JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 32, No. 5, September–October 2009

# Maneuvering Spacecraft Formations Using a Dynamically Adapted Finite Element Methodology

Laura Garcia-Taberner*
*Universitat de Girona, 17071 Girona, Spain*
and
Josep J. Masdemont†
*Universitat Politècnica de Catalunya, 08028 Barcelona, Spain*

DOI: 10.2514/1.41198

**The formation reconfiguration or the deployment of multiple spacecraft is a key issue that demands a robust methodology safely avoiding collision hazards. In this paper we present a systematic procedure to perform proximity maneuvering that could also be applied to other general situations such as for loose confinements of spacecraft clusters inside a particular region. The problem is formulated and solved using a finite element approach with optimal control. Although the methodology is quite general and suits many different types of problems, the examples that have been considered focus on some basic orbital maneuvers for libration point mission concepts in the sun–Earth system.**

## I. Introduction

IN RECENT years, the idea of constellations or formations of spacecraft have had an important role in technology for some science and astrophysical missions. The use of small satellites with baselines of hundreds or thousands of meters as a virtual satellite can provide a significant increase of the image resolution with respect to the capabilities of a single spacecraft. Representative mission concepts, such as Darwin in the European Space Agency and the Terrestrial Planet Finder (TPF) in NASA [1–3] need to be able to synthesize a much larger aperture than the one that could be obtained using a single satellite.

Nevertheless, this new technology demands the resolution of some problems from the astrodynamical point of view, such as the deployment, the control, or the reconfiguration of the formation avoiding collision risks between satellites. This paper focuses on the reconfiguration of spacecraft formations in a general way, a problem that presumably will be addressed many times during the lifetime of a mission.

In addition, missions about libration points have had a growing interest since the International Sun–Earth Explorer 3 (ISEE3) was launched in 1978. ISEE3 was in a halo orbit around the sun–Earth $L_1$ point from 1978 to 1982, studying the interaction between the Earth's magnetic field and the solar wind. Since then, several spacecraft, such as Advanced Composition Explorer, Solar and Heliospheric Observatory, or Genesis, have taken advantage of the singular dynamics about the $L_1$ or $L_2$ Lagrangian points for observatory missions of the sun or the sky [4,5]. $L_1$ located between the sun and Earth provides an unbeatable place to study the sun. $L_2$, which keeps the sun, Earth, and moon essentially in the same direction, implying an open field of half of the sky at all times with a thermal stable environment, provides an incomparable place for stellar and deep sky observations. It is for this reason that important project concepts such as TPF or Darwin consider the feasibility of formation flight about the $L_2$ libration point in the sun–Earth system.

Currently, the existing literature on the formation flight trajectory design about the collinear libration points focuses mainly on rough estimates of the mission cost, such as in the works of Beichman et al. [3] and Gómez et al. [6] or on the control strategies for the formation. With respect to this, Marchand and Howell [7] and Folta et al. [8] consider the formation control of the Microarcsecond X-Ray Imaging Mission about $L_2$ and more control techniques can be found in [9] and references therein. Also Gómez et al. [10], Sánchez et al. [11], and Perea et al. [12] have studies concerning the transfer of the formation, suitable geometries, and control procedures.

On the other hand, reconfigurations and deployments have been mostly considered for formations about the Earth. Representative techniques of proximity maneuvering have been studied by McInnes [13] and Badawy and McInnes [14] by means of Lyapounov functions and by Wang and Hadaegh [15] and Beard et al. [16], considering rotations of the formations or using a sequence of simple maneuvers. In general, these techniques have to be adapted in some way to the particular reconfiguration problem under consideration. They require some tuning of the parameters or to select an appropriate sequence of motions in a set of permutation possibilities. Moreover, rotations of the formation for reconfiguration purposes can be considerably expensive.

In this paper we study a direct way and provide systematic methodology for the reconfiguration of spacecraft formations without the need of prescribing parameters more than boundary conditions or geometric constraints.

Essentially we discretize the problem using a finite element approach. Following this, the result is analyzed and solved in terms of optimal control, where the objective function is directly related to the $\Delta v$ expenditure of the satellites, and collision avoidance enters as natural constraints in the procedure. The finite element underlying methodology guarantees a robust, stable, and consistent approach as it is well documented in the vast bibliography on the subject, and from its countless applications. In particular, the accuracy of the approximate solutions provided by the methodology can be systematically controlled using the adaptive capabilities and the convergence properties of the method.

We focus our study in some examples about the sun–Earth $L_2$ libration point. However, the procedure could also be applied to study proximity maneuvering in free space, or about the Earth, with few changes. Also, the same methodology can be suitable to design the deployment or to estimate the control cost of a given formation.

Presented as Paper 296 at the 2007 AIAA/AAS Astrodynamics Specialist Conference, Mackinac Island, MI, 19–23 August 2007; received 24 September 2008; revision received 17 June 2009; accepted for publication 18 June 2009. 

*Ph.D. Student, Departament d'Informàtica i Matemàtica Aplicada; laura.garcia@ima.udg.edu.

†Professor, Institute for Space Studies of Catalonia and Departament de Matemàtica Aplicada I, School of Industrial Engineering of Barcelona, Diagonal 647; josep@barquins.upc.edu.

Other than satellites, the procedure could also be generalized to control robot devices which are moving under dynamical laws and avoiding collisions.

## II. Modeling the Problem and Methodology

To set out the problem, the scenario and examples we consider in this paper assume that the formation is in a nominal halo orbit of 120,000 km of $z$ amplitude about $L_2$ in the sun–Earth system. Also, for the sake of simplicity, the model we use for the computations is the restricted three-body problem (RTBP), but the methodology can be implemented in any force field (or in free space) by changing a few technical matters.

Let us consider a spacecraft under the gravitational attraction of two main bodies (also called primaries), such as the sun and a planet or a planet and a moon. The influence of the spacecraft on the primaries can be ignored, so they move under Keplerian motion (a circular orbit is chosen), whereas the study of the motion of the spacecraft under the influence of the primaries is known as the RTBP [17].

The RTBP equations of motion are usually considered in a noninertial adimensional reference frame, known as the synodic system. The origin of this coordinate frame is located in the center of mass of the system. The $X$ axis is defined by the instantaneous line joining the two primaries directed from the smallest primary to the larger one. The $Z$ axis is normal to the orbital plane of the primaries, in the direction of their angular momentum and the $Y$ axis is chosen orthogonal to the previous ones to have a positively oriented coordinate system. Also, adimensional coordinates are selected in the way that the unit of mass is the sum of the masses of the primaries, the distance between the primaries is the unit of distance, and the time unit is such that the sidereal period of the primaries is equal to $2\pi$. With all these considerations the RTBP equations of motion in the synodic system are

$$\ddot{X} - 2\dot{Y} = \frac{\partial \Omega}{\partial X}, \qquad \ddot{Y} + 2\dot{X} = \frac{\partial \Omega}{\partial Y}, \qquad \ddot{Z} = \frac{\partial \Omega}{\partial Z} \tag{1}$$

where

$$\Omega(X, Y, Z) = (X^2 + Y^2)/2 + (1 - \mu)/r_1 + \mu/r_2 + (1 - \mu)\mu/2$$

with $\mu$ being the mass of the small primary and $r_1$ and $r_2$ the distances from the spacecraft to the big and small primaries, respectively.

As it is also well known, the RTBP equations have five equilibrium points, known as the Lagrangian points. All of them are on the plane $Z = 0$. Three of them ($L_1$, $L_2$, and $L_3$) are on the $X$ axis and the remaining ones ($L_4$ and $L_5$) form an equilateral triangle with the primaries.

In the neighborhood of the collinear libration points there exist different types of periodic and quasi-periodic orbits. Among them are the periodic halo orbits, which are the ones we use as nominal base orbits in this work [18]. As a general remark, libration point orbits have been used and considered for many missions and have also quite a bit of theoretical support (see [4,5,18,19], and references therein).

Let us assume a formation of $N$ spacecraft in a libration point orbit (for instance, the halo orbit stated above) and denote by $\mathbf{X}_i$ the state (position and velocity) of the $i$th spacecraft of the formation with respect to the nominal point on the halo orbit. Because we work with formations with diameters of a few hundred meters, the size of the formation with respect to the halo orbit is very small and it is feasible to use the linearized equations about the nonlinear orbit. The equations governing this relative motion can be written as

$$\dot{\mathbf{X}}_i(t) = A(t)\mathbf{X}_i(t) \tag{2}$$

where $A(t)$ is the $6 \times 6$ Jacobian matrix of the equations of motion about the halo orbit and is also periodic.

We now start considering the problem of the reconfiguration of a formation of $N$ spacecraft in a given time span $T$. This is, we essentially assume that at some initial epoch we have the initial states (position and velocity) of the $N$ spacecraft and we want to end up in a given final state after the time span $T$, avoiding collision risks. Because each spacecraft is under the influence of a control, the equations of motion corresponding to the reconfiguration problem are

$$\begin{cases} \dot{\mathbf{X}}_i(t) = A(t)\mathbf{X}_i(t) + \bar{\mathbf{U}}_i(t), \\ \mathbf{X}_i(0) = \mathbf{X}_i^0, \ \mathbf{X}_i(T) = \mathbf{X}_i^T, \quad i = 1, \ldots, N \end{cases} \tag{3}$$

where $\mathbf{X}_i^0$ and $\mathbf{X}_i^T$ are the initial and final states of the $i$th satellite inside the formation and the controls $\bar{\mathbf{U}}_i(t)$, affecting the acceleration of the $i$th satellite, are of the form $\bar{U}_i(t) = (0, 0, 0, \bar{U}_i^x(t), \bar{U}_i^y(t), \bar{U}_i^z(t))^t$.

The main goal to be solved is to find the controls $\bar{\mathbf{U}}_1, \ldots, \bar{\mathbf{U}}_N$ which minimize the fuel consumption under the fundamental restriction of collision avoidance between satellites. However, other geometrical restrictions can be eventually added to the problem.

### A. Uncoupling the System of Equations

Although we could continue our methodology using Eqs. (3), because we are centering our study in libration point trajectories, we can take advantage of the geometrical characteristics of these orbits to obtain a numerically more efficient algorithm in terms of computational time. One can check that, at any time $t$, the matrix $A(t)$ of the equations of motion (3) has six eigenvalues in the following form: two of them are real with opposite sign and the other four are complex pure imaginary and conjugated in pairs. Working with other types of libration point orbits, these main characteristics are also maintained.

