Proximity Maneuvering of Libration Point Orbit Formations Using Adapted Finite Element Methods

Laura Garcia Taberner

Departament d'Informàtica i Matemàtica Aplicada
Universitat de Girona

Programa de Doctorat en Ciència i Tecnologia Aeroespacial

Memòria per aspirar al grau de Doctor
per la Universitat Politècnica de Catalunya

Certifico que aquesta tesi ha estat realitzada per la Laura Garcia Taberner i dirigida per mi.

Barcelona, 17 de setembre de 2009.

Josep Joaquim Masdemont Soler

A l'avi,
sigui on sigui la seva estrella

Contents

Resum

Aquesta tesi doctoral està estructurada en quatre capítols. El primer capítol comprèn un resum dels projectes de vol en formació que s'han tingut en consideració els últims anys, especialment els que estan planejats de situar-se al voltant dels punts de libració. En aquest capítol també fem un estat de l'art de les principals tècniques de reconfiguració de formacions de satèl·lits.

Les principals contribucions noves d'aquesta tesi es troben als capítols 2, 3 i 4. En el capítol 2 introduïm la metodologia general que s'utilitzarà en tota la dissertació. Aquesta metodologia està basada en una discretització del temps usant una aproximació en elements finits, que al mateix temps la fa factible d'incorporar en problemes d'optimització. En aquest capítol es consideren les equacions linealitzades al voltant d'una òrbita Halo. El problema d'optimització minimitza el funcional obtingut per la suma dels quadrats de les maniobres. Encara que aquest funcional no estigui directament relacionat amb el consum de combustible, es comporta bé a l'hora de minimitzar.

En el capítol 3 es segueixen utilitzant les equacions linealitzades al voltant de l'òrbita Halo, però ara el funcional que es minimitza és la suma dels mòduls de les maniobres, que està directament relacionat amb el consum de combustible. Com a conseqüència, la metodologia permet que es pugui convergir a controls bang-bang en el cas que sigui possible, o a avanç continu en les altres situacions.

En aquest capítol, el nostre objectiu no consisteix només en estudiar com fer les reconfiguracions, sinó que també considerem el problema d'obtenir una bona discretització per al nostre problema d'elements finits, i decidir quina és la millor malla per cada tipus de problema.

Finalment, al capítol 4 considerem problemes no lineals i incloem perturbacions. Comencem considerant les reconfiguracions en el problema restringit de tres cossos, per després veure com es comporta usant les efemèrides JPL. Aquests nous models canvien una mica les trajectòries dels satèl·lits respecte les que havíem obtingut en els capítols anteriors. Per corregir aquestes desviacions implementem una metodologia basada en afegir petites correccions a les maniobres que estan donades. També estudiem la magnitud de les maniobres que cal aplicar quan es produeixen errors d'execució en les maniobres nominals. Finalment, aquest capítol acaba amb altres aplicacions que es poden dur a terme usant la metodologia que hem desenvolupat.

Agraïments

Han passat uns quants anys des de que vaig començar aquesta tesi fins al final. Durant aquests anys és molta la gent que m'ha ajudat, ja sigui en la confecció de la tesi, ja sigui fent-me la vida més fàcil, o més feliç.

En primer lloc, m'agradaria agrair al meu director de tesi, per guiar-me durant tot aquest camí. Per ajudar-me en els passos amb paciència. No podria haver tingut un director millor.

D'altra banda, he de donar especialment les gràcies als meus pares, que m'han ajudat i acompanyat tot aquest temps.

I als meus companys del departament d'Informàtica i Matemàtica Aplicada.

I a tu, amb qui tantes converses hem tingut tots aquests anys, que vas comenar una mica més tard que jo, i segueixes en el camí, però que aviat també arribaràs.

A tu, amb qui els últims anys també hem parlat molt, tot i no veure'ns gaire sovint. Que sempre acabes venint a visitar-me a la meva terra, i quan vinc per la teva sempre he de tornar aviat.

A tu, que m'has vingut a rescatar tantes tardes per anar a berenar.

A tu, que cada cop que parlaves amb mi, em deies: "I la tesi, què?"

A tu, que quan vas aparèixer em vas espantar, però que amb el temps has acabat essent com algú de la família. Tu, que des de la teva juventut m'has ensenyat tantes coses, com ara que hi ha molta gent a qui agraden els petits (que en realitat són els grans) detalls.

A tu, que acostumes a venir a donar-me la mà esperant que t'expliqui alguna cosa.

A tu, que t'has afartat de venir a dir-me "Què fas? Què és això? I per què?" A tu, que sempre que veig Mart et recordo dient "Mira, l'estrella de l'Avi!"

A vosaltres, que heu anat variant, apareixent, i desapareixent. Tots únics, i tots genials. Vam començar amb vergonya, però he anat creixent amb vosaltres, mentre us veia passar de la infantesa a l'adolescència, i que ara us veuré passar a ser adults. Havia de ser al revés, però crec que m'heu ensenyat més coses vosaltres a mi que jo a vosaltres.

I, finalment, i no per això menys important, a tu, que m'has acompanyat aquests últims mesos, i que ets l'exemple més clar d'ajuda fent-me la vida més fàcil, més feliç.

Abstract

This doctoral dissertation is structured in four chapters as follows. The first chapter contains a summary of formation flying projects that have been taken into consideration since few years ago. We specially focus on the missions that have been planned to be located in a libration point regime. For completeness, this chapter also contains a general state of the art about the main reconfiguration techniques for satellite formations.

The main new contributions of the thesis are contained in chapters 2, 3 and 4. Chapter 2 introduces the general methodology that will be considered in all the dissertation. It is based on a discretization in time by means of a finite element approximation, and at the same time, is suitable to incorporate optimal control problems. In this chapter we study the reconfigurations using linearized equations about a nominal Halo orbit minimizing the functional given by the sum of the square of the magnitude of the maneuvers. This functional is not directly related to the fuel consumption, but has good properties concerning minimization and regularity.

In chapter 3 we are still working with the linearized model about the nonlinear orbit, but the functional that we optimize, given by the sum of the modulus of the maneuvers, is directly related to fuel consumption. As a consequence, the methodology can be tuned in such a way that, if possible, the user can choose to converge to bang-bang optimal controls (when possible) or to low thrust trajectories in general situations.

In this chapter, our objective is not only to study how the reconfigurations can be accomplished. We also consider the problem of obtaining good meshes for our finite element discretization, and up to a certain extent, to decide which is the best mesh for each kind of problem.

Finally, in chapter 4, we deal with non-linear and perturbed problems. In a first step we consider reconfigurations in the Restricted Three Body Problem and in a second one with JPL ephemeris. This fact slightly changes the trajectories of the spacecraft with respect to the ones obtained in the previous chapters. To correct for such deviations we design and implement a methodology based on adding small corrective maneuvers on top of the nominal ones. We also study the magnitude of corrective maneuvers that will need to be applied in case of errors in the execution of the nominal ones. Finally, this chapter ends with some other applications that can be performed using the methodology we have developed.

Chapter 1

Introduction

In recent years, the interest in constellations of spacecraft and formation flight has increased. One application of formation flight technology is separated spacecraft interferometry. The aim of the formations of spacecraft is to fly them in harmony, as if they were a single instrument, obtaining an image with a resolution equivalent to a bigger telescope. This fact gives the capability to use a bigger virtual instrument instead of a heavy and complex one that could be launched from Earth.

There are many advantages in the use of multiple spacecraft formations, being one of the main ones to be easier to construct multiple small spacecraft than a larger one. There is also the possibility of having a formation which performs more than one task, or the chance to use the spacecraft in different projects, only changing the shape of the formation. Moreover we have always the possibility of adding more spacecraft to the formation to obtain a virtual bigger or more powerful instrument.

The spacecraft inside a constellation are connected virtually and must work together as a single one. This fact needs some new advanced techniques of control to implement, such as the ability to autonomously coordinate and synchronize multiple spacecraft.

The aim of this work points towards how to reconfigure a formation of spacecraft in different situations, like deployment, rotations or change the pattern of the formation. The main way to proceed is done modelling the trajectory of each spacecraft with finite elements in time. A methodology that also is provided with a strong theoretical background of literature.

In this chapter, we study the general background of reconfiguration of spacecraft. We start with a summary of the projects that are considered to be in orbits about libration points. Then, we present some other reconfiguration methods that can be found in the literature.

1.1 Formation flying

The formation flying concept has been taken into account in some projects and missions, due to the advantages which gives in the creation of virtual antennas. The main one is the possibility of coupling two or more telescopes together to synthetically build an aperture equal to the separation of the telescopes.

However, the use of these technologies gives rise to some new difficult problems that must be solved, such as the one of maintaining the spacecraft within a certain distance with high accuracy (less than centimeters) during the observations.

Formation flying differs from constellation of spacecraft. A formation flying system must maintain the relative position and attitude to small margins. In formation flying, the results of the observation are performed using the position of the spacecraft. On the other hand, in a constellation of spacecraft, each spacecraft is maintained in an orbit, and the corrections of position only affect to this single spacecraft. Also, in the constellations of spacecraft, the communication of observations is done by each single spacecraft.

1.1.1 Formation flying projects

There are many kinds of formations, and in the following section, we comment a summary of the more important future projects. In these projects, there are some small diameter formations, which are the ones we are centering our work, such as Darwin, where the distance between the spacecraft is of hundreds of meters. In the other way, other formations, such as MAXIM, need a bigger formation, of thousands of kilometers between satellites.

Search for terrestrial planets: Darwin and TPF

Darwin (ESA) and Terrestrial Planet Finder (NASA) have been two formation flying projects which were planned to find planets with the ability to support life. The goal of the projects was to measure the size, temperature, and placement of planets similar to Earth in the habitable zones of distant solar systems.

The use of formation flying in the search for terrestrial planets is essential. To see planets around nearby stars would require a telescope of roughly 30 meters in size and this is well beyond the current limits of technology. For this purpose, the two projects were based on a formation of spacecraft with a virtual aperture of hundreds of meters.

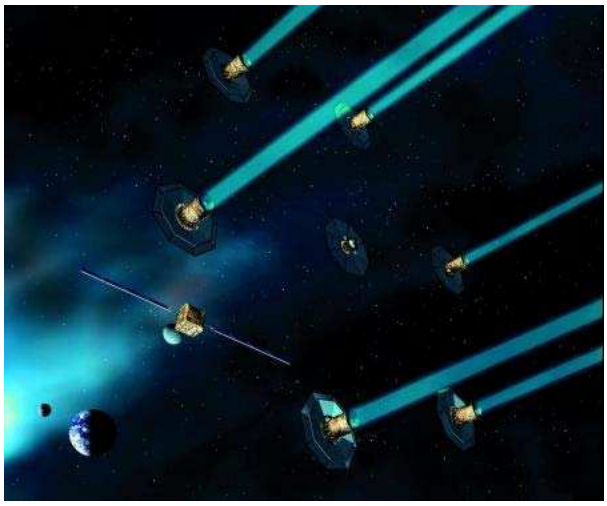

Figure 1.1: Schema of the Darwin formation with six telescopes. The telescopes are located in the vertices of an hexagon and the collector on the centroid of this hexagon.

Both projects were planned to be in a Halo orbit about the L_2 Sun - Earth+Moon point. Given the importance of the two projects and the similarity of them, it is possible that the two projects join in a single Darwin/Terrestrial Planet Finder. Unfortunately, due to technical problems, mainly in the thruster requirements, the join project has been deferred indefinitely since 2006, but the idea of the project is still valid to perform simulations of reconfigurations.

Darwin

The Darwin concept uses a formation of spacecraft composed by some space telescopes and a collector. The space telescopes are in the vertices of a regular polygon while the collector is located in the center of this polygon (see figure 1.1). The last design considered uses 4 spacecraft in the formation: three of the spacecraft should be space telescopes and should be at least three meters in diameter [31].

Terrestrial Planet Finder (TPF)

The Terrestrial Planet Finder concept sets a large baseline in the order of a hundred meters, composed by four telescope spacecraft with a diameter of 3 or 4 meters. These four spacecraft are combined with a collector, in

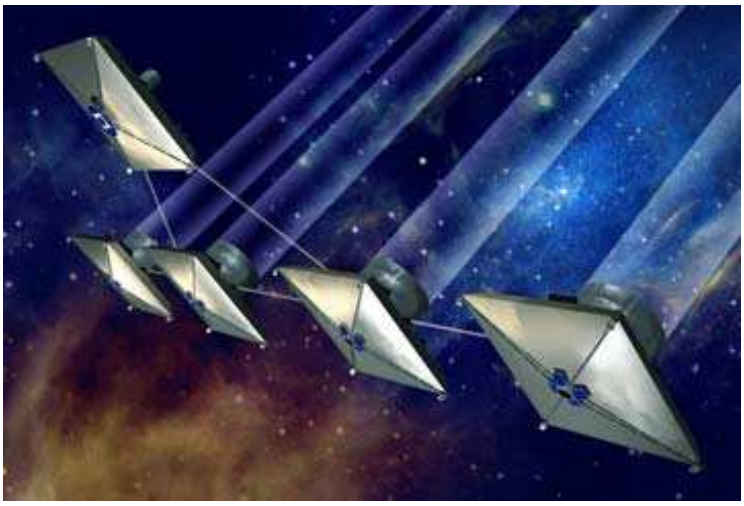

Figure 1.2: Artistic representation of the Terrestrial Planet Finder configuration.

an equilateral triangle with the two inner spacecraft of the baseline. The five spacecraft form a rotating formation (see figure 1.2) [32], and there were two target orbits initially considered for the formation: one was a Halo orbit about the L_2 Sun-Earth+Moon system, and the other one was a heliocentric orbit similar to the SIRTF heliocentric orbit. The Halo orbit about L_2 was finally chosen over the heliocentric orbit because the Halo orbit offers simpler telecommunications geometry, a lower insertion energy requirement and the option to launch ground-based spare spacecraft to orbit after the deployment of the original formation.

X-ray observatories

The observance of the galaxies of deep space are best done with X-ray telescopes. Nevertheless these telescopes need a long focal length, which is not possible to attain with a single spacecraft. For this reason, some projected missions to obtain images of deep space consider a formation of two spacecraft. The mirror, carried on one spacecraft, diffracts the rays and concentrates them onto the detectors, carried by the second spacecraft, which is orbiting behind the first.

There are several projects of X-ray observatories using formation flying to obtain a bigger focal length in this way: LISA (ESA and NASA), XEUS

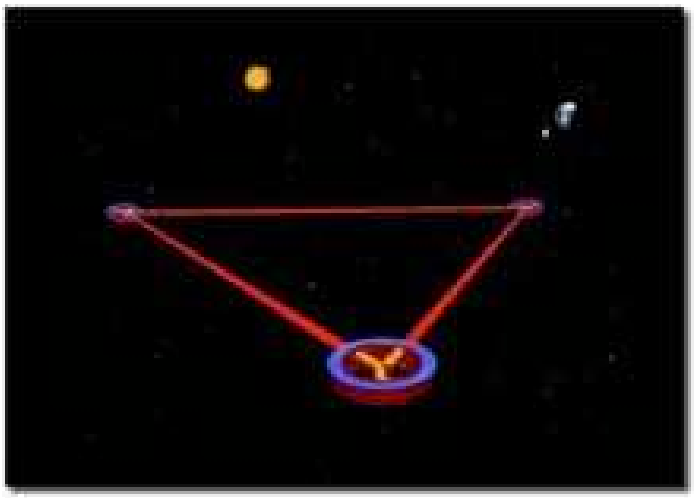

Figure 1.3: Representation of the LISA configuration.

(ESA), MAXIM (NASA), Simbol-X (CNES and ASI) and Constellation-X (NASA).

LISA

The Laser Interferometer Space Antenna (LISA) is a joint project of ESA and NASA to study the mergers of super massive black holes, tests Einstein's Theory of General Relativity, probes the early Universe, and searches for gravitational waves the primary objective [38].

The LISA formation is a large formation. It has three spacecraft in the vertices of an equilateral triangle, with a distance of five million kilometers between each spacecraft (see figure 1.3).

It is planned to put the formation in an orbit behind the Earth at distances of around 50 million kilometers (the orbit of LISA will be similar to the Earth orbit, but it would have a phase difference of 20 degrees from the Earth orbit). The plane of the triangle formed by the spacecraft is inclined at an angle of 60° to the plane of the ecliptic. This position is chosen to minimize the gravitational disturbances from the Earth-Moon system and to admit the communication with Earth.

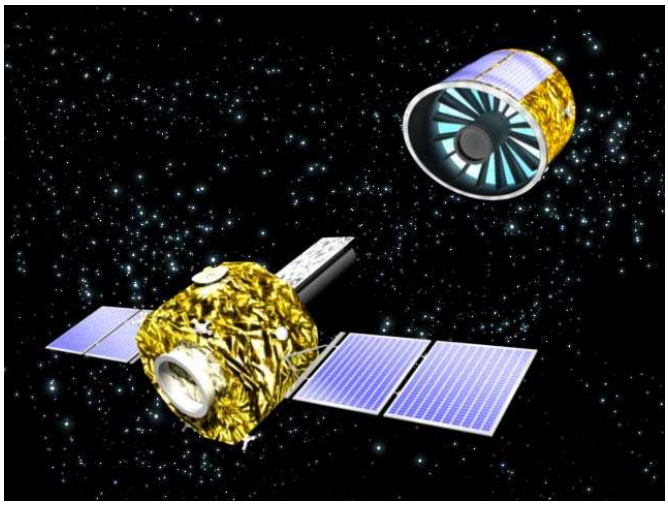

Figure 1.4: Representation of the XEUS formation with the two spacecraft: the mirror and the detector.

XEUS

The X-ray Evolving Universe Spectroscopy (XEUS) mission is designed to search for the first giant black holes that formed in the Universe. The objective is to look deep in space to see the galaxies that formed ten billion years ago. For this purpose, it needs a long focal length of around 50 meters, which is impractical for a single spacecraft. For this reason, the instrument is divided in two spacecraft [39].

The formation consists of two spacecraft, separated by 50 meters, to form a virtual X-ray observatory. One of the spacecraft carries the mirror and the other one carries the detectors (see figure 1.4). As Darwin and TPF, XEUS is planned to be placed into orbit at the L_2 Sun-Earth region.

MAXIM

MAXIM's goal is to image a black hole. The objective of the mission is to have a better resolution than the Chandra observatory currently in orbit about the Earth [44], which gives a 0.5 arc-seconds resolution. This purpose requires a big resolution, that will be obtained using a multitude of separate X-ray reflecting mirrors orbiting Earth in unison. The projected formation has 32 spacecraft with optical telescopes, located on the perimeter of a circumference of 200 meters and a spacecraft on the center of the circumference.

A collector is located 500 kilometers behind the mirrors, with a precision of 20 nanometer. This formation gives a 100 nano-arc-second resolution [40].

Simbol-X

Simbol-X is designed to study the interactions between black holes and their environment, or the acceleration of particles in the heart of pulsars and galaxy clusters.

For this purpose, a focal length of at least 30 meters is needed, and this is achieved considering a formation of two spacecraft flying separated by 30 meters [41].

Constellation-X

The Constellation-X Observatory is designed to investigate black holes, Einstein's Theory of General Relativity, galaxy formation, the evolution of the Universe on the largest scales, the recycling of matter and energy, and the nature of dark matter and dark energy.

The constellation is formed by a combination of four X-ray telescopes. The spacecraft have a diameter of 1.3 meters, and the data of the telescopes is collected by a collector satellite. The distance between the collector and the satellites with the lens is 10 meters, giving in this way a focal length of 10 meters [42].

1.1.2 Reconfiguration of formations

Most of the main objectives of the formations are to use the spacecraft as virtual telescopes. It is essential to make the observations maintaining constant relative distances between spacecraft. To maintain them, the formation must be controlled as it was a virtual single spacecraft.

However, when the spacecraft are not performing the observations, there is another kind of control maneuver which is critical for the tasks of the formation: the reconfiguration of the formation. The reconfiguration needs to be done with a minimum fuel consumption and without collision hazards between spacecraft.

In the initial position of the formation, spacecraft are usually in a cluster and they must achieve the final location inside the formation. In this first step, the deployment of the formation, needs some techniques to reconfigure them avoiding collision between spacecraft. This is the first time on the lifetime of the formation where the reconfiguration is needed. The formation might be reconfigured many times in its lifetime. Some reconfigurations, such as deployments and rotations are usual maneuvers, and other reconfigurations are more unusual, and must be done in exceptional cases.

Once the formation is working, there is also the need of reconfigurations. For telescope like missions, the most usual is the one which changes the pointing goal of the formation. This reconfiguration can be seen as a rotation of the formation [1], but in some cases there could be other trajectories of the spacecraft less expensive in terms of fuel consumption. We also study how to make rotations of the configuration using our methodology of finding optimal trajectories.

Other reconfigurations that can be useful are the ones which incorporate more spacecraft into a given formation, like in the case of adding a new spacecraft to perform a bigger antenna. One could also consider the possibility of suppressing a spacecraft of the formation, in case of failure.

Anyway, there are other scenarios where reconfigurations can be useful. During the lifetime of the formation, there could be some spacecraft that due to its relative position inside the formation spend a bigger consumption of fuel. Since the lifetime of the formation could run out when one of its spacecraft ends its fuel, the possibility of switching spacecraft during the lifetime of the formation, to equilibrate the fuel consumption of the members, may extend the total lifetime of the mission. This is also a critical case in the collision avoidance problem, because when swapping two spacecraft, if each spacecraft follow an optimal fuel consumption trajectory, the spacecraft would collide.

