the new coupling effects. Subtract $\tilde{\Phi}_{Di}^*$ from the column of the formation propagation matrix corresponding to the leader, and subtract $\hat{\Gamma}$ from that same column in the discrete convolution matrix for the full formation. Without loss of generality, we will let $i = 1$ be the leader. The new relative formation matrices, denoted by the subscript r , then become

$$\tilde{\Phi}_{r_i} = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ -\tilde{\Phi}_{Di}^* & \tilde{\Phi}_{Di}^* & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ -\tilde{\Phi}_{Di}^* & \cdots & 0 & \tilde{\Phi}_{Di}^* \end{bmatrix} \quad \tilde{\Gamma}_r = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ -\hat{\Gamma} & \hat{\Gamma} & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ -\hat{\Gamma} & \cdots & 0 & \hat{\Gamma} \end{bmatrix} \quad (18)$$

The cancellations in the first row results in a row of zeros that automatically discards the position constraint of the leader relative to itself. In fact, the row can be omitted entirely. The new cost, \tilde{C}_r , is adjoined to the global optimization, yielding the final cost function

$$\begin{aligned} \tilde{C}^* = \min_{\tilde{U}} \{ & Q_d \|\tilde{W}_d(\tilde{\Phi}_1 - I)(\tilde{\Phi}_0\delta\tilde{\mathbf{e}}(t_0) + \hat{\Gamma}\tilde{U})\| + Q_x \|\tilde{W}_x(\delta\tilde{\mathbf{e}}_d(t_1) - \tilde{\Phi}_0\delta\tilde{\mathbf{e}}(t_0) - \hat{\Gamma}\tilde{U})\| \\ & + Q_r \|\tilde{W}_r(\Delta\delta\tilde{\mathbf{e}}_d(t_1) - \tilde{\Phi}_{r_0}\delta\tilde{\mathbf{e}}(t_0) - \hat{\Gamma}_r\tilde{U})\| + Q_u \|\tilde{U}\| \} \end{aligned} \quad (19)$$

III. IMPLEMENTATION

A MATLAB code, called the ‘‘planner,’’ converts the optimization in Eq. 19 to a linear program (LP) and solves it. The LP is formulated as

$$\begin{aligned} & \text{minimize} && \mathbf{c}^T \mathbf{u} \\ & \text{subject to} && \mathbf{A}\mathbf{u} \leq \mathbf{b} \end{aligned} \quad (20)$$

The number of constraints and decision variables scales linearly with N , so the formation-wide IC problem can grow quickly with the number of spacecraft. The structure of the problem lends itself to the use of sparse matrices for the matrices in Eq. 19 as well as in the constraint matrix \mathbf{A} that results. For a formation of 7 spacecraft, with a planning horizon of 1 orbit and a discretization of 1000 time steps per orbit, \mathbf{A} is approximately $40,000 \times 40,000$. A full matrix representation would require a prohibitive 1.6 billion entries. Fortunately, the number of nonzero entries is roughly 600,000. Therefore, by switching to sparse matrices, the required storage space in memory for the constraint matrix alone can be reduced to less than 0.05% of the full matrix equivalent. Solving LPs of this size requires a solver that can handle sparse constraint matrices. Two LP solvers were used for solving the optimizations in this paper: `linprog`, which is part of the MATLAB Optimization Toolkit, and CLP [10]. The latter has been successfully tested on problems with up to 1.5 million constraints.

To validate the control sequences generated by the planner, a number of closed-loop simulations were conducted using a commercial, high fidelity, nonlinear orbit propagator [11]. These simulations included disturbances due to Earth’s oblateness (J_2 and higher order terms), absolute and differential atmospheric drag, and point mass effects from the sun, moon, and planets. The planning horizon of the planner was

set to 1 orbit (plus 1 additional orbit of drifting), and replanning occurred after every orbit. New plans were generated using the current spacecraft locations as the starting positions in the IC optimization. The duration of the simulations varied as required, and the weighting matrices and desired geometry could vary from one iteration of the planner to the next. The planner followed a predefined maneuver sequence, and at each replan step, the planning parameters were set to appropriate values for current maneuver. The procedure for the simulation is

1. Initialize the simulation by setting the starting locations of the spacecraft.
2. Generate a formation-wide plan for the next orbit.
3. Propagate the orbit by one time step in the simulator.
4. Pass the new states of the satellites from the simulator to MATLAB.
5. Add any control responses to the states. These responses are implemented as impulsive velocity changes.
6. Repeat steps 3 to 5 for all steps in the plan.
7. At the end of the plan, adjust the planning parameters as dictated by the current state and maneuver.
8. Go to step 2 and continue in this manner until the conclusion of the simulation.