Let us denote by $\lambda_1$ the modulus of the real eigenvalues (they are $\lambda_1$ and $-\lambda_1$), and by $\lambda_2$ and $\lambda_3$ the modulus of the imaginary eigenvalues (they are $\lambda_2 i$, $-\lambda_2 i$, $\lambda_3 i$, and $-\lambda_3 i$), being $\lambda_2 > \lambda_3$. The modulus of these eigenvalues does not change significantly in the family of halo orbits with practical applications and they are always different from zero as can be seen in Fig. 1.

Using these facts we can introduce a time-dependent change of coordinates $\mathbf{X} = P(t)\mathbf{Z}$ where Eqs. (3) for any satellite (the index $i$ denoting the satellite has been dropped for clarity) are expressed in the form,

$$\begin{cases} \dot{\mathbf{Z}}(t) = D(t)\mathbf{Z}(t) + \mathbf{U}(t) \\ \mathbf{Z}(0) = \mathbf{Z}^0, \qquad\qquad \mathbf{Z}(T) = \mathbf{Z}^T \end{cases} \tag{4}$$

where $D(t)$ is the Jordan form of the matrix $A(t)$:

$$D(t) = \begin{pmatrix} \lambda_1(t) & & & & & \\ & -\lambda_1(t) & & & & \\ & & 0 & \lambda_2(t) & & \\ & & -\lambda_2(t) & 0 & & \\ & & & & 0 & \lambda_3(t) \\ & & & & -\lambda_3(t) & 0 \end{pmatrix}$$

By means of introducing the uncoupled coordinates it has been observed that the methodology is simpler and more efficient, reducing the total time of computations to about 20%. Nevertheless, we note that in doing this, we have to consider a new control, $\mathbf{U}(t)$, which corresponds to the control $\bar{\mathbf{U}}(t)$ after the change of coordinates applied. Because the matrix $P$ of the change of variables is computed numerically integrating $\dot{P}(t) = A(t)P(t) - P(t)D(t)$, all the components of the control $\mathbf{U}(t)$ can be different from zero. However, when the inverse change of coordinates is applied, we return to the desired controls $\bar{\mathbf{U}}(t)$, where the first three position coordinate values are zero.

Taking derivatives in the new equations of motion (4) we can reduce them to six uncoupled equations for the components $z_j$ ($j = 1, \ldots, 6$) of $\mathbf{Z}$:

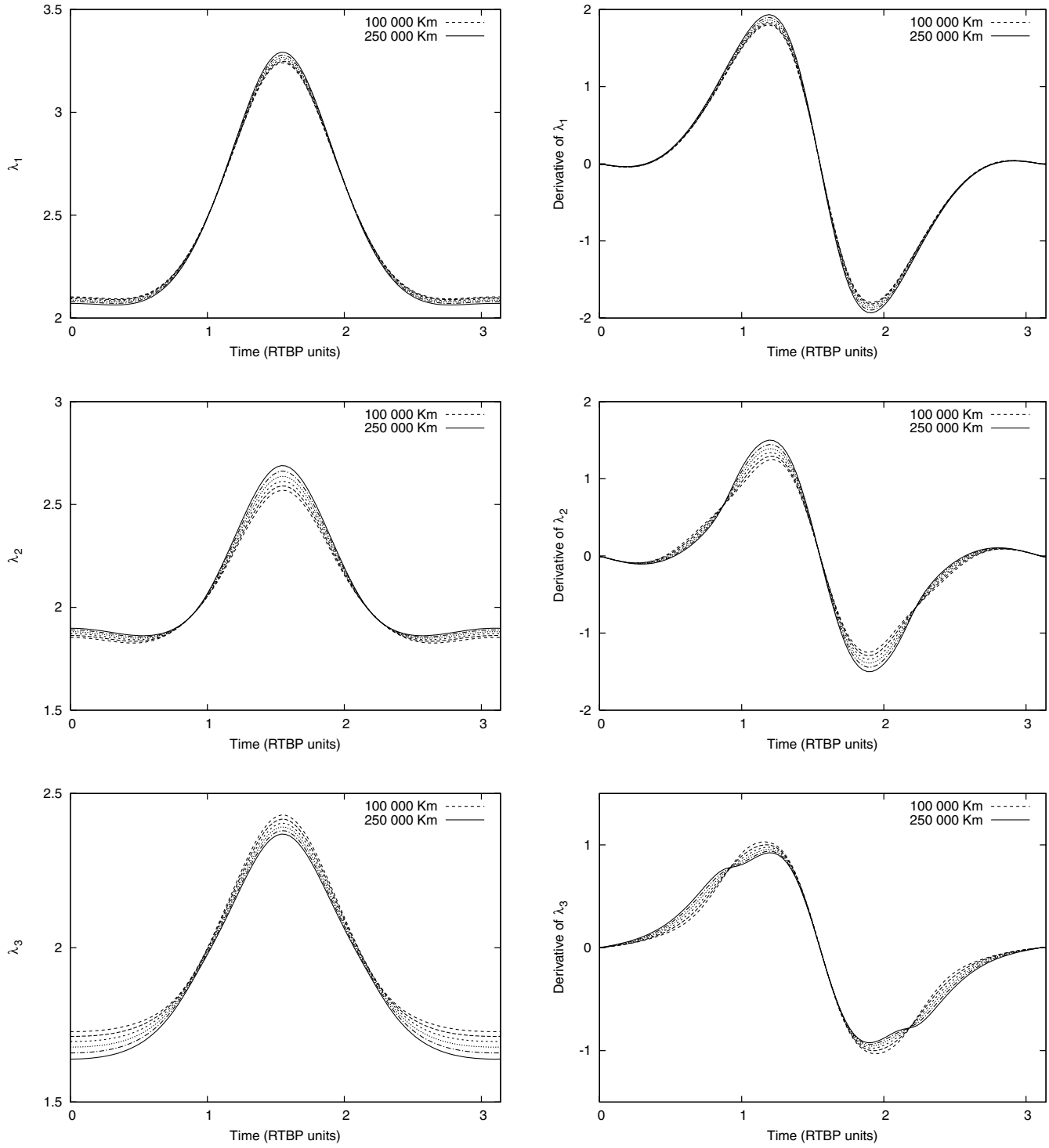


**Fig. 1 In the first column, from top to bottom, $\lambda_1$, $\lambda_2$, and $\lambda_3$, the modulus of the eigenvalues of $A(t)$ with respect to the RTBP time for halo orbits in the range from 100,000 to 250,000 km of $z$ amplitude. Note that they are always away from zero. In the second column of the panel we plot the derivatives of these eigenvalues.**

$$\begin{aligned}
\ddot{z}_1 &= (\dot{\lambda}_1 + \lambda_1^2)z_1 + u_1, & \ddot{z}_2 &= (-\dot{\lambda}_1 + \lambda_1^2)z_2 + u_2 \\
\ddot{z}_3 &= \frac{\dot{\lambda}_2}{\lambda_2}\dot{z}_3 - \lambda_2^2 z_3 + u_3, & \ddot{z}_4 &= \frac{\dot{\lambda}_2}{\lambda_2}\dot{z}_4 - \lambda_2^2 z_4 + u_4 \\
\ddot{z}_5 &= \frac{\dot{\lambda}_3}{\lambda_3}\dot{z}_5 - \lambda_3^2 z_5 + u_5, & \ddot{z}_6 &= \frac{\dot{\lambda}_3}{\lambda_3}\dot{z}_6 - \lambda_3^2 z_6 + u_6
\end{aligned} \tag{5}$$

Note that these equations do not have any problem of singularities, because the $\lambda_j$ ($j = 1, 2, 3$ are bounded away from zero as we have shown in Fig. 1. Moreover, because they are uncoupled, the computation of solutions will be simpler and the numerical implementation will be, in many aspects, independent from one equation to the other.

Again we note that the controls $u_j$ ($j = 1, \ldots, 6$) obtained in this second step of the procedure do not coincide with the ones of Eq. (4), but the components are related by means of simple formulas. For example, $u_1 = \lambda_1 U_1 + \dot{U}_1$.

When numerically implementing the computations it is convenient to use a Taylor approximation to have an expression for $U_1(u_1)$. The formula considers the value of the controls obtained from the current orbit in the iterative process of searching the optimal

value, $U_1(u_0) = U_0$. By using the local approximation it is only needed to provide good convergence to the methodology, and one can use

$$U_1(u_1) = U_0 + \left(-\frac{\lambda_1}{\dot{u}_1}U_0 + \frac{1}{\dot{u}_1}u_0\right)(u_1 - u_0)$$

truncated without small terms of order $(u_1 - u_0)^2$.

Similar formulas are obtained for the remaining components of $u_j$. In the case of controls associated with the still coupled variables in Eq. (4), such as with the case of $U_3$ and $U_4$, the expressions we find depend on the two controls, but again are well approximated with simple Taylor expressions in two variables.

### B. Finite Element Method Implementation

The six equations in (5) we obtain for each satellite after uncoupling the initial system (3) are essentially all of the same form (linear second-order equations but with different coefficients). Let us drop subscripts again, now corresponding to the coordinates. For each satellite we have shown how the initial problem has been reduced to find controls for a set of six scalar equations each of the form,

$$\begin{cases} \ddot{z}(t) + \lambda(t)\dot{z}(t) + \tau(t)z(t) = u(t), & \\ z(0) = z_0, & z(T) = z_T \\ \dot{z}(0) = v_0, & \dot{z}(T) = v_T \end{cases} \tag{6}$$

where $z(t)$ denotes one of the $z_j(t)$ $(j = 1, \ldots, 6)$ in Eq. (5) and $\lambda(t)$ and $\tau(t)$ are the functions representing the corresponding coefficients also in Eq. (5). We note that for each case, $\lambda(t)$ and $\tau(t)$ can be easily evaluated because they only depend on the eigenvalues of $A(t)$ represented in Table 1.

At this point it is also worth mentioning that if we want to consider the motion of the formation in free space we need only to take $\lambda(t) = \tau(t) \equiv 0$.

