JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 31, No. 2, March–April 2008

General Method for Optimal Guidance of Spacecraft Formations

Steven P. Hughes*
NASA Goddard Space Flight Center, Greenbelt, Maryland 20771

DOI: 10.2514/1.23731

One of the most interesting and challenging aspects of formation guidance algorithm design is the coupling of the orbit design and the science return. The effectiveness of the formation as a science instrument is intimately coupled with the relative geometry and evolution of the collection of spacecraft. Therefore, the science return can be maximized by optimizing the orbit design according to a performance metric relevant to the science mission goals. In this work, we present a general method for optimal formation guidance that is applicable to missions whose performance metric, requirements, and constraints can be cast as functions that are explicitly dependent upon the orbit states and spacecraft relative positions and velocities. The approach is fully nonlinear, accommodates orbital perturbations, and is applicable to multiple flight regimes including highly eccentric, hyperbolic, interplanetary, or libration point orbits. Furthermore, the method is applicable to small formations, as well as large formations and constellations. We present a general form for the cost and constraint functions, and derive their semi-analytic gradients with respect to the formation initial conditions. The gradients are broken down into two types. The first type are gradients of the mission-specific performance metric with respect to formation geometry. The second type are derivatives of the formation geometry with respect to the orbit initial conditions. The fact that these two types of derivatives appear separately allows us to derive and implement a general framework that requires minimal modification to be applied to different missions or mission phases. To illustrate the applicability of the approach, we conclude with applications to two missions: the Magnetospheric Multiscale Mission, and the Laser Interferometer Space Antenna.

Nomenclature

$\mathbf{A}$	=	upper left 3×3 partition of $\mathbf{\Phi}$
$\mathbf{B}$	=	upper right 3×3 partition of $\mathbf{\Phi}$
$\mathbf{C}$	=	lower left 3×3 partition of $\mathbf{\Phi}$
c_k	=	quadrature constant at point k
$\mathbf{D}$	=	lower right 3×3 partition of $\mathbf{\Phi}$
m	=	no. of unique sides in the formation
N	=	no. of path constraints
n	=	no. of spacecraft
n_k	=	no. of quadrature points
$\mathbf{r}$	=	position vector
$\mathbf{s}_{ik}$	=	vector defining side i at quadrature point k
$\dot{\mathbf{s}}_{ik}$	=	rate vector of side i at quadrature point k
s_{ik}	=	length of side i at quadrature point k
$\dot{s}_{ik}$	=	rate of change in length of side i at quadrature point k
$\mathbf{v}$	=	velocity vector
$\mathbf{\Phi}$	=	orbit state transition matrix

Subscripts

C	=	indicates association with constraints
i	=	side index
J	=	indicates association with cost function
j	=	spacecraft index
k	=	quadrature point index
ℓ	=	spacecraft index
p	=	dummy index

Received 8 March 2006; revision received 28 November 2006; accepted for publication 11 December 2006.

*Aerospace Engineer, Flight Dynamics Analysis Branch.

Introduction

DEVELOPING guidance algorithms for spacecraft formations and constellations is a challenging problem that can seemingly involve mutually exclusive goals. From the perspective of a mission analyst working on orbit design for distributed spacecraft missions, it is desirable to have a flexible method that can be applied to many problems and that permits modification of the cost and constraints with as little effort as possible. We also desire an approach that can handle real-world mission constraints, find an optimal solution in a reasonable amount of time, and converge consistently with relatively poor initial guesses. The work presented in this paper was motivated by the need for an algorithm that can provide formation initial conditions for many different phases in the mission design process. These phases include, but are not limited to, developing initial conditions to use as target states to achieve after separation from the launch vehicle and separation from the spacecraft stack, developing new conditions for formation maintenance after perturbations and errors in navigation and control have caused a significant degradation in mission return, and developing new formation initial conditions for a new mission phase when the desired formation geometry is significantly different. For each of these mission phases we need to determine initial conditions for the orbits of multiple, cooperating spacecraft that optimally satisfy mission requirements and constraints. This is a long list of goals, but we can meet them all by developing effective design algorithms.

In recent years, significant effort has been devoted to all aspects of the spacecraft formation flying problem. New relative motion theories have been developed by numerous researchers. Garrison et al. [1] investigated relative motion with respect to an elliptic reference orbit. A time-explicit solution to the relative motion between elliptic orbits was presented by Melton [2]. Karlgard and Lutze [3] developed a second-order solution to the relative motion problem for a circular reference orbit. Gurfil and Kasdin [4] presented a general approach for determining higher order relative motion equations and developed a second-order solution as an example. Alfriend and Yan [5] presented a detailed comparison of several state transition matrices.

Scharf et al. [6] presented a survey of formation flying guidance research. Schaub et al. [7–10] presented methods for determining

formation initial conditions. Many researchers have developed methods that simultaneously solve the guidance and control problem. Hocken and Schoenmaekers [11] presented an optimization approach to minimize ΔV while simultaneously satisfying guidance constraints for the Cluster constellation [12,13]. Huntington et al. [14,15] used the Gauss pseudospectral method to solve the optimal control problem while satisfying guidance constraints. Richards et al. [16] employed mixed-integer linear programming to find the optimal control while satisfying avoidance constraints. Guzman [17] and Clemente and Atkins [18] developed heirarchical optimization schemes. Zanon and Campbell [19] presented a search space reduction technique to solve the coupled guidance and control problem.

This paper complements the existing body of work in the following ways: It presents a general approach to generate science-optimal initial conditions for numerous classes of formation missions. The method is illustrated by applying it to two missions in significantly different dynamic regimes and with significantly different cost and constraint functions. The method is fully nonlinear and can include orbital perturbations to any necessary degree. As a result, the approach is applicable to multiple flight regimes including highly eccentric, hyperbolic, interplanetary, or libration point orbits. Furthermore, the method is applicable to small formations, as well as large formations and constellations. The algorithm is directly applicable to all formation flying or constellation missions that impose specific requirements on the relative motion of spacecraft and the method accommodates complicated path constraints such as close approach constraints or interspacecraft separation requirements. The implementation requires little modification for applicability to new problems. Finally, the method applies especially well to the large class of missions where natural orbits must be employed because frequent, active control is prohibited due to mission requirements, power limitations, thruster plume impingement, or environmental cleanliness constraints [20,21].

The approach presented in this paper is a direct parameter optimization method which assumes that at a given instant in time, one can formulate a performance metric that is an explicit function of the inertial Cartesian states, the relative position vectors, relative velocity vectors, range and range rates of all spacecraft in formation. The performance metric allows us to quantify the effectiveness of the relative dynamics of the spacecraft at a particular point in the orbit. The cost function J is simply the integral of the instantaneous performance metric over regions of interest to the particular mission. The goal is to maximize or minimize the mission-specific cost function while simultaneously satisfying equality and inequality constraints consistent with mission requirements and real-world limitations.

We begin by posing a continuous-time set of cost and constraint functions that define a general framework for the problem. The continuous-time system is the problem we would like to solve. However, there are certain advantages to discretizing the problem. Hence, we discretize the problem to enable a nonlinear programming (NLP) routine to solve for an optimal solution. The discrete cost function can be calculated by quadrature, or a simple trapezoid rule, using numerical integration of the nonlinear orbit equations of motion, with perturbations, to give us the state of the formation at discrete times in the orbit. The constraints are calculated in a similar manner. We show how to calculate the derivative of the cost function and the Jacobian of the constraint functions using the state transition matrix (STM). The STM is found by numerically integrating an additional 36 equations per spacecraft with terms that include perturbations. There is a significant advantage to this approach. Numerical integration, while nontrivial, is a solved problem and something that modern computers perform efficiently. Secondly, as we will see in a later section, the derivatives of the cost and constraints contain two types of terms. The first type of terms are composed of portions of the STM and inertial states. The second type of terms contain derivatives of algebraic functions with respect to the formation geometry. Because the terms are separated in this way, we can implement the algorithm in a very general way so as to allow minimal modification to new problems. We conclude the paper with two applications that illustrate how the method is applied to real-world problems. The first example is the Magnetospheric Multiscale (MMS) Mission. The second example is the Laser Interferometer Space Antenna (LISA). Let us begin by defining some notation.