IV. RESULTS

To verify the correct operation of the cost function in Eq. 19, consider a formation of four spacecraft. Take the reference orbit to be roughly at GEO:

$$\mathbf{e}_{ref}(t_0) = \begin{bmatrix} 6.6107 & 0.005 & .01 & 0 & 0 & 0 \end{bmatrix}^T \quad (21)$$

where the elements of \mathbf{e}_{ref} are semimajor axis (normalized by the radius of Earth), eccentricity, inclination (radians), right ascension of the ascending node (radians), argument of perigee (radians), and mean anomaly (radians). The initial positions of the spacecraft are:

$$\begin{aligned} \delta \mathbf{e}_1(t_0) &= \begin{bmatrix} 3.13571 \times 10^{-5} & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\ \delta \mathbf{e}_2(t_0) &= \begin{bmatrix} -1.56785 \times 10^{-5} & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\ \delta \mathbf{e}_3(t_0) &= \begin{bmatrix} -3.13571 \times 10^{-5} & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\ \delta \mathbf{e}_4(t_0) &= \begin{bmatrix} -4.70356 \times 10^{-5} & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \end{aligned} \quad (22)$$

The only elements that differ from the reference orbit are the semimajor axes, so the spacecraft begin in a straight line along the radial direction in the LVLH frame. The first spacecraft begins at an altitude 200 meters above the reference orbit (\mathbf{e}_{ref}), the second spacecraft at an altitude 100 meters below \mathbf{e}_{ref} , the third spacecraft 200 meters below \mathbf{e}_{ref} , and the fourth spacecraft 300 meters below \mathbf{e}_{ref} . In addition, the first spacecraft is designated as the leader ($\delta \mathbf{e}_1 = \delta \mathbf{e}_L$), and the other three spacecraft as its followers. The desired geometry is specified as a combination of both the absolute and relative geometry conditions. The leader's target location is chosen to lie on the reference orbit (although this is unnecessary):

$$\delta \mathbf{e}_{d1} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \quad (23)$$

The followers are given desired LVLH states relative to the leader, using C_r .

$$\begin{aligned} \Delta \mathbf{x}_{d2} &= \begin{bmatrix} 0 & -200 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\ \Delta \mathbf{x}_{d3} &= \begin{bmatrix} 0 & -400 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\ \Delta \mathbf{x}_{d4} &= \begin{bmatrix} 0 & -600 & 0 & 0 & 0 & 0 \end{bmatrix}^T \end{aligned} \quad (24)$$

Table 1: Numerical data from the GEO simulation. Position and Velocity columns are error values.

	1st Orbit			2nd Orbit		
	Pos. (m)	Vel. (mm/s)	Fuel (mm/s)	Pos. (m)	Vel. (mm/s)	Fuel (mm/s)
Leader	29.8247	0.5488	27.2799	2.3926	0.0836	1.6001
SC2	51.5167	0.8865	18.4177	4.4724	0.0765	1.1501
SC3	73.2449	1.2562	36.8346	6.5525	0.0824	2.3009
SC4	94.8905	1.6270	55.2539	8.6285	0.0989	3.4498

	3rd Orbit			Drift Orbit		
	Pos. (m)	Vel. (mm/s)	Fuel (mm/s)	Pos. (m)	Vel. (mm/s)	Fuel (mm/s)
Leader	0.1205	0.0144	0.0877	0.2648	0.0142	0
SC2	0.2062	0.0537	0.0698	0.4804	0.0539	0
SC3	0.2919	0.1213	0.1395	0.6962	0.1215	0
SC4	0.37706	0.1890	0.2091	0.9105	0.1891	0

Here, $\Delta \mathbf{x}_{di}$ has the form in Eq. 17 with positions in meters and velocities in meters per second. Note that $\Delta \delta \mathbf{e}_{di}$ (used in Eq. 19) can be recovered by applying the rotation in Eq. 16. The weighting matrices W_x and W_r are used to select the absolute geometry penalty for the leader, and the relative geometry penalty for the followers.

The simulation had a duration of four orbits (roughly four days), with a planning horizon of one orbit. Replanning was executed after every orbit, and the linear optimization never took more than 40 seconds to solve. Control was allowed at any time during the first three orbits, but the fourth orbit was specified as a drift orbit, with no control allowed. The reason for this decision was to observe how well the planner minimized drift. The scalar weights Q_u , Q_x , Q_d , and Q_r were chosen to achieve a balance between fuel use and maintaining the target formation geometry. Drift was penalized even during periods with allowable control; if it was not, the planner would have little reason to maneuver the spacecraft to semi-invariant orbits. Instead, it would only try to minimize fuel use and deviation from the desired geometry. With a planning horizon of one orbit, this might result in short-sighted plans that, while using little fuel initially and positioning the satellites correctly at the end of the orbit, require more fuel on subsequent orbits to correct drift tendencies.