Let us deal with these equations using a finite element methodology with some special features. As it is well known, the finite element method (see, for instance, [20] for a general introduction) has been applied to many technical problems and benefits from a lot of theoretical support and active research in many aspects. However, in contrast to most implementations of this methodology, in our case we mesh the problem using a discretization in time instead of in space variables which is the common implementation. In addition, and following this approach, we treat and formulate the result of the modeling as an optimization problem.

We divide the time interval $[0, T]$ in a number $M$ of smaller intervals (elements). The mesh of elements does not need to be uniform; we can have elements of different lengths in the same trajectory and different time meshes for each spacecraft. This fact is useful when considering problems which are not symmetric with respect to the roles of the spacecraft inside the formation. For instance, some satellites have a bang–bang trajectory as the optimum way to archive the final state, whereas some others require intermediate controls. We note, however, that for the equations corresponding to a particular spacecraft we will take the same time mesh for its six associated coordinates (5).

We consider the elements indexed from 0 to $M - 1$. We focus the computations in the $k$th element corresponding to a time span $[t_k, t_{k+1}]$ (in particular, $t_0 = 0$ and $t_M = T$) and we start applying the Galerkin, Rayleigh–Ritz, finite element method in time to Eqs. (6). This is a very common implementation of the finite element methodology which is particularly suitable to our purposes, attending the regularity of the truncation. Moreover, for the second-order part of the equations it has associated a symmetric bilinear form, implying good numerical properties and condition number of the system of equations obtained from the discretization.

**Table 1 $\Delta v$ expenditures (cm/s) in the reconfiguration case of example III. Computations have been carried out using 20 elements for each trajectory**

| Security distance, m | Sat 1 | Sat 2 | Sat 3 | Sat 4 | Sat 5 | Total |
|---|---|---|---|---|---|---|
| 20 | 1.26 | 1.35 | 1.38 | 1.26 | 0.51 | 5.76 |
| 30 | 1.48 | 1.40 | 1.59 | 1.51 | 0.62 | 6.60 |
| 40 | 1.89 | 2.08 | 2.09 | 1.96 | 0.92 | 8.94 |
| 50 | 2.37 | 2.44 | 2.53 | 2.47 | 1.18 | 10.99 |

This implementation starts considering weighted residuals. This is, products of the residual times weight functions $\omega(t)$. Then the usual weak form on each element will be computed from the expression [20]:

$$\int_{t_k}^{t_{k+1}} \omega(t)(\ddot{z}(t) + \lambda(t)\dot{z}(t) + \tau(t)z(t))\,\mathrm{d}t = \int_{t_k}^{t_{k+1}} \omega(t)u(t)\,\mathrm{d}t, \qquad k = 0, \ldots, M-1 \tag{7}$$

According also with the local RTBP equations (2) considered for modeling the problem, linear elements are used for the discretization. Higher order elements could have been implemented in a similar way, but linear elements are already consistent with the truncated equations and provide good accuracy in the results we obtain.

Then, following the finite element formulation, we define the two interpolation functions in the $k$th element by means of the Lagrange polynomials:

$$\psi_1^{(k)}(t) = \frac{t_{k+1} - t}{t_{k+1} - t_k}, \qquad \psi_2^{(k)}(t) = \frac{t - t_k}{t_{k+1} - t_k}$$

where $t_k$ and $t_{k+1}$ correspond to the location of the two nodes of the linear element and are also the times at which the maneuvers will be applied. If we denote by $z^{(k)}(t)$ the approximation of the variable $z(t)$ inside the $k$th element that we are considering, the variational method states that this approximation has to be of the form

$$z^{(k)}(t) = z_k\psi_1^{(k)}(t) + z_{k+1}\psi_2^{(k)}(t)$$

where $z_k$ and $z_{k+1}$ are the nodal values that approximate $z(t_k)$ and $z(t_{k+1})$, respectively. Note that because $z^{(k)}(t)$ is linear with respect to time, the approximated trajectory in each element will be a straight line, and for each satellite, the final trajectory will be approximated by a linear spline, that is, a piecewise linear continuous function in time that converges toward the true solution of the problem when the diameter of the mesh decreases.

Let us also note how we can deal with the functions $\lambda(t)$ and $\tau(t)$. As it has been said they are zero if we consider the problem in free space, when the problem is formulated about a halo orbit they are related to the eigenvalues of $A(t)$ in Eq. (2). In this case a convenient way to handle them is to consider an approximation of the same order as the truncation error we are taking in our computations; that is, to use again the element interpolation functions and to write

$$\lambda(t) \simeq \lambda(t_k)\psi_1^k(t) + \lambda(t_{k+1})\psi_2^k(t)$$
$$\tau(t) \simeq \tau(t_k)\psi_1^k(t) + \tau(t_{k+1})\psi_2^k(t)$$

in which the numerical implementation only requires one to tabulate $\lambda(t)$ and $\tau(t)$ at the nodes of the elements.

Returning to the weighted residuals formula (7), and integrating by parts the term $\omega\ddot{x}$, we obtain the weak form,

$$\int_{t_k}^{t_{k+1}} (-\dot{\omega}\,\dot{z} + \lambda\omega\dot{z} + \tau\omega z)\,\mathrm{d}t + [\omega\dot{z}]_{t_k}^{t_{k+1}} = \int_{t_k}^{t_{k+1}} \omega u\,\mathrm{d}t$$

Then, the two equations associated with a given element are computed substituting $z(t)$ for $z^{(k)}(t)$ and selecting $\omega(t)$ as each one of the element interpolation functions. For the right-hand side we assume also that the control $u(t)$ inside the element is distributed in two nodal impulses at times $t_k$ and $t_{k+1}$, that we express in terms of the Dirac delta function:

$$u^{(k)}(t) = \delta_{t_k}(t)\Delta u_k^{(k)} + \delta_{t_{k+1}}(t)\Delta u_{k+1}^{(k)}$$

Performing the integrations we finally end up with system of equations associated to an element of our finite element model,

$$\begin{pmatrix} K_{1,1}^{(k)} & K_{1,2}^{(k)} \\ K_{2,1}^{(k)} & K_{2,2}^{(k)} \end{pmatrix} \begin{pmatrix} z_k \\ z_{k+1} \end{pmatrix} = \begin{pmatrix} \Delta u_k^{(k)} \\ \Delta u_{k+1}^{(k)} \end{pmatrix} \tag{8}$$

where

$$K_{1,1}^{(k)} = \frac{1}{t_{k+1}-t_k} - \frac{3\lambda(t_k)+\lambda(t_{k+1})}{12}(t_{k+1}-t_k) + \frac{2\tau(t_k)+\tau(t_{k+1})}{6}$$

$$K_{1,2}^{(k)} = \frac{-1}{t_{k+1}-t_k} - \frac{\lambda(t_k)+\lambda(t_{k+1})}{12}(t_{k+1}-t_k) - \frac{2\tau(t_k)+\tau(t_{k+1})}{6}$$

$$K_{2,1}^{(k)} = \frac{-1}{t_{k+1}-t_k} - \frac{\lambda(t_k)+\lambda(t_{k+1})}{12}(t_{k+1}-t_k) + \frac{\tau(t_k)+2\tau(t_{k+1})}{6}$$

$$K_{2,2}^{(k)} = \frac{1}{t_{k+1}-t_k} - \frac{\lambda(t_k)+3\lambda(t_{k+1})}{12}(t_{k+1}-t_k) - \frac{\tau(t_k)+2\tau(t_{k+1})}{6}$$

This system essentially states a relation between the nodal values $z_k$ and $z_{k+1}$, which are directly related to the positions of the reconfiguration trajectory, and the maneuvers $\Delta u_k^{(k)}$ and $\Delta u_{k+1}^{(k)}$, applied at the nodes of an element. We note that currently, $z_k$, $z_{k+1}$, $\Delta u_k^{(k)}$, and $\Delta u_{k+1}^{(k)}$ are unknowns, while the $K_{ij}^k$ are $3\times 3$ matrices that can be easily evaluated.

It is also important to remark that this system is valid for all the elements in the mesh of the trajectory, except for the first and last ones, where we must correct, respectively, for the initial departing and final arrival velocities of the spacecraft. In the first element, we have $\Delta u_0^{(0)} - V_0$ instead of $\Delta u_0^{(0)}$, and in the last element, the $\Delta u_M^{(M-1)}$ must be substituted by $\Delta u_M^{(M-1)} + V_T$, where $V_0$ and $V_T$ are the corresponding component [according to the represented $z_j(t)$ of Eq. (4)] of the departure and arrival velocities of the satellite in Eq. (3) but taking into account the changes of coordinates.

Assembling the element equations and taking into account that the positions at epochs $t_0 = 0$ and $t_M = T$ are given, we obtain a tridiagonal linear system linking the nodal values $z_k$ of the whole trajectory with the controls $\Delta u_k = \Delta u_k^{(k)} + \Delta u_k^{(k+1)}$ applied at the times corresponding to the interior nodes $t_k$, $k = 1, \ldots, M-1$:

$$\begin{pmatrix} K_{2,2}^{(0)} + K_{1,1}^{(1)} & K_{1,2}^{(1)} & 0 & & \\ K_{2,1}^{(1)} & K_{2,2}^{(1)} + K_{1,1}^{(2)} & K_{1,2}^{(2)} & & \\ \ddots & \ddots & \ddots & & \\ & K_{2,1}^{(M-3)} & K_{2,2}^{(M-3)} + K_{1,1}^{(M-2)} & K_{1,2}^{(M-2)} & \\ & 0 & K_{2,1}^{(M-2)} & K_{2,2}^{(M-2)} + K_{1,1}^{(M-1)} & \end{pmatrix}$$

$$\times \begin{pmatrix} z_1 \\ z_2 \\ \vdots \\ z_{M-2} \\ z_{M-1} \end{pmatrix} + \begin{pmatrix} K_{2,1}^{(0)} z_0 \\ 0 \\ \vdots \\ 0 \\ K_{1,2}^{(M-2)} z_T \end{pmatrix} = \begin{pmatrix} \Delta u_1 \\ \Delta u_2 \\ \vdots \\ \Delta u_{M-2} \\ \Delta u_{M-1} \end{pmatrix} \tag{9}$$

while for the first and last node, corresponding to the initial and final times of the reconfiguration process, the maneuvers associated are given by

$$\begin{aligned} \Delta u_0 &= K_{1,1}^{(0)} z_0 + V_0 + K_{1,2}^{(0)} z_1 \\ \Delta u_M &= K_{2,1}^{(M-1)} z_{M-1} + K_{2,2}^{(M-1)} z_M - V_T \end{aligned} \tag{10}$$

## III. Computational Procedure

We have seen that for each satellite ($i = 1, \ldots, N$) of the formation we can consider an associated mesh of $M_i$ elements in the time interval $[0, T]$, and for each trajectory, using some suitable variables, a set of $6 \times N$ uncoupled equations of the form (9) and (10) relate the maneuvers we apply with the nodal states of the orbits via a change of variables. That is, a set of

$$6N \sum_{i=1}^{N} (M_i - 1)$$

nodal positions $\mathbf{x}_{i,k}$ determine velocities $\mathbf{v}_{i,k}$ and maneuver $\Delta\mathbf{v}_{i,k}$ ($\Delta u$ in the uncoupled variables) for the $i$th spacecraft at the corresponding $k$th nodal time inside its mesh.