In any case, the reconfiguration of the formations should avoid collision risks, and it must do it taking into account that the fuel consumption must be the minimum possible. The aim of this work is to design a methodology to compute trajectories in all of these cases.

1.2 Reconfiguration methods in the literature

The reallocation of spacecraft inside a formation is an special case of control strategy for multiple objects. The objective of the reconfiguration problems is to move multiple objects (the spacecraft), giving each of the objects a final goal, and avoiding collisions between them. We note that these kind of problems of reconfiguration of spacecraft formations are similar to other optimization problems, such as the ones of guidance of robotic objects or underwater vehicle guidance.

In this work, we focus on the reconfiguration problem of spacecraft for orbits around libration points or in free space. Many of the science projects of formation flying are thought to be in these two scenarios. However there are some other formations (for example MAXIM [40] or CloudSat [43]) that are considered about the Earth. Basically, the differences between an scenario

and the others lies in the obtainment of the equations of motion for the spacecraft, but the methodology that we develop for reconfiguration could, in principle, be applied to all of them.

The bibliography referent to this subject can be divided in different sections, depending on the kind of reconfiguration. Here we report a summary of the most significant ideas in each of the cases of the vast bibliography related to formation flying.

1.2.1 Basic maneuvers

When we talk about basic maneuvers, we consider some kind of maneuvers that must be done many times in the lifetime of the formation and do not have collision risk in general. Maneuvers like this are the rotation of the formation (when the rotation angle is small, a bang-bang trajectory if it does not have collision risk), changes in the pointing direction (that can be considered as a kind of rotation), changes in the size of the formation (contractions and expansions, which again do no have collision risk with the bang-bang trajectory), or the deployment of a set of spacecraft.

Using geometrical properties.

Beard, McLain and Hadaegh [1], use the geometrical properties of rotations to obtain the reconfiguration. They consider the formation as a rigid body, and optimize the rotation point to arrive to the next configuration. After this, they calculate the trajectory of each spacecraft, and this gives the delta-v that must be applied to each spacecraft. Using this method, there is no problem of collision between the spacecraft, because mutual distances are maintained constant.

Initialization of a formation.

Hadaegh et al [15] have studied the problem of initializing a formation of N spacecraft into a pattern. In a first step, the spacecraft must find the location of the other spacecraft in the formation. The communication of the spacecraft is done in three phases: first of all an in-plane search, then and out-of-plane search and finally a near field search. This assures the communication between spacecraft and makes possible the final maneuvers to initialize the formation.

Initialization and resize of a formation using genetic algorithms.

Mailhé and Guzmán [22] study the problem of initializing a formation of spacecraft and to resize the formation. The trajectories are found in two

steps. First of all, a genetic algorithm is applied to find a first approach to the solution. This genetic algorithm gives a rough idea of the trajectory for the spacecraft, but in this first step there is no attempt to optimize the location or the number of maneuvers. Once they have this first approach, this solution is refined using Lawden's primer vector theory, to determine the location and number of maneuvers.

1.2.2 Methods using artificial potentials

The main idea of these methods is to add an artificial potential to the equations of motion, such as the goal of the spacecraft locations has a minimum on the potential and the other spacecraft (or the objects and places that must be avoided) have a maximum of potential. This is an usual method in control and guidance of robots.

Obtainment of the analytical solution via the second method of Lyapunov.

The idea of the artificial potential functions can be seen on [23], where C. R. McInnes uses this artificial potential function (which essentially are Lyapunov functions) to control proximity maneuvers in the presence of perturbations, avoiding collisions. In this case, the problem is solved analytically and the solution is found via the second method of Lyapunov. This method may require some tricky scaling of the potential in the final phase of the reconfiguration, because when the spacecraft is near the final goal, the potential reduces slowly and the time needed to arrive to the goal is high. This method can also add some artificial local minimum, which must be avoided in order to reach the goal. This procedure can be done by changing dynamically the potential functions.

Artificial potential functions to control a robotic arm.

The method of using artificial potential functions is used in many problems of robotic control. Richard Volpe uses the artificial potential in the guidance of a robotic arm [34]. In this case, the potential function has three different parts:

1. Attraction potential: he uses superquadrics to draw isopotential surfaces for the potential function. This potential creates a bowl shaped energy well centered in the goal point, which drives the arm to it.

2. Avoidance potential: uses functions which surround the objects and that have a limited range of influence and a spherical symmetry for large distances.

3. Approach potential: this potential is used when there is an object in the goal point and it is used to slow the arm velocity to obtain less impact forces.

Volpe adds all these potentials to obtain the artificial potential function, which guides the robotic arm. However, this process can create extra local minimums in the potential. This could give the problem that the robotic arm arrives to an extra minimum without achieving the goal. Again, he solves this problem by dynamically changing the potential shape, assuring the desired final position.

Use artificial potentials as a penalty term.

Junge and Ober-Blöbaum [18] include an artificial potential to control the formation. The potential is a logarithm of the distance between any two spacecraft, which assures that the two spacecraft repel each other. The trajectory for the spacecraft is found by minimizing a functional of the cost, and this potential is added to the functional as a penalty term.

1.2.3 Reconfigurations by minimizing a functional

The last group of reconfigurations are the ones which minimize a functional of the delta-v, and where the collision avoidance enters in the problem as constraints.

Model the trajectory using splines

Singh and Hadaegh [29] use a minimization problem to find the optimal trajectory. In this case, the trajectory of each spacecraft is modelled using cubic splines. The parameters of minimization for the functional are the coefficients of the splines and the function to minimize is a functional of the accelerations which they must apply to each spacecraft to follow the trajectory.

To avoid the collision between spacecraft, they define an exclusion sphere centered on each spacecraft and impose that these spheres can not intersect, except for a single point.

Deployment for LEO spacecraft formations

Thevenet and Epenoy [33] study the deployment of formations in orbits about the Earth, basically with LEO orbits. The problem of reconfiguration is performed in a time interval big enough to note the effect of J_2. The solution of the problem is found by applying an optimal control problem, using the optimality conditions derived from the Pontryagin's maximum principle.

Use of the Gauss pseudo-spectral method

Huntington, Benson and Rao [17] study the guidance of a formation of spacecraft in a tetrahedron. This method is studied to perform both in orbits about Earth or in the vicinity of the Lagrangian points. The trajectories can be the deployment of the formation, or the change of orbit.

The problem is formulated via a reconfiguration problem, and the solution is found via a pseudospectral method.

1.2.4 Our scenario: orbits about libration points

Most of the projects presented in section 1.1.1 are planned to be in orbits around the Sun-Earth libration points, specially in L_2. The Lagrangian points and the orbits about them are easy and inexpensive to reach from the Earth.

The libration point orbits also give other advantages for the future of the spacecraft located there. First of all, they can provide ballistic planetary captures, such as the one used by the Hiten mission in 1991. They can also provide Earth transfer and return trajectories, such as the one used for the Genesis mission. And they can provide interplanetary transport, such as with the Petit Grand Tour mission concept designing a low energy cost transfer orbit to visit Jupiter and several of its moons.

Moreover, the orbits about the Lagrangian points do not have only advantages from the astrodynamical point of view. They are also a good choice because of the communication system between them and the Earth (it is also cheap, since the libration orbits of the Earth-Sun system always remain close to the Earth - at a distance of roughly 1.5 million kilometers-). And also, both L_1 and L_2 have good properties in the observance of different targets (L_1 for the observance of the Sun, and L_2 for the observance of the space).

The Lagrangian point L_1 is a good site to study the Sun. Its position, between the Sun and the Earth, is a good place to study solar wind, and the Halo orbits, which avoid the interference from the Sun in the communication with Earth provide good nominal places to accomplish these observations. The first spacecraft to use this location was ISEE3, which 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 [7]. Since then, several spacecraft, such as ACE, SOHO or Genesis, have used the dynamics about the L_1 point as their nominal orbits in order to use the dynamics around the Lagrangian points to perform some observation missions [6].

On the other hand, the Lagrangian L_2 point of the Sun-Earth system is a good location to observe the space out of the Solar System, since the Earth,

Moon and Sun remain behind the telescope all the time. As the L_2 point moves around the Sun, during the course of one year, the entire celestial sphere can be observed. Some of the projects of formation flying, such as GAIA [16], Darwin [31] and Terrestrial Planet Finder [32] are planning to orbit around this point. The orbits about L_2 Sun-Earth system give great thermal stability, which is suitable for non-cryogenic missions, such as the ones with highly precise visible light telescopes. Additionally, the libration orbits about the L_2 of the Earth-Moon system can be used to establish a permanent communications link between the Earth and the hidden part of the Moon.

Currently, the existing literature on formation flight trajectory design about the collinear libration points focuses mainly on rough estimates of the mission cost, like in the works of Beichman, Gómez, Lo, Masdemont, Museth and Romans [2, 14] or on the control strategies for the formation. With respect to this, Folta, Hartman, Howell, Marchand [20, 10] consider the formation control of the MAXIM mission about L_2 and more control techniques can be found in Farrar, Thein and Folta [8] and references therein. Also Elosegui, Gómez, Marcote, Masdemont, Mondelo, Perea and Sánchez have studies concerning the transfer of the formation, suitable geometries and control procedures (see [13, 28, 26]).

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 [23, 24] by means of Lyapounov functions and by Hadaegh, Beard, Wang and McLain [35, 1] considering rotations of the formations or using a sequence of simple maneuvers as it has been discussed previously.

Chapter 2

Fundamentals of the Methodology

In this chapter, we consider the reconfiguration of N spacecraft about a nominal Halo orbit, and compute some reconfigurations of the formation in a fixed time T. These reconfigurations give a first approach for the reconfiguration using a complete model. The methodology to obtain the trajectories takes into account the collision avoidance between spacecraft and minimizes a functional related to fuel consumption. In this chapter, we present the general methodology to compute the reconfiguration, applied to linearized equations about a Halo orbit. In next chapters we will use this methodology applied to a more realistic model, including the JPL ephemeris and a functional more related to fuel consumption. We also present some results of the different cases we have studied, that will be revisited in the following chapters.

2.1 Modelling the problem

2.1.1 The Restricted Three Body Problem

Consider a spacecraft under the gravitational attraction of two big bodies, such as the Sun and a planet or a planet and a moon. The attraction of the spacecraft on the big bodies can be ignored, so the two large objects (also called primaries) move in a Keplerian motion, while the spacecraft moves attracted by the two larger bodies, this is known as the Restricted Three Body Problem (RTBP).

To study the RTBP, it is usually chosen an adimensional reference frame, known as the synodic system, where the units are such as the sum of the

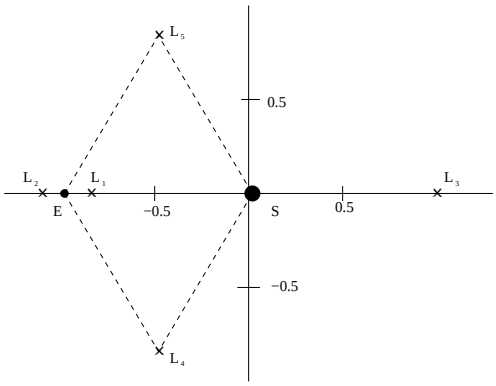

Figure 2.1: The five Lagrangian points

masses of the primaries and the distance between the primaries are equal to 1 and the period of the primaries is equal to 2π. In this synodic coordinate system, the origin is on the center of mass of the system. The X axis is located in the line defined by the two primaries, from the smallest primary to the larger one. The Z axis is normal to the rotation plane, in the direction of the angular momentum of the primaries, and the Y axis is chosen orthogonal to the previous ones in order to have a positively orientated coordinate system.

In this system of coordinates, the equations of motion are:

$$\ddot{X} - 2\dot{Y} = \frac{\partial \Omega}{\partial X}, \quad \ddot{Y} + 2\dot{X} = \frac{\partial \Omega}{\partial Y}, \quad \ddot{Z} = \frac{\partial \Omega}{\partial Z}, \tag{2.1}$$

where $\Omega(X, Y, Z) = (X^2+Y^2)/2+(1-\mu)/r_1+\mu/r_2+(1-\mu)\mu/2$, being μ the mass of the small primary and r_1 and r_2 the distances from the spacecraft to the big and small primaries respectively.

2.1.2 Lagrange points and Halo Orbits

As it is well known, the RTBP equations have five equilibrium points, known as the Lagrangian Points. All of them are in the plane Z=0. Three of them (L_1, L_2 and L_3) are in the X axis and the remaining ones (L_4 and L_5) form an equilateral triangle with the primaries (see figure 2.1).

The L_1 point is located in (X_1,0,0), where $X_1 = -1+(\mu/3)^{1/3}-1/3(\mu/3)^{2/3}+O(\mu)$, and L_2 is located in (X_2,0,0), with $X_2 = -1-(\mu/3)^{1/3}-1/3(\mu/3)^{2/3}+O(\mu)$. In the Sun-Earth+Moon system, both L_1 and L_2 are located approximately 1.5×10^6 km from the Earth.

The Lagrange points L_1 and L_2 are in good places for doing astronomical observations. Since the L_1 point is located between the Earth and the Sun, it

is a good place for observations of the Sun, like the ones made by the SOHO [45]. On the other hand, the L_2 point is located in a point which has always the Sun, the Earth and the Moon in the same direction, and a spacecraft in that point can view half of the sky without these interferences all time.

Spacecraft that have been sent to these two Lagrangian points are not positioned exactly in the point. In the case of the L_1 Lagrange point, the communication between the spacecraft and the Earth would be impossible, due to the interference from the Sun: the signal of a spacecraft would be indistinguishable compared with the radiation of the Sun. In the case of the Sun - Earth+Moon L_2 point, the communication would be impossible because the Moon would be in between the spacecraft and the Earth.

For these reasons, spacecraft are positioned in orbits about these Lagrangian points, in order to avoid eclipses from the Sun and from the Moon. We can also consider the advantage to put more than one spacecraft near a Lagrangian point, in different orbits. The orbits which are usually chosen are Halo orbits or Lissajous orbits [6, 3].

Halo orbits are a family of periodic three-dimensional orbits that have been taken into account for nominal orbits in missions such as SOHO [6] or ISEE3 [7]. The equations of motion about this orbit can be linearized, and we obtain:

$$\dot{\mathbf{X}}(t) = \mathbf{A}(t)\mathbf{X}(t). \tag{2.2}$$

Since Halo orbits are periodic orbits, $\mathbf{A}(t)$ is also a periodic matrix. The matrix $\mathbf{A}(t)$ has as well some properties related to the characteristics of this kind of orbits: for a fixed value of t, it has six eigenvalues, two of them are real with opposite sign (the ones which give the hyperbolic part to the Halo orbit) and the other 4 ones are pure imaginary numbers and conjugated in pairs (the ones which are related with the rotations about the orbit), as can be seen in [12]. In the case of other libration orbits, this is not exactly in this way, but the hyperbolic and rotation characteristics are maintained.

Let us denote by λ_1 the modulus of the real eigenvalues (they are λ_1 and $-\lambda_1$), and by λ_2 and λ_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 do not change significantly in the family of Halo orbits with practical applications and they are always different from zero as can be seen in figure 2.2.

Using this fact, we obtain the Jordan form of the $\mathbf{A}(t)$ matrix:

$$\mathbf{D}(t) = \begin{pmatrix} \lambda_1(t) & & & & & \\ & -\lambda_1(t) & & & & \\ & & 0 & \lambda_2(t) & & \\ & & -\lambda_2(t) & 0 & & \\ & & & & 0 & \lambda_3(t) \\ & & & & -\lambda_3(t) & 0 \end{pmatrix}. \quad (2.3)$$

2.1.3 The equations of motion

Let us consider the problem of the reconfiguration of a fixed number of spacecraft, N, in a fixed time T. We consider initial and final position of all the spacecraft also determined. The spacecraft of the formation are moving in the vicinity of a given Halo orbit, and since we work with formations of a diameter of few hundreds meters, the size of the formation with respect to the Halo orbit is very small so it is feasible to use the linearized equations about the nonlinear orbit to find a first approximation of the trajectories. In chapters 3 and 4, we study the trajectories using more precise modelling equations.

Let us fix a nominal Halo orbit for the trajectory of the formation. We denote by $\mathbf{X}_i$ the position and velocity of the i-th spacecraft of the formation with respect to the nominal point on the Halo orbit at time t. Considering the equations for relative motion, the governing equations for this satellite inside the formation can be written as:

$$\dot{\mathbf{X}}_i(t) = \mathbf{A}(t)\mathbf{X}_i(t), \quad (2.4)$$

where $\mathbf{A}(t)$ is the Jacobian matrix of the equations of motion about the Halo orbit.

Since the aim of the work is to perform reconfigurations of a set of spacecraft, the spacecraft must be subjected to a control. Let us consider a control applied to the spacecraft in a form that it modifies the acceleration. The equations will be of the form

$$\dot{\mathbf{X}}_i(t) = \mathbf{A}(t)\mathbf{X}_i(t) + \bar{\mathbf{U}}_i(t),$$

where the control $\bar{\mathbf{U}}_i(t)$ only affects to the acceleration, i. e. it is of the form $\bar{\mathbf{U}}_i(t) = (0, 0, 0, \bar{u}_i^x(t), \bar{u}_i^y(t), \bar{u}_i^z(t))^t$.

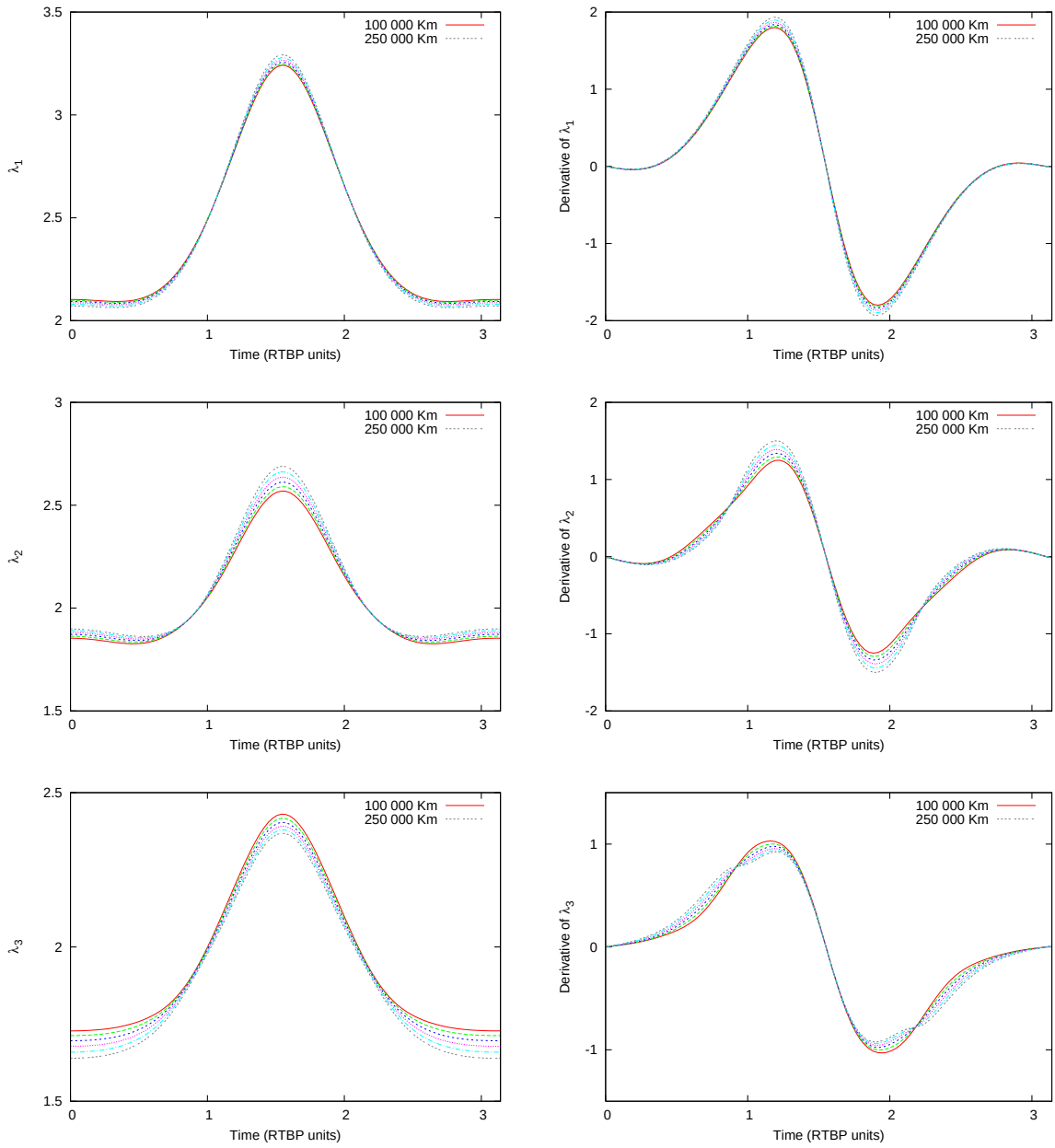

Figure 2.2: In the first column, from top to bottom, λ_1, λ_2 and λ_3, the modulus of the eigenvalues of $\mathbf{A}(t)$ with respect to the RTBP time for Halo orbits in the range from 100000 km to 250000 km of z-amplitude. Note that they are always away from zero. In the second column of the panel we plot the derivatives of these eigenvalues.