General Problem Statement

The reliability and effectiveness of an optimization algorithm is intimately dependent upon the parameterization of the cost and constraint functions. In this section we pose a set of cost and constraint functions for a formation of spacecraft as explicit functions of the Cartesian states of the spacecraft in formation, and the relative motion of the spacecraft. Let us begin with some definitions. Recall that our goal is to find a set of initial Cartesian states for a set of n spacecraft. The vector defining the initial Cartesian states for all spacecraft is

$$\mathbf{X}_o = \left[\mathbf{x}_{o1}^T \mathbf{x}_{o2}^T \cdots \mathbf{x}_{on}^T\right]^T \tag{1}$$

where

$$\mathbf{x}_{o1} = \begin{bmatrix}\mathbf{r}_{o1}^T & \mathbf{v}_{o1}^T\end{bmatrix}^T = \begin{bmatrix}x_{o1} & y_{o1} & z_{o1} & \dot{x}_{o1} & \dot{y}_{o1} & \dot{z}_{o1}\end{bmatrix}^T \tag{2}$$

and so on. For our implementation, we choose Earth's mean equator and equinox at the J2000 epoch as the coordinate system in which to express the Cartesian states. However, it is possible to work in other coordinate systems, including rotating systems, as long as you are consistent. We assume that the equations of motion for the spacecraft are explicit functions of the spacecraft state and time, or

$$\dot{\mathbf{X}}(t) = \mathbf{f}(\mathbf{X}, t) \tag{3}$$

Let us define the ith unique side vector of the formation $\mathbf{s}_i$ as the vector from the ℓth to the jth spacecraft

$$\mathbf{s}_i(t) = \mathbf{r}_j(t) - \mathbf{r}_\ell(t) \tag{4}$$

The subscripts j and ℓ can be chosen arbitrarily as long as for each value of i there is a given j, ℓ pair; the pairs result in a unique side, $1 \le j \le n$ and $1 \le \ell \le n$, and the chosen definition is used consistently throughout the implementation. Similarly to the side vector, the side rate vector $\dot{\mathbf{s}}_i$, the side length s_i, and the side rate $\dot{s}_i$ are defined as

$$\dot{\mathbf{s}}_i(t) = \dot{\mathbf{r}}_j(t) - \dot{\mathbf{r}}_\ell(t) \tag{5}$$

$$s_i(t) = (\mathbf{s}_i(t)^T \mathbf{s}_i(t))^{1/2} \tag{6}$$

$$\dot{s}_i(t) = \frac{\mathbf{s}_i(t)^T}{s_i(t)} \dot{\mathbf{s}}_i(t) \tag{7}$$

Let us pose a continuous-time cost function of the form

$$J(\mathbf{X}_o) = \int_{t_o}^{t_f} F_J(\mathbf{X}(t), \mathbf{s}_i(t), \dot{\mathbf{s}}_i(t), s_i(t), \dot{s}_i(t), \mathbf{C}_J)\, \mathrm{d}t \tag{8}$$

where $\mathbf{C}_J$ is a set of constants. (See Appendix A for a discussion on the coupling of terms in the cost function.) Note that we assume the cost function is an explicit function of the relative geometry of the spacecraft. In Eq. (8), F_J is the algebraic performance metric that contains information on how well the geometry satisfies mission goals at a particular instant. F_J is a mission-specific function and must be formulated according to the requirements of the mission being addressed. In a later section we present two examples for illustrative purposes. We can define nonlinear equality constraints $\mathbf{G}_E$ and nonlinear inequality constraints $\mathbf{G}_I$ as

$$\mathbf{G}_E(\mathbf{X}_o) = \mathbf{F}_E(\mathbf{X}(t), \mathbf{s}_i(t), \dot{\mathbf{s}}_i(t), s_i(t), \dot{s}_i(t), \mathbf{c}_E) = \mathbf{C}_E \tag{9}$$

$$\mathbf{G}_I(\mathbf{X}_o) = \mathbf{F}_I(\mathbf{X}(t), \mathbf{s}_i(t), \dot{\mathbf{s}}_i(t), s_i(t), \dot{s}_i(t), \mathbf{c}_I) \leq \mathbf{C}_I \tag{10}$$

where $\mathbf{C}_E$ and $\mathbf{C}_I$ are vectors of constants associated with the nonlinear equality constraints and nonlinear inequality constraints, respectively. Finally, we also impose linear equality constraints, linear inequality constraints, and bound constraints, respectively, as

$$\mathbf{A}_E\mathbf{X}_o = \mathbf{B}_E \tag{11}$$

$$\mathbf{A}_I\mathbf{X}_o \leq \mathbf{B}_I \tag{12}$$

$$\mathbf{L} \leq \mathbf{X}_o \leq \mathbf{U} \tag{13}$$

We seek a solution that minimizes J, while simultaneously satisfying the constraints in Eqs. (9–13). However, there are several issues that make this difficult, if not impossible, to solve exactly. The first is that it is not possible to develop closed form solutions of $\mathbf{X}$ in terms of t. Similarly, we cannot express the relative geometry as explicit functions of time. By discretizing the problem, we can avoid these complications. Yet, we must be satisfied with the fact that the NLP code will now find an optimal solution to a new problem that approximates the original problem. For the missions addressed later in this paper, the difference between the continuous-time formulation and the discrete formulation is academic when compared to other real-world error sources such as navigation and control inaccuracies. Now let us take a look at how to discretize the continuous-time problem.

Discretization of the Cost and Constraints

In general, we cannot solve the integral in Eq. (8). Also, it is often difficult to evaluate the path constraint functions in Eqs. (9–13) along the entire trajectory. Furthermore, differentiating the equations with respect to $\mathbf{X}_o$ is usually difficult. For these reasons, it is convenient to approximate the integral in the cost function by using a quadrature rule. Similarly, it is convenient to evaluate path constraints at discrete points along the trajectory. Once we have recast the cost and constraints as a discrete system, we can take the derivatives relatively easy as we will see near the end of this section.

Let us begin by writing the cost function as a quadrature rule as follows:

$$J(\mathbf{X}_o) \approx C_J \sum_{k=1}^{n_k} c_k F_J(\mathbf{X}_k, \mathbf{s}_{ik}, \dot{\mathbf{s}}_{ik}, s_{ik}, \dot{s}_{ik}) \tag{14}$$

where k is the quadrature point index, $\mathbf{X}_k = \mathbf{X}(t_k)$, $\mathbf{s}_{ik} = \mathbf{s}_i(t_k)$, and so on.

We have imposed many types of constraints in the continuous-time system. We limit the discussion here to the most difficult of these constraints: nonlinear path constraints. Nonlinear point constraints are a subset of nonlinear path constraints, whereas linear and bound constraints are trivial and we do not formally address them further in this work. The nonlinear path constraints are constraints that must be satisfied along a section of the trajectory and not at a specific point. We can group the equality and inequality path constraints together for convenience as

$$\mathbf{G}(\mathbf{X}_o) = \begin{pmatrix} \mathbf{G}_E(\mathbf{X}_o) \\ \mathbf{G}_I(\mathbf{X}_o) \end{pmatrix} \approx \begin{pmatrix} P_{1k}(\mathbf{X}_k, \mathbf{s}_{ik}, \dot{\mathbf{s}}_{ik}, s_{ik}, \dot{s}_{ik}) \\ P_{2k}(\mathbf{X}_k, \mathbf{s}_{ik}, \dot{\mathbf{s}}_{ik}, s_{ik}, \dot{s}_{ik}) \\ \vdots \\ P_{\ell k}(\mathbf{X}_k, \mathbf{s}_{ik}, \dot{\mathbf{s}}_{ik}, s_{ik}, \dot{s}_{ik}) \\ \vdots \\ P_{Nk}(\mathbf{X}_k, \mathbf{s}_{ik}, \dot{\mathbf{s}}_{ik}, s_{ik}, \dot{s}_{ik}) \end{pmatrix} \tag{15}$$

where the function $P_{\ell k}$ is the ℓth path constraint function evaluated at the kth quadrature point. Hence, we now have $(N \cdot k)$ point constraints in the discrete time system, whereas we only had N path constraints in the continuous-time system.

NLP algorithms are almost always more efficient when they are supplied with analytic derivatives as opposed to using finite differencing to find approximations for the derivatives. Recall that the parameters in which we allow the NLP algorithm to vary are the initial Cartesian states, $\mathbf{X}_o$. We need to determine the derivatives of the cost and constraints with respect to $\mathbf{X}_o$. We can write the derivative of the cost function as

$$\frac{\partial J}{\partial \mathbf{X}_o} \approx C_J \sum_{k=1}^{n_k} c_k \frac{\partial}{\partial \mathbf{X}_o} F_J(\mathbf{X}_k, \mathbf{s}_{ik}, \dot{\mathbf{s}}_{ik}, s_{ik}, \dot{s}_{ik}) \tag{16}$$

Using the chain rule we can expand Eq. (16) to read

$$\frac{\partial J}{\partial \mathbf{X}_o} \approx C_J \sum_{k=1}^{n_k} c_k \left(\frac{\partial F_J}{\partial \mathbf{X}_k}\frac{\partial \mathbf{X}_k}{\partial \mathbf{X}_o} + \frac{\partial F_J}{\partial \mathbf{s}_{ik}}\frac{\partial \mathbf{s}_{ik}}{\partial \mathbf{X}_o} + \frac{\partial F_J}{\partial \dot{\mathbf{s}}_{ik}}\frac{\partial \dot{\mathbf{s}}_{ik}}{\partial \mathbf{X}_o} + \frac{\partial F_J}{\partial s_{ik}}\frac{\partial s_{ik}}{\partial \mathbf{X}_o} + \frac{\partial F_J}{\partial \dot{s}_{ik}}\frac{\partial \dot{s}_{ik}}{\partial \mathbf{X}_o} \right) \tag{17}$$

Similarly, for the constraints we can write

$$\frac{\partial \mathbf{G}}{\partial \mathbf{X}_o} = \begin{pmatrix} \frac{\partial P_{1k}}{\partial \mathbf{X}_o}(\mathbf{X}_k, \mathbf{s}_{ik}, \dot{\mathbf{s}}_{ik}, s_{ik}, \dot{s}_{ik}) \\ \frac{\partial P_{2k}}{\partial \mathbf{X}_o}(\mathbf{X}_k, \mathbf{s}_{ik}, \dot{\mathbf{s}}_{ik}, s_{ik}, \dot{s}_{ik}) \\ \vdots \\ \frac{\partial P_{\ell k}}{\partial \mathbf{X}_o}(\mathbf{X}_k, \mathbf{s}_{ik}, \dot{\mathbf{s}}_{ik}, s_{ik}, \dot{s}_{ik}) \\ \vdots \\ \frac{\partial P_{Nk}}{\partial \mathbf{X}_o}(\mathbf{X}_k, \mathbf{s}_{ik}, \dot{\mathbf{s}}_{ik}, s_{ik}, \dot{s}_{ik}) \end{pmatrix} \tag{18}$$