From now on we treat the problem using optimal control where the functional we minimize is directly related to the $\Delta v$ expenditure of the spacecraft. Collision avoidance and any other type of requirements enter in the method as constraint functions of the positions $\mathbf{x}_{i,k}$, velocities $\mathbf{v}_{i,k}$, or size in the maneuvers $\Delta\mathbf{v}_{i,k}$.

### A. Objective Function

The main goal of our procedure is to optimize the fuel consumption of the spacecraft. This is essentially to minimize

$$J_1 = \sum_{i=1}^{N} \sum_{k=0}^{M_i} \rho_{i,k} \| \Delta\mathbf{v}_{i,k} \| \tag{11}$$

where $\| * \|$ denotse the Euclidean norm and $\rho_{i,k}$ are weight parameters that can be used, for instance, to penalize the fuel consumption of selected spacecraft with the purpose of balancing fuel resources (here for clarity we consider that $\rho_{i,k}$ multiplies the modulus of the $\Delta v$, but in a similar way we can impose a weight on each component).

However, as is well known, this functional has numerical problems in the computation of the derivatives when a $\Delta v$ norm is small (and this is exactly what the procedure intends to produce ...). For this reason it is usual to consider

$$J_2 = \sum_{i=1}^{N} \sum_{k=0}^{M_i} \rho_{i,k} \| \Delta\mathbf{v}_{i,k} \|^2 \tag{12}$$

Essentially, our procedure starts finding an optimal solution for the $J_2$ functional and the result is taken as an initial seed for the minimization of $J_1$ by means of a remeshing algorithm that is discussed in the following sections.

However, some important remarks have to be said about obtaining the initial seed for $J_2$. For a given mesh, taking into account the assembled equations (9) and (10), and assuming no constraints, the optimal trajectory of $J_2$ is the solution of a linear system of equations for each spacecraft. Moreover, one can prove that the determinants of these systems are different from zero when the nodes of the mesh are evenly distributed [21].

This fact provides a natural way to proceed. For a given initial mesh with an even distribution of nodes, we first compute the optimal solution of $J_2$ without taking into account the constraints. This initial seed, in some way, sketches the "most convenient" trajectories of the reconfiguration process. Then we look for a solution of $J_2$ including collision avoidance constraints, and eventually others, using an optimization procedure starting with this initial seed. If necessary, we can also easily implement a continuation process with respect to the security distance, for instance, but for practical applications, and, in particular, for the examples considered in the last section of this paper, there is no need to follow this way (although it is convenient for improving CPU time).

#### 1. *General Approach to Remeshing*

Now, as a result of the minimization of $J_2$, we have an approximation for the optimal reconfiguration of the formation considering the $J_1$ functional. At the same time, we are able to detect the small

maneuvers that can affect its ill conditioning. Considering these facts, and with the purpose of computing the optimum of $J_1$, we have developed a technique which adds or subtracts nodes of the meshes in a controlled way.

In other words, in a first iteration the algorithm suppresses some of the nodes where the associated maneuvers are less than a given threshold $D_m$ and adds more nodes near the ones with maneuvers bigger than another threshold $D_M$. The idea of taking out only some of the nodes, and not all of them at once, is related to the fact that there exists the possibility that some consecutive small maneuvers could be replaced by a bigger one. Proceeding in this way, the algorithm also solves another practical problem: in case a spacecraft has a thrust level to maneuver restricted inside a certain range (i.e., a constraint on the minimum and maximum $\Delta v$ allowed in a node), the addition of neighbor nodes splits the maneuver in longer time spans.

#### 2. *Test and Procedure I: Taking Out Nodes Associated with Small Maneuvers*

Nodes with associated maneuvers near zero cause ill conditioning in the calculus of the derivative of Eq. (11). Test I is directed to detect them. To this end, the test compares all the maneuvers in the trajectories with a given threshold $D_m$. The current mesh discretization passes the test if all of them are bigger than $D_m$. Otherwise, some of the nodes are removed using procedure I.

Taking out nodes in procedure I is conditioned by the current $\Delta v$ values in neighbor nodes. When the candidate nodes to be removed are isolated, they are taken off. But in case of having a sequence of consecutive nodes failing test I, one must be careful, because we could end up with a discretization with a long time interval without nodes. For example, in Fig. 2 we represent a sequence of maneuvers corresponding to a certain spacecraft in a case. In the central part there are five consecutive maneuvers with magnitude smaller than the threshold. If we suppress all the nodes at once there would be a long time span without nodes, and this might not be realistic; moreover, it could add some bad behavior in the next iteration. Because of this, procedure I counts first the total amount of $\Delta v$ inside the time span. If the total amount is still less than the threshold, we assume that the nodes can be removed from the mesh; otherwise we decrease the density of nodes by a factor depending on the ratio between the total amount of $\Delta v$ and the threshold, as can be seen in the figure.

#### 3. *Test and Procedure II: Adding Nodes to Split Big Maneuvers*

In some cases, similar to when in an iteration we have maneuvers larger than the maximum ones allowed, $D_M$, it is necessary to add more nodes to the time mesh discretization. The task of test II is to detect maneuvers bigger than the threshold $D_M$, while procedure II has been implemented with the capability of splitting elements in smaller parts when we find nodes failing this test. This procedure is also the seed idea for computing the low-thrust continuous trajectory controls.

Essentially, if we have only one node with a maneuver greater than $D_M$, say at time $t_j$, we can rearrange the two elements sharing this node (the time interval from node $t_{j-1}$ to node $t_{j+1}$) in three equal elements. This is an efficient and easy way to reduce the value of the $\Delta v$ in the nodes. In case we have a sequence of maneuvers greater than the threshold, all the elements containing these nodes are split in two elements of the same length.

#### 4. *Test and Procedure III: Remeshing Using Relative Neighbor Values*

In our methodology, tests I and II, and their respective procedures, are iterated while nodes exist with maneuvers smaller than $D_m$ or bigger than $D_M$. However, after the application of this algorithm, one could end up with some undesired discretization. For example, if we consider a reconfiguration problem in free space without collision hazards, the optimal solution is a bang–bang control for each spacecraft. So in the final solution for this case all maneuvers are zero, except the first and the last ones.

Again the relative small maneuvers could cause Eq. (11) to be ill conditioned for the derivatives and the procedure to suffer because of convergence problems. For this reason, when tests I and II are passed, we check to see if in test III there exist small maneuvers when compared with the ones in their neighborhoods. In a positive case we apply procedure III to remove the nodes corresponding to these relatively small maneuvers.

However, there is another major advantage of implementing test III and its corresponding procedure for the limit case of bang–bang controls, or in cases where parts of the solution are close to bang–bang controls.

We note that the minimization of the functional $J_2$ cannot give the bang–bang solution. Moreover, after the first iteration of the methodology, most of the nodes containing the maneuvers that minimize $J_2$ cannot be removed, as will be shown in an example of Sec. IV, because they are over the minimum threshold $D_m$ of test I. Typically, the value of the functional $J_2$ evaluated in the bang–bang optimal control is very big. However, test III and its associated procedure provide a robust and efficient way to improve the convergence, and even if only a few nodes are eliminated by procedure I, all the interior nodes are removed in a few iterations.

To summarize, our methodology of optimal control search finite elements for formation flight (FEFF-DV) for the $J_1$ functional (11) follows the scheme represented in Fig. 3. First of all, we find an initial guess of the problem using the $J_2$ functional (12). Using it as an initial seed, and applying a remeshing methodology, we minimize the $J_1$ functional at the same time that we control the density of nodes in the time meshes of the spacecraft. Moreover, convergence can either be

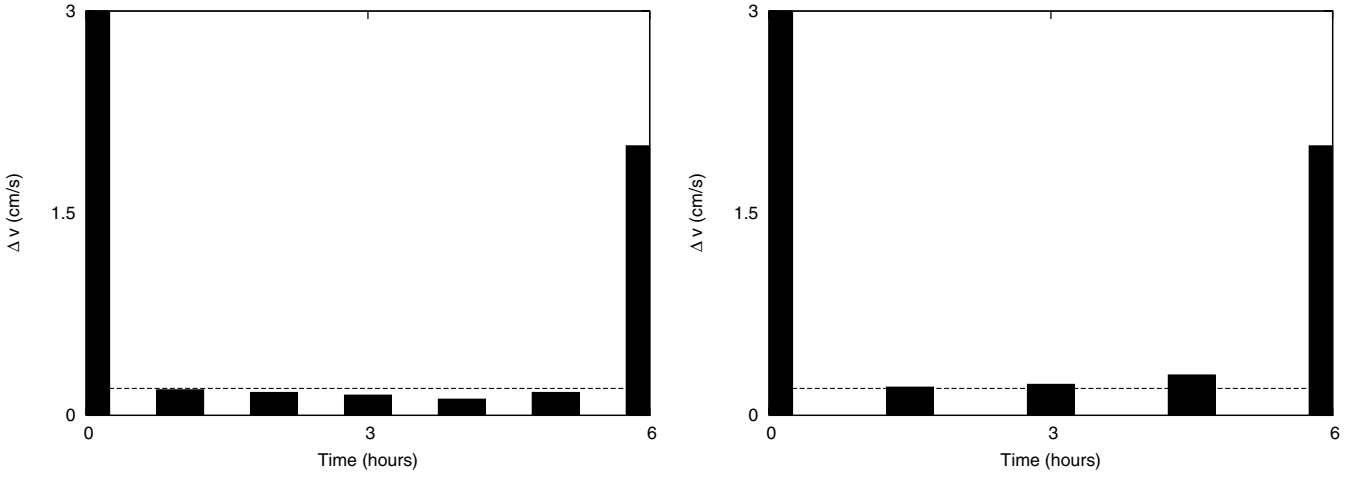


**Fig. 2 On the left, we represent a sequence of maneuvers obtained at some iteration of the optimization procedure for a particular satellite. We have some maneuvers smaller than the threshold (dotted lines) in the central part. In this case, instead of removing all the nodes, we change their density as displayed in the panel on the right.**