Adding the initial and final positions of the spacecraft in the orbit, we obtain the equations of the reconfiguration problem:

$$\begin{cases} \dot{\mathbf{X}}_i(t) = \mathbf{A}(t)\mathbf{X}_i(t) + \bar{\mathbf{U}}_i(t) \\ \mathbf{X}_i(0) = \mathbf{X}_i^0 \\ \mathbf{X}_i(T) = \mathbf{X}_i^T \end{cases} \tag{2.5}$$

where $\mathbf{X}_i^0$ is the initial state and $\mathbf{X}_i^T$ is the final state of the formation. The goal is to find optimal controls, $\bar{\mathbf{U}}_1, \ldots, \bar{\mathbf{U}}_N$, subjected to certain restrictions. The fundamental restriction is that the spacecraft cannot collide. We will discuss how to find this control and how to avoid collisions in section 2.4.

2.1.4 The equations for the uncoupled problem

The solution of the problem consists on finding suitable controls $\bar{\mathbf{U}}_i(t)$ satisfying the requirements (2.5). But the system of equations we obtain for each spacecraft is coupled with the ones corresponding to the other spacecraft. Our first objective is to separate this big problem in some simpler ones, which will be more efficient to solve.

In order to do this, we try to uncouple the system of equations using the properties of the Halo orbits. As we have seen in section 2.1.2, the matrix of the equation, $\mathbf{A}(t)$, has a specified Jordan form, $\mathbf{D}(t)$. Using this form, we introduce a change of coordinates, $\mathbf{X} = \mathbf{P}(t)\mathbf{Z}$. In this new set of coordinates, $\mathbf{Z}$, the equations of motion are

$$\dot{\mathbf{Z}} = \mathbf{D}(t)\mathbf{Z}. \tag{2.6}$$

The advantage of this new set of equations is that from them we can obtain a set of six equations which are uncoupled, and this fact simplifies the problem.

To obtain the uncoupled equations, we work with pairs of variables. Considering the first two variables, which are related to the real eigenvalues (λ_1 and $-\lambda_1$), we obtain the equations:

$$\dot{Z}_1 = \lambda_1(t)Z_1, \qquad \dot{Z}_2 = -\lambda_1(t)Z_2 \tag{2.7}$$

We can differentiate these equations in order to obtain a second order equation. After doing this, we have the uncoupled equations:

$$\ddot{Z}_1 = \dot{\lambda}_1(t)Z_1 + \lambda_1(t)\dot{Z}_1, \qquad \ddot{Z}_2 = -\dot{\lambda}_1(t)Z_2 - \lambda_1(t)\dot{Z}_2.$$

Note that now we can use the relations in equation (2.7) to uncouple the two equations and obtain:

$$\ddot{Z}_1 = (\dot{\lambda}_1(t) + \lambda_1^2(t))Z_1, \qquad \ddot{Z}_2 = (-\dot{\lambda}_1(t) + \lambda_1^2(t))Z_2. \tag{2.8}$$

We can see in figure 2.2 the values of the real eigenvalue. Its range is inside the interval $[2.0, 3.5]$, whereas its derivative moves in the range $[-2.5, 2.5]$. So, the equations obtained with this process can be always considered of the form

$$\ddot{\mathbf{z}} = \lambda(t)\mathbf{z},$$

with λ a positive parameter.

The other equations are similar in pairs, and the uncoupled equations are obtained in a similar way as the previous ones. Taking the variables corresponding to eigenvalue λ_2,

$$\dot{Z}_3 = \lambda_2(t)Z_4, \qquad \dot{Z}_4 = -\lambda_2(t)Z_3, \tag{2.9}$$

and differentiating them, as we have done with the previous ones,

$$\ddot{Z}_3 = \dot{\lambda}_2(t)Z_4 + \lambda_2(t)\dot{Z}_4, \qquad \ddot{Z}_4 = -\dot{\lambda}_2(t)Z_3 - \lambda_2(t)\dot{Z}_3,$$

and finally using the relations (2.9), we obtain the final uncoupled equations:

$$\ddot{Z}_3 = \frac{\dot{\lambda}_2}{\lambda_2}\dot{Z}_3 - \lambda_2^2 Z_3, \quad \ddot{Z}_4 = \frac{\dot{\lambda}_2}{\lambda_2}\dot{Z}_4 - \lambda_2^2 Z_4. \tag{2.10}$$

Note that, for a Halo orbit with an adequate size for astronomical purposes (see [12]), the eigenvalues given by the $\lambda_i(t)$ functions are away from zero and there is no numerical problem with the equations we have obtained.

Summarizing, we have seen that, if we use suitable variables Z which transform equation (2.5) into equation (2.6), we can reduce the problem to a set of six equations, all of them of the form:

$$\ddot{z} = \mu(t)\dot{z} + \lambda(t)z.$$

Note also that if we are working on free space, we also have these equations, but now with $\lambda \equiv \mu \equiv 0$.

A note for the controls after uncoupling coordinates

The change of coordinates that uncouple the equations of motion applies also to the controls and make them to be expressed in a more complex form.

For instance, considering the first equation in (2.7), we have,

$$\dot{Z}_1 = \lambda_1(t)Z_1 + U_1, \tag{2.11}$$

were U_1 is the first component of the control $\mathbf{U}$ which results after the application of the change of coordinates to the controls $\bar{\mathbf{U}}$ in (2.5).

Now if we differentiate it to obtain a second order equation, the equation with controls is transformed into

$$\ddot{Z}_1 = (\dot{\lambda}_1(t) + \lambda_1^2(t))Z_1 + \lambda_1(t)U_1 + \dot{U}_1.$$

Since we will deal with this type of second order equations, the control that we will compute will be in fact $u_1 = \lambda_1 U_1 + \dot{U}_1$, while U_1 needs to be obtained from this one. This is conveniently achieved considering $U_1(u_1)$ and its Taylor expansion up to first order:

$$U_1(u_1) = U_0 + (-\frac{\lambda_1}{\dot{u_1}}U_0 + \frac{1}{\dot{u_1}}u_0)(u_1 - u_0), + O((u_1 - u_0)^2),$$

where U_0 is the change of velocity on the node, that will be obtained via the piecewise linear function of the finite element method and u_0 is the control which $U_1(u_0) = U_0$.

Of course when we consider the equation for Z_2, the procedure is similar, and in the case of Z_3, Z_4 in (2.10) we have attached respectively controls u_3 and u_4 since the equations are already second order which are related to U_3 and U_4 of (2.10),

$$\dot{Z}_3 = \lambda_2(t)Z_4 + U_3, \qquad \dot{Z}_4 = -\lambda_2(t)Z_3 + U_4,$$

in the form $U_3(u_3, u_4)$, $U_4(u_3, u_4)$ that can be also expanded in Taylor series.

We remark that these approximations will be inside an iterative process where the base values u_0 and U_0 will be updated at each iteration in such a way that U_0 tends to be U_1.

Obtainment of the change of coordinates

We know that looking for a set of coordinates where the equations of the reconfiguration problem are (2.6), our initial coupled equations can be reduced to six uncoupled equations. The objective is to find this change of coordinates, $\mathbf{X} = \mathbf{P}(t)\mathbf{Z}$.

We note that the matrix $\mathbf{P}(t)$ must satisfy the differential equation

$$\dot{\mathbf{P}}(t) = \mathbf{A}(t)\mathbf{P}(t) - \mathbf{P}(t)\mathbf{D}(t), \tag{2.12}$$

since we can differentiate the change of coordinates, $\mathbf{X} = \mathbf{P}(t)\mathbf{Z}$, obtaining

$$\dot{\mathbf{X}} = \dot{\mathbf{P}(t)}\mathbf{Z} + \mathbf{P}(t)\dot{\mathbf{Z}},$$

and using equation (2.12) we obtain,

$$\dot{\mathbf{X}} = (\mathbf{A}(t)\mathbf{P}(t) - \mathbf{P}(t)\mathbf{D}(t))\,\mathbf{Z} + \mathbf{P}(t)\mathbf{D}(t)\mathbf{Z} = \mathbf{A}(t)\mathbf{X}.$$

In section 2.2.2 we will see that we use the finite element method to find the control. We need to know this change of coordinates in the nodes of the mesh. For this reason, the matrix $\mathbf{P}(t)$ is obtained on each of the nodes of the mesh numerically integrating the differential equation $\dot{\mathbf{P}} = \mathbf{AP} - \mathbf{PD}$, with a Runge-Kutta-Fehlberg method. The matrices which appear in the equation, $\mathbf{A}(t)$ and $\mathbf{D}(t)$ can be computed on each node: $\mathbf{A}(t)$ is the matrix (2.2) of the linearized equations of motion, and $\mathbf{D}(t)$ is the matrix (2.3) which contains the eigenvalues of $\mathbf{A}(t)$.

We find the initial condition of the differential equation by imposing that $\mathbf{P}(t)$ in the first node ($\mathbf{P}(0)$) satisfies

$$\mathbf{P}^{-1}(0)\mathbf{A}(0)\mathbf{P}(0) = \mathbf{D}(0).$$

This $\mathbf{P}(0)$ is constructed using the eigenvectors of the $\mathbf{A}(0)$ matrix: the first and second column are the eigenvectors of eigenvalues λ_1 and $-\lambda_1$. The third and fourth columns are the real and complex part of the eigenvectors of eigenvalues λ_2 and $-\lambda_2$, which are conjugated. The fifth and sixth columns are obtained in the same way as the third and fourth. This matrix $\mathbf{P}(0)$ has the property that $\dot{\mathbf{P}}(0) = \mathbf{A}(0)\mathbf{P}(0) - \mathbf{P}(0)\mathbf{D}(0) = 0$, which gives the initial condition to solve the differential equation (2.12).

Once the matrix $\mathbf{P}(t)$ is determined, we can use it to change the position coordinates:

$$\mathbf{X} = \mathbf{PZ}, \qquad \mathbf{Z} = \mathbf{P}^{-1}\mathbf{X}.$$

We obtain the change of coordinates for the velocities and delta-v differentiating the equations of the change of position:

$$\dot{\mathbf{X}} = \dot{\mathbf{P}}\mathbf{Z} + \mathbf{P}\dot{\mathbf{Z}}, \qquad \dot{\mathbf{Z}} = \dot{\mathbf{P}}^{-1}\mathbf{X} + \mathbf{P}^{-1}\dot{\mathbf{X}}. \tag{2.13}$$

2.2 The finite element method

As is well known, the finite element method is a method for obtaining approximate solutions of differential equations subject to boundary conditions. The resolution of differential equations is not always possible analytically. In order to have numerical results, there are different discretization methods, which transform these problems into discrete problems that approximate the real solution. These methods find an approximation of the solution, which can be, up to the numerical limits, as close as we want to the continuous solution, by increasing the number of nodes of the mesh.

The discretization of the finite element method consists on dividing the domain of the problem in a finite number of subdomains, which are called

elements. The set of all elements is called the *mesh*. These elements are connected to each other by a finite number of points called *nodes*. We note that this discretization is inherent to the problem, and can be a difficult problem to be obtained when the domain is a two-dimension or a three-dimension domain. The equations of each element are expressed with a finite number of parameters, which correspond to the value of some physical variables at the nodes and the forces or other variables applied to each of the nodes of the element. The finite element method consists on finding a solution which satisfies some form of the equation in the elements. This solution is a discrete approximation of the solution of the initial differential equation.

The solution of the problem is approximated by looking for a function in a set of functions which best approximates the real solution. The solution on each element is found by finding a combination of these base functions which best approximates the real solution of the problem. In order to do this, the set of functions must be chosen. It is possible to choose different kinds of form functions, giving different approximations of the finite element method. These functions must be independent, but they also have to represent simple functions. A good choice for these functions are the polynomials.

The solutions obtained using this method, converge to the real solution of the differential equation, when adding more elements (increasing the density of the mesh) or increasing the order of the elements.

Once one has a system of equations for each element, all the equations are merged together, obtaining a system of assembled equations, which only depends on the nodal values. Solving the assembled system of equations with suitable boundary conditions we obtain an approximation of the nodal values, which are used to approximate the real solution of the differential equation.

In the assembling process, the variables of the problem must be numbered inside the whole mesh. In two and three dimensional problems this is not an easy problem: the variables of the system correspond to values of a determinate magnitude in the elements. As the number of elements increase, the assembled matrix increases in size but is more sparse. The objective for a good numbering is to obtain a matrix with a small band, and this implies that the nodes which belong to neighbor elements be near in the numeration.

2.2.1 A summary of finite element method in dimension one

Let us recall that each equation of our reconfiguration problem is of the form:

$$\begin{cases} \ddot{x}(t) + \lambda(t)\,\dot{x}(t) + \tau(t)\,x(t) = u(t), \\ x(0) = x_0, \qquad x(T) = x_T, \\ \dot{x}(0) = v_0, \qquad \dot{x}(T) = v_T. \end{cases}$$

In order to solve a problem like this, we consider a variational method that will be applied to a general second order differential equation of the form,

$$a(t)\frac{d^2u}{dt^2} + b(t)\frac{du}{dt} + c(t)u(t) = g(t), \tag{2.14}$$

where $t \in [0, T]$.

It is convenient to consider this equation written in the self-adjoint form,

$$-\frac{d}{dt}(a_1(t)\frac{du}{dt}) + a_0(t)u = f(t).$$

that it is accomplished introducing some auxiliary functions [21]:

$$a_1(t) = e^{\int \frac{b(s)}{a(s)}ds}, \quad a_0(t) = \frac{-c(t)}{a(t)}a_1(t), \quad f(t) = \frac{-g(t)}{a(t)}a_1(t).$$

In general words, a variational method considers the *residual*,

$$R(t) = R(u(t), t) = -\frac{d}{dt}(a_1(t)\frac{du}{dt}) + a_0(t)u - f(t),$$

and tries to find the solution $u(t)$ in such a way that $R(t) \equiv 0$.

Since one considers the solution $u(t)$ approximated in a given finite dimensional space of functions (i.e. $u(t)$ will be of the form $\bar{u}(t) = \sum_{i=1}^{N} u_i\psi_i(t)$ where $\psi_i(t)$ are given and u_i to be determined), it is not possible to accomplish $R(\bar{u}(t), t) \equiv 0$ inside this finite-dimensional space and one works with the concept of weighted residuals. This is to consider,

$$\int_0^T \omega(t)R(u, t)dt = 0,$$

for a set of functions $\omega(t)$. (Note that this condition for all $\omega(t)$, for instance continuous functions, is equivalent to $R(u, t) \equiv 0$).

The finite element method is a variational method of approximation that facilitates in a systematic way the functions $\psi_i(t)$ that traditionally were very

difficult to obtain in the classical variational methods, due to the properties they are required to have [25]. The main idea relies in the fact that a complex domain can be decomposed or approximated by the union of simpler subdomains called elements (for instance triangles for a 2D domain) where one can find suitable $\psi_i(t)$ easily. Then essentially, the weighted residuals concept can be translated to each element, Ω^k making

$$\int_{\Omega^k} \omega_i(t) R(u^k, t) dt = 0, \text{ for a set of } \omega_i(t). \tag{2.15}$$

where u^k is the expression for $\bar{u}$ inside Ω^k.

Also, the choice of the set $\{\omega_i\}$ gives rise to different methods of the finite element methodology. A very common one is the Garlekin method, which takes as $\{\omega_i\}$ the set of base functions $\{\phi_i\}$ used for the approximation of u^k, and is the one selected for our work.

The number of base functions ϕ_i considered in each element is related to the degree of the approximation inside the element. This lets us to talk about the *order of the element* which is basically given by the number of selected points (*the nodes*) inside the element, and as we will see, they essentially play the role of interpolating points. Typically, in the finite element methodology, one can improve the approximate solution either considering elements of smaller size or considering higher order elements (the so called h and p-convergences).

As a main difference with respect to most finite element applications, our variable is time instead of spatial coordinates. This is, we are dealing with one-dimensional problems in the domain $[0, T]$ and the elements are sub-intervals of this time span. If we consider linear elements (order 1 approximation) each element has two nodes located at the ends of the interval. When considering bigger orders more nodes have to added inside the element. In all this dissertation we consider linear elements which have given very satisfactory results, although some aside computations have also be done using second order elements. Since the order of the elements is kept fixed, the convergence towards the solution will be controlled by means of the size of the elements. This is specially studied in chapter 3.

The weak form

Let us consider an element of the mesh, $\Omega^k = [t_A, t_B]$ with n nodes. The finite element theory provides a way to determine the set of base functions $\{\psi_i^k(t)\}$ to be considered in the definition of $u^k(t)$. These are selected in order $u^k(t)$ be an interpolant for the nodal values in terms of a polynomial of degree $n-1$ and it is convenient to choose $\psi_i^k(t)$ as a polynomial of degree $n-1$

such that $\psi_i^k(t_j) = \delta_{ij}$ (δ_{ij} is Kronecker's delta), for all $\{t_j\}$, $j = 1, \ldots, n$, the nodes inside Ω^k, this is, we consider Lagrange interpolation and for the case of linear elements we have,

$$\psi_1^k(t) = \frac{t_B - t}{t_B - t_A}, \quad \psi_2^k(t) = \frac{t - t_A}{t_B - t_A}.$$

Then, the approximation of the solution inside Ω^k is given by,

$$u^k(t) = \sum_{j=0}^{n} u_j^k \psi_j^k(t),$$

where u_j^k are constants to be computed and are known as *nodal values*.

As stated previously, the application of the weighted residuals method makes us to consider,

$$\int_{t_A}^{t_B} \omega[-\frac{d}{dt}(a_1 \frac{du}{dt}) + a_0 u - f]dt = 0, \tag{2.16}$$

that it is usually formulated in the *weak form* (see [36]) obtained by means of the formula of parts:

$$\int_{t_A}^{t_B} (a_1 \frac{d\omega}{dt}\frac{du}{dt} + a_0 \omega u - \omega f)dt - \left[\omega a_1 \frac{du}{dt}\right]_{t_A}^{t_B}$$

Taking into account that the desired condition $R(u,t) = 0$ has been substituted by $R(u^k,t) = 0$ in the discretization process, and the set of $\omega(t)$ functions are taken as the set $\{\psi_i^k(t)\}$ in the Galerkin approximation, the system of equations (2.15) turns to be,

$$\begin{pmatrix} K_{11}^k & K_{12}^k & \ldots & K_{1n}^k \\ K_{21}^k & K_{22}^k & \ldots & K_{2n}^k \\ \vdots & & \vdots & \vdots \\ K_{n1}^k & K_{n2}^k & \ldots & K_{nn}^k \end{pmatrix} \begin{pmatrix} u_1^k \\ u_2^k \\ \vdots \\ u_{k+1}^k \end{pmatrix} = \begin{pmatrix} F_1^k + Q_1^k \\ F_2^k + Q_2^k \\ \vdots \\ F_n^k + Q_n^k \end{pmatrix}, \tag{2.17}$$

with

$$K_{ij}^k = K_{ij}^{k,1} + K_{ij}^{k,0},$$

$$K_{ij}^{k,1} = \int_{t_A}^{t_B} a_1 \frac{d\psi_i^k}{dt}\frac{d\psi_j^k}{dt}dt, \quad K_{ij}^{k,0} = \int_{t_A}^{t_B} a_0 \psi_i^k \psi_j^k dt,$$

$$F_i^k = \int_{t_A}^{t_B} f\psi_i^k dt,$$

$$Q_1^k = -\omega(t_A)a_1(t_A)\frac{du}{dt}(t_A), \quad Q_n^k = \omega(t_B)a_1(t_B)\frac{du}{dt}(t_B),$$

and $Q_j^k = 0$ for $j = 2, \ldots, n$, but for convenience when working with higher order approximations are interesting to be written in this way. Equation (2.17) can be written in a compact form as

$$[K^k]u^k = F^k + Q^k,$$

where $[K^k]$ is the *elemental matrix*.

2.2.2 Applying the finite element method to reconfigurations

We have seen in section 2.1.4 that the problem we want to solve reduces to find suitable controls for some uncoupled equations, all of them of the form

$$\begin{cases} \ddot{x}(t) + \lambda(t)\,\dot{x}(t) + \tau(t)\,x(t) = u(t), \\ x(0) = x_0, \qquad x(T) = x_T, \\ \dot{x}(0) = v_0, \qquad \dot{x}(T) = v_T. \end{cases} \tag{2.18}$$

where $\lambda(t)$ and $\tau(t)$ are computed from the corresponding $\lambda_i(t)$ set.

At this point it is also worth to mention that if one wants to consider the motion of the formation in free space, which is common in many studies of formation flight, we need only to take $\lambda(t) = \tau(t) \equiv 0$.

As stated previously, our problem is one-dimensional in time and our domain is the time interval considered for the reconfiguration ($[0, T]$). We consider this time span split in M sub-interval elements. The elements can be of different length and can be different for each spacecraft. This use of different length elements is interesting also to put more elements where the delta-v needed is bigger, and elements of a bigger length (i. e. less elements) where the delta-v needed is smaller and in this one can control the h-convergence someway.

Let us consider linear elements. So taking a set of time values $\{t_k\}$ inside $[0, T]$ with $t_0 = 0$ and $t_M = 0$, we have $\Omega^k = [t_k, t_{k+1}]$ and a phase-space coordinate, $x(t)$, of the reconfiguration trajectory is approximated inside Ω^k by,

$$x^k(t) = x_k\psi_1^k(t) + x_{k+1}\psi_2^k(t),$$

where x_k, x_{k+1} are the nodal values corresponding respectively to times t_k, t_{k+1} and,

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

We note that in fact the trajectory $x(t)$ is approximated by a piecewise linear function forming a polygonal curve with vertices the set of nodes $\{t_k\}$. These nodes are also the places where it will be allowed to apply (instantaneous) maneuvers.