Using the chain rule we can write the derivative of the ℓth constraint at the kth quadrature point as

$$\frac{\partial \mathbf{P}_{\ell k}}{\partial \mathbf{X}_o} \approx \frac{\partial \mathbf{P}_{\ell k}}{\partial \mathbf{X}_k}\frac{\partial \mathbf{X}_k}{\partial \mathbf{X}_o} + \frac{\partial \mathbf{P}_{\ell k}}{\partial \mathbf{s}_{ik}}\frac{\partial \mathbf{s}_{ik}}{\partial \mathbf{X}_o} + \frac{\partial \mathbf{P}_{\ell k}}{\partial \dot{\mathbf{s}}_{ik}}\frac{\partial \dot{\mathbf{s}}_{ik}}{\partial \mathbf{X}_o} + \frac{\partial \mathbf{P}_{\ell k}}{\partial s_{ik}}\frac{\partial s_{ik}}{\partial \mathbf{X}_o} + \frac{\partial \mathbf{P}_{\ell k}}{\partial \dot{s}_{ik}}\frac{\partial \dot{s}_{ik}}{\partial \mathbf{X}_o} \tag{19}$$

Inspecting Eqs. (17) and (19) we see that two types of terms appear and that these terms appear as pairs in a product. The first type of term is the derivative of the mission-specific function with respect to the formation geometry. Examples of this type of constraint are

$$\frac{\partial F_J}{\partial \mathbf{X}_k}, \qquad \frac{\partial F_J}{\partial s_{ik}}, \qquad \frac{\partial \mathbf{P}_{\ell k}}{\partial \mathbf{s}_{ik}} \tag{20}$$

The second type of term is the derivative of the geometry at a specific quadrature point, with respect to the formation initial conditions. Examples of this type of term are

$$\frac{\partial \mathbf{X}_k}{\partial \mathbf{X}_o}, \qquad \frac{\partial \dot{\mathbf{s}}_{ik}}{\partial \mathbf{X}_o}, \qquad \frac{\partial s_{ik}}{\partial \mathbf{X}_o} \tag{21}$$

What makes this method "simple" is twofold. First, the terms of type one contain derivatives of the mission performance metric function with respect to the formation geometry. For many missions, the performance metric can be cast as an algebraic function of the formation geometry. As we will see in the examples presented in later sections, these derivatives are trivial. The second reason this method is simple is that the second type of terms contain derivatives of the geometry with respect to the initial conditions, and the form of these derivatives is independent of the mission-specific performance metric, and therefore they only have to be derived and implemented once. It is very convenient that the derivatives are separated in this way, and the fact that they are is simply due to the application of the chain rule to a cost function that is explicitly dependent upon the formation geometry. Let us look at the derivatives of type two, which are the derivatives of the formation geometry and its evolution, with respect to the formation initial conditions.

Derivatives of Formation Relative Geometry with Respect to Orbit Initial Conditions

The derivatives of the formation geometry with respect to the formation initial conditions contain information on the sensitivity of the state of the formation at some time t_k, with respect to the state of the formation at the initial epoch. We would expect portions of the STM to appear in these derivatives. Recall that the definitions for the geometry given in Eqs. (4–7) express the relative geometry, such as the relative position vector between two spacecraft, in terms of the Cartesian states of the spacecraft. To calculate the derivative of side vector $\mathbf{s}_{ik}$ with respect to the initial position of the pth spacecraft, $\mathbf{r}_{op}$, we can write

$$\frac{\partial \mathbf{s}_{ik}}{\partial \mathbf{r}_{op}} = \frac{\partial}{\partial \mathbf{r}_{op}}(\mathbf{r}_{jk} - \mathbf{r}_{\ell k}) = \begin{cases} \mathbf{A}_j(t_k, t_o) & p = j \\ -\mathbf{A}_\ell(t_k, t_o) & p = \ell \\ \mathbf{0} & \text{otherwise} \end{cases} \tag{22}$$

where $\mathbf{A}_j$ is the upper left 3×3 component of the STM for the jth spacecraft and so on. We define the four 3×3 subcomponents of the STM as

$$\mathbf{\Phi}(t, t_o) = \begin{pmatrix} \mathbf{A}_{3\times3}(t, t_o) & \mathbf{B}_{3\times3}(t, t_o) \\ \mathbf{C}_{3\times3}(t, t_o) & \mathbf{D}_{3\times3}(t, t_o) \end{pmatrix} \tag{23}$$

The derivative of $\mathbf{s}_{ik}$ with respect to $\mathbf{v}_{op}$ can be written as

$$\frac{\partial \mathbf{s}_{ik}}{\partial \mathbf{v}_{op}} = \frac{\partial}{\partial \mathbf{v}_{op}}(\mathbf{r}_{jk} - \mathbf{r}_{\ell k}) = \begin{cases} \mathbf{B}_j(t_k, t_o) & p = j \\ -\mathbf{B}_\ell(t_k, t_o) & p = \ell \\ \mathbf{0} & \text{otherwise} \end{cases} \tag{24}$$

The remaining derivatives of the formation relative geometry with respect to the initial conditions are

$$\frac{\partial \dot{\mathbf{s}}_{ik}}{\partial \mathbf{r}_{op}} = \frac{\partial}{\partial \mathbf{r}_{op}}(\mathbf{v}_{jk} - \mathbf{v}_{\ell k}) = \begin{cases} \mathbf{C}_j(t_k, t_o) & p = j \\ -\mathbf{C}_\ell(t_k, t_o) & p = \ell \\ \mathbf{0} & \text{otherwise} \end{cases} \tag{25}$$

$$\frac{\partial \dot{\mathbf{s}}_{ik}}{\partial \mathbf{v}_{op}} = \frac{\partial}{\partial \mathbf{v}_{op}}(\mathbf{v}_{jk} - \mathbf{v}_{\ell k}) = \begin{cases} \mathbf{D}_j(t_k, t_o) & p = j \\ -\mathbf{D}_\ell(t_k, t_o) & p = \ell \\ \mathbf{0} & \text{otherwise} \end{cases} \tag{26}$$

$$\frac{\partial s_{ik}}{\partial \mathbf{r}_{op}} = \frac{\partial}{\partial \mathbf{r}_{op}}(\mathbf{s}_{ik}^T \mathbf{s}_{ik})^{1/2} = \frac{\mathbf{s}_{ik}^T}{s_{ik}} \frac{\partial \mathbf{s}_{ik}}{\partial \mathbf{r}_{op}} \tag{27}$$

$$\frac{\partial s_{ik}}{\partial \mathbf{v}_{op}} = \frac{\partial}{\partial \mathbf{v}_{op}}(\mathbf{s}_{ik}^T \mathbf{s}_{ik})^{1/2} = \frac{\mathbf{s}_{ik}^T}{s_{ik}} \frac{\partial \mathbf{s}_{ik}}{\partial \mathbf{v}_{op}} \tag{28}$$

$$\frac{\partial \dot{s}_{ik}}{\partial \mathbf{r}_{op}} = \frac{1}{s_{ik}}\left(\dot{\mathbf{s}}_{ik}^T - \mathbf{s}_{ik}^T \dot{\mathbf{s}}_{ik} \frac{\mathbf{s}_{ik}^T}{s_{ik}^2}\right) \frac{\partial \mathbf{s}_{ik}}{\partial \mathbf{r}_{op}} + \frac{\mathbf{s}_{ik}^T}{s_{ik}} \frac{\partial \dot{\mathbf{s}}_{ik}}{\partial \mathbf{r}_{op}} \tag{29}$$

$$\frac{\partial \dot{s}_{ik}}{\partial \mathbf{v}_{op}} = \frac{1}{s_{ik}}\left(\dot{\mathbf{s}}_{ik}^T - \mathbf{s}_{ik}^T \dot{\mathbf{s}}_{ik} \frac{\mathbf{s}_{ik}^T}{s_{ik}^2}\right) \frac{\partial \mathbf{s}_{ik}}{\partial \mathbf{v}_{op}} + \frac{\mathbf{s}_{ik}^T}{s_{ik}} \frac{\partial \dot{\mathbf{s}}_{ik}}{\partial \mathbf{v}_{op}} \tag{30}$$

The derivatives contained in Eqs. (22–30) only contain terms that involve the spacecraft states and their associated STMs. Therefore, these equations do not change from one problem to the next. What will change is the actual numbers in the STMs and the spacecraft states. This enables us to implement the algorithm in a general way to minimize the amount of recoding to apply the algorithm to a new mission or mission phase. Now let us take a look at some aspects of the implementation we use. Then, we will look at two mission applications.

Implementation

The implementation of a numerical optimization approach must be carefully considered to ensure speed and ease of use. There are several aspects of the implementation that will have an influence on the speed of the approach we present in this paper. The first is how the orbit states and STMs are propagated. It is recommended that all propagation is performed in a compiled language as opposed to an interpreted language such as MATLAB®. This is primarily due to the fact that we must propagate 42 differential equations per spacecraft. For our implementation, we coded most of the algorithm in MATLAB, and use MATLAB's fmincon SQP routine. However, all propagation is performed in compiled C code that can be called directly from MATLAB through a MEX interface.