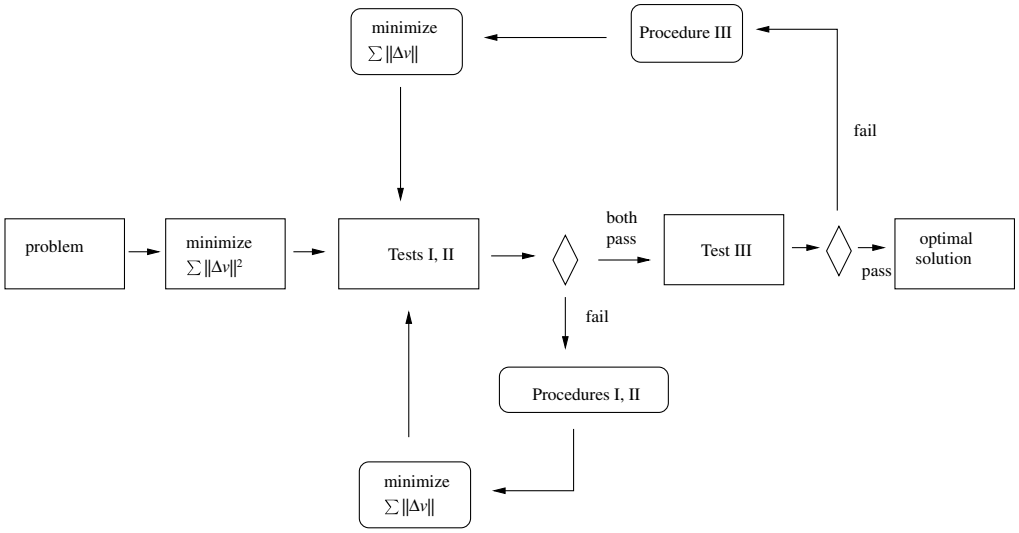


**Fig. 3 Scheme of the optimization methodology FEFF-DV for the $J_1$ functional.**

easily directed toward the limit cases of bang–bang control (when it is feasible) or toward low-thrust controls in general situations.

### B. Some Comments About the Collision Avoidance Implementation

During the reconfiguration process of the formation it is of utter importance to keep the satellites far from collision hazards. To this end, in our methodology we essentially consider a sphere centered on each spacecraft and we accomplish the requirement imposing that the spheres do not intersect in the final trajectories of the solution. Of course, one can select a different radius for each spacecraft, and in the deployment cases the radius can be considered time dependent.

Conceptually, the numerical implementation of this requirement, like others, inside our methodology is simple but the technical details are somewhat cumbersome. Here we just sketch the fundamental points. The interested reader can see [21] for further details.

Essentially, for a feasible point in the optimization process (that gives us the trajectories of the reconfiguration maneuver), one has to check whether the distance between any pair of spacecraft is big enough to fulfill the requirements or not. As a consequence, collision avoidance is a constraint that is introduced inside the optimization procedure in a "natural way"; that is, providing the optimizer of the function giving the distance between any pair of satellites as a function of the nodal position values $\mathbf{x}_{i,k}$ computed taking into account the changes of variables discussed in the methodology. Also, as it is very common in this class of problems, it is convenient to provide also the derivatives of the constraints with respect to the nodal values.

In the numerical implementation, both the distance function and the differential with respect to the nodal values are again computed in a "finite element manner" because, to determine these functions, it is just a matter of knowing them for pairs of elements inside the time mesh. In general, any constraint that can be evaluated by means of locally considering the elements of the mesh can be introduced in the process in a similar way.

## IV. Illustrative Examples of Reconfigurations

The FEFF-DV methodology has been coded using the C programming language and the problems are defined with an input file. In this section we present a few representative cases and the following results are obtained:

1) The first example models a limit case: we only have one spacecraft and, because there is no collision risk, the optimal trajectory is a bang–bang controlled solution.

2) In the second example we present a case based on the TPF formation concept where a bang–bang trajectory would produce a collision between two spacecraft. In this case we also show how we can direct the solution toward the other limit case: a low-thrust optimal trajectory.

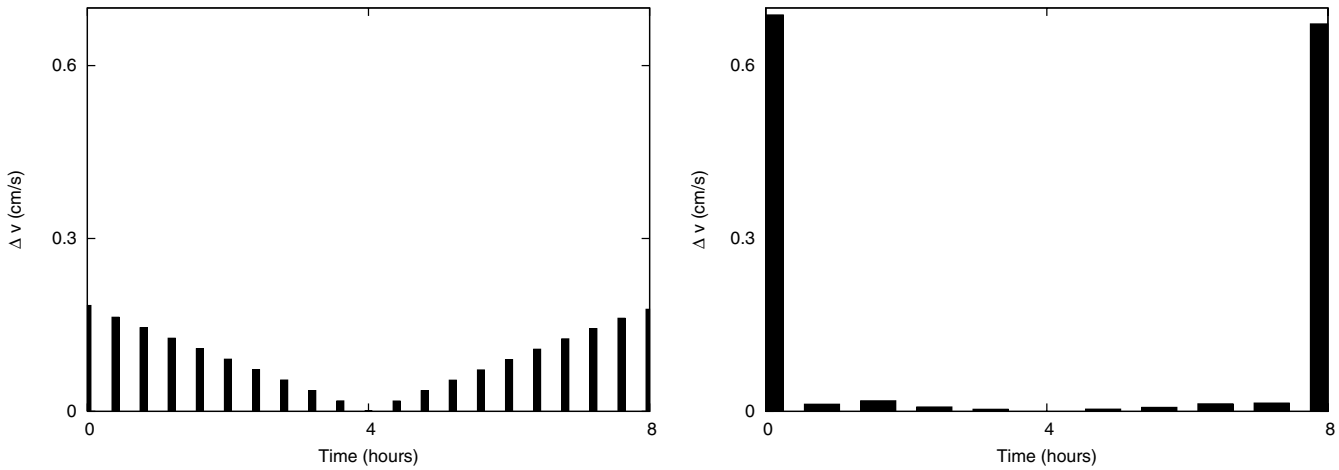


**Fig. 4 Two $\Delta v$ profiles obtained during the minimization sequence of example I, where the optimum is a bang–bang control. On the left we display the $\Delta v$ profile obtained when minimizing the functional $J_2$. Note that in this case all the maneuvers are significant, except the one corresponding to time 4. In the panel on the right, we show the $\Delta v$ profile obtained in the next iteration of FEFF-DV. It is the result of suppressing the node corresponding to time 4 by procedure I and then minimizing the functional $J_1$.**

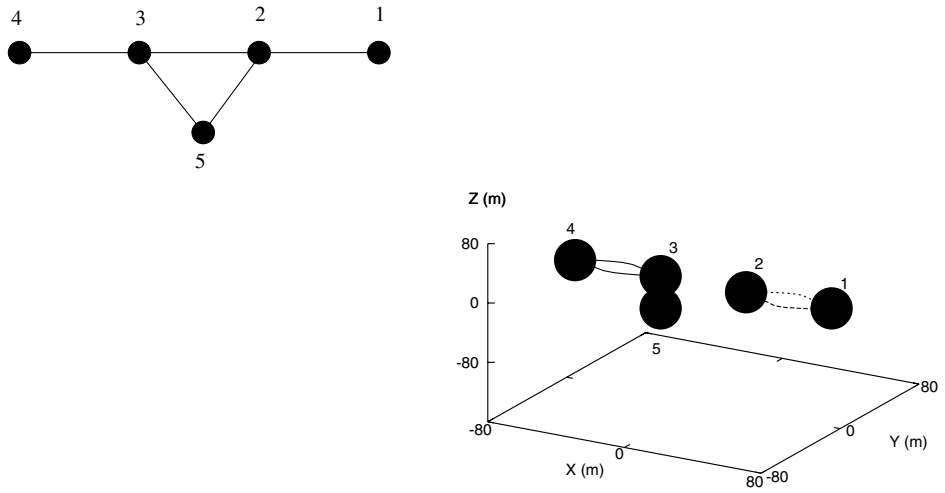


**Fig. 5 On the left a sketch the TPF configuration. On the right we show the orbits resulting from the optimal reconfiguration considered in example II.**

3) Finally, we consider a test case with five spacecraft and multiple collision risks, to stress test the algorithm.

### A. Example I: Parallel Shift Without Collision Risk

In many cases without collision risk, one can consider the orbits of the spacecraft independent from the others. In this example we consider a single spacecraft performing a parallel shift.

The reference frame for Eqs. (2) and (3) is aligned with respect to the RTBP reference frame, but the origin is located on the nominal point of the base halo orbit (when $t = 0$ this point corresponds to the "upper" position of the halo orbit, and this is when it crosses the RTBP plane $Y = 0$ with $Z > 0$). The initial condition for this example is taken 100 m from the base nominal halo orbit in the $X$ direction, and the goal is to transfer it to a symmetrical position with respect to the halo orbit in 8 h (this is 100 m in the opposite $X$ direction doing a parallel shift of 200 m for a formation).