Let us consider the weighted residual approximation briefly explained in section 2.2 for the equations (2.18) inside Ω^k:

$$\int_{t_k}^{t_{k+1}} w(t)\,(\ddot{x}(t)+\lambda(t)\,\dot{x}(t)+\tau(t)\,x(t))dt = \int_{t_k}^{t_{k+1}} w(t)\,u(t)dt, \quad k=0,\ldots,M-1. \tag{2.19}$$

Note that we can compute the values of the functions $\lambda(t)$ and $\tau(t)$ at the nodes: they are 0 in free space and in a Halo orbit we can compute them using the eigenvalues of $A(t)$. For this purpose we can consider again the base functions and their nodal values to obtain their approximate expression inside Ω^k with the same truncation accuracy as the one used for the solution $x(t)$:

$$\lambda(t) \cong \lambda^k(t) = \lambda(t_k)\psi_1^k(t) + \lambda(t_{k+1})\psi_2^k(t),$$
$$\tau(t) \cong \tau^k(t) = \tau(t_k)\psi_1^k(t) + \tau(t_{k+1})\psi_2^k(t).$$

Let us proceed with the weak form of (2.19). In order to incorporate the impulsive controls in the nodes of the mesh, we assume that in each element Ω^k the control $u^k(t)$ is of the form,

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

where $\delta_{t_j}(t)$ stands for the Dirac delta at node t_j. For clarity we consider this weak form split in two parts:

The associated system considering $\lambda(t) \equiv 0$

In section 2.2 we have obtained an expression of the system for this case. The elemental matrix associated with this part is symmetrical. If we denote $h_k = t_{k+1} - t_k$, and $\lambda_k = \lambda(t_k)$, the corresponding system is:

$$\begin{pmatrix} K_{1,1}^k & K_{1,2}^k \\ K_{2,1}^k & K_{2,2}^k \end{pmatrix} \begin{pmatrix} x_k \\ x_{k+1} \end{pmatrix} = \begin{pmatrix} \Delta u_k^k + Q_1^k \\ \Delta u_{k+1}^k + Q_2^k \end{pmatrix},$$

where

$$K_{1,1}^k = \frac{1}{h_k} - \frac{3\lambda_k + \lambda_{k+1}}{12}h_k, \qquad K_{1,2}^k = \frac{-1}{h_k} - \frac{\lambda_k + \lambda_{k+1}}{12}h_k,$$

$$K_{2,1}^k = \frac{-1}{h_k} - \frac{\lambda_k + \lambda_{k+1}}{12}h_k, \qquad K_{2,2}^k = \frac{1}{h_k} - \frac{\lambda_k + 3\lambda_{k+1}}{12}h_k,$$

and we note also that

$$\Delta u_k^k = \int_{t_k}^{t_{k+1}} u^k(t)\psi_1^k(t)dt, \qquad \Delta u_{k+1}^k = \int_{t_k}^{t_{k+1}} u^k(t)\psi_2^k(t)dt.$$

The part of the system related with the term $\lambda(t)\dot{x}(t)$

A space-phase coordinate $x(t)$ inside Ω^k is approximated by means of $x^k(t) = x_k\psi_1^k(t) + x_{k+1}\psi_2^k(t)$. The derivative $\dot{x}(t)$ can be approximated inside Ω^k as

$$\dot{x}(t) \simeq \dot{x}^k(t) = \frac{x_{k+1} - x_k}{h}.$$

We note that since the elements are linear, $\dot{x}^k(t)$ is constant.

Thus, the contribution of this term to the weak form of the equations is obtained integrating the part of the residual corresponding to,

$$\int_{t_k}^{t_{k+1}} \psi_i^k(t)(-\lambda^k(t)\dot{x}^k(t))dt, \quad i = 1, 2.$$

This is,

$$\int_{t_k}^{t_{k+1}} \psi_1^k(t)(-\lambda^k(t)\dot{x}^k(t))dt \simeq \frac{-2\lambda_k - \lambda_{k+1}}{6}(x_{k+1} - x_k).$$

$$\int_{t_k}^{t_{k+1}} \psi_2^k(t)(-\lambda^k(t)\dot{x}^k(t))dt \simeq \frac{-\lambda_k - 2\lambda_{k+1}}{6}(x_{k+1} - x_k),$$

where $\lambda_k = \lambda(t_k)$. Observe that when $\lambda(t)$ is constant, these terms are both $-\lambda(x_{k+1} - x_k)/2$.

As a consequence, the part $\lambda(t)\dot{x}(t)$ adds to the matrix of the system of elemental equations $[K^k]u^k = F^k + Q^k$ the contribution,

$$\begin{pmatrix} \frac{2\lambda_k+\lambda_{k+1}}{6} & \frac{-2\lambda_k-\lambda_{k+1}}{6} \\ \frac{\lambda_k+2\lambda_{k+1}}{6} & \frac{-\lambda_k-2\lambda_{k+1}}{6} \end{pmatrix}$$

and the elemental matrix is no longer symmetric.

The assembled system

As we have just seen, for each element Ω^k of the mesh, we have associated a linear system of the form,

$$\begin{pmatrix} K_{1,1}^k & K_{1,2}^k \\ K_{2,1}^k & K_{2,2}^k \end{pmatrix} \begin{pmatrix} x_k \\ x_{k+1} \end{pmatrix} = \begin{pmatrix} \Delta u_k^k + Q_1^k \\ \Delta u_{k+1}^k + Q_2^k \end{pmatrix},$$

where,

$$K^k_{1,1} = \frac{1}{h_k} - \frac{3\lambda_k + \lambda_{k+1}}{12} h_k + \frac{2\tau_k + \tau_{k+1}}{6}, \quad K^k_{1,2} = \frac{-1}{h_k} - \frac{\lambda_k + \lambda_{k+1}}{12} h_k - \frac{2\tau_k + \tau_{k+1}}{6},$$

$$K^k_{2,1} = \frac{-1}{h_k} - \frac{\lambda_k + \lambda_{k+1}}{12} h_k + \frac{\tau_k + 2\tau_{k+1}}{6}, \quad K^k_{2,2} = \frac{1}{h_k} - \frac{\lambda_k + 3\lambda_{k+1}}{12} h_k - \frac{\tau_k + 2\tau_{k+1}}{6},$$

Next step in the finite element methodology is to assemble all the elemental equations in order to obtain a system of equations $[K]x = \Delta u$ involving all the nodal values in the mesh and their corresponding maneuvers.

Note that this mesh is in dimension one (the time) and the global numbering for the nodes is the natural numeration: the nodes can be ordered in time.

Thus, node k only belongs to elements Ω^{k-1} and Ω^k. We note that the equations on Ω^{k-1} are:

$$K^{k-1}_{1,1} x_{k-1} + K^{k-1}_{1,2} x_k = \Delta u^{k-1}_1 + Q^{k-1}_1,$$

$$K^{k-1}_{2,1} x_{(k-1)} + K^{k-1}_{2,2} x_k = \Delta u^{k-1}_2 + Q^{k-1}_2,$$

and on Ω^k,

$$K^k_{1,1} x_k + K^k_{1,2} x_{k+1} = \Delta u^k_1 + Q^k_1,$$
$$K^k_{2,1} x_k + K^k_{2,2} x_{k+1} = \Delta u^k_2 + Q^k_2.$$

Adding the second equation corresponding to element Ω^{k-1} and the first one of element Ω^k, and taking into account the flux matching conditions in secondary variables of the finite element method that impose $Q^{k-1}_2 + Q^k_1 = 0$ see [27, 21], we obtain:

$$K^{k-1}_{2,1} x_{k-1} + (K^{k-1}_{2,2} + K^k_{1,1}) x_k + K^k_{1,2} x_{k+1} = \Delta u^{k-1}_2 + \Delta u^k_1.$$

Note that, for each node, we have two parts of the control, one for each of the elements where the node belongs to. The total control on node k can be obtained by

$$\Delta u_k = \Delta u^{k-1}_2 + \Delta u^k_1$$

The matrix obtained in this way is a tridiagonal matrix, since each node belongs to a maximum of two elements.

This expression is valid for all the interior elements, but for the first and last element, we must add to the equations the contribution of the initial velocity in the first one and the desired final velocity in the last one. In the first element, the delta v that must be applied in the first node is $\Delta u_0 - \bar{v}_0$, and

in the last element, the delta-v of the last node is supplied by $\Delta u_M + \bar{v}_T$, where $\bar{v}_0$ and $\bar{v}_T$ are the initial and final velocity after the change of coordinates which uncouples the system.

Using the fact that the position at epochs 0 and T are fixed in our problem, the assembled system is:

$$\begin{pmatrix} K^0_{2,2} + K^1_{1,1} & K^1_{1,2} & 0 & & \\ K^1_{2,1} & K^1_{2,2} + K^2_{1,1} & K^2_{1,2} & & \\ \ddots & \ddots & \ddots & & \\ & K^{M-3}_{2,1} & K^{M-3}_{2,2} + K^{M-2}_{1,1} & K^{M-2}_{1,2} & \\ & 0 & K^{M-2}_{2,1} & K^{M-2}_{2,2} + K^{M-1}_{1,1} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_{M-2} \\ x_{M-1} \end{pmatrix} +$$

$$+ \begin{pmatrix} K^0_{2,1} x_0 \\ 0 \\ \vdots \\ 0 \\ K^{M-2}_{1,2} x_T \end{pmatrix} = \begin{pmatrix} \Delta u_1 \\ \Delta u_2 \\ \vdots \\ \Delta u_{M-2} \\ \Delta u_{M-1} \end{pmatrix}. \tag{2.20}$$

We obtain the Δu for the first and last node with the use of the first and the last equation:

$$\Delta u_0 = K^0_{1,1} x_0 + v_0 + K^0_{1,2} x_1,$$

$$\Delta u_M = K^{M-1}_{2,1} x_{M-1} + K^{M-1}_{2,2} x_M - v_T.$$

Other parameters of the equation

In order to apply the finite element method, we need the values of $\lambda_i(t)$ and their derivatives at the nodes. As we have discussed, we find the values of $\lambda_i(t)$ at node k computing the matrix $A(t_k)$ and its eigenvalues.

The values of the derivative of $\lambda_i(t)$, are obtained using an approximation of the same order of the approximated solution using the base functions of the finite element method.

If we use linear elements, we write $\lambda_i(t)$ inside Ω^k by means of

$$\lambda_i(t) \simeq \lambda^k(t) = \lambda_i(t_k)\psi^k_1(t) + \lambda_i(t_{k+1})\psi^k_2(t).$$

Then, the derivative of λ_i in this element is:

$$\dot{\lambda}_i(t) \simeq \dot{\lambda}^k_i(t) = \frac{\lambda_i(t_{k+1}) - \lambda_i(t_k)}{h_k}.$$

We note that the derivative of $\lambda_i(t)$ is not continuous when we use finite elements. Except for the first and last nodes, each node belongs to two different elements and the derivative at the node is different depending on the element used to calculate it. We use a ponderate combination of the two expressions to obtain the approximation of the derivative:

$$\dot{\lambda}_i(t_k) \simeq \frac{h_{k-1}}{h_{k-1}+h_k}\dot{\lambda}_i^{k-1} + \frac{h_k}{h_{k-1}+h_k}\dot{\lambda}_i^{k},$$

which can also be written as:

$$\dot{\lambda}_i(t_k) \simeq \frac{\lambda_i(t_{k+1}) - \lambda_i(t_{k-1})}{h_{k-1}+h_k}. \tag{2.21}$$

2.3 Optimization problems

We reduce our reconfiguration problem to an optimization problem with constraints. Optimization problems can be formulated as

$$\begin{array}{ll} \text{minimize} & f(\mathbf{x}), \quad \mathbf{x} \in R^n \\ \text{subject to} & \mathbf{l} \leq \mathbf{c}(\mathbf{x}) \leq \mathbf{u}, \end{array} \tag{2.22}$$

where $\mathbf{x}$ is a set of variables (in the case of reconfigurations, $\mathbf{x}$ contains variables related to the position of the spacecraft in the nodes), f($\mathbf{x}$) is the function to minimize (which is related to fuel consumption) and $\mathbf{c}(\mathbf{x})$ are some constraints (we have two kinds of constraints: one of them has to be in all the reconfigurations with the objective of avoiding collisions and the other ones are geometrical constraints, which will be added depending on the reconfiguration problem). The constraints can be nonlinear, and $\mathbf{l}$ and $\mathbf{u}$ are sets of upper and lower bounds defining them.

Note that in this general formulation we include all the cases of constrained optimization. Constraints on the values of the optimization variables (such as imposing that the spacecraft has a given state on a node), enter easily in the problem with $\mathbf{c}_i(\mathbf{x}) = \mathbf{x}_j$. Equality constraints enter in the problem using the same upper and lower bound.

The constrained optimization problems are usually solved by transforming the initial problem into an optimization problem without constraints, which is easier to solve. The new problem has a penalty function for the constraints which do not satisfy restrictions. The optimal of the problem is found as the limit of a sequence of solutions which converge to a feasible solution of the problem.

The method we use is a sequential quadratic programming (SQP) method (see [9], [11]). These methods reduce the initial problem to problems which can be solved by Newton's method.

The SQP methods have a procedure that involves major and minor iterations. The major iterations are the sequence of solutions which converge to the solution. On each major iteration, the procedure must solve an optimization problem. This optimization sub-problem is solved iteratively, and the iterations to find this solution are the minor iterations.

The objective of the SQP method is to generate a sequence of iterates which converge to a point satisfying two conditions:

- There exists a vector of Lagrange multipliers μ such that the gradient of the Lagrangian $f(\mathbf{x}) - \mu^T \mathbf{c}(\mathbf{x})$ is zero.
- The Lagrange multiplier μ_j associated with the jth constraint satisfies $\mu_j = 0$ if $l_j < \mathbf{c}_j(\mathbf{x}) < u_j$; $\mu_j \geq 0$ if $l_j = c_j(x)$; $\mu_j \leq 0$ if $\mathbf{c}_j(\mathbf{x}) = u_j$; and μ_j can have any value if $l_j = u_j$.

For this purpose, every new major iteration is found by

$$\bar{\mathbf{x}} = \mathbf{x} + \alpha \mathbf{p},$$

where α is the step length, α is a non negative scalar, and $\mathbf{p}$ is the search direction.

The search direction, $\mathbf{p}$, is obtained by means of the minor iterations, is the solution of the minimization problem,

$$\begin{array}{rl} \text{minimize} & f(\mathbf{x}) + g(\mathbf{x})^T \mathbf{p} + \frac{1}{2} \mathbf{p}^T H \mathbf{p}, \quad \mathbf{p} \in R^n \\ \text{subject to} & \mathbf{l} \leq \mathbf{c}(\mathbf{x}) + J(\mathbf{x}) \mathbf{p} \leq \mathbf{u}, \end{array} \tag{2.23}$$

where H is an approximation to the Hessian of the Lagrangian, $g(x)$ denotes the gradient vector of first derivatives of the objective function and $J(x)$ is the Jacobian matrix of first derivatives of $r(x)$.

2.4 The general procedure FEFF

Our reconfiguration problem consists on finding a trajectory for a set of spacecraft which have a fixed initial state and must arrive to a known final goal in a fixed time T.

To obtain the trajectories, all the spacecraft must be subjected to a control law in the acceleration. In the initial problem (2.5),

$$\begin{cases} \dot{\mathbf{X}}_i(t) = \mathbf{A}(t) \mathbf{X}_i(t) + \bar{\mathbf{U}}_i(t), \\ \mathbf{X}_i(0) = \mathbf{X}_i^0, \quad \mathbf{X}_i(T) = \mathbf{X}_i^T, \end{cases} \tag{2.24}$$

are of the form $\bar{\mathbf{U}}_i(t) = (0, 0, 0, u_i^x(t), u_i^y(t), u_i^z(t))^t$ and essentially our objective is to find the optimal functions for $\bar{U}_i(t)$.

In section 2.1.4, we have studied how to split the general equation $\dot{\mathbf{X}}_i(t) = \mathbf{A}(t)\mathbf{X}_i(t)$ in six uncoupled equations. Using the same idea we uncoupled (2.24) in six equations ($j = 1 \dots 6$) for each satellite, namely:

$$\begin{cases} \ddot{x}_{ij}(t) + \lambda(t)\,\dot{x}_{ij}(t) + \tau(t)\,x_{ij}(t) = u_{ij}(t), \\ x_{ij}(0) = x_{i0}, \qquad x_{ij}(T) = x_{iT}, \\ \dot{x}_{ij}(0) = v_{i0}, \qquad \dot{x}_{ij}(T) = v_{iT}. \end{cases} \tag{2.25}$$

With the objective of finding the optimal controls $\bar{\mathbf{U}}_{ij}$ but in terms of u_{ij}, we must note that in 2.25 the variables of the problem are some functions of the states of the spacecraft, but not the states. For the controls of equation (2.24), we know that the first three components of the control must be 0 and the controls $u_{1j}, \dots, u_{Nj}$ do not have this property. However the computations will be carried out in such a way that only controls u_{ij} giving feasible $\bar{\mathbf{U}}_{ij}$ will be considered.

We also note that the controls are also uncoupled, and so they can be obtained independently for each spacecraft if we do not take into account constraints between them.

The optimal solution will be found via an optimal problem. The functional to be minimized is related to fuel consumption being the most important restriction the one which avoids collision between the spacecraft.

2.4.1 The objective function

Our objective when computing the reconfiguration of a spacecraft is the consumption of a minimum amount of fuel. The fuel expenditure of the spacecraft is directly related to the norm of the delta-v maneuvers. We note that the Δv used in the functional are the delta-v that must be applied to the spacecraft, and are obtained of the Δu via the change of coordinates of controls. This functional is given by:

$$J_1 = \sum_{i=1}^{N} \sum_{k=0}^{M_i} \rho_{i,k} ||\Delta \mathbf{v}_{i,k}||, \tag{2.26}$$

where $||*||$ denote 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 we can impose a weight on each component).

FEFF	Generic name for the reconfiguration (Finite Elements for Formation Flying)
FEFF-DV2	Optimization of the functional (2.27) (minimization of sum of Δv-square)
FEFF-DV	Optimization of the functional (2.26) (sum of Δv-modulus minimization)
FEFF-A	Obtain ad-hoc meshes using adaptive remeshing

Table 2.1: Names of the methodologies used to obtain the reconfiguration trajectories.

However, as it 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 intents to produce...). For this reason is usual to consider,

$$J_2 = \sum_{i=1}^{N} \sum_{k=0}^{M_i} \rho_{i,k} ||\Delta \mathbf{v}_{i,k}||^2, \tag{2.27}$$

This function is also related to fuel expenditure, because it is small when the delta-v are small, but its derivative do not have ill conditioning problems. Additionally, we have introduced parameters $\rho_{i,k}$. These are penalty parameters, which can be used to penalize the consumption of a spacecraft with critical fuel resources, for instance. It can also be variable on time, to penalize the fuel consumption in a determinate time span.

Using the two functionals and some techniques to obtain suitable and good meshes, we define different methodologies to obtain the trajectories. The names of the methodologies and a brief description of them are shown in table 2.1.

In this chapter, we study the methodology associated with FEFF-DV2. In the following chapter, we study some remeshing techniques to avoid the ill-conditioned problem using the functional of equation (2.26).

2.4.2 Collision avoidance

Like the minimization of fuel consumption, collision avoidance is a key problem to be considered in the reconfiguration of spacecraft. Collision avoidance enters in the optimal problem as a set of constraints.

One of the advantages of the methodology we use is that we can include easily constraints in the problem. Due to the nature of the optimization problems, these constraints can be equality constraints, this is $c(\mathbf{x}) = 0$, or

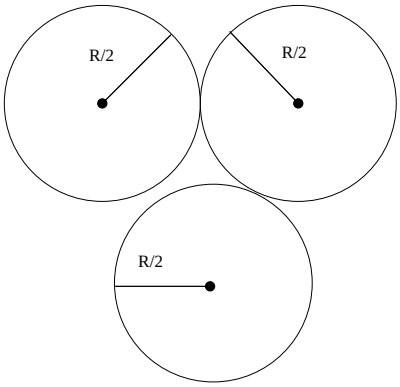

Figure 2.3: We surround each spacecraft by an imaginary sphere of a radius $R/2$, equal to half the security distance between spacecraft. In all the reconfiguration process, the spheres must not intersect, except maybe for a tangency point.

inequality constraints, $c(\mathbf{x}) > 0$. Our most important set of constraints is collision avoidance, but we can add some other constraints to the optimization problem, to obtain trajectories with geometrical confinements.

For simplicity, in this first approach of the computation of the collision avoidance constraints, we assume that all the spacecraft have the same mesh: i. e., a node of the mesh of a given spacecraft is also a node of the mesh of any other spacecraft.

The methodology to avoid collision between spacecraft consists on assuring a minimum distance between the spacecraft during the reconfiguration time. To accomplish this objective, we surround each spacecraft by an imaginary sphere with a radius half the security distance. The constraint we impose is that in all the reconfiguration process the spheres do not intersect. We only accept a tangency point between spheres, as can be seen in figure 2.3.