Data on the performance of the planner is tabulated in Table 1. The error values in the table are taken from the actual positions and velocities of the spacecraft during the simulation. Because the velocity error and fuel expenditure were much smaller than the position error, they are expressed in millimeters per second. Figure 1(a) shows the leader's separation from its desired position. It begins 200 meters away from the reference orbit, but moves 170 meters closer during the first orbit. Over the next two orbits, the actual position converges to within 12 centimeters of the desired position. The complete trajectory is shown in the absolute LVLH frame in Figure 1(b). Figure 2 shows the positions of the three followers relative to the leader in the LVLH frame. Because the cross-track direction is out of the plane of the maneuver, and the orbit is nearly circular, the fluctuations in Figure 2(c) are small. After one orbit, the position errors are large (between 50 and 90 meters), but they converge on the second and third orbits to within 50 centimeters. As expected, the formation uses the most fuel during the first orbit, when the in-track formation was initializing. Fuel use declines over time as the new formation stabilizes (see Figure 3). By the start of the drift orbit, each spacecraft in the formation is established on a stable, semi-invariant orbit, and the desired positions are maintained absent any control. The position and velocity errors increase only slightly over that duration. The ability to maintain the relative geometry without using thrust is advantageous because thruster activity is often discouraged for missions that perform science.

The success of the maneuver is best illustrated in the LVLH frame of Figure 2(d). The frame is centered on the leader, and the relative trajectories of the followers are shown. The switch from the initial radial alignment to the final in-track formation is visible from this perspective.

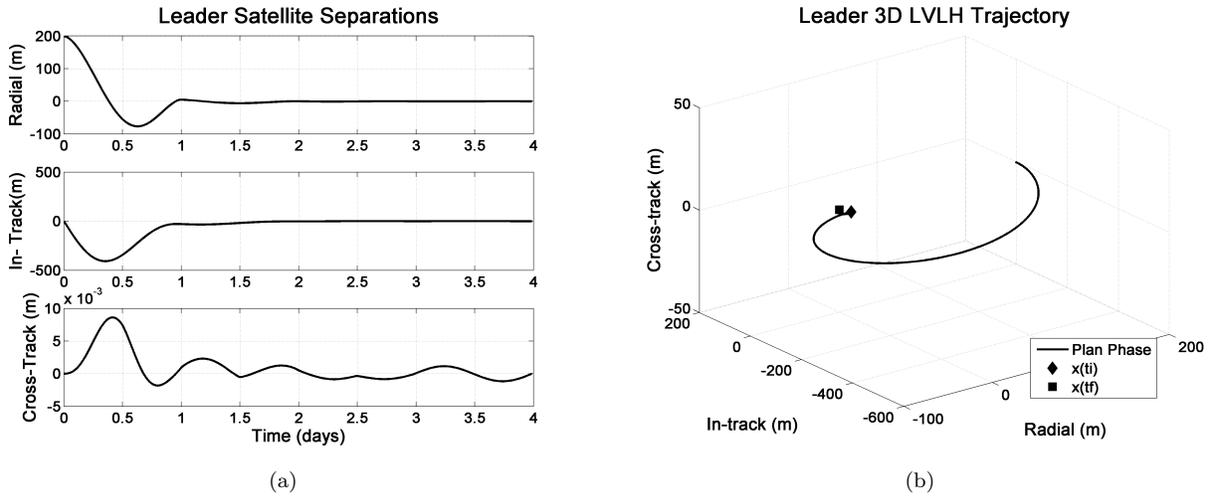


Fig. 1: The LVLH separation from the leader to the desired orbit for the GEO case.