The second issue that influences speed is the number of quadrature points chosen and the considerable amount of bookkeeping which must be performed if the orbits are propagated and stored before evaluating the cost and constraints. For the examples presented in the next few sections, we used a simple trapezoid rule as opposed to a complex quadrature rule, in the summation of the discrete cost function. The number of points in the summation of the cost function was chosen to maximize the speed while yielding an acceptable approximation to the exact, integral form of the cost function.

Another issue to consider is the organization of the code. To enable convenient application to new problems, it is helpful to isolate code that is particular to the mission-specific function. By separating and organizing the code carefully, it is relatively simple to provide a new function containing cost and derivative terms. For example, for the MMS example in the next section, there are only 32 lines of code that are specific to the MMS problem and they are all located in a single MATLAB function that is easy to modify to solve a new problem.

Now that we have discussed a few of the important issues in implementation, let us take a look at a specific mission example, the Magnetospheric Multiscale Mission.

Application: MMS

In the preceding portions of this paper we posed a general parameter optimization approach, and an approach to its implementation, to find optimal initial conditions of spacecraft formations. In this section, we will discuss an application of the method to the MMS mission. We will begin with a brief overview of the MMS mission. Then we will present the cost and constraint functions we choose for MMS and derive the necessary derivatives. We conclude the section with a discussion of results for a particular phase of the MMS mission.

The MMS mission is one of several missions in NASA's solar terrestrial probes (STP) program. MMS will employ a four spacecraft formation to make fundamental advancements in our understanding of the Earth's magnetosphere and its dynamic interaction with the solar wind. MMS will not be the first mission to use multiple spacecraft to study magnetospheric dynamics. The Cluster II mission was successfully launched by the European Space Agency (ESA) in the summer of 2000 and has already provided fascinating results on magnetospheric dynamics [11–13]. There are several phases in the MMS mission. In this work we focus on phase 1, which consists of a highly elliptic reference orbit with a perigee radius of 1.2 Re and an apogee radius of 12 Re. The mission design objective is to provide a near regular tetrahedron formation between true anomalies of 160 and 200 deg. There are numerous ways to pose a set of cost and constraint functions to achieve this goal and in the next section we will discuss the method we choose. There have been many important contributions to the literature on tetrahedron formations and magnetospheric missions, and a detailed discussion of all the past efforts is beyond the scope of this work. References [20,22–28] contain information on the science of the MMS mission, reference orbit design, launch window analysis, tetrahedron design, and maneuver design, among other topics.

MMS Cost and Constraint Functions

In phase 1 of the MMS mission, one of the mission design objectives is to provide a near regular tetrahedron, with sides of 10 km, within a region defined by ±20 deg in true anomaly about orbit apogee. This requires casting a set of cost and constraint functions that take into account the size of the tetrahedron and its likeness to a regular tetrahedron. Let us begin by investigating some useful properties of tetrahedrons. If we assume one of the spacecraft is the reference, then there are three relative position vectors that describe the relative geometry of the spacecraft and we will define them as $\mathbf{s}_{1k}, \mathbf{s}_{2k}$, and $\mathbf{s}_{3k}$ to be consistent with our previous definitions. Knowing these quantities, we can calculate the volume of the tetrahedron at the kth quadrature point using

$$V_a = \tfrac{1}{6}|\mathbf{s}_{1k} \cdot (\mathbf{s}_{2k} \times \mathbf{s}_{3k})| \tag{31}$$

where V_a stands for the volume of the actual tetrahedron formed by the spacecraft, as opposed to the desired volume which we discuss later. Their are six unique sides in the formation, and the side lengths can be used to calculate the average side length L^*, using

$$L^* = \tfrac{1}{6}(s_{1k} + s_{2k} + s_{3k} + s_{4k} + s_{5k} + s_{6k}) \tag{32}$$

at time t_k. Paschmann [29] and Guzman [30] present several methods for evaluating how close a tetrahedron is to being a regular tetrahedron. One approach is to compare the volume of the actual tetrahedron V_a with the volume of a regular tetrahedron V_r with side lengths equal to the average side length of the actual tetrahedron. The volume of a regular tetrahedron of side length L can be calculated using

$$V_r = \frac{\sqrt{2}}{12} L^3 \tag{33}$$

Using V_a and V_r, Paschmann [29] suggests an instantaneous volumetric performance metric Q_v for a tetrahedron of spacecraft as

$$Q_v^k = \frac{V_a^k}{V_r^k} = \frac{\sqrt{2}}{L^{*3}} |\mathbf{s}_{1k} \cdot (\mathbf{s}_{2k} \times \mathbf{s}_{3k})| \tag{34}$$

where the superscripts k indicate values at the kth point. Q_v has the useful property: $0 \le Q_v \le 1$. However, it does not take into account the actual size of the tetrahedron. We propose a simple polynomial function Q_s that has the properties below, to take into account the size of a tetrahedron. The constants ℓ_1, ℓ_2, ℓ_3, and ℓ_4 are used to change the shape of the function. A plot of Q_s for a 10 km tetrahedron, with $\ell_1 = 4$, $\ell_2 = 6$, $\ell_3 = 18$, and $\ell_4 = 25$, is shown in Fig. 1. The important property of Q_s is that it is zero for tetrahedrons with an average side length of less than 4 km or greater than 25 km. Between the values of 6 and 18 km, Q_s is equal to 1:

$$Q_s^k(L^*) = \begin{cases} 0 & L^* < \ell_l \\ (L^* - \ell_1)^2 (L^* + \ell_1 - 2\ell_2)^2 / (\ell_2 - \ell_1)^4 & \ell_l < L^* < \ell_2 \\ 1 & \ell_2 < L^* < \ell_3 \\ (L^* - \ell_4)^2 (L^* - 2\ell_3 + \ell_4)^2 / (\ell_4 - \ell_3)^4 & \ell_3 < L^* < \ell_4 \\ 0 & L^* > \ell_4 \end{cases} \tag{35}$$

By making a composite function involving both Q_v and Q_s, we can create a performance metric that evaluates both the size and shape of a tetrahedron. Let us use the following continuous-time form:

$$P(t) = (1 - Q_v(t) Q_s(t))^2 \tag{36}$$

$P(t)$ is zero for a regular tetrahedron with a side length between 6 and 18 km. For unacceptable tetrahedrons, $P(t)$ is 1. We can formulate a cost function by integrating $P(t)$ over the region of interest, or

$$J = \int_{t_o}^{t_f} P(t)\, \mathrm{d}t = \int_{t_o}^{t_f} (1 - Q_v(t) Q_s(t))^2\, \mathrm{d}t$$

where t_o is the time when the reference spacecraft is at a true anomaly of 160 deg, and t_f is the time when the reference spacecraft is at a true anomaly of 200 deg. However, we cannot evaluate this integral exactly, so we have to approximate it. We use a simple trapezoid rule, as opposed to a complicated quadrature rule, where

$$\begin{aligned} J &\approx C \sum_{k=1}^{n_k} (1 - Q_v^k Q_s^k)^2 \Delta t_k \\ &= C \sum_{k=1}^{n_k} \left(1 - \sqrt{2} \frac{|\mathbf{s}_{1k} \cdot (\mathbf{s}_{2k} \times \mathbf{s}_{3k})|}{\left(\frac{1}{6} \sum_{i=1}^{6} s_{ik}\right)^3} Q_s^k \right)^2 \Delta t_k \end{aligned} \tag{37}$$

For MMS, the independent variables are the initial Cartesian states of the four spacecraft in formation. In addition to minimizing the cost function above, we also must satisfy the nonlinear constraints where the semimajor axes of all spacecraft in formation are equal. We can write this as

$$a_p = \frac{1}{2/r_{op} - v_{op}^2/\mu} = a_d \tag{38}$$

where a_p is the semimajor axis of the pth spacecraft, a_d is the desired semimajor axis of all spacecraft, r_{op} is the magnitude of the initial position vector of the pth spacecraft, and v_{op} is the magnitude of the initial velocity vector of the pth spacecraft.

In Eq. (37), we have a performance metric for MMS that is an explicit function of the relative geometry of the formation. In Eq. (38), we have four nonlinear constraint functions in terms of the initial spacecraft states. Now all we require are the derivatives of the cost and constraint functions in order to apply the method to find an optimal solution.