The collision avoidance constraint can be checked in this way: the time span is divided in linear elements, and on each element, the trajectory is a straight line. The idea is that the distance between the spacecraft, $d(t)$, must be greater than R in at all times. But, since all trajectories are straight lines inside an element, collision avoidance reduces to a check, for each pair of spacecraft, and each pair of elements, that

$$d_{ij}^k(t) \geq R_{ij}. \tag{2.28}$$

This is, for each pair (i, j) of spacecraft and for each element Ω^k, we have a constraint:

$$c_{ij}^k(\mathbf{x}) = d_{ij}^k(t) - R_{ij} \geq 0.$$

We compute the minimum distance between spacecraft on each element and then we check whether this distance is greater than R.

Computing the minimum distance between two spacecraft in an element

Let us focus on an element Ω^k. Let us note the initial nodal position of spacecraft by $\mathbf{p}_i$, and the final nodal position by $\mathbf{q}_i$. We consider a variable $\mathbf{v}_i = \mathbf{q}_i - \mathbf{p}_i$, which indicates the direction of motion of the spacecraft inside the element. Using this notation, the position of the spacecraft in the element is given by

$$\mathbf{x}_i = \mathbf{p}_i + \mathbf{v}_i s$$

where $s \in [0, 1]$ is a parameter related to time.

We look for the parameter s which makes the minimum distance between two spacecraft. Instead of minimizing the distance, we equivalently minimize the square of the distance.

$$\begin{aligned} f(s) = \quad & (\mathbf{p}_i + s\mathbf{v}_i - \mathbf{p}_j - s\mathbf{v}_j)^T \cdot (\mathbf{p}_i + s\mathbf{v}_i - \mathbf{p}_j - s\mathbf{v}_j) = \\ & ((\mathbf{p}_i - \mathbf{p}_j) + s(\mathbf{v}_i - \mathbf{v}_j)^T \cdot ((\mathbf{p}_i - \mathbf{p}_j) + s(\mathbf{v}_i - \mathbf{v}_j). \end{aligned}$$

Differentiating this equation, we obtain the minimum of the distance. The derivative is

$$f'(s) = 2((\mathbf{p}_i - \mathbf{p}_j) + s(\mathbf{v}_i - \mathbf{v}_j))^T \cdot (\mathbf{v}_i - \mathbf{v}_j).$$

We observe that the extreme, in case that the function has a minimum, is really a minimum, since

$$f''(s) = 2(\mathbf{v}_i - \mathbf{v}_j)^T \cdot (\mathbf{v}_i - \mathbf{v}_j),$$

is always positive, except for the case when $\mathbf{v}_i = \mathbf{v}_j$, which corresponds to the case where the two lines are parallel.

When the two lines are parallel, the minimum distance between the two straight lines is the distance taken for any value of the parameter s. When they are not parallel, the value of s where the distances are minimum is,

$$s = -\frac{(\mathbf{p}_i - \mathbf{p}_j)^T \cdot (\mathbf{v}_i - \mathbf{v}_j)}{(\mathbf{v}_i - \mathbf{v}_j)^T \cdot (\mathbf{v}_i - \mathbf{v}_j)}.$$

We are only interested in the minimum distance inside the elements, so this parameter only will be useful if it is inside $[0, 1]$. If not, the minimum distance between the straight lines will be outside of the element, but inside of the elements it is achieved in one of the nodes. When the parameter s is

less than 0, the minimum will be achieved in the first node of the element. When s is greater than 1, it will be achieved in the second one.

Effectively, if we write the value of the parameter s where the distance is minimum as $-A/v^2$, for a given A and v, the distances in the first and last node are related as:

$$f(1) = f(0) + 2A + v^2.$$

When $s < 0$, then $A > 0$ and $f(0) < f(1)$, so the minimum distance between the spacecraft is achieved in the first node of the element.

When $s > 1$, then $A < 0$ and $|A| > v^2$, so $f(0) > f(1)$ and the minimum is in the second node of the element.

In the case that $s \in [0, 1]$, let us write, in order to simplify the notation, $\mathbf{p} = \mathbf{p}_i - \mathbf{p}_j$ and $\mathbf{v} = \mathbf{v}_i - \mathbf{v}_j$. The minimum distance in this case is given by:

$$\frac{(p_1v_2 - p_2v_1)^2 + (p_1v_3 - p_3v_1)^2 + (p_2v_3 - p_3v_2)^2}{v_1^2 + v_2^2 + v_3^2}.$$

Collision avoidance when the security distance is variable

In some cases, like in the case of the deployment of the formation, we need a security distance which is not constant, but changing with time, because in the first steps of the deployment, the spacecraft are so close that would not satisfy the constraints.

In this case, the restriction is of the form $f(s) \leq R(s)$, and it is not enough to find the minimum distance between the spacecraft to know whether the spheres intersect.

Since we have the values of the distances in the nodes, R_1 and R_2, and since we are using linear elements, we also consider a linear function to express the security distance inside the elements:

$$R(s) = R_1 + s(R_2 - R_1).$$

Now, the point where the restriction is minimum is found by solving the equation,

$$\frac{\partial(d^2 - R^2)}{\partial s} = 0,$$

which gives us the parameter s corresponding to the minimum distance:

$$s = -\frac{(\mathbf{p}_i - \mathbf{p}_j)^T \cdot (\mathbf{v}_i - \mathbf{v}_j) - R_1(R_2 - R_1)}{(\mathbf{v}_i - \mathbf{v}_j)^T \cdot (\mathbf{v}_i - \mathbf{v}_j) - (R_2 - R_1)^2}. \tag{2.29}$$

When the security distance is constant, this parameter s can be computed in any case, except when the spacecraft have parallel trajectories, and it is

always a minimum of the distance function. But when the security distance is not constant, the parameter s gives us the minimum distance only when the denominator of (2.29) is positive. Effectively, the second derivative of the function is,

$$\frac{\partial^2(f(s) - R^2(s))}{\partial s^2} = 2(\mathbf{v}_i - \mathbf{v}_j)^T \cdot (\mathbf{v}_i - \mathbf{v}_j) - 2(R_2 - R_1)^2.$$

So, if the denominator of the equation (2.29) is not positive, the minimum distance must be achieved in one of the nodes of the element.

In case that the denominator is positive, we can compute s, and as in the case with constant distances, when $s \in [0, 1]$, this is the time where the restriction is minimum. If $s < 0$, the minimum is in the first node of the element, while when $s > 1$, the minimum is achieved in the second node.

2.4.3 Obtainment of the initial seed for FEFF-DV2

All the numerical methods which compute optimal values of a function subjected to constrains need an initial seed to start the computations. It is important that the initial seed of the reconfiguration process be near the optimum searched, in order to assure the convergence of the method, and also to converge faster.

In our case, for each spacecraft we must provide a trajectory near the optimum. In order to find this initial seed, we consider each spacecraft as if it were alone, without taking into account other spacecraft of the formation. For a given spacecraft, we can find easily the minimum fuel-consumption trajectory. This gives the optimum of the problem without taking into account possible collisions between spacecraft. This procedure has the advantage that in problems where the minimum fuel-consumption trajectories do not collide, this initial seed is already the optimum.

For a given spacecraft i, we are going to minimize the contribution of the delta-v in a coordinate l. Since the problem is uncoupled, minimizing the contribution for each coordinate, minimizes the delta-v of the spacecraft. The functional we consider to be optimized is

$$J_{i,l} = \sum_{k=0}^{M_i} (\Delta u_{i,l,k})^2,$$

where M_i is the number of elements of the mesh for spacecraft i. From now on, in order to simplify the notation, we do not write the indices for the satellite (i) and their components (l), in all the formulae of the procedure.

Let us write the delta-v values using the expression obtained applying the finite element methodology. For the interior nodes, the assembled matrix of the finite element methodology, relates the nodal values of the trajectory, u_k, with the maneuvers applied at the nodes:

$$\begin{pmatrix} \Delta u_1 \\ \\ \vdots \\ \\ \Delta u_{M-1} \end{pmatrix} = \begin{pmatrix} A_{1,0} \\ 0 \\ \vdots \\ 0 \\ A_{M-1,M} \end{pmatrix} + \\ + \begin{pmatrix} A_{1,1} & A_{1,2} & 0 & & \\ A_{2,1} & A_{2,2} & A_{2,3} & & \\ & \ddots & \ddots & \ddots & \\ & & A_{M-2,M-3} & A_{M-2,M-2} & A_{M-2,M-1} \\ & & & A_{M-1,M-2} & A_{M-1,M-1} \end{pmatrix} \begin{pmatrix} u_1 \\ \\ \vdots \\ \\ u_{M-1} \end{pmatrix}. \tag{2.30}$$

The Δu corresponding to the first and last node are obtained with the equations resulting from the boundary conditions:

$$\Delta u_0 = A_{0,0}u_0 + A_{0,1}u_1 + \bar{v}_0$$

$$\Delta u_M = A_{M,M-1}u_{M-1} + A_{M,M}u_M - \bar{v}_M,$$

where u_0, u_M, v_0 and v_M are constant, because they correspond to the states at the initial and final time.

The value of the objective function can be written in a compact form. Denoting by A the assembled matrix of equation (2.30) and by $\mathbf{b}$ the vector $(A_{1,0}, 0, \ldots, 0, A_{M-1,M})$, the function we want to minimize is,

$$J = (A\mathbf{u} + \mathbf{b})^T(A\mathbf{u} + \mathbf{b}) + (\Delta u_0)^2 + (\Delta u_M)^2,$$

where $\mathbf{u}$ is the vector of nodal positions.

The values of u_k which minimize J are computed taking into account that the partial derivatives of J respect all the interior u_k must be 0. This gives us a system of equations,

$$(A^TA + C)\mathbf{u} + (A^T\mathbf{b} + \mathbf{d}) = 0.$$

The matrix of the system $(A^TA + C)$ is a pentadiagonal symmetrical matrix, because A is a tridiagonal matrix and C is a matrix with zeros in all the components, except the one of the first row and first column and the one of the last row and last column.

Let us see that the matrix of the system, $A^T A + C$, is non singular, so the system has a unique solution.

C is a matrix which has only two elements different from 0 (the elements C_{11} and C_{MM}). These two elements are positive. We can calculate the determinant of the matrix $A^T A+C$, expanding it along the first row. Denoting by $M_{i,j}$ the minors of the matrix, the determinant is

$$det(A^T A+C) = ((A^T A)_{1,1}+C_{1,1})M_{1,1}+(A^T A)_{1,2}M_{1,2}+\cdots+(A^T A)_{1,M}M_{1,M},$$

which is the same as:

$$det(A^T A + C') + C_{1,1}M_{1,1},$$

where C' is a sparse matrix with all the elements zero, except for C_{MM}.

Then, the determinant of the matrix $A^T A + C'$ can be expanded along the last row, and we obtain:

$$det(A^T A + C) = det(A^T A) + C_{1,1}M_{1,1} + C_{M,M}M_{M,M}.$$

The matrix A is not singular, so $det(A^T A)$ is positive. The terms of the matrix C are also positive, and the minors are also positive, because they come from the assembled matrix of the finite element method. So, the determinant of the matrix is positive, and the matrix $A^T A+C$ is not singular.

2.5 Simulations using FEFF-DV2

When treating different reconfiguration problems, we can consider that there are four main kinds of reconfigurations, with similar behavior. This four groups are:

- Basic maneuvers without collision risk, such as translations and rotations.
- Switches or transfers with collision risk.
- Reconfigurations with additional restrictions on the trajectories.
- Deployments of formations.

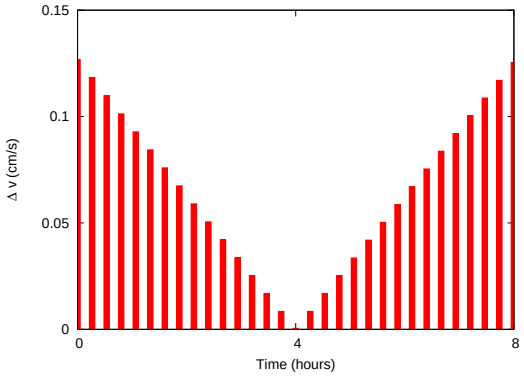

Figure 2.4: The delta-v profile that we obtain when we compute the optimal solution for a problem without collision risk using FEFF-DV2.

2.5.1 Basic maneuvers without collision risk

There are some reconfigurations where the spacecraft moving in a straight line have no collision risk at all. For example, a reconfiguration of a formation where the satellites keep the same shape, but the distances between spacecraft are longer (expansions) or shorter (contractions), or the case of parallel shifts. In all of these cases the optimal solution is the solution which minimizes the fuel consumption of each single spacecraft from its initial position to the final one.

The solution to this problem in free space is known: a bang-bang control. In our problem, since we work with formations near the Halo orbit in a short span of time, the optimal solution is similar to a bang-bang control. Nevertheless, in the first approach of the problem (FEFF-DV2), we do not consider the functional of equation (2.26), and the optimal solution we obtain is not the bang-bang solution.

Note that our minimization method always starts with the optimal trajectory for each spacecraft, without taking into account collision risks. Since in this case the optimal trajectory does not end up with collision, the initial seed of the minimization algorithm is the solution we are searching, and the procedure of minimization does not perform any more iterations.

When minimizing the functional of equation (2.27), the typical distribution of delta-v maneuvers that we obtain is the one of figure 2.4.

The case of rotations can be considered a limiting case between this first group of reconfigurations and the second one (the ones with collision risk). When the rotation is of a small angle, the straight-line trajectory gives us trajectories free of collision risk between the spacecraft being the solution

essentially the same one as in translations. When for instance the rotation is of an angle of 180 degrees, then the rotation is in fact a switch of spacecraft in pairs.

2.5.2 Switches or transfers avoiding collision risk

When we work with reconfigurations where the bang-bang solution is affected of collision risk, is when the optimization procedure must be specially robust. A typical case of this is the switch in position of two spacecraft: changing their position with a bang-bang maneuver, they will produce collision.

To exemplify this situation, we can consider the TPF formation. In the TPF formation, the spacecraft of the baseline usually will not have the same fuel consumption: the ones which are near the Halo orbit consume less fuel than the other ones. In order to correct this situation and equilibrate the fuel consumption at some point, we can switch the positions of the inner spacecraft with the ones of the outer positions. But this maneuver implies a collision risk.

In figure 2.5, we show the maneuver profile and in figure 2.6 the trajectories of the spacecraft performing the switch between spacecraft in the TPF formation using the FEFF-DV2 procedure. We can observe that the delta-v expenditure of each spacecraft are similar when doing the reconfiguration, and that the spacecraft chose a different plane to switch positions.

This kind of reconfigurations let us work with test bench examples. As an example, we consider reconfigurations of a big number of spacecraft, that with a bang-bang control all of them would collide in a point. A simple case of this considers five spacecraft: four of them located in the vertices of a square, and the fifth one in the center of the square. Let us assume that we want to change the position of the spacecraft in opposite vertices. Using a bang-bang control all of them would collide in the center of the square in half the transfer time. In this case, to avoid collision, procedure FEFF-DV2 switches the spacecraft using different planes. In figure 2.7, we display the positions of the spacecraft inside their trajectories for different times.

2.5.3 Reconfigurations with trajectory restrictions

Taking advantage of the properties of procedure FEFF-DV2, that allow us the possibility of introducing some restrictions in addition to collision avoidance, we could also consider adding some restrictions in the trajectories or in the fuel consumption.

The functional we are minimizing (2.27) has a parameter, ρ, that we use to penalize the fuel consumption of some spacecraft in a time span, or in

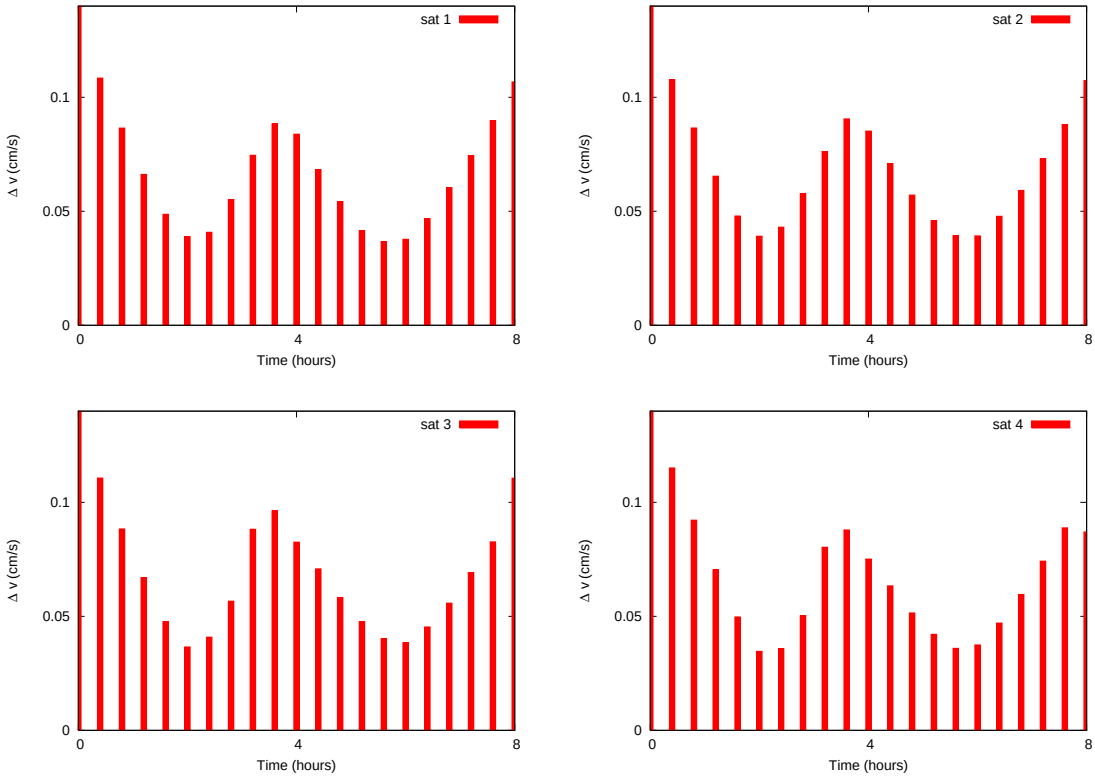

Figure 2.5: The maneuvers (in cm/s) with respect to time (in hours) we must apply to the spacecraft of the TPF formation to switch the two pairs of spacecraft with 20 elements.

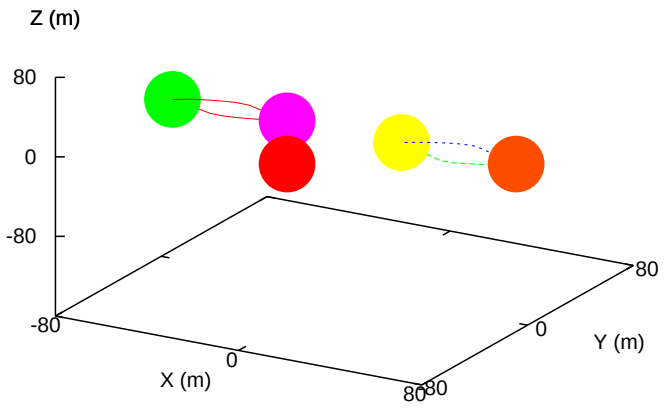

Figure 2.6: Trajectories of the spacecraft of the TPF formation which perform a reconfiguration of switching two pairs of spacecraft.

all the reconfiguration time. However, using the restrictions, we can also stop maneuvers during a certain time span. At this point, we note that this procedure is equivalent to use a new mesh, without certain nodes, and that this procedure can be implemented using the techniques that we develop in the following chapter.

We can consider the same example of the previous section: switching the inner and outer spacecraft of the TPF formation, but now adding the restriction that two nodes (at times between 0.6 and 1 hours) must have a zero delta-v (for instance, due to operational reasons). In this case, we observe (see figure 2.8) that the delta-v profile that we obtain is similar to the case without any restriction, only slightly bigger values near the nodes of zero delta-v, to compensate the lack of corrections.

We could also add other restrictions to the problem. For instance other natural restrictions are geometrical considerations, which would bind the spacecraft to do some prescribed kind of trajectories. Examples of them are:

- Maintaining three spacecraft in the vertices of an equilateral triangle or four spacecraft in the vertices of a tetrahedron. In this case, there is no collision problem, because the natural distances between the spacecraft are always the same. The spacecraft move as a solid.