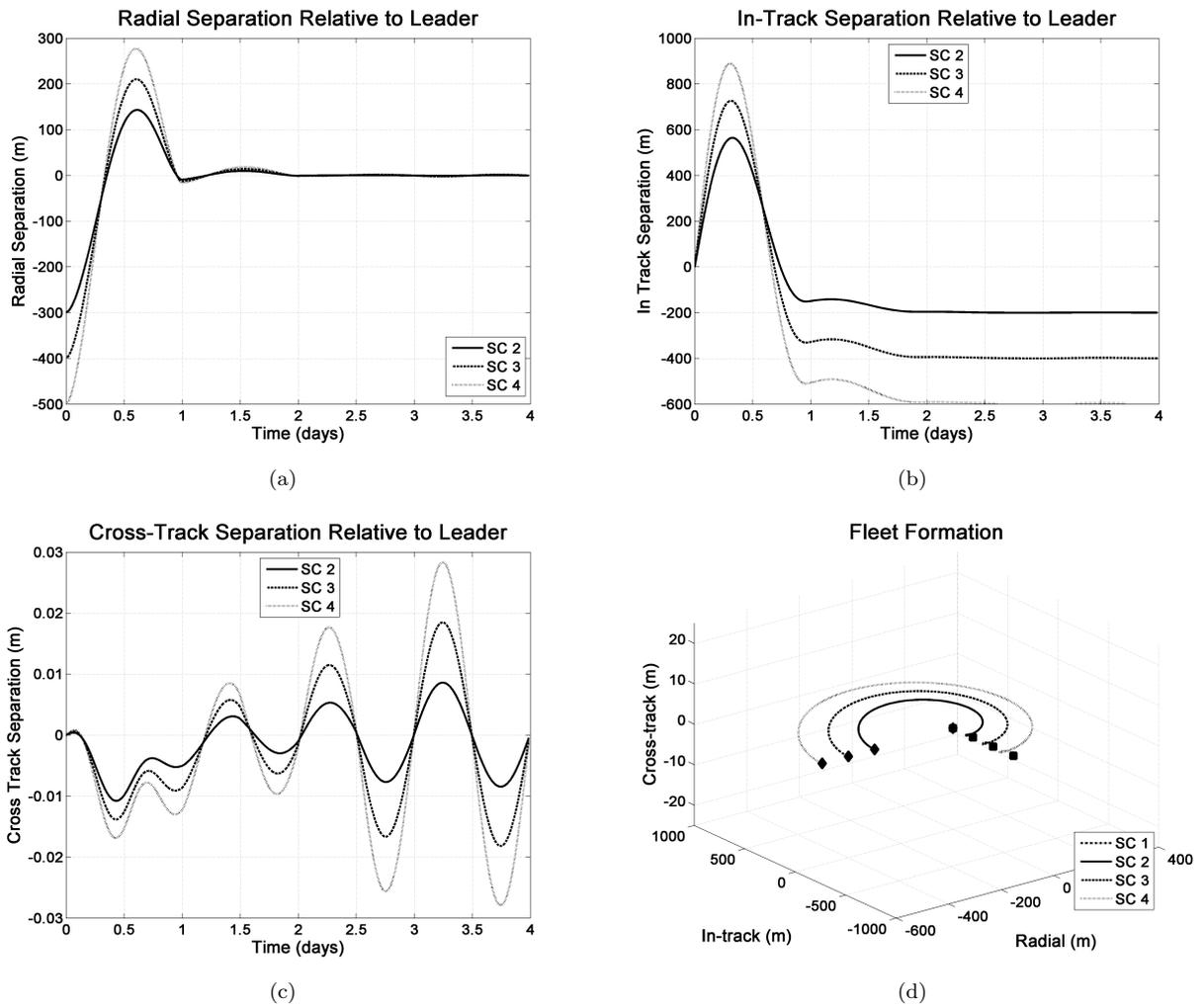


Fig. 2: The relative positions of the followers with respect to the leader for the GEO case. Figure 2(d) shows the trajectories of the spacecraft in the relative LVLH frame during the control phase. A \blacklozenge represents a location at t_0 , and a \blacksquare represents a location at the start of the drift orbit.

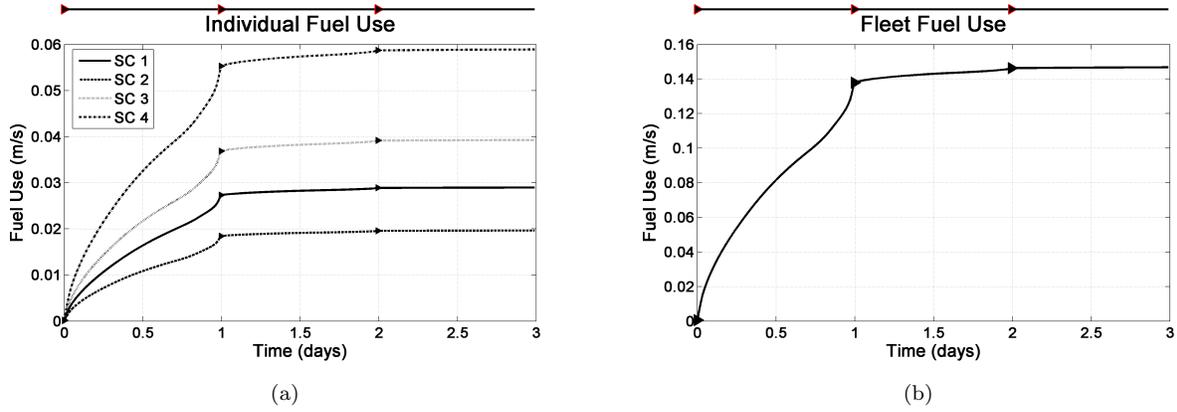


Fig. 3: Fuel use of the formation for the GEO case. A \blacktriangleright indicates a replanning point.