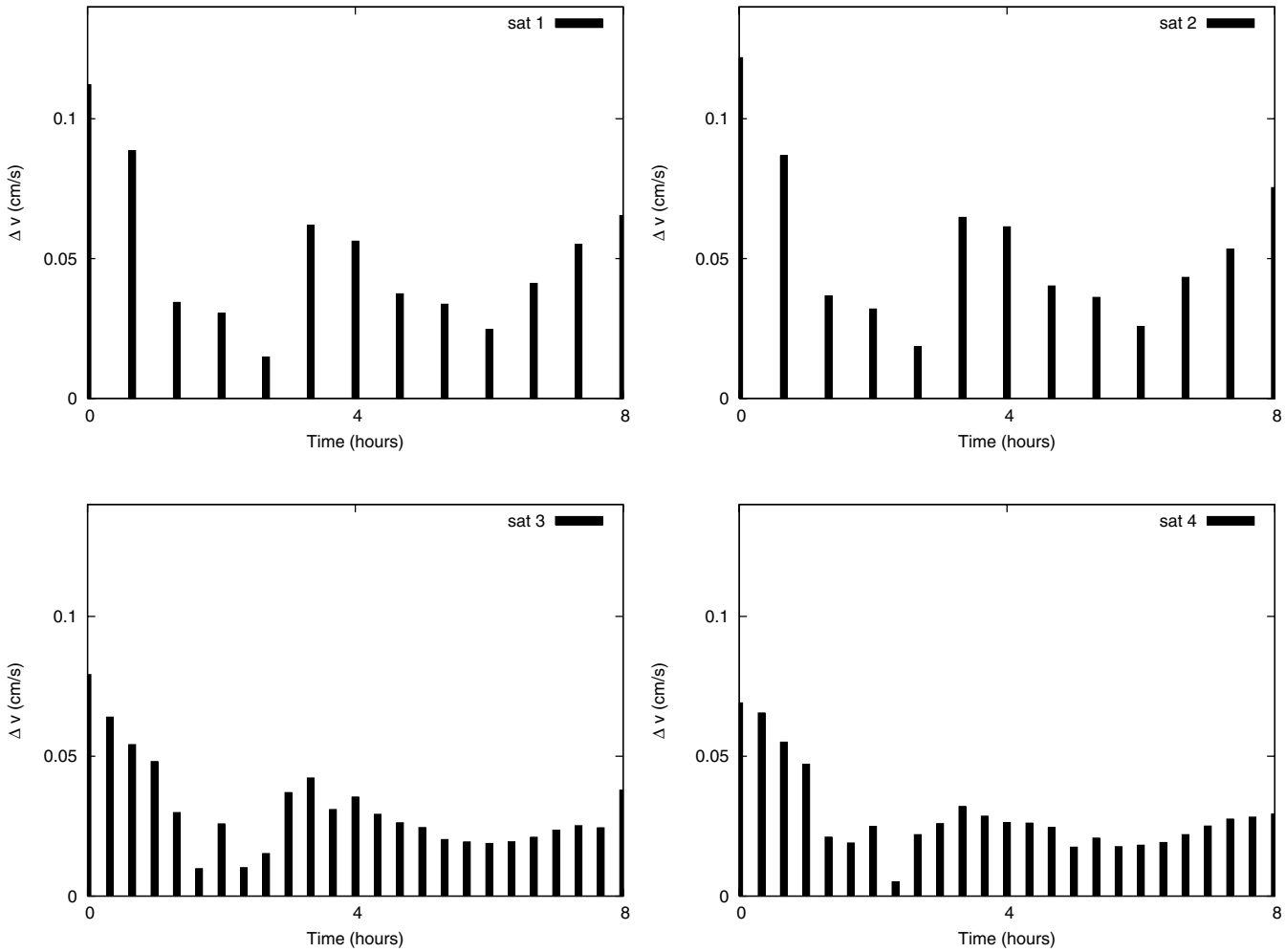


**Fig. 6 Maneuvers (cm/s) that must be applied to the spacecraft of example II when the number of elements chosen for the trajectories of satellites 1 and 2 is 12 and for satellites 3 and 4 is 24.**

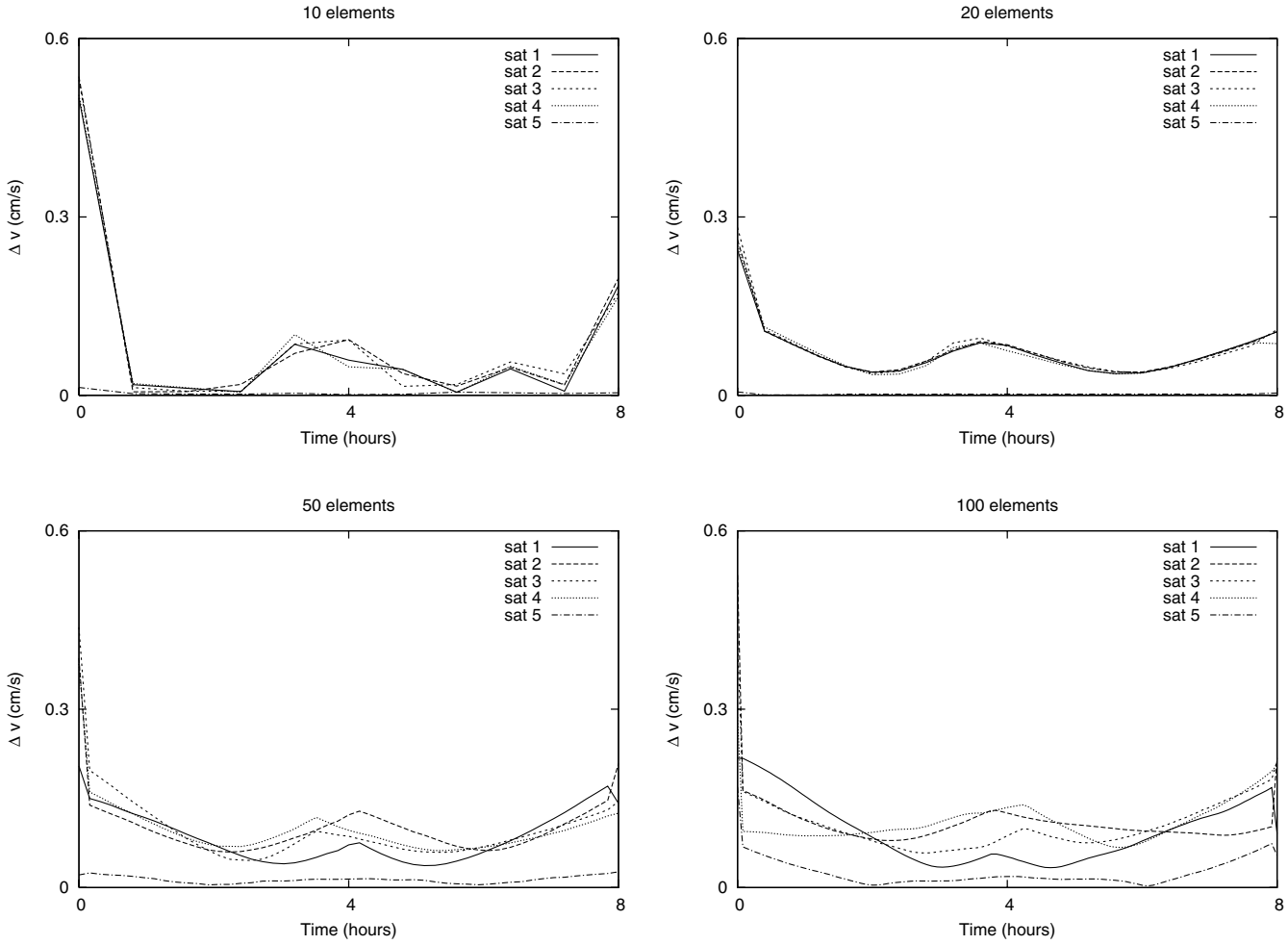


**Fig. 7 $\Delta v$ divided by the element length for the case example II, using 10, 20, 50, and 100 elements. The results tend to a low-thrust profile.**

For this particular case we obtain as a solution a bang–bang control with maneuvers of 0.69 cm/s at departure and arrival. However, it is not the purpose of this paper to study the costs associated with parallel shifts of this form, or the particular costs of the other examples shown in this section. They have to be considered as illustrative cases to comment on the convergence of the methodology. Depending on the direction and initial time, the cost of the parallel shift can change slightly, and, of course, they depend in a stronger way on the time spent in the maneuver.

As has been discussed in the previous section, a bang–bang control is a limit case, because the maneuvers have to be zero in all the nodes except for the first and last one, and small values of $\Delta v$ make the functional $J_1$ very ill conditioned for the calculus of its derivatives.

We start minimizing the $\Delta v$ square functional $J_2$ of Eq. (12). This minimization has no problems in the computation of the derivatives, and the optimal trajectory using 20 elements is found in less than a second. This trajectory, with the $\Delta v$ profile represented in the first panel of Fig. 4, is used as an initial seed for the minimization of $J_1$ following the scheme FEFF-DV of Fig. 3.

We note that this initial seed has only a maneuver near zero (the one at time 4 h). Test I detects it and the corresponding node is removed by procedure I (only this node was eliminated in this step). Then the result is sent to the minimization procedure of $J_1$ and we obtain the $\Delta v$ profile displayed in the second panel of Fig. 4 which is already rather close to the objective). In the following three iterations all the interior nodes are suppressed by procedure III delivering the bang–bang control in this way. For this example the value of $D_m$ has been set to 0.01 cm/s, and because the convergence has been very fast, no attempts to optimize the number of iterations with respect to $D_m$ have been pursued.

### B. Example II: Switching a Pair of Satellites in the Terrestrial Planet Finder

For this example we consider the TPF formation concept [2] of the NASA Origins Program. In general, it consists of five spacecraft contained in a plane. Four of them are evenly distributed in a baseline of about 150 m. The other one forms an equilateral triangle with the two interior ones of the baseline (see Fig. 5).