- Maintaining the spacecraft inside an sphere. The spacecraft are con-

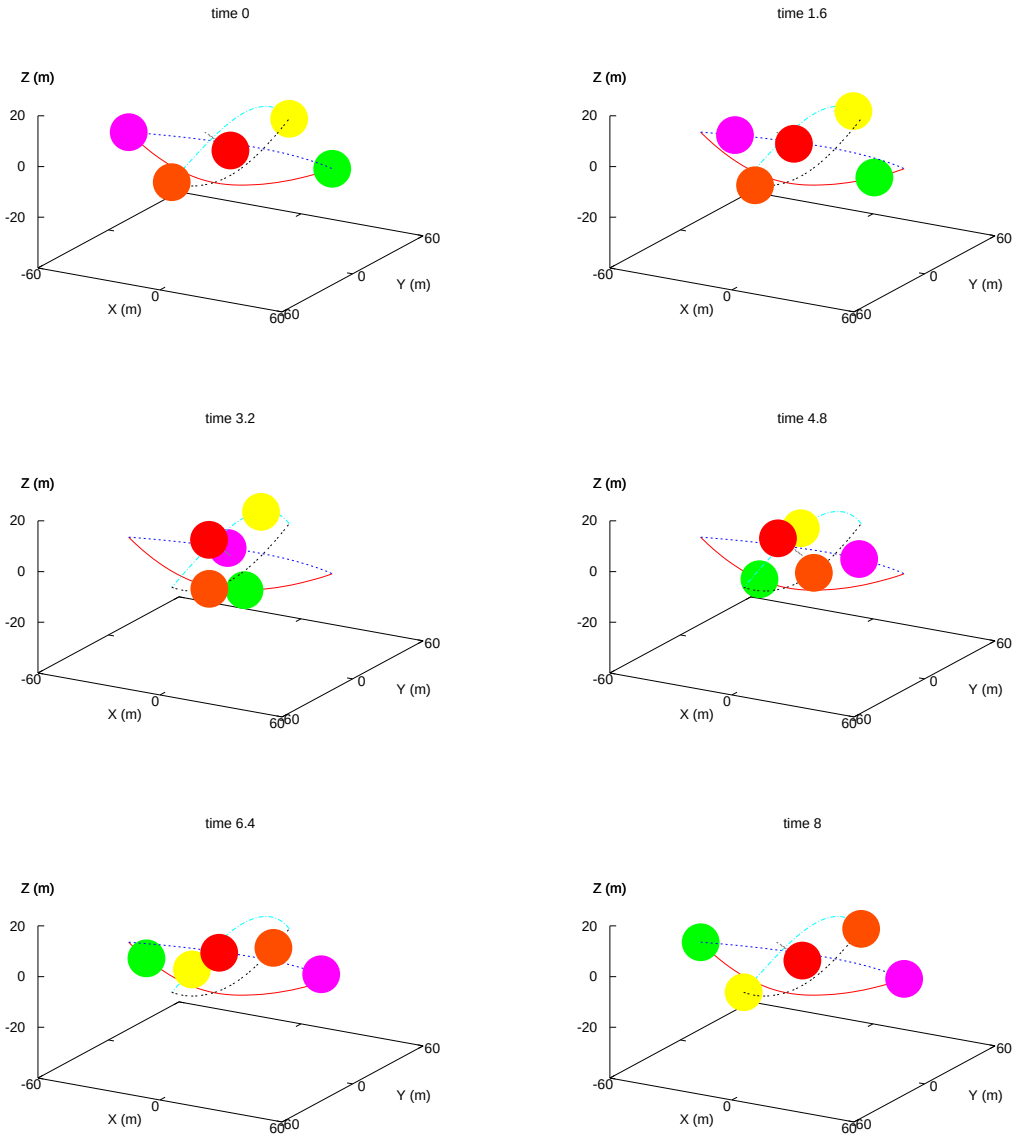

Figure 2.7: Trajectories of the five spacecraft switching positions in opposite vertices of a square. The procedure FEFF-DV2 has made two different planes of motion in order to avoid collisions.

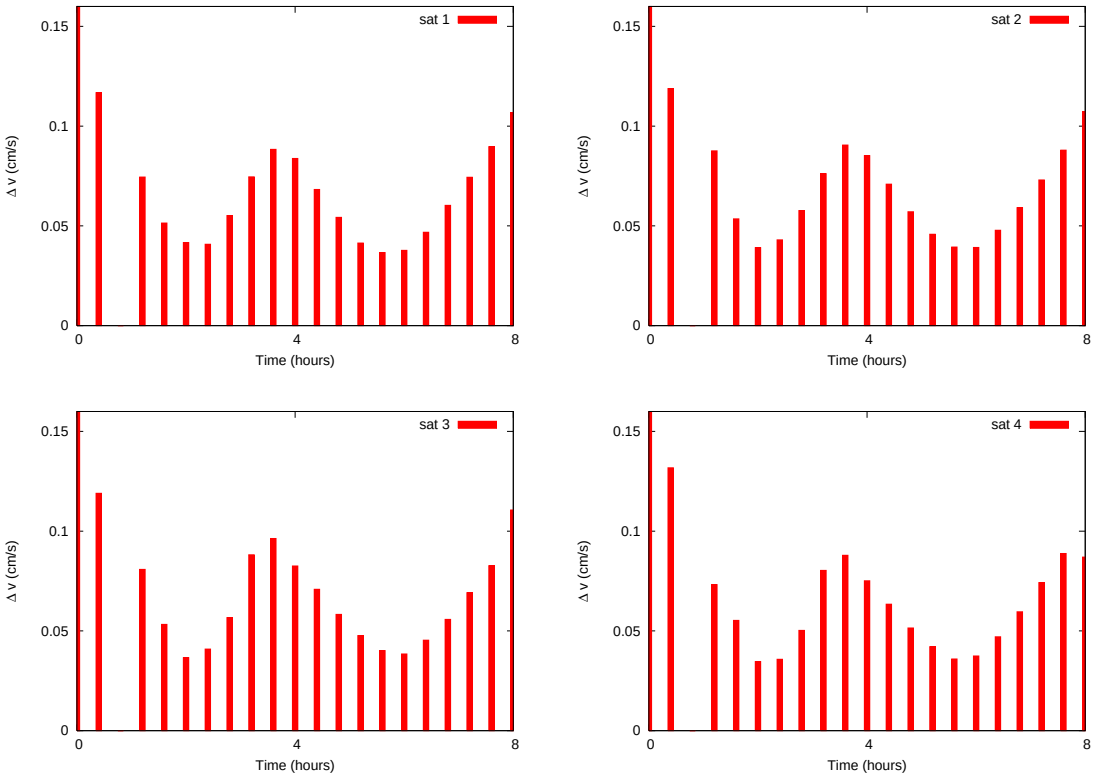

Figure 2.8: Delta-v (in cm/s) possible obtained in the problem of switching the spacecraft of the TPF formation, maintaining the delta-v in the time interval [0.5,1] (in hours) equal to 0.

fined in a sphere centered in the Halo orbit.

- Maintaining the spacecraft in the surface of a paraboloid (in order to make them work as an antenna).

2.5.4 Deployments

In the first steps of the formation, the spacecraft must achieve the final pattern. In some cases, due to the number of spacecraft of the formation, the spacecraft are not in a single stack and the deployment phase might follow after a rendezvous of several motherships. In this case, we can consider deployments as special cases of reconfigurations where the satellites depart from different locations (one location for each mothership) and the final configuration is the pattern of the formation.

In all the reconfiguration examples we have treated before, the security distance is a fixed value for all the time span. However, when performing deployments of formations that are in different stacks, in the first steps of the procedure, the spacecraft are too close (below the usual security distance required) and procedure FEFF-DV2 could not find a feasible solution. In order to avoid this fact, we consider a security distance which is a function of time: it is zero or very small at the beginning (when the spacecraft are near the mothership) and should increase during the reconfiguration phase.

If properly executed, deployments do not have collision risk using bang-bang controls, so the final solutions we obtain are similar to the translations.

Once we have two or more stacks with spacecraft, we can check which is the best place to form the pattern. Let us consider an example with the Darwin configuration: we consider that the 7 spacecraft of the formation are in two different stacks, separated by a distance of 1000 m, and they must form the Darwin pattern in an undefined point in the line between the two stacks.

It turns out that the optimum place for the rendezvous is the point in the line joining the two stacks which is located in the center of mass of the initial configuration, this is, if we work with a stack of 4 spacecraft and another one of 3 spacecraft, the optimal place to assemble the Darwin formation is the point of the line joining the initial stacks located 571.4 m far from the stack of 3 spacecraft and 428.6 m far from the stack of 4 spacecraft. We show the trajectory in figure 2.9 and the profiles of the delta-v expenditures for stacks of 3 and 4 spacecraft in figure 2.10 and for 2 and 5 spacecraft in figure 2.11. Total amounts of Δv are given in table 2.2.

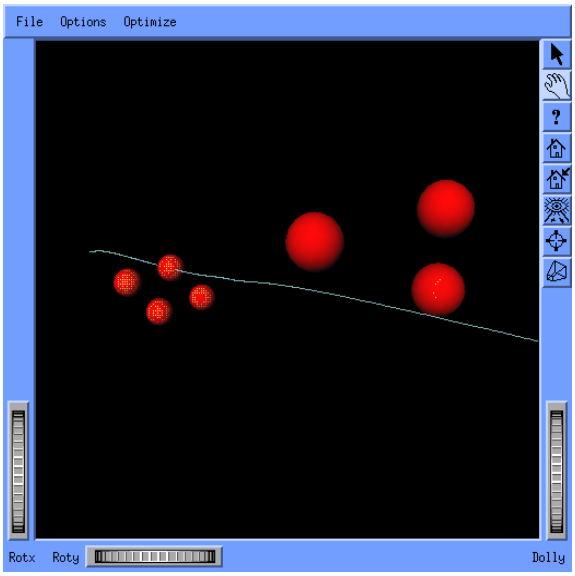

Figure 2.9: A snapshot of a rendezvous trajectory for the Darwin mission.

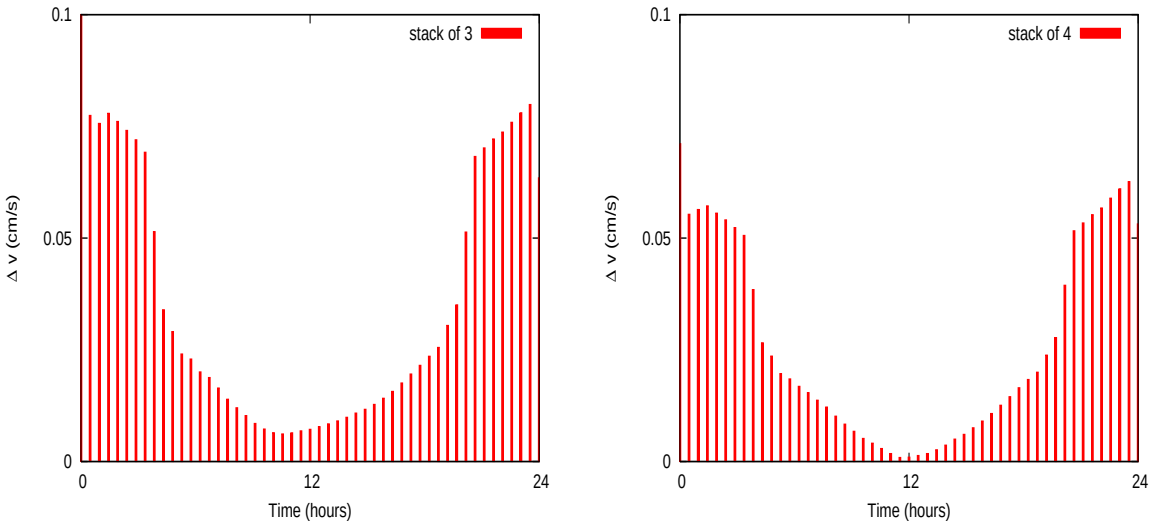

Figure 2.10: The profiles of the Δv expenditure (in cm/s) for a spacecraft of the stack of 3 spacecraft (left-hand plot) and for a spacecraft of the stack of 4 (right-hand plot). Total amounts of Δv are given in table 2.2

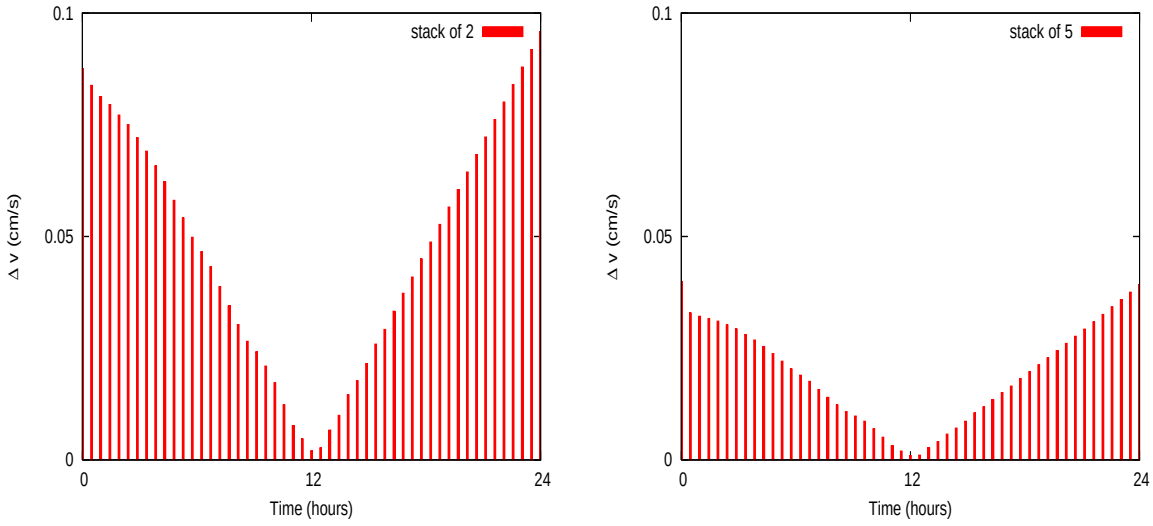

Figure 2.11: The profiles of the Δv expenditure (in cm/s) for a spacecraft of the stack of 2 spacecraft (left-hand plot) and for a spacecraft of the stack of 5 (right-hand plot).

Satellite	1	2	3	4	5	6	7	Total (cm/s)
Δv	1.83	1.83	1.83	1.33	1.33	1.33	1.33	10.81
Δv on-off	1.32	1.32	1.32	0.99	0.99	0.99	0.99	7.92

Table 2.2: Δv cost corresponding to the rendezvous example for the Darwin formation. The two groups of satellites depart from 1000 m apart and perform rendezvous in one day. The example uses 50 linear elements for each satellite.

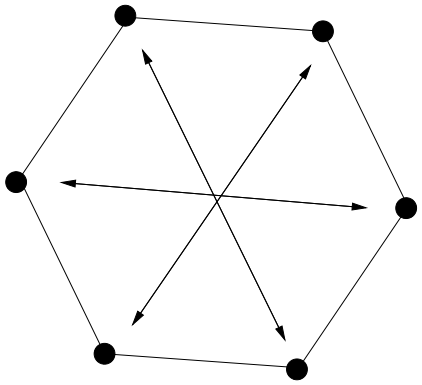

Figure 2.12: Test problem for an hexagon, where all the spacecraft move to the position symmetrical with respect to the center of the hexagon.

2.6 Some comments about computational cost of FEFF-DV2

To analyze in detail the cost of FEFF-DV2, is a complex problem where we must take into account that when we have a lot of spacecraft or security distances similar to the initial distance between the spacecraft, the procedure may need quite a few iterations to find the optimum solution. In this section, we study how the increment of nodes or of security distances affects to the total computational load of the algorithm.

The computations are made with a Pentium at 1.73 GHz, with 1GB of RAM, using a Linux-Debian distribution.

The test problem

In order to calculate the influence of the number of elements and the security distance on the problem, we have considered some problems affected of multiple collision risk. In these examples we consider the spacecraft located in a regular polygon (an n-gon), and the reconfiguration consists on changing each spacecraft of the polygon, to a symmetrical position with respect to the center of the polygon (see figure 2.12).

This kind of reconfigurations are subjected to multiple collision risks. With a bang-bang control, all the spacecraft would collide in the center of the polygon in half the reconfiguration time.

Computational cost increasing the number of elements

In general, when we carry out the optimization using few elements, we obtain the solution in a small number of iterations. When we have more

elements, the number of variables grow, increasing the computational cost of the procedure. For this purpose, we study how we can obtain a refined solution using as initial seed the solution obtained using a small number of elements (that we choose between 6 and 10).

We have considered two different ways to increase the number of elements. The first one consists on increasing a fixed number of elements on each iteration, and the second one consists on multiplying the number of elements by a fixed number.

Using the first method, when we add a fixed number of elements on each iteration, we must take into account that if we add a small number of elements at each step (for example, 1 or 2), on each step we get the solution with few iterations, but the total sum is high, because many steps could be required. On the other hand, if we add a big number of elements in a single step, the cost to obtain the corresponding optimum can be high.

To exemplify this situation, we have considered a particular case: we have 4 spacecraft in the vertices of a square and the reconfiguration consists on switching the spacecraft in the opposite vertices. We have added a fixed number of elements on each step and the results are displayed in figure 2.13. Is it clear that the worst case is when we add one element at each step. The cost for each step is small, but this does not compensate the large amount of steps needed. We also observe that for small number of elements, the cost is lower if we do not add a big number of elements to the previous mesh, but when the number of elements increase, is better to add bigger numbers.

The idea of the second way is a consequence of the results obtained with the first method. When we have a small number of elements, the optimum procedure consists in adding a small number of elements, and when we have already a bigger number of elements, is better to increase with bigger number of elements. The purpose is then to multiply the number of nodes of the mesh by a fixed number (i. e., increasing the number in an exponential way).

We have used the same example as in the previous case, and we have multiplied the number of elements by 1.5, 2, 2.5 and 3. We can see in figure 2.14 that the computational cost is similar with the different values, but that the best ones are obtained doubling the number of elements at each step.

Once we have the two results, we can compare the best results of each algorithm, and we observe that the results obtained with the first method are better in general: using the exponential method, we make less iterations, but the total cost is bigger, as it can be seen in figure 2.15.

In conclusion, when we want to reduce the computational time of the algorithm for a given problem with a lot of nodes, the best strategy is to sum a fixed quantity of nodes, which is less than the number of nodes of the best trajectory we have, but this quantity must be significant when compared

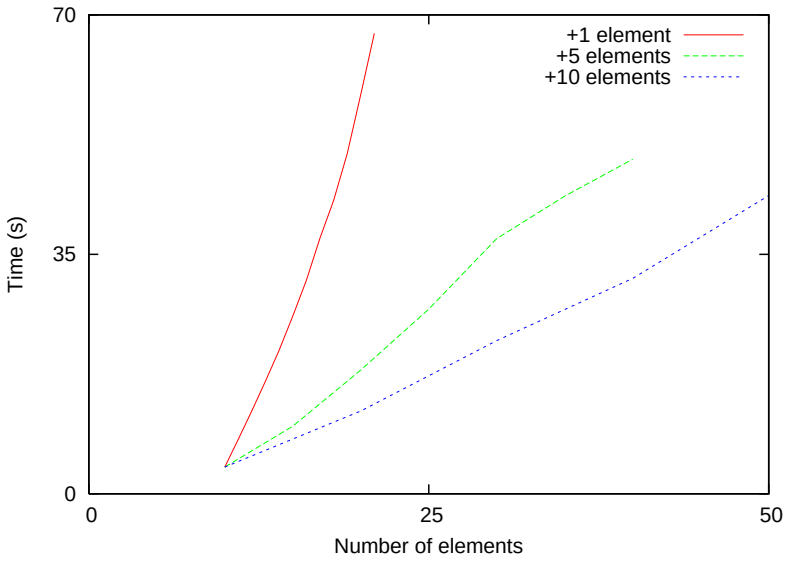

Figure 2.13: Computational cost (in seconds) of the reconfiguration of a formation of spacecraft depending on the number of elements. On each step, we increase the number of elements by adding a fixed number of elements.

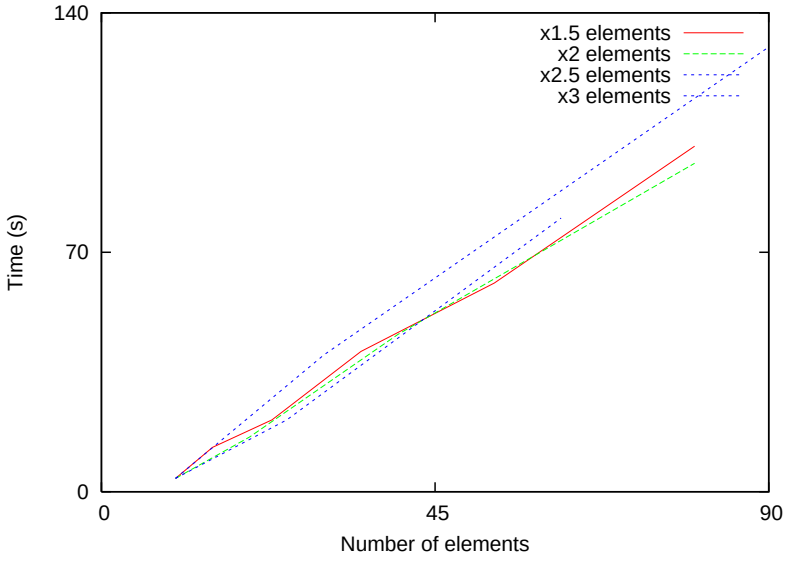

Figure 2.14: Computational cost (in seconds) of the reconfiguration of a formation of spacecraft increasing the number of elements by multiplying the number of elements by a fixed value at each step.