Control of a Formation in an Elliptical Orbit

A more difficult task is to control a formation in an elliptical orbit. One orbit of this type is

$$\mathbf{e}_{ref}(t_0) = \begin{bmatrix} 4.69549 & 0.471 & 1.10497 & 4.24115 & 3.7350 & 0 \end{bmatrix}^T \quad (25)$$

To directly compare and contrast the type of trajectories planned for a circular orbit and for an elliptical orbit, the initial and desired conditions for the elliptical simulation were set to the same values as in Eqs. 22–24. Because the elliptical orbit was expected to be harder to control, the simulation was extended to eight orbits to ensure steady-state performance was reached. The time to solve the optimizations varied between 40 and 250 seconds. With an orbit period of approximately 50,000 seconds and a step size of 50 seconds, the planner is active for at most 5 time steps per orbit. In a practical application, this would be addressed by disallowing control inputs for the first part of the orbit.

The general trends of fuel use and convergence to the desired geometry were the same for both the GEO and elliptical cases, as shown by Figures 5 and 6. However, Table 2 indicates that the actual magnitudes for fuel use and error are larger for the elliptical orbit. After the first four orbits, all spacecraft in the formation were regulated to within 1.5 meters of their desired positions. With target formation separations of 200 meters, this represents about 1% error. The error arises from two main sources. First, there is some inaccuracy that results from using linearized dynamics in the planner to control a nonlinear system. The ability to solve linear optimizations quickly (necessary for real time control) is an acceptable tradeoff. Second, the planner is not forced to drive the geometry error completely to zero because it is also trying to minimize drift and fuel use. If more accuracy is required, a smaller time step with more frequent replanning could be used. Slight improvements might also be possible by varying the weights on the individual elements of the cost function.

Once the formation is established, the fuel required to maintain it is only a few millimeters per second per orbit (see Figure 4). A characteristic that differentiates the elliptical trajectories from the GEO trajectories are the oscillations in Figure 6(b). The oscillations have a period of 1 orbit, and the peaks occur at the end of every plan. The relative separations at these peaks is as desired. The oscillations are not present in the radial and cross-track directions; this is characteristic of an eccentric orbit, where the in-track velocity varies with the true anomaly. The scenarios considered in this paper only constrain the geometry at the end of each planning period (corresponding to the start of the next orbit). The oscillations in the in-track direction could be suppressed by applying geometry constraints at regular intervals during each of the plans, or by using error-boxes, like those discussed in Ref. [6].

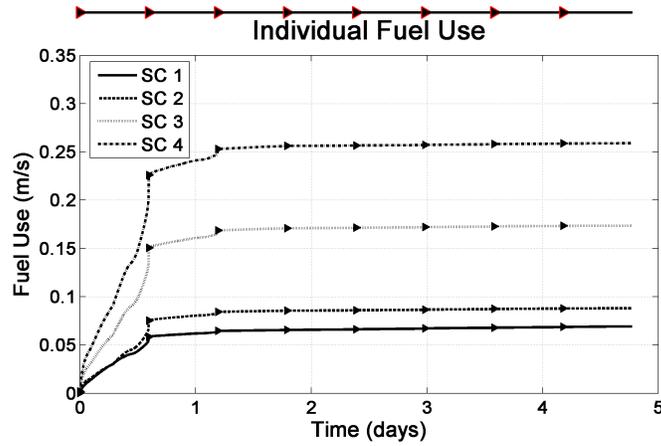
Further Uses for the Controller

Although the two examples already presented demonstrate the basic functionality of the formation-wide planner, they do not exhaust its possibilities. The initial condition optimization can be solved for formations larger than four spacecraft; an example of a seven spacecraft simulation is shown in Figure 7, and solutions

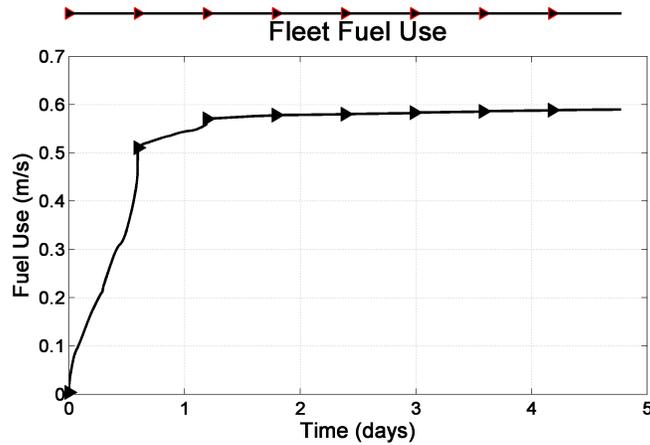
Table 2: Numerical data from the elliptical simulation. Position and Velocity columns are error values.