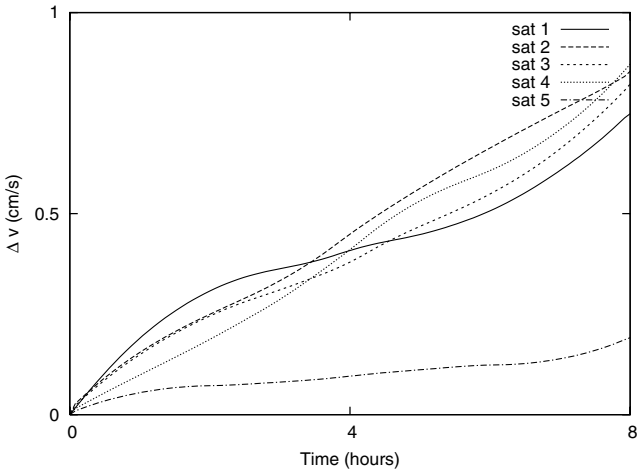


**Fig. 8 $\Delta v$ accumulated (cm/s) for each satellite in the TPF reconfiguration of example II. Computations have been done using 100 elements for each trajectory.**

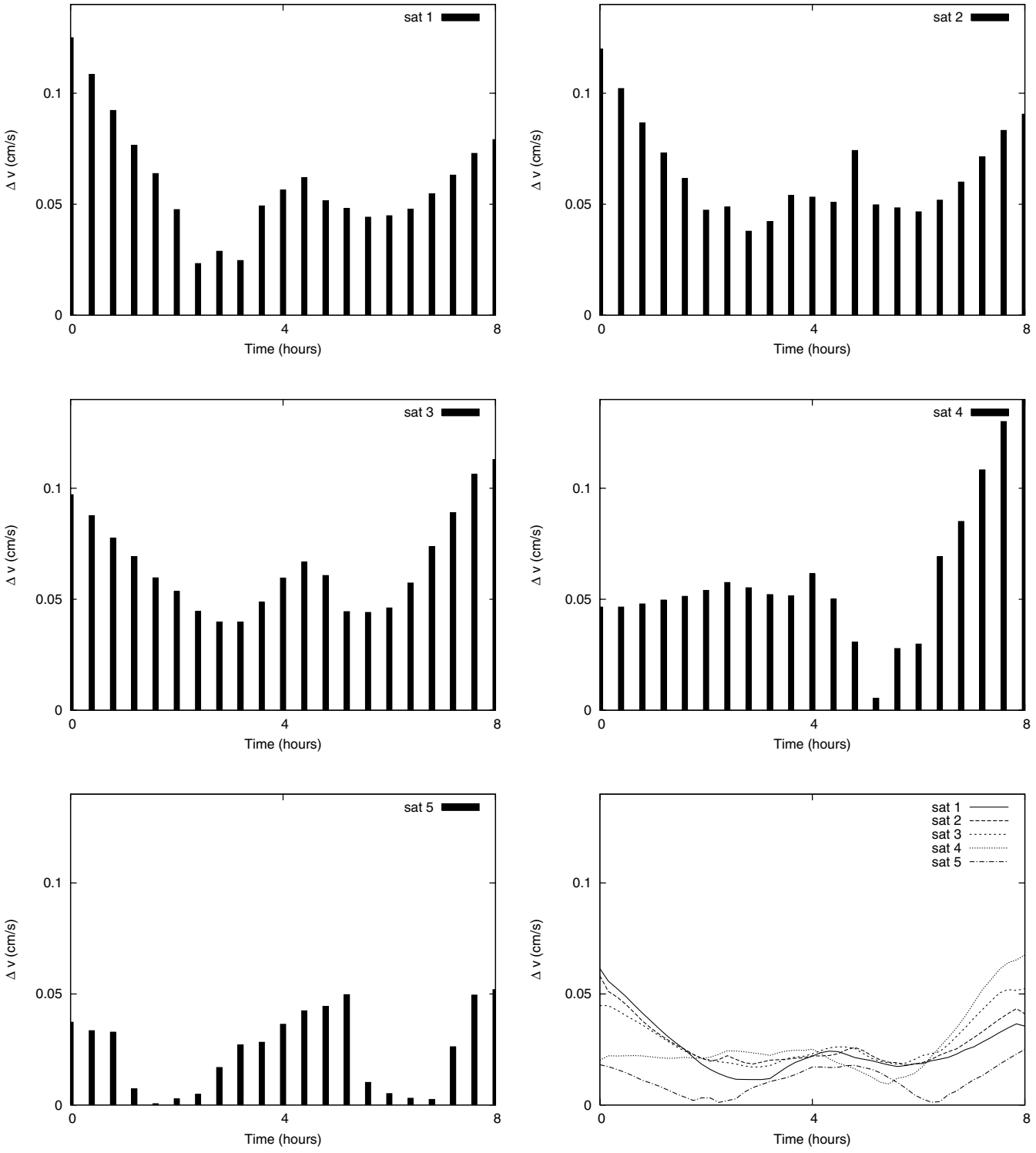


**Fig. 9 $\Delta v$ (cm/s) expenditures for the spacecraft considered in example III. Computations have been carried out using 20 elements in each trajectory and with a security distance of 20 m. As a reference, the plot on the right of the last row shows the optimal low-thrust profile for the reconfiguration.**

For this case we assume that the satellites are initially contained in the local plane $Z = 0$ discussed in the former example, with the interferometry baseline aligned on the $X$ axis (see Fig. 5). We simulate the switch between two pairs of satellites in the baseline: each inner satellite changes its location with the outer satellite which is closest in position (these inner satellites are maneuvered to attain outer positions and vice versa). Again we consider 8 h for the reconfiguration. The process of switching positions has a collision risk and simple bang–bang controls are no longer valid.

In this particular example, the optimal trajectories found by the algorithm are the ones also represented in Fig. 5, where the spacecraft move apart from the straight line to switch their positions. To illustrate the procedure we have considered a different number of elements in each leg. In fact, starting from the optimal seed resulting from the optimization of $J_2$ using six elements, FEFF-DV goes directly to the minimization of $J_1$. Then we consider 12 elements in each leg and the procedure again goes directly to the minimization of $J_1$. Finally, in one leg, we have taken 12 elements, and the total $\Delta v$ obtained is 0.77 cm/s, while for the other leg we have used 24 elements, and the total $\Delta v$ is reduced to 0.65 cm/s. In Fig. 6, we represent the maneuvers needed for this reconfiguration. Of course, when we increase the number of nodes in a mesh, but keep the ones of

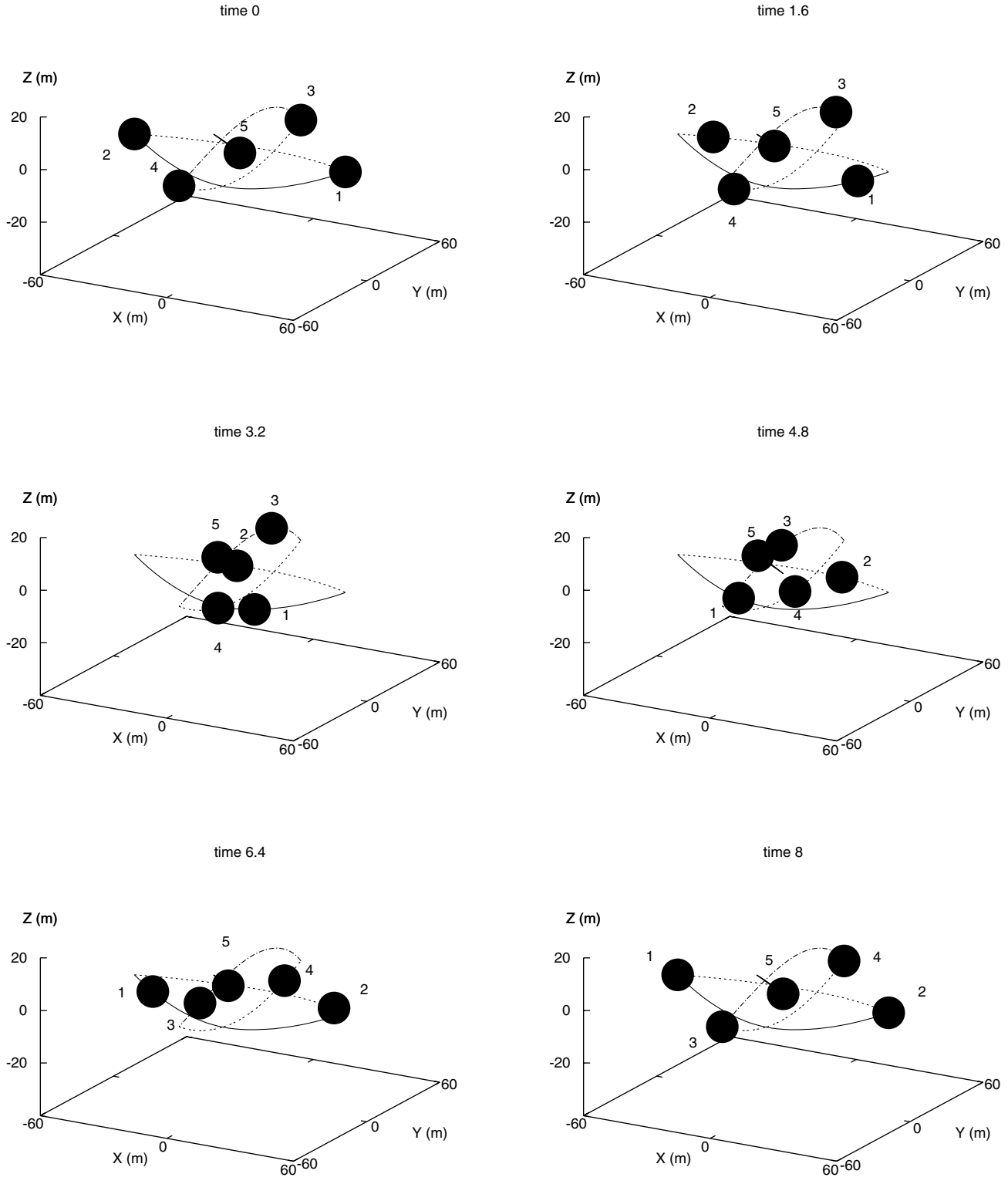


**Fig. 10 Sequence of the reconfiguration process for the spacecraft of example III. The time span between frames is 2 h.**

a previous step, the cost of the reconfiguration decreases (up to the low-thrust limit) and the maneuvers tend to be smaller. In this example, the leg of 24 elements has a double number of maneuvers of the leg with 12 elements, but roughly half of the size.

In this example of switching two spacecraft, the fuel consumption is similar for the four satellites participating in the process, due to near symmetry reasons. However, as in example I, we obtain slightly different results depending on the initial epoch or initial angle of the baseline inside the plane. Note that we could add some other constraints to this common scenario, such as geometrical constraints, such as maintaining the spacecraft inside a given zone or avoiding certain zones, or time constraints, not allowing maneuvers inside a certain period of time (just by means of not including nodes in the time span considered).