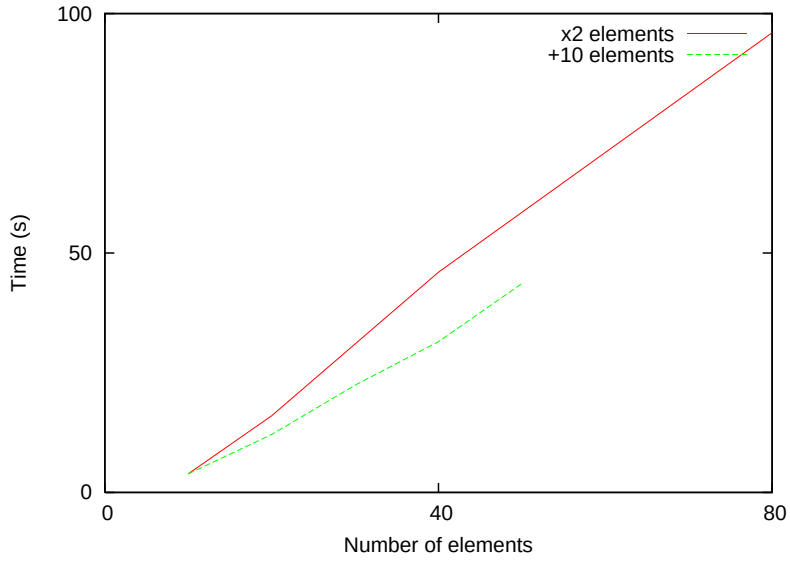

Figure 2.15: Computational cost (in seconds) of the reconfiguration of a formation of spacecraft increasing the number of elements by adding a fixed number of elements on each iteration or by multiplying the number of elements by a fixed number.

with the number of nodes.

We note also that the procedure of adding nodes to the existing mesh can also be understood as a way to refine the trajectory towards a low thrust control.

Computational cost increasing the security distance

In problems with multiple collisions, we must take into account that the FEFF-DV2 procedure may have problems to reach the optimum, specially when the security distance is big when compared to the size of the formation. In this case, we can use a continuation method to obtain the reconfiguration trajectories: we can start computing a solution with a small security distance, and then to increase it by steps using the previous solution as initial seed.

When computing reconfiguration trajectories in this way there is no need to have accurate intermediate results since, in fact, they are just initial seeds for the next step. For this reason, if we want to obtain the solution with a big number of elements, we first find an approximation of the solution using few elements and then we refine the obtained solution increasing the mesh density once the required security distance is attained.

Similar to the study of the computational cost increasing the number of elements, we study the process of increasing the security distance adding a fixed amount at each step and trying to find in this way the number of

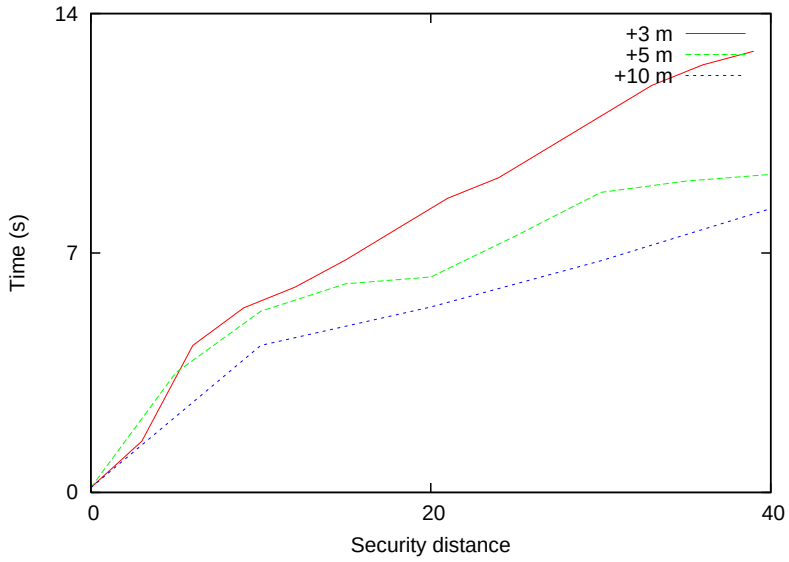

Figure 2.16: Computational cost (in seconds) of the reconfiguration of a formation of spacecraft increasing the security distance with different number of intermediate steps.

intermediate steps that optimize the process. To illustrate it let us consider 6 spacecraft in the vertices a regular hexagon of 100 meters of edge and the problem consists in switching the spacecraft located in opposite vertices. We evaluate the computational time needed to obtain the final solution, with a security distance of 90 meters, using a continuation method with different number of intermediate steps. The results are presented in figure 2.16.

We observe that the first iteration is always of the ones with a bigger cost, because we start from a situation with multiple collision risks and we end up with a situation without collision. For this particular case, we see that there is a limit, near the security distance of 40 meters, where the solution cannot be continued. The local continuation to obtain the optimum can end in a point with no feasible solution and, in this case, we must obtain the optimum trajectory for the spacecraft using the initial guess, without the continuation method.

Chapter 3

Remeshing strategies

In the previous chapter, we have developed a methodology to obtain an approximation for the reconfiguration of a spacecraft formation, using a fixed mesh, which has been taken the same one for all of the spacecraft. In this chapter, we consider elements of different length and different meshes adapted to each one of the trajectories.

The use of elements of different length is introduced for several reasons. However, and first of all, there is a main motivation on finding what we could say "an optimal mesh". As it can be seen in the examples of chapter 2, the maneuvers that must be applied to the spacecraft are bigger in some time spans and very small in some other. In the first case, the use of more elements inside the time span, could make nodal maneuvers smaller, while in the second case, the elements could be longer, because there are no big changes in the velocities as a consequence of the applied delta-v.

With the underlying idea that the requirements are different for each spacecraft and depending on its trajectory, for each spacecraft we are going to consider its own mesh in time.

Taking advantage of our finite element methodology, we have introduced a remeshing method, FEFF-A, with the goal to control and optimize the mesh and the computations involved in reconfigurations.

Moreover, in parallel with the remeshing method, we have also introduced a modification of the methodology FEFF-DV2, consisting in minimizing the sum of the modulus of the delta-v applied, not of the sum of squares of the Δv. This is a functional more related to the fuel consumption of the satellites. However, the method which minimizes this new penalty function (3.1), also needs elements of different length, in order to avoid the problems of computing the derivatives of the function near zero.

3.1 Using elements of different length

To use elements of different length, makes no essential difference in the formulation of our finite element methodology. The matrix of equation (2.20) associated with each element that we have obtained in section 2.2.2, does not take into account the fact that all the elements have the same length, and so it is also valid when the elements are of different size.

In section 2.4.3 we considered all elements of the same length to prove that matrix $A^TA + C$ in this case is not singular. When the elements are of different length, we cannot prove this fact. However, this matrix would be singular only in few (degenerate) meshes, and in most of the ordinary cases we will still be able to find an initial seed following the same algorithm. Anyway, there is also another way to proceed: we first obtain an initial seed considering elements of the same length and then we use the methodology of the previous chapter to continue working with uneven meshes.

We also note that since the value of the penalty function (2.27) is computed for each spacecraft independently, the computation of this function and its derivatives is carried out in the same way as if all the elements were of the same length. In fact, the essential difference in this case is related to collision avoidance. Again, as in section 2.4.2, collision avoidance enters in the problem in the form of constraints. In this case, since each spacecraft has a different mesh, in order to compute whether constraints are violated or not, we consider another mesh considering the union of nodes of the meshes of the two spacecraft under consideration. We call this mesh *the extended mesh*.

With this new mesh, the approximation of the trajectory with the finite element method for the two spacecraft in all the checking time spans is again linear. Then, on each element, the functional giving the minimum distance between the spacecraft can be computed, and we can check whether this distance is greater than a given security distance or not.

Computing derivatives of the constraints

Consider two different spacecraft, i and j and the extended mesh associated with them. When we need to compute the derivatives of the constraints in one of the elements of the extended mesh, we have to consider different cases depending on the provenance of the nodes of the extended mesh:

- both nodes of an element Ω^n are also nodes of the mesh of satellite i
- one of the nodes of an element Ω^n is also a node of the mesh of spacecraft

i, but the other one is only a node of the mesh of spacecraft j

- none of the nodes of the element Ω^n of the extended mesh belongs to the mesh of spacecraft i

This fact generates different ways of computation of derivatives:

Case 1: The nodes of the element Ω^n of the extended mesh are nodes of the mesh of spacecraft i.

This case is the same one we dealt in section 2.4.2, since in fact we are computing the derivatives in an element that belongs to the mesh of spacecraft i.

Case 2: The first node of Ω^n belongs to the mesh of satellite i, but the second one is a node only of the mesh of satellite j.

We consider an element Ω^n of the extended mesh of satellites i and j. This element has two nodes, at epochs t_n and t_{n+1}. In this case, t_n is also a node of the mesh of spacecraft i (the k-th node of this mesh), but the node at time t_{n+1} does not belong to mesh of spacecraft i.

The distance between spacecraft (and also the restriction) depends on the position of spacecraft i at node k, but also depends on the position of the spacecraft i at the following node of the mesh of satellite i: $k+1$, which does not belong Ω^n (see figure 3.1).

Inside Ω^n the position of spacecraft i is obtained by interpolation of the spacecraft positions at nodes k and $k+1$ of its own mesh:

$$\mathbf{X}(t) = \mathbf{X}(t_k)\frac{t_{k+1}-t}{t_{k+1}-t_k} + \mathbf{X}(t_{k+1})\frac{t-t_k}{t_{k+1}-t_k}.$$

In particular, the position at node $n+1$ of the extended mesh (the node of Ω^n which does not belong to the mesh of spacecraft i) is:

$$\mathbf{X}(t_{n+1}) = \mathbf{X}(t_k)\frac{t_{k+1}-t_{n+1}}{t_{k+1}-t_k} + \mathbf{X}(t_{k+1})\frac{t_{n+1}-t_k}{t_{k+1}-t_k}.$$

Since the position of spacecraft i inside Ω^n depends on the position of spacecraft i at nodes k and $k+1$, we have to compute two derivatives for the spacecraft i: the derivative of the function (2.28) with respect to the position at node k and with respect to the position at node $k+1$ of the mesh of satellite i.

Let us also consider the constraint $c(\mathbf{x}) = d(t) - R \geq 0$ as in section 2.4.2. We want to compute $\partial c/\partial r_n$, the derivative of the constraint with respect to the nodal value of the function at the first node. We note that this value is

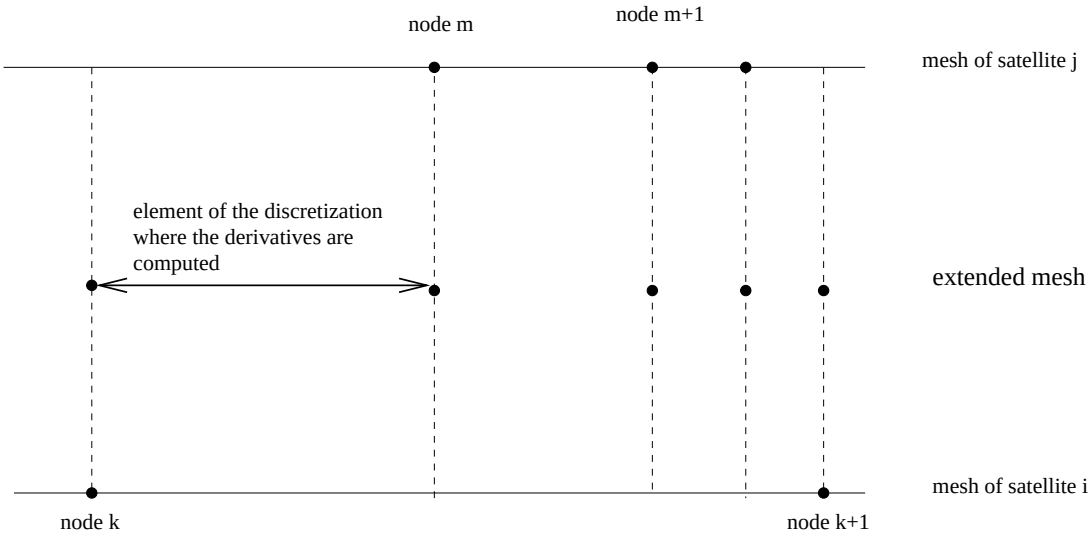

Figure 3.1: The extended mesh related to spacecraft i and j is obtained by the union of nodes of meshes of satellite i and satellite j. In this case we are computing the derivatives with respect to the nodal values of spacecraft i. The first node of the element (node k) belongs to the mesh of spacecraft i, but the second one (node m) only belongs to the mesh of spacecraft j. However, the value of the differential depends on the nodal value k, but also on the nodal value $k+1$.

related to the position of spacecraft, but after uncoupling the equations, it does not have a physical significance. This derivative can be computed as in section 2.4.2. $\partial c/\partial r_{n+1}$, the derivative with respect to the value of the function on the second node, must be computed in a different way. In this case, we obtain the expression for the derivatives of the constraint:

$$\frac{\partial c}{\partial x_k} = \frac{\partial c}{\partial r_n} + \frac{\partial c}{\partial r_{n+1}} \frac{t_{k+1} - t_m}{t_{k+1} - t_k},$$

$$\frac{\partial c}{\partial x_{k+1}} = \frac{\partial c}{\partial r_{n+1}} \frac{t_m - t_k}{t_{k+1} - t_k}.$$

Case 3: The second node of Ω^n belongs to the mesh of satellite i, but the first one does not.

This case is the opposite of the previous one, and derivatives are computed in a similar way. Now, the element Ω^n has two nodes: n and $n+1$. The node $n+1$ of the element belongs to the mesh of spacecraft i, but the node n only belongs to the mesh of spacecraft j.

When computing the derivatives of the constraint in Ω^n with respect to the nodal positions of spacecraft i, we must consider the nodal value $n+1$ of the extended mesh for spacecraft i (we can denote it as k-th node of the mesh of spacecraft i) and the previous node of the mesh of spacecraft i, this is, the node $k-1$.

Analogous to the previous case, we can compute the derivatives of the constraint with respect to the nodal position on nodes k and $k+1$, obtaining:

$$\frac{\partial c}{\partial x_k} = \frac{\partial c}{\partial r_n} \frac{t_k - t_n}{t_k - t_{k-1}},$$

$$\frac{\partial c}{\partial x_{k+1}} = \frac{\partial c}{\partial r_n} \frac{t_n - t_{k-1}}{t_k - t_{k-1}} + \frac{\partial c}{\partial r_{n+1}}.$$

Case 4: None of the nodes of Ω^n belong to the mesh of satellite i.

Also when the nodes n and $n+1$ of element Ω^n of the extended mesh belong only to the mesh of spacecraft j, we will need to compute the derivatives with respect to the nodal positions of the mesh of spacecraft i. The nodes to be considered inside the mesh of spacecraft i are consecutive. Let us note them as k and $k+1$, where node k is before the node n of the extended mesh, and node $k+1$ is after the node $n+1$ of the extended mesh.

Using similar calculations as in the previous cases, the derivatives that will be needed are:

$$\frac{\partial c}{\partial x_k} = \frac{\partial c}{\partial r_n} \frac{t_{k+1} - t_n}{t_{k+1} - t_k} + \frac{\partial c}{\partial r_{n+1}} \frac{t_{k+1} - t_{n+1}}{t_{k+1} - t_k},$$

$$\frac{\partial c}{\partial x_{k+1}} = \frac{\partial c}{\partial r_n}\frac{t_n - t_{k-1}}{t_{k+1} - t_k} + \frac{\partial c}{\partial r_{n+1}}\frac{t_{n+1} - t_k}{t_{k+1} - t_k}.$$

3.2 The minimization of $\sum |\Delta v|$ (FEFF-DV)

Our goal is the reconfiguration of the formation using the minimum fuel consumption of the spacecraft. Attending the way the maneuvers are made, the fuel expenditure of the spacecraft i is directly related to the sum of the modulus of the maneuvers,

$$\sum_{i=0}^{N}\sum_{k=0}^{M_i} \rho_{i,k}||\Delta v_{i,k}||, \tag{3.1}$$

where $||*||$ denotes the Euclidean norm, M_i is the total number of nodes in the trajectory of spacecraft i and ρ is a penalty parameter. However, the drawback of this expression is the numerical problem when computing derivatives for small values of delta-v (and this is what we want to achieve in our computations).

In the previous chapter, we have considered the functional to be optimized

$$\sum_{i=0}^{N}\sum_{k=0}^{M_i} \rho_{i,k}||\Delta v_{i,k}||^2. \tag{3.2}$$

This functional is also related to fuel expenditure, moreover it is smooth and derivable and it is not ill conditioned.

The general idea of the procedure that we are going to develop is to use the functional of equation (3.2) to obtain a first approximation of the optimal trajectory, as we have seen in previous chapter. This preliminary solution avoids collision, and minimizes the fuel in some way, so we can think that the trajectory we are looking for is close to this one. Once we have this first guess for the optimal trajectory we are looking for, the idea is to change the mesh for each spacecraft, depending on the properties of the trajectory we have obtained when minimizing the delta-v square functional (3.2). Using this new mesh, we will compute the trajectory which minimizes the delta-v norm functional (3.1), avoiding ill-conditioned problems..

3.2.1 Remeshing

As it has been stated before, minimizing the delta-v square functional (3.2), we obtain a rough approximation for the delta-v to be applied on each node.

In order to make the algorithm more efficient when minimizing the functional (3.1) and at the same time avoiding problems in the derivatives of the functional when they are near zero, we have developed a technique which adds or subtracts nodes of the meshes by means of an iterative process in a controlled way.

In general words, in a first iteration the algorithm suppress some of the nodes where the associated maneuvers are less than a given threshold V_m and adds more nodes near the ones with maneuvers bigger than another threshold V_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 that 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.

The methodology consists on a serial of tests and procedures according to the results. The tests check whether the constraints on the delta-v are achieved or not, and in case they are not achieved, the procedures change the mesh trying to achieve them. Tests and procedures are iterated until the mesh we obtain satisfies all the constraints.

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 (3.1). Test I is directed to detect them. To this end, the test compares all the maneuvers in the trajectories with a given threshold V_m. The current mesh passes the test if all them are bigger than V_m. Otherwise, some of the nodes are removed using procedure I.

Taking out nodes in procedure I is conditioned by the current Δv 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 be careful, since we could end up with a mesh with a long time interval without nodes. For example, in figure 3.2 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 to the next iteration. Because of this, the 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

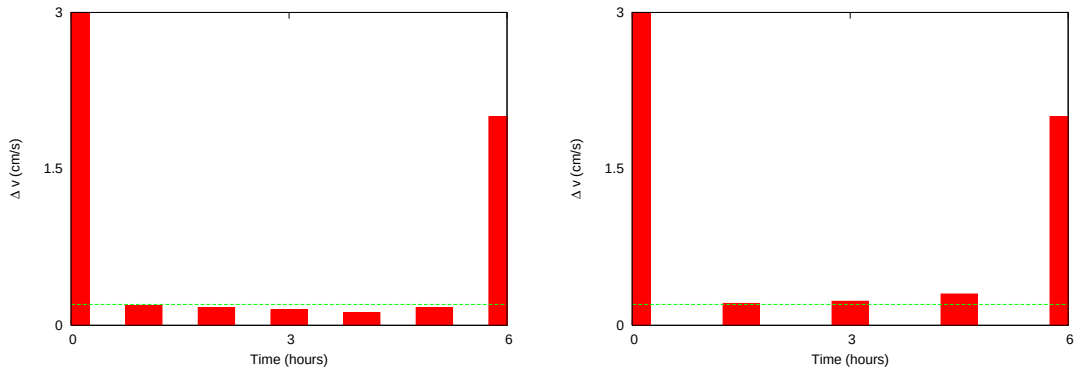

Figure 3.2: On the left, we represent a sequence of maneuvers obtained at some iteration of the optimization procedure for a particular satellite. We have some maneuvers smaller than the threshold (dotted line) in the central part. In this case, instead of removing all the nodes, we change their density as it is displayed in the panel on the right.

by a factor depending on the ratio between the total amount of delta-v and the threshold, as can be seen in the figure.

Test and Procedure II: Adding nodes to split big maneuvers

In some cases, like when in an iteration we have maneuvers larger than the maximum ones allowed, V_M, it is necessary to add more nodes to the time mesh. The task of Test II is to detect maneuvers bigger than the threshold V_M, whereas the 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 V_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.

Test III and Procedure III: Re-meshing using relative neighbor values

In our methodology, tests I and II, and their respective procedures, are iterated while there exist nodes with maneuvers smaller than V_m or bigger than V_M. However, after the application of this algorithm, one could end up with some undesired mesh. 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 (3.1) to be ill conditioned for the derivatives and the procedure suffer of convergence problems. For this reason, when test I and II are passed, we check in test III if there exist small maneuvers when compared to the ones in their neighborhoods (see figure 3.3). In a positive case we apply procedure III to remove the nodes corresponding to these relatively small maneuvers.

Procedure III compares the ratio between the delta-v of the trajectory in the current iteration and the length of the elements where the node belongs to:

$$\delta_k = \frac{\Delta v_k}{t_{k+1} - t_{k-1}}.$$

The test considers that the maneuver is small enough to be removed when

$$\delta_k < \frac{\delta_{k+1}}{C} \quad \text{and} \quad \delta_k < \frac{\delta_{k-1}}{C},$$

where C is a constant. If this constant is chosen small, then we could remove more nodes than we expected. If C is too big, then we may not remove some nodes that are small enough to be removed. Some explorations have shown that suitable values for C are inside the range [8,10].

When the procedure does not pass test III, it means that there are small maneuvers when compared with the neighboring ones. The nodes that do not pass test III are isolated, because if the delta-v of a node is small with compared with its neighbors, the neighboring ones can not be small when compared with it. Procedure III suppress all the nodes that have not passed test III.