	1 st Orbit			2 nd Orbit		
	Pos. (m)	Vel. (mm/s)	Fuel (mm/s)	Pos. (m)	Vel. (mm/s)	Fuel (mm/s)
Leader	128.3244	15.6734	58.5065	18.2688	2.6505	6.0559
SC2	299.7792	36.3731	75.2726	50.4123	7.4583	9.0463
SC3	471.2018	57.0688	150.5417	82.5095	12.2580	18.0848
SC4	642.6396	77.7658	225.8022	114.6381	17.0617	27.1397

	3 rd Orbit			4 th Orbit		
	Pos. (m)	Vel. (mm/s)	Fuel (mm/s)	Pos. (m)	Vel. (mm/s)	Fuel (mm/s)
Leader	2.3007	0.3307	0.9258	1.4944	0.1718	0.724627
SC2	4.7470	0.7804	1.1612	2.2762	0.4906	0.454604
SC3	6.9262	1.2724	2.2395	2.0818	0.9388	0.480238
SC4	8.8307	1.7649	3.2489	1.3908	1.4090	0.377241



(a)



(b)

Fig. 4: Fuel use of the formation for the elliptical case. A ► indicates a replanning point.

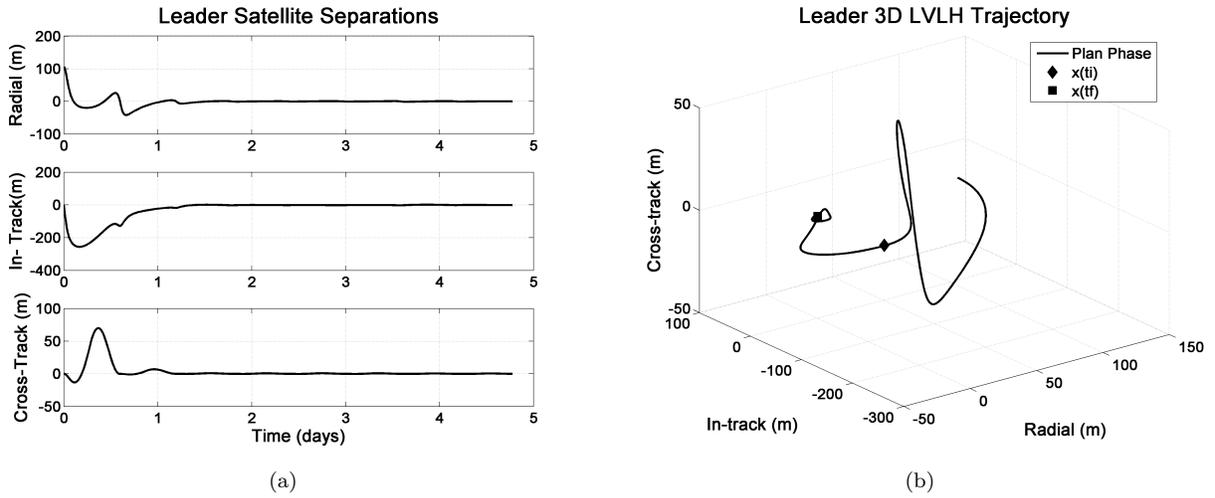


Fig. 5: The LVLH separation from the leader to the desired orbit for the elliptical case.