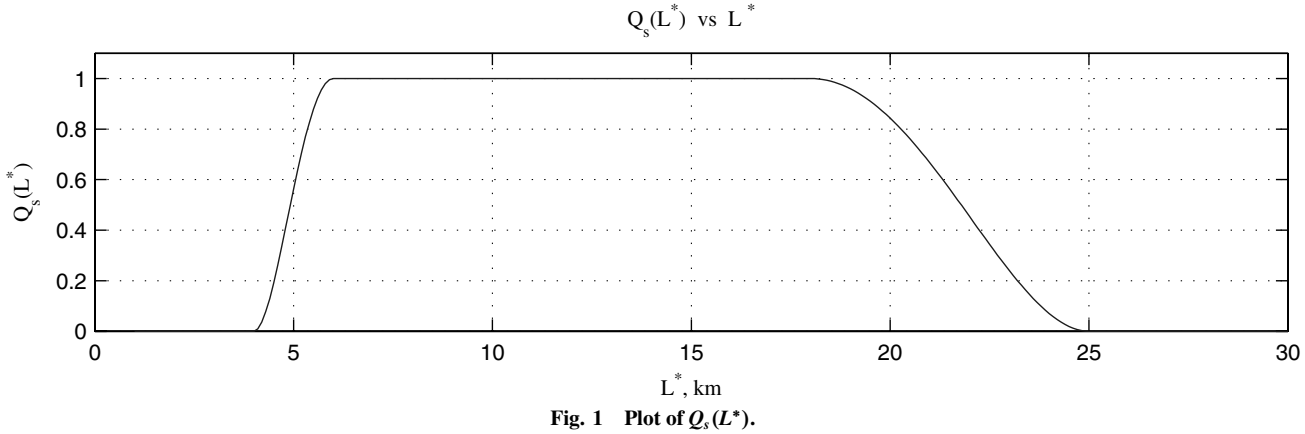

Fig. 1 Plot of $Q_s(L^*)$.

Derivatives of MMS Cost and Constraint Functions

The last step to perform before solving for an optimal formation for MMS is to determine the derivatives of the cost and constraints with respect to $\mathbf{X}_o, \mathbf{s}_{ik}, \dot{\mathbf{s}}_{ik}, s_{ik}$, and $\dot{s}_{ik}$. The kth term in the summation in the cost function J_k has the form

$$J_k = \left(1 - Q_v^k(\mathbf{s}_{ik}, s_{ik}) Q_s^k(s_{ik})\right)^2 \tag{39}$$

Notice that the cost function does not depend on the side rates, only on the side vectors and side lengths. Therefore

$$\frac{\partial J_k}{\partial \dot{\mathbf{s}}_{ij}} = \mathbf{0} \tag{40}$$

and

$$\frac{\partial J_k}{\partial \dot{s}_{ij}} = 0 \tag{41}$$

The nonzero derivatives of the cost function with respect to $\mathbf{s}_{ik}$ are (see Appendix A for more detail)

$$\frac{\partial J_k}{\partial \mathbf{s}_{1k}} = -2\sqrt{2}\left(1 - Q_v^k Q_s^k\right) \frac{Q_s^k}{L^3} \frac{\mathbf{s}_{1k} \cdot \mathbf{s}_{2k} \times \mathbf{s}_{3k}}{|\mathbf{s}_{1k} \cdot \mathbf{s}_{2k} \times \mathbf{s}_{3k}|} \left(\mathbf{s}_{2k}^x \mathbf{s}_{3k}\right)^T \tag{42}$$

$$\frac{\partial J_k}{\partial \mathbf{s}_{2k}} = 2\sqrt{2}\left(1 - Q_v^k Q_s^k\right) \frac{Q_s^k}{L^3} \frac{\mathbf{s}_{1k} \cdot \mathbf{s}_{2k} \times \mathbf{s}_{3k}}{|\mathbf{s}_{1k} \cdot \mathbf{s}_{2k} \times \mathbf{s}_{3k}|} \left(\mathbf{s}_{1k}^T \mathbf{s}_{3k}^x\right) \tag{43}$$

$$\frac{\partial J_k}{\partial \mathbf{s}_{3k}} = -2\sqrt{2}\left(1 - Q_v^k Q_s^k\right) \frac{Q_s^k}{L^3} \frac{\mathbf{s}_{1k} \cdot \mathbf{s}_{2k} \times \mathbf{s}_{3k}}{|\mathbf{s}_{1k} \cdot \mathbf{s}_{2k} \times \mathbf{s}_{3k}|} \left(\mathbf{s}_{1k}^T \mathbf{s}_{2k}^x\right) \tag{44}$$

Now let us look at the derivatives of the cost function with respect to s_{ik}. We start with

$$\frac{\partial J_k}{\partial s_{ik}} = \frac{\partial}{\partial s_{ik}}\left(1 - Q_v^k Q_s^k\right)^2 = -2(1 - Q_v^k Q_s^k)\left(Q_v^k \frac{\partial Q_s^k}{\partial s_{ik}} + \frac{\partial Q_v^k}{\partial s_{ik}} Q_s^k\right) \tag{45}$$

We can show that

$$\frac{\partial J_k}{\partial s_{ik}} = -2\left(1 - Q_v^k Q_s^k\right)\left(\frac{Q_v^k}{6} \frac{\partial Q_s^k}{\partial L^*} - \frac{Q_s^k}{\sqrt{2}} \frac{|\mathbf{s}_{1k} \cdot \mathbf{s}_{2k} \times \mathbf{s}_{3k}|}{L^{*^4}}\right) \tag{46}$$

where

$$\frac{\partial Q_s^k}{\partial L^*} = \begin{cases} 0 & L^* < \ell_l \\ 4(L^* - \ell_1)(L^* + \ell_1 - 2\ell_2)(L^* - \ell_2)/(-\ell_2 + \ell_1)^4 & \ell_l < L^* < \ell_2 \\ 0 & \ell_2 < L^* < \ell_3 \\ 4(L^* - 2\ell_3 + \ell_4)(L^* - \ell_4)(L^* - \ell_3)/(\ell_4 - \ell_3)^4 & \ell_3 < L^* < \ell_4 \\ 0 & L^* > \ell_4 \end{cases} \tag{47}$$

Finally, the nonzero derivatives of the semimajor axis constraints are

$$\frac{\partial a_p}{\partial \mathbf{r}_{op}} = \frac{\partial a_p}{\partial r_{op}} \frac{\partial r_{op}}{\mathbf{r}_{op}} = 2\left(\frac{a_p}{r_{op}}\right)^2 \frac{\mathbf{r}_{op}^T}{r_{op}} \tag{48}$$

$$\frac{\partial a_p}{\partial \mathbf{v}_{op}} = \frac{\partial a_p}{\partial v_{op}} \frac{\partial v_{op}}{\mathbf{v}_{op}} = \frac{2a_p^2}{\mu} \mathbf{v}_{op}^T \tag{49}$$

We now have all of the information to use an NLP code to find optimal solutions. In the next section, we discuss the results for an optimal MMS formation.

Results of Optimization for MMS

Recall that the primary formation flying goal for the phase of MMS we consider is to provide a near regular tetrahedron, with side lengths near 10 km, in the orbit region between 160 and 200 deg true anomaly. We also must satisfy the constraint that the semimajor axes for all spacecraft are the same. Figure 2 shows several plots that illustrate an optimal solution. The top figure shows the evolution of Q_s, Q_v, and $Q_s \cdot Q_v$ for one Keplerian orbit. The lower plot shows the evolution of the six individual sides and the average side length. We see that the quality factor is near 1 during the region of interest, which is bounded by the vertical dashed lines. The optimal solution is significantly better than the initial guess which was calculated using an algorithm described in [27]. For the initial guess, the time-averaged quality factor $Q_v \cdot Q_s$ was about 0.55. The NLP algorithm found an optimal solution with a time-averaged quality factor of about 0.94 in the orbit region of interest, and required only 40 cost function evaluations. From inspection of the bottom plot in Fig. 2 we see that the average side length is on the order of 10 km. The orbit states for this solution are found in Appendix B.

Now let us take a look at a more complicated example, the Laser Interferometer Space Antenna, which has a large number of nonlinear path constraints, in a deep-space flight regime.

Application: LISA

LISA is a NASA/ESA mission to detect and study gravitational waves from massive black hole systems and galactic binaries. Gravitational waves are ripples in space-time caused by massive objects. They are a prediction of general relativity, but are yet to be directly detected. The detection and understanding of gravitational waves can provide breakthroughs and refinements in current relativistic theory. For a more detailed discussion of the LISA mission, see [21,31,32].

The nominal LISA formation consists of three spacecraft in heliocentric orbits trailing the Earth by about 20 deg, with inclinations near 1 deg with respect to the ecliptic plane. The mission design goals for LISA are challenging. The primary goal is to provide a formation that maintains a nearly equilateral triangle with sides near 5×10^6 km for the entire life of the mission, which is currently about 8 years. This has to be achieved entirely through careful orbit design, as continuous feedback control of the orbits is not permitted because it will interfere with the science measurements. We also must ensure that the sides of the triangle remain within 1% of 5×10^6 km and that the side rates never exceed 15 m/s. There is a secondary, and competing goal, that we keep the formation as close to Earth as possible for power reasons. Let us look at how we can cast these goals and constraints in a manner consistent with the algorithm presented in previous sections.