In the cases where the solution is not a bang–bang control, the methodology can be directed to compute the optimal value using low-thrust control which is the limit case. For this purpose we just need to increase the number of elements in the time mesh, maintaining, for instance, an even distribution. In this example, we do the computations with 10, 20, 50, and 100 elements (it is also much more efficient in terms of CPU time to gradually increase the number of elements rather than computing at once the low-thrust

control solution with a big number of them). In Fig. 7 we plot the $\Delta v$ divided by the element length (acceleration thrust) to see the convergence toward the low-thrust optimal profile. Additionally, the accumulated $\Delta v$ is shown in Fig. 8.

### C. Example III: Test Case with Multiple Collision Risks

The purpose of this example is to present a stress test case where multiple collision risks are avoided. At $t = 0$ we consider a square of 100 m of edge longitude contained in the same local plane, $Z = 0$, of the previous example. The square is centered on the nominal point of the base halo orbit and with the diagonals aligned with the coordinate axis.

A satellite is located on each vertex of the square and another one in the barycenter. The reconfiguration consists of switching the satellites of opposite vertices while keeping the one in the barycenter. In principle, without taking into account the collision avoidance constraints, all the spacecraft would collide in the center of the square.

The solution we obtain using 20 elements for each trajectory and considering a security distance of 20 m (10 m for the radius of the exclusion spheres) can be seen in Figs. 9 and 10. The six frames showing the position of the spacecraft are evenly distributed in the 8 h of reconfiguration time. We note that to avoid collisions the spacecraft in the center moves out of the plane while the ones in the vertices do similar motions to the case of the TPF example II. It is also interesting to note that for the two pairs of switching satellites the procedure has converged itself to a solution where each pair essentially uses a different plane for the trajectories.

It is not the purpose of this paper to deliver further conclusions on this particular case example, more than the robustness of the procedure and the affordable fuel expenditures we might have even when stressing the parameters. In Tables 1–3, we summarize the $\Delta v$ needs we obtain when we increase the security distance up to a level comparable with the size of the formation, and so, the reconfiguration trajectories are pretty far from the straight lines of a bang–bang control. The procedure is still able to obtain very reasonable costs, without any sign of divergence. Moreover, the computations have been repeated considering 20, 50, and 100 elements with few remarkable changes in the results. This implies that, in many situations, a few elements are enough to catch the main details of the process.

Although it is not necessary in this example (and in many other more complex scenarios that we have tested, but we have not included here due to a heavy representation or description), as in example II, a continuation process in the implementation of the procedure is feasible and natural. It can help in some extraordinary situations and in general speeds up the process. For instance, in this case example, optimal trajectories obtained with a security distance of 20 m have been taken as an initial seed to compute the solutions when this parameter has been increased. Finally, just to say that, proceeding this way, the computational cost associated with problems like this one is quite low in terms of CPU time and resources, needing only a few minutes on an ordinary laptop (Pentium V, 1.73 G-Hz, 1 MB RAM).

**Table 2 Same as in Table 1 but using 50 elements in each trajectory**

| Security distance, m | Sat 1 | Sat 2 | Sat 3 | Sat 4 | Sat 5 | Total |
|---|---|---|---|---|---|---|
| 20 | 1.19 | 1.29 | 1.31 | 1.21 | 0.50 | 5.50 |
| 30 | 1.43 | 1.38 | 1.56 | 1.47 | 0.61 | 6.45 |
| 40 | 1.83 | 2.01 | 2.03 | 1.92 | 0.89 | 8.68 |
| 50 | 2.34 | 2.41 | 2.49 | 2.45 | 1.15 | 10.84 |

**Table 3 Same as in Table 1 but using 100 elements in each trajectory**

| Security distance, m | Sat 1 | Sat 2 | Sat 3 | Sat 4 | Sat 5 | Total |
|---|---|---|---|---|---|---|
| 20 | 1.17 | 1.28 | 1.29 | 1.19 | 0.49 | 5.42 |
| 30 | 1.41 | 1.35 | 1.54 | 1.43 | 0.60 | 6.33 |
| 40 | 1.79 | 1.98 | 2.00 | 1.89 | 0.88 | 8.54 |
| 50 | 2.31 | 2.39 | 2.45 | 2.43 | 1.13 | 10.71 |

## V. Conclusions

In this paper we present a systematic way to compute reconfigurations of spacecraft formations implementing an optimal control problem in the basis of a finite element methodology. Although we focus on libration point orbits and we provide some numerical results, the main purpose of the paper is not to discuss particular data concerning the dynamics about the libration points, but to show the general capabilities of the methodology. Also, the methodology is general enough to be applied for spacecraft formations in other scenarios, or even to analyze the motion of vehicles under certain laws and constraints.

The vast amount of theoretical support underlying the finite element method has been the key point in developing a rigorous and robust algorithm, very suitable for small size formations like TPF or Darwin. The methodology we present is able to deal with complex reconfiguration trajectories at a low computational cost and can also be easily adapted to confinement problems for loose type formations or spacecraft clusters. Moreover, the convergence can be easily directed toward low-thrust or bang–bang controls, when possible. In the near future further research will be directed to incorporate adaptive remeshing techniques and nonlinear extensions, especially for large diameter formations.

## Acknowledgments

This research has been supported by the Spanish Ministry of Science and Technology European Regional Development Fund grant MTM2006-00478 and the European Marie Curie grant Astronet. The authors thank the referees for their comments that improved the manuscript in many aspects.

## References

[1] The Science and Technology Team of Darwin and Alcatel Study Team, "Darwin. The Infrared Space Interferometer. Concept and Feasibility Study Report," European Space Agency, ESA-SCI(2000)12, 2000, pp. iii+218.

[2] TPF Science Working Group, "The Terrestrial Planet Finder (TPF): A NASA Origins Program to Search for Habitable Planets," Jet Propulsion Lab. Publ. 99-003 [online database], http://planetquest.jpl.nasa.gov/TPF/tpf_book/index.cfm [retrieved May 1999].

[3] Beichman, C., Gómez, G., Lo, M. W., Masdemont, J. J., and Romans, L., "Searching for Life with the Terrestrial Planet Finder: Lagrange Point Options for a Formation Flying Interferometer," *Advances in Space Research*, Vol. 34, No. 3, 2004, pp. 637–644. doi:10.1016/j.asr.2003.05.032

[4] Dunham, D., and Farquhar, R., "Libration Point Missions, 1978–2002," *Libration Point Orbits and Applications*, World Scientific, Singapore, 2003, pp. 45–73.doi:10.1142/9789812704849-0003

[5] Canalias, E., Gómez, G., Marcote, M., and Masdemont, J. J., "Assessment of Mission Design Including Utilization of Libration Points and Weak Stability Boundaries. Final Report," ESA-ESTEC Ariadna Study [online database], http://www.esa.int/gsp/ACT/doc/ARI/ [retrieved 2007].

[6] Gómez, G., Lo, M., Masdemont, J., and Museth, K., "Simulation of Formation Flight Near Lagrange Points for the TPF Mission," American Astronautical Society, Paper 01-305, July 2001.

[7] Marchand, B. G., and Howell, K. C., "Control Strategies for Formation Flight in the Vicinity of the Libration Points," *Journal of Guidance, Control, and Dynamics*, Vol. 28, No. 6, 2005, pp. 1210–1219. doi:10.2514/1.11016

[8] Folta, D., Hartman, K., Howell, K., and Marchand, B., "Formation Control of the MAXIM L2 Libration Orbit Mission," AIAA Paper 2004-5211, Aug. 2004.

[9] Farrar, M., Thein, M., and Folta, D. C., "A Comparative Analysis of Control Techniques for Formation Flying Spacecraft in an Earth/Moon-Sun L2-Centered Lissajous Orbit," AIAA Paper 2008-7358, Aug. 2008.

[10] Gómez, G., Marcote, M., Masdemont, J. J., and Mondelo, J. M., "Zero Relative Radial Acceleration Cones and Controlled Motions Suitable

for Formation Flying," *Journal of the Astronautical Sciences*, Vol. 53, No. 4, 2005, pp. 413–431.

[11] Sánchez, M., Gómez, G., Masdemont, J. J., and Peñin, L. F., "Design and Analysis Capabilities of LODATO for Formation Flying Libration Point Missions," ESA Special Publication SP-654, 2008, pp. 1–8.

[12] Perea, L., Gómez, G., and Elosegui, P., "Extension of the Cucker-Smale Control Law to Space Flight Formations" *Journal of Guidance, Control, and Dynamics*, Vol. 32, No. 2, 2009, pp. 526–536.doi: 10.2514/1.36269

[13] McInnes, C. R., "Autonomous Proximity Manoeuvring Using Artificial Potential Functions," *ESA Journal*, Vol. 17, No. 2, 1993, pp. 159–169.

[14] Badawy, A., and McInnes, C. R., "On-Orbit Assembly Using Superquadric Potential Fields," *Journal of Guidance, Control, and Dynamics*, Vol. 31, No. 1, 2008, pp. 30–43. doi:10.2514/1.28865

[15] Wang, P. K. C., and Hadaegh, F. Y., "Minimum-Fuel Formation Reconfiguration of Multiple Free-Flying Spacecraft," *Journal of the Astronautical Sciences*, Vol. 47, No. 1, 1999, pp. 77–102.

[16] Beard, R. W., McLain, T. W., and Hadaegh, F. Y., "Fuel Optimization for Constrained Rotation of Spacecraft Formations," *Journal of the Astronautical Sciences*, Vol. 43, No. 3, 1999, pp. 339–346.doi:10.2514/2.4528

[17] Szebehely, V., *Theory of Orbits*, Academic Press, New York, 1967, Chap. 2.

[18] Gómez, G., Jorba, A., Masdemont, J., and Simó, C., "Dynamics and Mission Design Near Libration Points. Vol. 3, Advanced Methods for Collinear Libration Points," World Scientific, Singapore, 2000.

[19] Gómez, G., Llibre, J., Martínez, R., and Simó, C., "Dynamics and Mission Design Near Libration Points. Vol. 1, Fundamentals: The Case of Collinear Libration Points," World Scientific, Singapore, 2001.

[20] Zienkiewicz, O. C., and Taylor, R. L., *The Finite Element Method*, McGraw–Hill, New York, 1994, Chaps. 1, 9.

[21] Garcia-Taberner, L., Ph.D. Dissertation, Universitat Politècnica de Catalunya, 2009 (to be published).