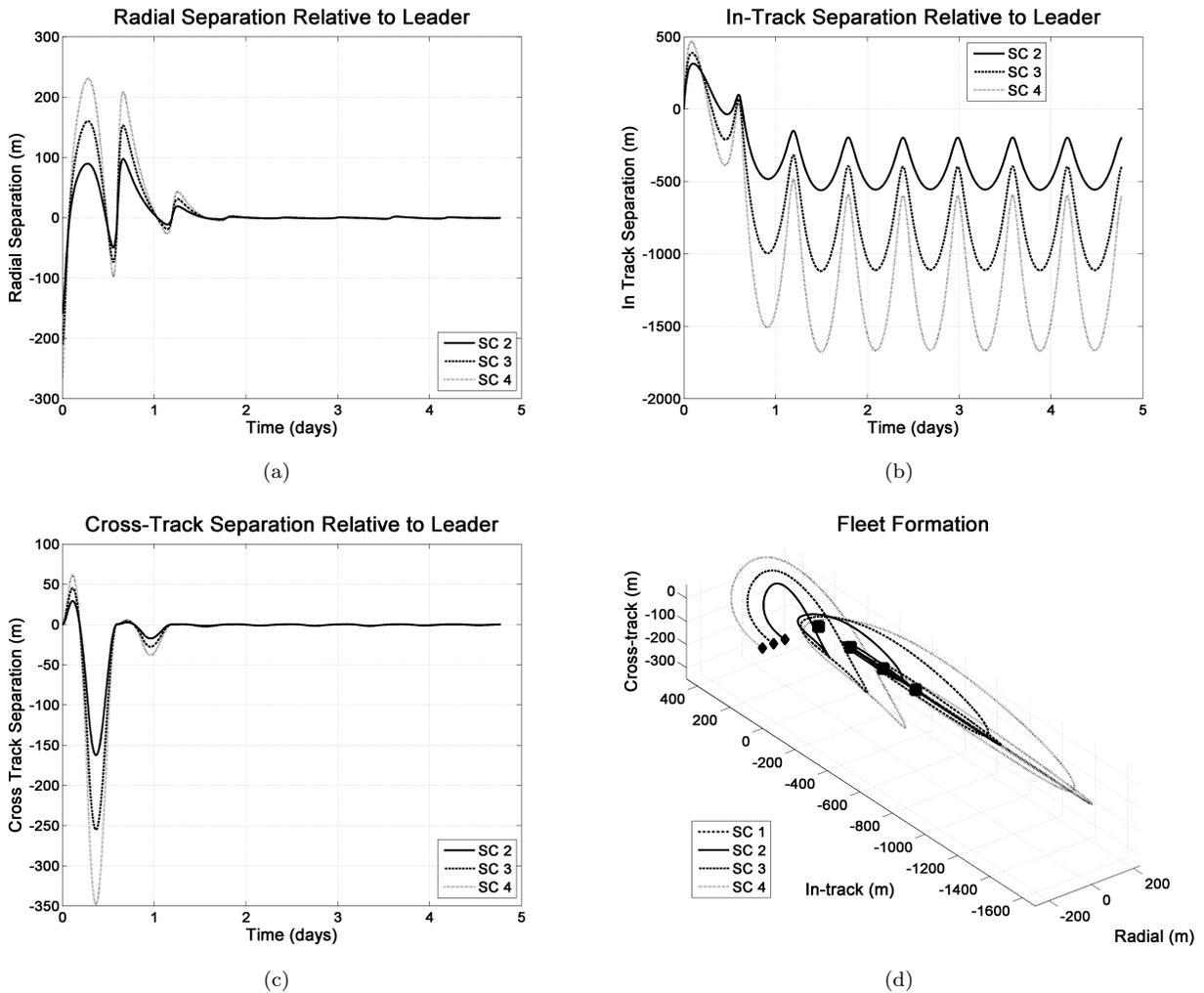
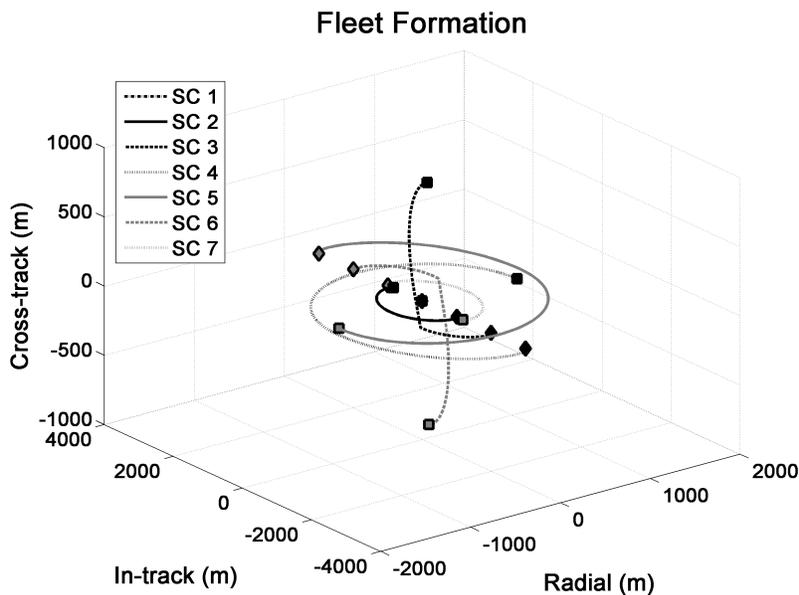
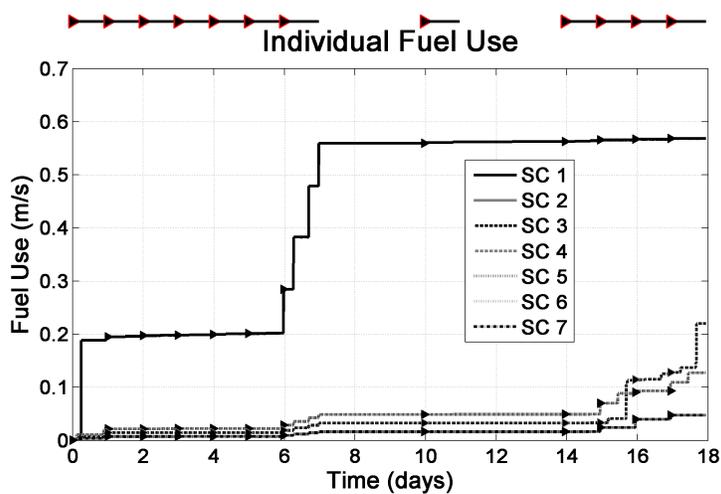