LISA Cost and Constraint Functions

There are essentially two goals for LISA that can be formulated as a cost function. The first is the desire to keep the side lengths as close to 5×10^6 km as possible. The second is to minimize the distance between the LISA formation and the Earth. We choose to pose the cost function in terms of the Earth distance, and cast the side length goals as constraints. The following cost function permits us to minimize the distance of the LISA formation from Earth:

$$J = c \sum_{k=1}^{n_k} (r_{1k} + r_{2k} + r_{3k}) \tag{50}$$

where r_{1k} is the distance from Earth of spacecraft 1 at quadrature point k and is calculated by $r_{1k} = (x_{1k}^2 + x_{2k}^2 + x_{3k}^2)^{1/2}$, and so on. To ensure that the sides of the formation never exceed a 1% variation from the desired side length, we can apply the following set of $i \cdot k$ path constraints:

$$(s_{ik} - L)^2 \le g_1(t_k)^2 \tag{51}$$

where $L = 5 \times 10^6$ and

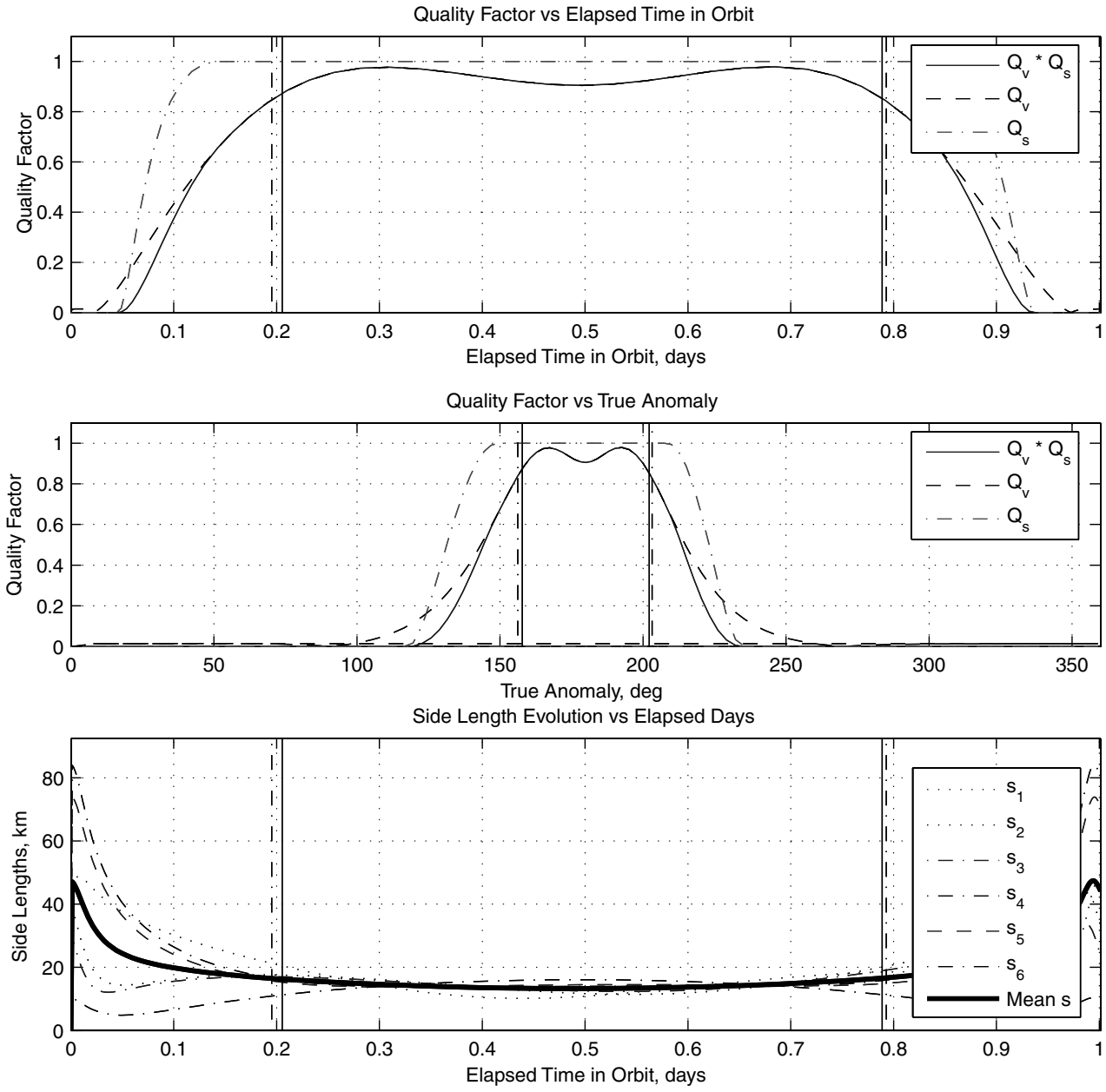

Fig. 2 Results of MMS orbit optimization.

$$g_1(t_k) = \begin{cases} -2000t_k + 50,000 \text{ km} & t_k < 5 \text{ yrs} \\ 40,000 \text{ km} & t_k \geq 5 \text{ yrs} \end{cases} \tag{52}$$

where t_k is the time at the kth point, in units of years. The function $g_1(t_k)$ above, and $g_2(t_k)$ below, were suggested by Sweetser.[†] $g_1(t_k)$ and $g_2(t_k)$ are chosen so that the range and range rate requirements are still satisfied with expected orbit insertion errors. The following set of $i \cdot k$ constraints ensures that the side rates never exceed 15 m/s:

$$(\dot{s}_{ik})^2 \leq g_2(t_k)^2 \tag{53}$$

where

$$g_2(t) = \begin{cases} -\frac{3}{5}t_k + 15 \text{ m/s} & t_k < 5 \text{ yrs} \\ 12 \text{ m/s} & t_k \geq 5 \text{ yrs} \end{cases} \tag{54}$$

Finally, we require that the ecliptic inclination i_{ec} of each orbit be less than 1 deg:

$$i_{\text{ec}} < 1 \text{ deg} \tag{55}$$

Let us reformulate this constraint so that the derivatives are simpler:

$$\cos i_{\text{ec}} = \hat{\mathbf{z}}^T \hat{\mathbf{h}}_{\text{ec}} > \cos 1 \text{ deg} \tag{56}$$

where

$$\hat{\mathbf{z}} = [0 \quad 0 \quad 1]^T \tag{57}$$

$$\mathbf{h}_{\text{ec}} = \mathbf{r}_{\text{ec}} \times \mathbf{v}_{\text{ec}} \tag{58}$$

and $\mathbf{r}_{\text{ec}}$ and $\mathbf{v}_{\text{ec}}$ are the initial position and velocity of a LISA spacecraft, in the sun-centered mean ecliptic of the J2000 frame, and are given by

$$\mathbf{r}_{\text{ec}} = \mathbf{r}_{E/S} + \mathbf{R}\mathbf{r}_{ok} \tag{59}$$

$$\mathbf{v}_{\text{ec}} = \mathbf{v}_{E/S} + \mathbf{R}\mathbf{v}_{ok} \tag{60}$$

[†]Sweetser, T. H., private communication, April 2005.

Table 1 Derivatives of LISA cost and constraint functions

Function	∂s_{ik}	$\partial \dot{s}_{ik}$	$\partial \mathbf{r}_{op}$	$\partial \mathbf{v}_{op}$
$\partial c \sum_{k=1}^{n_k}(r_{1k}+r_{2k}+r_{3k})$	0	0	$\frac{\partial \mathbf{r}_{pk}}{\partial \mathbf{r}_{op}} = \mathbf{A}_p(t_k,t_o)$	$\frac{\partial \mathbf{r}_{pk}}{\partial \mathbf{v}_{op}} = \mathbf{B}_p(t_k,t_o)$
$\partial(s_{ik}-L)^2$	$2(s_{ik}-L)$	0	0	0
$\partial(\dot{s}_{ik})^2$	0	$2\dot{s}_{ik}$	0	0
$\partial(\hat{\mathbf{z}}^T)\hat{\mathbf{h}}_{\text{ec}}$	0	0	$\mathbf{z}^T(-\mathbf{v}^\times_{E/S}\mathbf{R}-\mathbf{R}\mathbf{v}^\times_{op})$	$\mathbf{z}^T(\mathbf{r}^\times_{E/S}\mathbf{R}+\mathbf{R}\mathbf{r}^\times_{op})$

where $\mathbf{r}_{E/S}$ and $\mathbf{v}_{E/S}$ are the position and velocity of the Earth with respect to the sun, at the initial epoch, and $\mathbf{R}$ is the rotation matrix from the Earth's mean equator of J2000 to the mean ecliptic of J2000.

The derivatives of the cost and constraint functions are presented in Table 1. The leftmost column in the table indicates the function, and the rows are the derivatives with respect to the term that appears in the column labels. The superscript x indicates the skew symmetric matrix. For derivatives with respect to the initial Cartesian states, the p subscript is a dummy variable associated with the state of the pth spacecraft.

Let us discuss some practical considerations before moving on to the results of the LISA optimization. The initial epoch we use is 1 January 2015 12:00:00.000 UTC. The force model used in the LISA optimization includes point masses from all planets and the Earth's moon. The coordinate system is Earth's mean J2000 equator. However, the converged solution is presented in Appendix B in the sun-centered mean ecliptic of J2000. Now let us look at details of a representative initial guess and optimal solution for LISA.

Results of Optimization for LISA

Figure 3 contains plots illustrating characteristics of the initial guess and converged solution for an 8.5 year LISA trajectory. The plots associated with the initial guess are in the left column, and the plots associated with the converged solution are in the right column. The independent variable for each plot is the elapsed mission time in years. The first row of plots shows the side lengths between spacecraft, or s_i. The second row of plots shows the side rates, or $\dot{s}_i$. Finally, the third row contains plots illustrating the evolution of the spacecraft-to-Earth distance.

Let us begin by taking a closer look at some properties of the initial guess. The first point we see is that the initial guess is not a feasible solution. We see that the maximum variation in range between any two LISA spacecraft is around 700,000 km. This is over an order of magnitude larger than the desired maximum range variation of 50,000 km. Similarly, the maximum range rate for the initial guess is around 140 m/s, which is about an order of magnitude higher than the 15 m/s constraint. From inspection of the plots, we see that the initial guess is unstable. The instability is due to the fact that the initial guess has been placed too close to the Earth. We see that the distance from Earth to the formation varies from a minimum of about 31×10^6 km to about 53×10^6 km.

The optimal solution is found to have significantly improved properties over the initial guess. Table 2 contains a comparison of relevant statistics for the initial guess and optimal solution. We see that the maximum variation in range between any two spacecraft is 50,000 km for the converged solution. This indicates that the range constraint is active at the solution. The range rate between any two spacecraft is 13.7 m/s. The penalty we have paid to satisfy these constraints is that the formation was moved farther away from Earth. For the optimal solution, the minimum and maximum distance from any LISA spacecraft to the Earth is about 49×106 km and 64×106 km, respectively.

In summary, we have seen that the method we present here can solve a complex mission design problem, with multiple real-world constraints, in a deep-space flight regime. We began with a set of goals and requirements for the LISA mission and cast them mathematically as a set of cost and constraint functions. Next, we presented the derivatives of the cost and constraint functions with

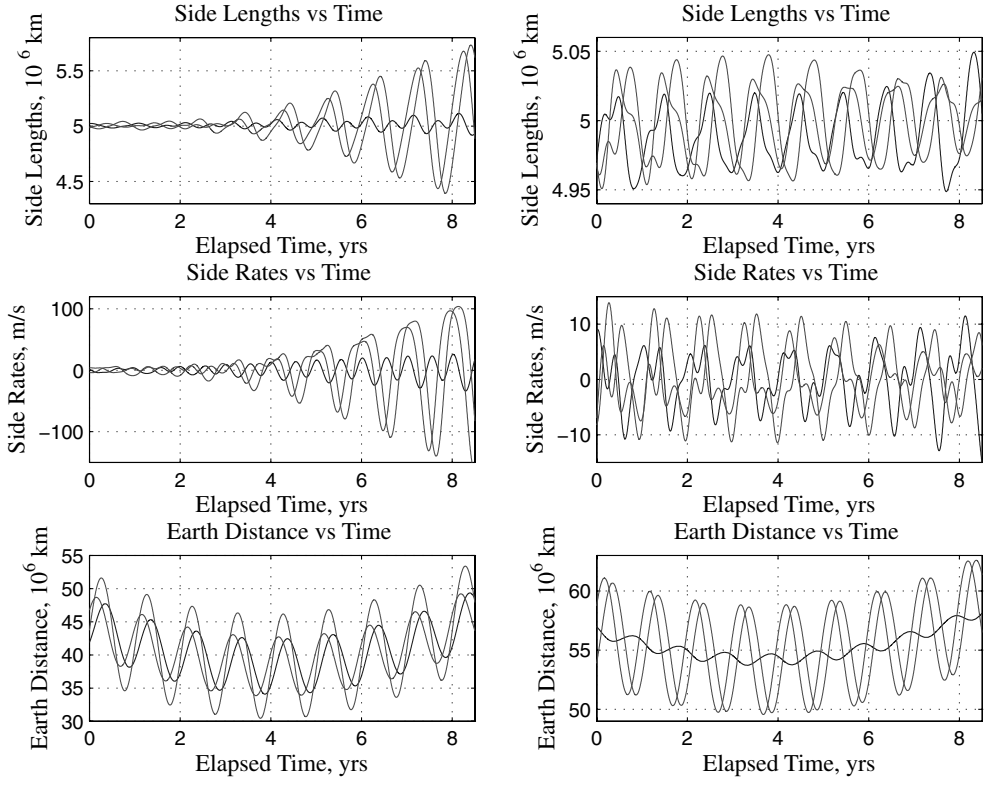

Fig. 3 Initial guess and results for LISA optimization (initial guess data in left column, optimal solution data in right column).

Table 2 Statistics for LISA initial guess and converged solution

Trajectory	$\max(\lvert\dot{s}_{ik}\rvert)$, m/s	$\max(s_{ik})$, 10^6 km	$\min(s_{ik})$, 10^6 km	$\max(r_{ik})$, 10^6 km
Initial guess	140	5.7	4.4	53
Optimal solution	13.7	5.04	4.95	64

respect to the formation geometry. Comparing the initial guess to the converged solution indicates that we can satisfy the current mission requirements for LISA, and that we do not need a feasible guess to find a solution. Now let us look at some general conclusions we can draw from this work.

Conclusions

We began this work with a long list of objectives that is motivated by the need for a practical, efficient method to solve complex, real-world guidance problems to support a diverse set of formation flying missions. The approach we presented was developed in a general manner without regard to any particular mission performance metric and is therefore applicable to numerous formation flying guidance problems. The approach is nonlinear and not restricted to specific flight regimes or small interspacecraft separations. By carefully implementing the method, we can minimize the amount of work that is required to solve new problems. For example, the entire MATLAB implementation of the cost and constraint functions for the MMS example, neglecting code for propagation and the NLP solver, consists of only 354 lines of code. Only 36 lines are specific to the MMS problem. The reason so few lines of code are specific to the mission problem is due to the fact that we assumed the cost function is an explicit function of the spacecraft relative positions, velocities, ranges, and range rates. This simple assumption allows us to perform half of the derivation of the derivatives a priori in a general way and implement them in software only once. This reduces the work that must be performed to solve new analysis problems. Furthermore, the derivatives that can be calculated a priori are the derivatives of the dynamics with respect to the initial conditions. The derivatives that are problem dependent are often simply derivatives of algebraic functions with respect to the formation geometry.

We illustrated the applicability of the approach to two different missions: MMS and LISA. Both of these missions have a complex set of cost and constraint functions that are explicitly dependent upon the formation geometry. We presented optimal orbit solutions for both missions.

Appendix A

This Appendix rigorously addresses the issue that some of the independent variables that appear in Eq. (14) actually have a dependency upon each other. It is important to illustrate how the algorithm, and most notably the derivatives of the cost and constraints, take this fact into account. Without loss of generality, we can begin by looking at a simplified cost function that does not contain the dependence upon $\dot{\mathbf{s}}_{ik}$ or $\dot{s}_{ik}$. We assume that the component of the cost function at the kth quadrature point can be written as

$$J^k = F_J(\mathbf{X}_k, \mathbf{s}_{ik}, s_{ik}, \mathbf{C}_J) \tag{A1}$$

Writing Eq. (A1), showing all of the functional dependencies explicitly, we obtain

$$J^k = F_J(\mathbf{X}_k, \mathbf{g}_i(\mathbf{X}_k), \mathbf{h}_i(\mathbf{g}_i(\mathbf{X}_k)), \mathbf{C}_J) \tag{A2}$$

where

$$\mathbf{s}_{ik} = \mathbf{g}_i(\mathbf{X}_k) \tag{A3}$$

and

$$s_{ik} = \mathbf{h}_i(\mathbf{g}_i(\mathbf{X}_k)) \tag{A4}$$

Taking the derivatives of the two forms of the cost function shown in Eqs. (A1) and (A2) we obtain the following equations:

$$\frac{\partial J^k}{\partial \mathbf{X}_o} = \frac{\partial F_J}{\partial \mathbf{X}_k}\frac{\partial \mathbf{X}_k}{\partial \mathbf{X}_o} + \frac{\partial F_J}{\partial \mathbf{s}_{ik}}\frac{\partial \mathbf{s}_{ik}}{\partial \mathbf{X}_o} + \frac{\partial F_J}{\partial s_{ik}}\frac{\partial s_{ik}}{\partial \mathbf{X}_o} \tag{A5}$$

$$\frac{\partial J^k}{\partial \mathbf{X}_o} = \frac{\partial F_J}{\partial \mathbf{X}_k}\frac{\partial \mathbf{X}_k}{\partial \mathbf{X}_o} + \frac{\partial F_J}{\partial \mathbf{g}_i}\frac{\partial \mathbf{g}_i}{\partial \mathbf{X}_k}\frac{\partial \mathbf{X}_k}{\partial \mathbf{X}_o} + \frac{\partial F_J}{\partial \mathbf{h}_i}\frac{\partial \mathbf{h}_i}{\partial \mathbf{g}_i}\frac{\partial \mathbf{g}_i}{\partial \mathbf{X}_k}\frac{\partial \mathbf{X}_k}{\partial \mathbf{X}_o} \tag{A6}$$

Equating like terms in Eqs. (A5) and (A6) we arrive at

$$\frac{\partial \mathbf{s}_{ik}}{\partial \mathbf{X}_o} = \frac{\partial \mathbf{g}_i}{\partial \mathbf{X}_k}\frac{\partial \mathbf{X}_k}{\partial \mathbf{X}_o} \tag{A7}$$

$$\frac{\partial s_{ik}}{\partial \mathbf{X}_o} = \frac{\partial \mathbf{h}_i}{\partial \mathbf{g}_i}\frac{\partial \mathbf{g}_i}{\partial \mathbf{X}_k}\frac{\partial \mathbf{X}_k}{\partial \mathbf{X}_o} \tag{A8}$$

These equations explain a subtlety that can be overlooked when applying the method to a new problem. The subtlety is that when providing the derivatives of the type $\partial F_J/\partial s_{ik}$ for a new problem, you must ignore the coupling of s_{ik} and $\mathbf{s}_{ik}$. This is because Eqs. (A7) and (A8) illustrate that the coupling between s_{ik} and $\mathbf{s}_{ik}$ is taken into account in Eqs. (22–30). In other words, the coupling of s_{ik} and $\mathbf{s}_{ik}$, and similarly $\dot{s}_{ik}$ and $\dot{\mathbf{s}}_{ik}$, is handled in the portion of the algorithm that is general to any problem. As a brief example, if a mission cost function is of the form

$$J^k = (\mathbf{s}_{1k}^T \mathbf{s}_{2k}) s_{1k} \tag{A9}$$

then the correct portion of the derivative to be supplied to the algorithm is simply

$$\frac{\partial J^k}{\partial \mathbf{s}_{ik}} = \mathbf{s}_{2k}^T s_{1k} \tag{A10}$$

Appendix B

Optimal orbit states for MMS and LISA are given in Tables B1 and B2.