Fig. 6: The relative positions of the followers with respect to the leader for the elliptical case. Figure 6(d) shows the trajectories of the spacecraft in the relative LVLH frame for the first four orbits (corresponding to Table 2). A \blacklozenge represents a location at t_0 , and a \blacksquare represents a location at the end of the fourth orbit.

to problems with as many as 9 spacecraft have been obtained. As the formations grow, so does the solving time. Likewise, as more coupling is introduced between the spacecraft, through relative geometry constraints or otherwise, the computational difficulty increases. Eventually, the planner requires too large a fraction of the planning period to implement real-time control. This problem was not encountered for any of the simulations presented in this paper. Another possible application of the formation-wide planner is to design plans that evenly distribute fuel use across the formation. This would help prevent a formation's mission from ending prematurely due to uneven fuel use.



(a)



(b)

Fig. 7: An example reconfiguration maneuver for a seven spacecraft formation. The trajectory shown in 7(a) is taken from part of a simulation that tested several different relative geometry constraints. The depicted maneuver occurred between day 14 and 15, where the fuel use in 7(b) increases noticeably as the spacecraft are repositioned. Breaks in the horizontal line above the graph indicate drift periods.

V. CONCLUSIONS

This paper extended the linear optimization techniques that had been previously used for single spacecraft to multi-spacecraft formations. This new algorithm enables the selection of the spacecraft initial conditions (ICs) that minimize the global fuel use and drift, while considering fleet-wide performance objectives and configuration constraints. The IC optimization was also used to perform closed-loop control for the fleet, as demonstrated in a series of fully nonlinear simulations. In these simulations, the controller regulated a four spacecraft formation to within 1% of the desired vehicle separations, both in circular and elliptical orbits. Once established, these formations only required 1–3 mm/s of ΔV per orbit, even in the presence of significant disturbances.

References

- ¹J. Leitner, F. Bauer, D. Folta, M. Moreau, R. Carpenter, and J. P. How, “Distributed Spacecraft Systems Develop New GPS Capabilities,” in *GPS World: Formation Flight in Space* Feb. 2002.
- ²H. Schaub and K. T. Alfriend, “ J_2 Invariant Relative Orbits for Spacecraft Formations,” *Flight Mechanics Symposium*, (Goddard Space Flight Center, Greenbelt, Maryland), January 18-20, 2002, Paper No. 11.
- ³S. R. Vadali, K. T. Alfriend, and S. Vaddi, “Hills Equations, Mean Orbital Elements, and Formation Flying of Satellites,” American Astronautical Society, AAS 00-258, March 2000.
- ⁴L. S. Breger, J. P. How, and K. T. Alfriend, “Partial J_2 -Invariance for Spacecraft Formations,” *AIAA Guidance, Navigation, and Control Conference Conf.*, August 2006.
- ⁵L. S. Breger and J. P. How, “Gauss’s Variational Equation-Based Dynamics and Control for Formation Flying Spacecraft,” *AIAA Journal of Guidance, Control and Dynamics*, accepted for publication April 18, 2006.
- ⁶M. Tillerson, G. Inalhan, and J. P. How, “Co-ordination and control of distributed spacecraft systems using convex optimization techniques,” *International Journal of Robust and Nonlinear Control*, Vol.12, John Wiley & Sons, 2002, p. 207-242.
- ⁷D. W. Gim and K. T. Alfriend, “State Transition Matrix of Relative Motion for the Perturbed Noncircular Reference Orbit,” *AIAA Journal of Guidance, Control, and Dynamics*, vol. 26, no. 6, Nov.-Dec. 2003, p. 956-971.
- ⁸S. Curtis, “The Magnetospheric Multiscale Mission Resolving Fundamental Processes in Space Plasmas,” NASA GSFC, Greenbelt, MD, Dec. 1999. NASA/TM2000-209883.
- ⁹Schaub, Hanspeter and Junkins, John L., *Analytical Mechanics of Space Systems*, AIAA Education Series, Reston, VA, 2003.
- ¹⁰R. Lougee-Heimer, “The Common Optimization INterface for Operations Research,” *IBM Journal of Research and Development*, vol. 47(1), p. 57-66, January 2003.
- ¹¹F. Perkins, *FreeFlyer User’s Documentation*. AI Solutions: Lanham, Maryland, April 2000.