Table B1 States for a representative MMS phase 1, 10 km formation

Property	MMS1	MMS2	MMS3	MMS4
a, km	42,095.7	42,095.7	42,095.7	42,095.7
e	0.81818	0.81798411	0.81799342	0.81800317
i, deg	27.8	27.800078	27.801283	27.798145
ω, deg	15.000001	15.008968	15.002286	14.982171
Ω, deg	0	359.99962	359.98943	0.014249285
ν, deg	180	179.99657	180.00303	180.00225

Table B2 States for a representative LISA formation (heliocentric MJ2000 ecliptic, 1 January 2015 12:00:00.000 UTC)

Property	LISA1	LISA2	LISA3
a, km	149,457,958	149,457,278	149,457,372
e	0.018265676	0.00949038651	0.00887263317
i, deg	0.890058699	0.993559828	0.993293398
ω, deg	90.6009132	32.5150215	148.24379
Ω, deg	3.64713792	126.757235	240.28666
ν, deg	343.945562	278.418085	51.016237

References

[1] Garrison, J. L., Gardner, T. G., and Axelrad, P., "Relative Motion in Highly Elliptical Orbits," *Advances in the Astronautical Sciences*, Vol. 89, No. 2, 1995, pp. 1359–1376.

[2] Melton, R. G., "Time-Explicit Representation of Relative Motion Between Elliptical Orbits," *Journal of Guidance, Control, and Dynamics*, Vol. 23, No. 4, 2000, pp. 604–610.

[3] Karlgard, C. D., and Lutze, F. H., "Second Order Relative Motion Equations," *Journal of Guidance, Control, and Dynamics*, Vol. 26, No. 1, 2003, pp. 41–49.

[4] Gurfil, P., and Kasdin, N. J., "Nonlinear Modelling of Spacecraft Relative Motion in the Configuration Space," *Journal of Guidance, Control, and Dynamics*, Vol. 27, No. 1, 2004, pp. 154–157.

[5] Alfriend, K. T., and Yan, H., "Evaluation and Comparison of Relative Motion Theories," *Journal of Guidance, Control, and Dynamics*, Vol. 28, No. 2, 2005, pp. 254–261.

[6] Scharf, D. P., Hadaegh, F. Y., and Kang, B. H., "A Survey of Spacecraft Formation Flying Guidance," *Proceedings of the International Symposium on Formation Flying Missions and Technologies*, CNES, Toulous, Oct. 2002.

[7] Schaub, H., "Relative Orbit Geometry Through Classical Orbit Element Differences," *Journal of Guidance, Control, and Dynamics*, Vol. 27, No. 5, 2004, pp. 839–848.

[8] Chichka, D. F., "Satellite Clusters with Constant Apparent Distribution," *Journal of Guidance, Control, and Dynamics*, Vol. 24, No. 1, 2001, pp. 117–122.

[9] Schaub, H., and Alfriend, T., "J_2 Invariant Relative Orbits for Spacecraft Formations," *Celestial Mechanics and Dynamical Astronomy*, Vol. 79, No. 2, 2001, pp. 77–95.

[10] Vaddi, S. S., Vadali, S. R., and Alfriend, K. T., "Formation Flying: Accommodating Nonlinearity and Eccentricity Perturbations," *Journal of Guidance, Control, and Dynamics*, Vol. 26, No. 2, 2003, pp. 214–223.

[11] Hocken, D., and Schoenmaekers, J., "Optimization of Cluster Constellation Manoeuvres," *Proceedings of the 16th International Symposium on Space Flight Dynamics*, NASA/JPL, Pasadena, CA, Dec. 2001.

[12] Dow, J., Matussi, S., Dow, R. M., Schmidt, M., and Warhaut, M., "The Implemetation of the Cluster II Constellation," *Acta Astonautica*, Vol. 54, No. 9, 2004, pp. 657–669.

[13] Seig, D., "Optimization of New 4 S/C Formations Considering Operational Constraints of the Extended Cluster Mission," *Proceedings of the 18th International Symposium on Space Flight Dynamics*, GSOC/ESA, Munich, Oct. 2004.

[14] Huntington, G. T., and Rao, A. V., "Optimal Reconfiguration of a Tetrahedral Formation via a Gauss Pseudospectral Method," *Advances in the Astronautical Sciences*, Vol. 123, No. 2, 2004, pp. 1337–1358.

[15] Huntington, G. T., and Benson, D. A., "Post Optimality Evaluation and Analysis of a Formation Flying Problem via a Gauss Pseudospectral Method," *Advances in the Astronautical Sciences*, Vol. 123, No. 2, 2004, pp. 1359–1378.

[16] Richards, A., Schouwenaars, T., How, J. P., and Fenson, E., "Spacecraft Trajectory Planning with Avoidance Constraints Using Mixed-Integer Linear Programming," *Journal of Guidance, Control, and Dynamics*, Vol. 25, No. 4, 2002, pp. 755–764.

[17] Guzman, J., "Tetrahedron Formation Control," *Proceedings of the 2003 Flight Mechanics Symposium*, NASA Goddard Space Flight Center, Greenbelt, MD, Oct. 2003.

[18] Clemente, D. C., and Atkins, E. M., "Optimization of a Tetrahedral Satellite Formation," *Journal of Spacecraft and Rockets*, Vol. 42, No. 4, 2005, pp. 699–710.

[19] Zanon, D. J., and Campbell, M. E., "Mission Objectives in Tetrahedral Formation Maneuvering," *Proceedings of the 2005 AIAA Guidance, Navigation, and Control Conference and Exhibit*, AIAA, Reston, VA, Aug. 2005.

[20] Curtis, S., "The Magnetospheric Multiscale Mission ... Resolving Fundamental Processes in Space Plasmas," NASA Science and Technology Definition Team for the MMS Mission, Rept. NASA/TM-2000-209883, 1999.

[21] Astrium, "Study of the Laser Interferometer Space Antenna: Final Technical Report," ESTEC Rept. LI-RP-DS-009, Dornier Satelletensysteme GmbH, 2000.

[22] Guzman, J., and Edery, A., "Mission Design for the MMS Tetrahedron Formation," *Proceedings of the IEEE Aerospace Conference*, IEEE, Piscataway, NJ, 2004.

[23] Petruzzo, C., "Orbit Design Status for Mission Phases 0,1,2," MMS Trajectory Design Status Review, NASA/GSFC, April 2001.

[24] Stern, D. P., "Systematic Indentification of Preferred Orbits for Magnetosperic Missions: 1. Single Satellites," *Journal of the Astronautical Sciences*, Vol. 49, No. 4, 2001, pp. 559–583.

[25] Stern, D. P., "Systematic Indentification of Preferred Orbits for Magnetosperic Missions: 2. The "Profile" Constellation," *Journal of the Astronautical Sciences*, Vol. 50, No. 2, 2001, pp. 149–171.

[26] Gim, D., and Alfriend, K., "Criteria for Best Configuration and Sub-Optimal Reconfiguration for MMS Mission," *Advances in the Astronautical Sciences*, Vol. 119, No. 1, 2004, pp. 947–968.

[27] Hughes, S., "Formation Tetrahedron Design for Phase I of the Magnetospheric Multiscale Mission," *Proceedings of the 2003 Flight Mechanics Symposium*, NASA Goddard Space Flight Center, Greenbelt, MD, Oct. 2003.

[28] Hughes, S. P., "Orbit Design for Phase I and II of the Magnetospheric Multiscale Mission (MMS)," *Advances in the Astronautical Sciences*, Vol. 118, No. 1, 2004, pp. 255–274.

[29] Paschmann, G., and Daly, P. W., *Analysis Methods for Multi-Spacecraft Data*, ESA Publications Division, The Netherlands, 1998.

[30] Guzman, J., and Schiff, C., "A Prelimary Study for a Tetrahedron Formation: Quality Factors and Visualization," *Proceedings of the AIAA/AAS Astrodynamics Specialist Conference*, AIAA, Reston, VA, Aug. 2002.

[31] Hughes, S. P., "Preliminary Optimal Orbit Design for the Laser Interferometer Space Antenna (LISA)," *Advances in the Astronautical Sciences*, Vol. 111, No. 1, 2002, pp. 61–78.

[32] Folkner, W. M., Hechler, F., Sweetser, T. H., Vincent, M. A., and Bender, P. L., "LISA Orbit Selection and Stability," *Classical and Quantum Gravity*, Vol. 14, 1997, pp. 1405–1410.