The limiting case of a bang-bang solution

In a case without collision risk using rectilinear trajectories, as we pointed out, we know that the optimum trajectory for each spacecraft is a bang-bang control. Because of the number of nodes with zero delta-v associated, this is a critical case for the methodology FEFF-DV.

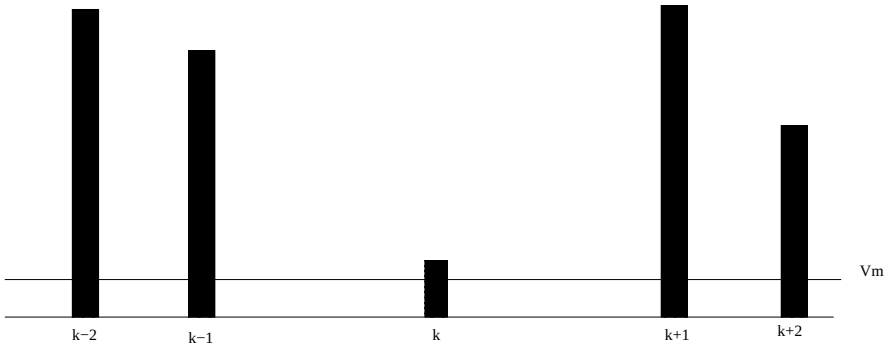

Figure 3.3: Delta-v corresponding to a trajectory which has passed tests I and II, and is considered for test III. In this case, we compare the relative value of the maneuver on node k with the ones on nodes $k+1$ and $k-1$. In test III we do no take into account only the amount of delta-v; we also take into account the length of the elements. In this case, test III detects that the maneuver in node k is small when compared with neighboring ones, and procedure III will eliminate the node.

We remark that the procedure of minimization of the delta-v square functional (3.2) does not provide a bang-bang solution. Generically, it gives a trajectory with a delta-v profile similar to the one of figure 3.4. Moreover, after the first iteration of the algorithm, nodes corresponding to delta-v values minimizing (3.2) cannot be removed, because the threshold criterion does not apply to them (it only applies to the node or nodes near the middle zone of the reconfiguration time interval). Typically, the value of the functional (3.2) evaluated in the bang-bang optimal control trajectory is very large when compared with its optimum. However, the algorithm FEFF-DV is robust enough even in this case. Once we have the solution of (3.2), only few nodes can be eliminated by procedure I, but using the test and procedure III, all the interior nodes are eliminated in few iterations.

To summarize, our methodology FEFF-DV of optimal control search follows the scheme represented in figure 3.5. First of all, we find the solution of the problem using the delta-v square functional (3.2). Using this trajectory as a initial seed, we minimize the delta-v norm functional (3.1) at the same time that we control the density of nodes in the mesh. Moreover, convergence can be directed towards the limiting cases of bang-bang control (when it is feasible) or towards low thrust controls in general situations.

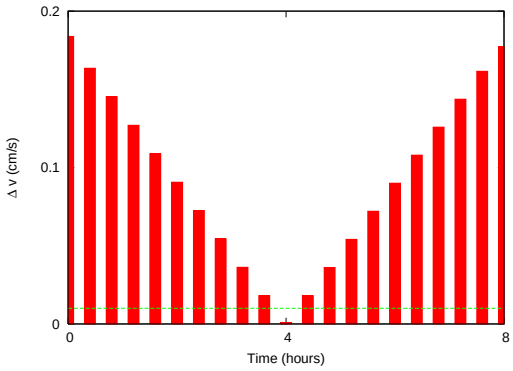

Figure 3.4: In the first step of the computation of the reconfiguration of spacecraft without collision hazards, the profile of delta-v has always this shape. If we apply the tests and procedures I and II to obtain a better mesh, we only suppress the delta-v corresponding to a node, the one which is smaller than the threshold represented with a bold line.

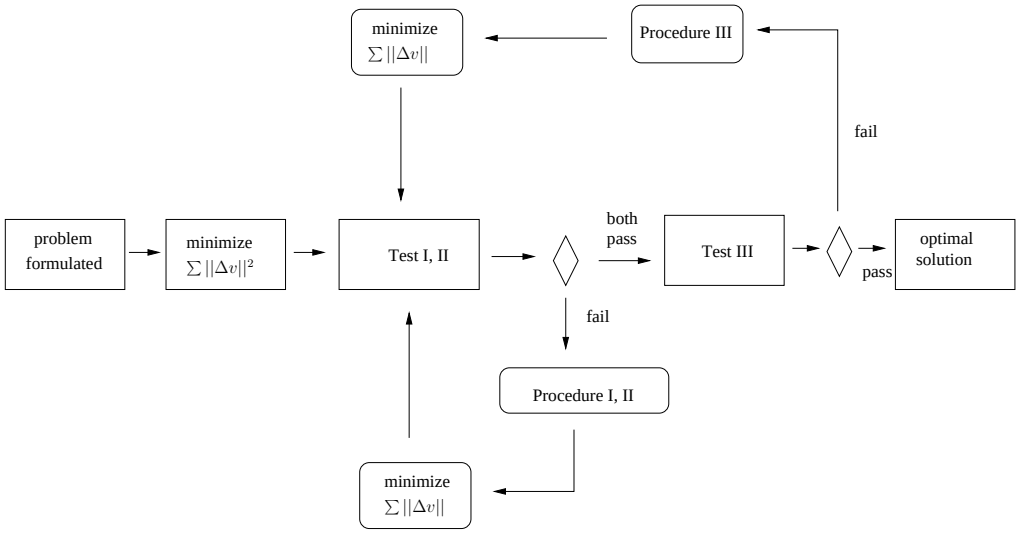

Figure 3.5: Scheme of the procedure FEFF-DV.

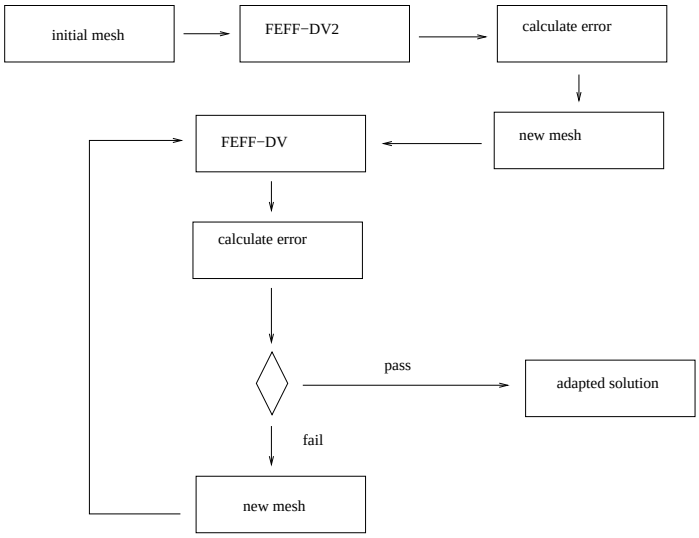

Figure 3.6: Schema of the procedure of adaptive remeshing (FEFF-A).

3.3 Adaptive remeshing (FEFF-A)

Adaptive remeshing is a methodology which can be applied to solving differential equations using the finite element method allowing to control both the error in the obtained solution and the computational cost [21].

The approximation of the solution via the finite element method, gives some errors associated to each one of the elements of the mesh. The general idea of adaptive remeshing is that, given a value e, we want to find a mesh that provides an approximate solution with error (understood as the difference between the solution of the problem and its approximation inside of an element) less than e.

In figure 3.6 we give a schema of its general idea. The procedure has two different phases. In the first one, it starts computing the solution of a problem using the finite element method with a given mesh, and then it computes an estimation of the error (this error is essentially obtained by comparison between the gradient obtained using the finite element model and the one obtained by integration of the equations of motion, as we will see in the following section). Once we have this approximation of the error, we must decide whether the error is below a given tolerance or if we need to remesh and to recompute the approximate solution. When remeshing is necessary, the new mesh is adapted using the estimation of errors of the previous mesh.

Adaptive remeshing methods penalize the elements where the error is

considered big, dividing them in smaller elements. On the other hand, if the estimation of the error is small in an element, then this element is made bigger in the next iteration. Since, essentially our estimation of the error is related to the value of the delta-v that we must implement, this method tends to increase the length of the elements which have small delta-v associated and tends to decrease the length of the elements which have big delta-v associated. We note that, in fact, the idea is similar to some parts of our FEFF-DV methodology.

3.3.1 Error estimates

In adaptive remeshing techniques, for each element, the associated error is computed by comparison of two gradients. The one obtained with the solution given by finite element interpolation (the estimated gradient from the finite element interpolation) and the one obtained by integration of the equations of motion (the exact gradient). This is an ordinary procedure that gives us an error estimation for each element, which is suitable in order to decide if our the mesh is good enough.

In our case, the computation of these gradients is simpler than in the general finite element implementations because our elements are in dimension one and in fact corresponds to the velocities. Since we work with linear elements, the estimated gradient is constant on each element. In order to compute the exact gradient, we integrate the equations of motion, $\dot{\mathbf{x}} = A(t)\mathbf{x}$ (equation 2.4) on each element (from t_k to t_{k+1}) and we obtain a gradient function for each of the elements, that now is not constant.

Let us define $\mathbf{v}_1$ as the estimated gradient obtained by means of the approximated solution of the finite element method. This gradient is constant on each element and has only one component because our elements are one-dimensional. Let us denote $\mathbf{v}_2$ the gradient obtained integrating the equations. The error estimation in the element Ω^k is defined as a norm of the difference between these two gradients:

$$e_k = ||\mathbf{v}_1 - \mathbf{v}_2||_{L_2} = \left(\int_{t_k}^{t_{k+1}} (\mathbf{v}_1 - \mathbf{v}_2) \cdot (\mathbf{v}_1 - \mathbf{v}_2) dt \right)^{1/2}, \qquad (3.3)$$

computed by means of numerical quadrature.

Acceptability criteria

Once we have an estimation of the errors, we need to decide whether the mesh is good enough for our purposes. We define the global error of the mesh, $||e||$ as the modulus of the vector containing all the elemental errors, this is,

$$||e|| = \sqrt{e_1^2 + e_2^2 + \ldots + e_M^2} \tag{3.4}$$

Essentially to decide whether the current mesh is good enough or not is based in two different criteria [5]: to compare the modulus of the errors with the modulus of the obtained solution or to compare it with the length of the elements.

- **Comparison with the value of the integrated solution:** The first criterion consists on comparing the modulus of the elemental errors with the total gradient of the solution,

  $$||\bar{u}|| = \int_0^T \mathbf{v}_2 dt.$$

  We accept the mesh when

  $$||e|| \leq \nu ||\bar{u}||,$$

  where ν is the acceptability criteria. This value is usually chosen as 0.05 in the literature.

- **Comparison with the length of the elements:** We can impose a criterion to compare the total error with the length of the elements in the mesh. Since the elements have different lengths, we consider h, the length of the longest element in the mesh (also known as the diameter of the mesh) and we impose the acceptability criterion:

  $$||e|| \leq \alpha h^m,$$

  where m is a constant that depends on the functions of the problem, but it is related with the maximum degree of the polynomials included in the interpolation. In second-order elliptic problems, m is usually taken as the degree of the interpolating polynomials. In our case, we take $m = 1$. α depends on the distortion of the elements, and it is proportional to a measure of the $m + 1$ derivatives of function. In our case, α is proportional to the second derivative of the function and we have considered it 1.

We note that we must deal with different types of reconfigurations. One of them is the reconfiguration of a formation where the optimal solution could be a bang-bang control. In this case, we want to end up with a solution with a big h. Another class of reconfigurations are the ones that end up with a

low-thrust trajectory. In this case, the length of the elements will be very small.

In the reconfigurations that end up in a bang-bang control, we expect that our final mesh would be a mesh with only one element, with a length equal to the reconfiguration time. This final mesh, using the criterion of comparison with the length of the elements, would accept errors proportional to the reconfiguration time. To avoid this fact, we use the first criterion.

3.3.2 The obtainment of a new mesh

In the literature we can find different methods for remeshing when the error is bigger than a given threshold. In all of them, the objective is to obtain a new length for the elements. The calculus of the new length depends on the elemental error e_k of the elements.

In all this section, we denote by $||e||$ the modulus of the error as defined in equation (3.4) for a given spacecraft, and e_k the estimated error in the k-th element (3.3). M is the number of elements of the mesh for the spacecraft. To differentiate the old or previous mesh (the one on which we have just applied the finite element method) and the new one (the mesh that we recompute depending on the errors and that will be used in the next iteration of the finite element method), the parameters of the new mesh are labeled with a hat.

- **Li and Bettess remeshing strategy:** The Li and Bettess strategy [19] is based on the idea that the error distribution on an optimal mesh is uniform,

$$||\hat{e}_k|| = \nu||\bar{u}||/\sqrt{\hat{M}},$$

where ν is again the acceptability criteria. This method consists on finding the new length of the elements using the number of elements of the new mesh, $\hat{M}$. Let us denote d the dimension of the problem and m the maximum degree of the polynomials used in the interpolation. Then, according to Li and Bettess, the number of elements needed by the new mesh is,

$$\hat{M} = (\nu||\bar{u}||)^{-d/m} \left(\sum_{k=1}^{M} ||e_k||^{d/(m+d/2)} \right)^{(m+d/2)/m}.$$

Working with linear elements ($m = 1$) in dimension one ($d = 1$), as is the case of our study, the recommended number of elements of the new

mesh is

$$\hat{M} = (\nu||\bar{u}||)^{-1} \left(\sum_{k=1}^{M} ||e_k||^{2/3} \right)^{3/2}.$$

Once we have the estimation of the number of elements, we can find the length of the new elements:

$$\hat{h}_k = \left(\frac{\nu||\bar{u}||}{\sqrt{\hat{M}}||e_k||} \right)^{1/m+d/2} h_k,$$

that in our case, turns out to be

$$\hat{h}_k = \left(\frac{\nu||\bar{u}||}{\sqrt{\hat{M}}||e_k||} \right)^{3/2} h_k.$$

- **Zienkiewicz and Zhu remeshing strategy:** The Zienkiewicz and Zhu remeshing strategy [37] can be seen as a simplified version of the Li and Bettess strategy. In this case, the new mesh is optimal if the error distribution of the new mesh is uniform referred to the elements of the old mesh. With this criterion, the length of the new elements are:

$$\hat{h}_k = \left(\frac{\nu||\bar{u}||}{\sqrt{M}||e_k||} \right)^{1/m} h_k,$$

so in our case,

$$\hat{h}_k = \frac{\nu||\bar{u}||}{\sqrt{M}||e_k||} h_k.$$

- **Oñate and Bugeda remeshing strategy:** The idea of this strategy is to refine elements according to the density of errors: we consider that a mesh is optimal if the square of the error per unit area or volume is the same over the whole mesh. We note that, working with one dimensional problems, this area is equal to the length of the elements. We denote the area of the elements, A_k and the total area $A = \sum A_k$. Then, the length for the new elements is then computed with

$$\hat{h}_k = \left(\frac{\nu||\bar{u}||A_k^{1/2}}{||e_k||A^{1/2}} \right)^{1/m} h_k.$$

We note that with one dimensional problems ($A_k = h_k$ and $A = T$) and using linear elements ($m = 1$), the new length can be written as

$$\hat{h}_k = \left(\frac{\nu||\bar{u}||}{||e_k||T^{1/2}} \right) h_k^{\frac{3}{2}}.$$

Like in the choice of the acceptability criterion, we do not use the Oñate and Bugeda strategy, because in reconfigurations that end up in a bang-bang solutions, there is only one element, with a length equal to the reconfiguration time. In our case, the other two criteria give similar results as we will see in following sections, but we finally recommend the Li and Bettess strategy.

Remeshing in dimension one

The theory based on adaptive remeshing methods (see [5]) is developed for meshes in two or three dimensions. However in our case, the elements are in time, and the remeshing method only needs to assign a new length for the different elements of the mesh.

For our procedure FEFF-A, we have considered the Li and Bettess remeshing strategy (see [19]), applied to dimension one. We accept the mesh if

$$||e|| < \nu||\bar{u}||,$$

with ν the acceptability criteria for the algorithm.

Then the number of elements of the new mesh ($\hat{M}$) is then computed by:

$$\hat{M} = (\nu||\bar{u}||)^{-1}(\sum_{k=1}^{M} ||e_k||^{2/3})^{3/2},$$

and, for each element, the new length is:

$$\hat{h_k} = (\frac{\nu||\bar{u}||}{\sqrt{\hat{M}}||e_k||})^{2/3} h_k.$$

3.4 Some simulations using remeshing techniques

In the previous sections, we have considered two different techniques with the purpose of finding a mesh that is optimal in some sense. The first technique, FEFF-DV, is developed taking into account the fact that we need meshes where the computation of derivatives be not ill conditioned. The second technique, FEFF-A, comes from the general theory of the finite element method.

Now we are going to check both techniques in different kinds of reconfigurations we are dealing with. These reconfigurations essentially can be reduced to two different classes: cases where there are no collision hazards (where we obtain bang-bang solutions) or cases with collision risk (where we can obtain low-thrust solutions).

3.4.1 Cases converging to bang-bang solutions

When there are no collision hazards with a rectilinear trajectory, as we pointed out, we know that the optimum trajectory for each spacecraft is a bang-bang control. This is a critical case for the methodology, since after a first iteration of FEFF-DV or FEFF-A, all the delta-v in the interior nodes of the mesh must be zero, which is the case that gives ill-conditioned problems.

Our objective is to check if FEFF-DV and FEFF-A converge to the bang-bang control, and which are the differences between these methods.

In this kind of problems, as there are no collision hazards, the trajectory for each spacecraft is independent from the others. For this reason, we can reduce the computations to obtain the optimal trajectory for a single spacecraft.

In order to exemplify the procedure, we consider a single spacecraft. The reference frame for the equations (2.5) 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, this is when it crosses the RTBP plane $Y = 0$ with $Z > 0$). The initial condition for this example is taken 100 meters far 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 hours. This is to 100 meters in the opposite X direction doing a parallel shift of 200 meters for a formation (see figure 3.7).

For this particular case we obtain as a solution a bang-bang control with maneuvers of 0.69 cm/s at departure and arrival.

Both procedures must deal with the same problem: the initial seed for all of them is obtained minimizing the functional (3.2). The optimal trajectory for this case is not a bang-bang solution (see figure 3.8).

We use this initial seed, with a mesh of 20 elements of the same length, to exemplify the different methodologies.

Solution obtained by the procedure FEFF-DV

The procedure FEFF-DV2, which uses functional (3.2) gives us a trajectory with the delta-v profile of figure 3.8. This is a typical profile in reconfigurations without collision risk.

We note that this initial seed has only a maneuver near zero (the one at time 4 hours). 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 (3.1) and we obtain the delta-v profile displayed in the first row of figure 3.9 which is already rather close to the objective. In the following four iterations all the interior nodes are suppressed by procedure III delivering this way the bang-bang control.

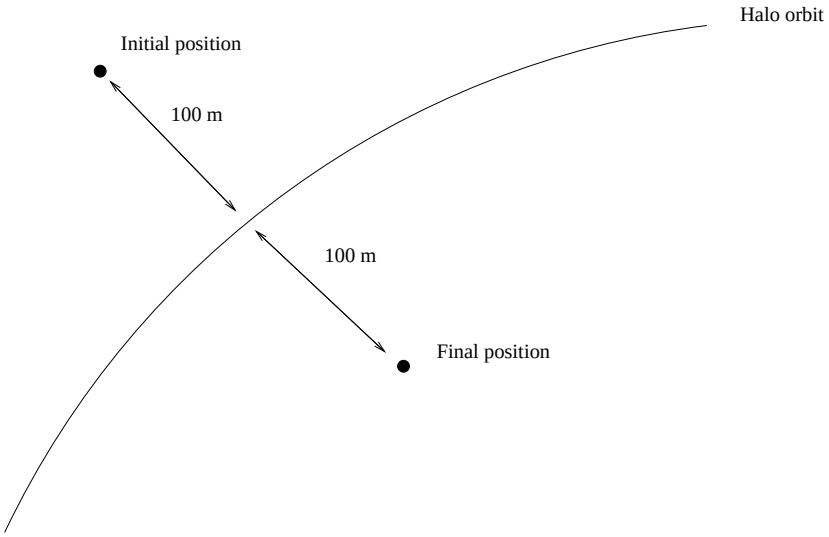

Figure 3.7: Reconfiguration of a "formation" with a single spacecraft to exemplify the reconfiguration of formations with no collision risk: the spacecraft starts in a position 100 m far from the Halo orbit and ends up in the symmetrical position with respect to the orbit (parallel transfer).

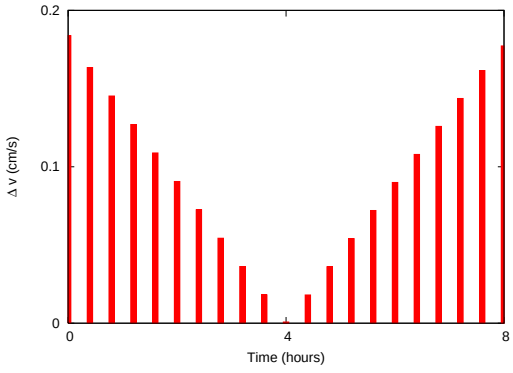

Figure 3.8: Delta-v obtained with the minimization of the delta-v square of functional (3.2) in the case of no collision risk. The optimal trajectory is not a bang-bang trajectory however the methodologies FEFF-DV and FEFF-A must converge to a bang-bang solution with this initial seed.

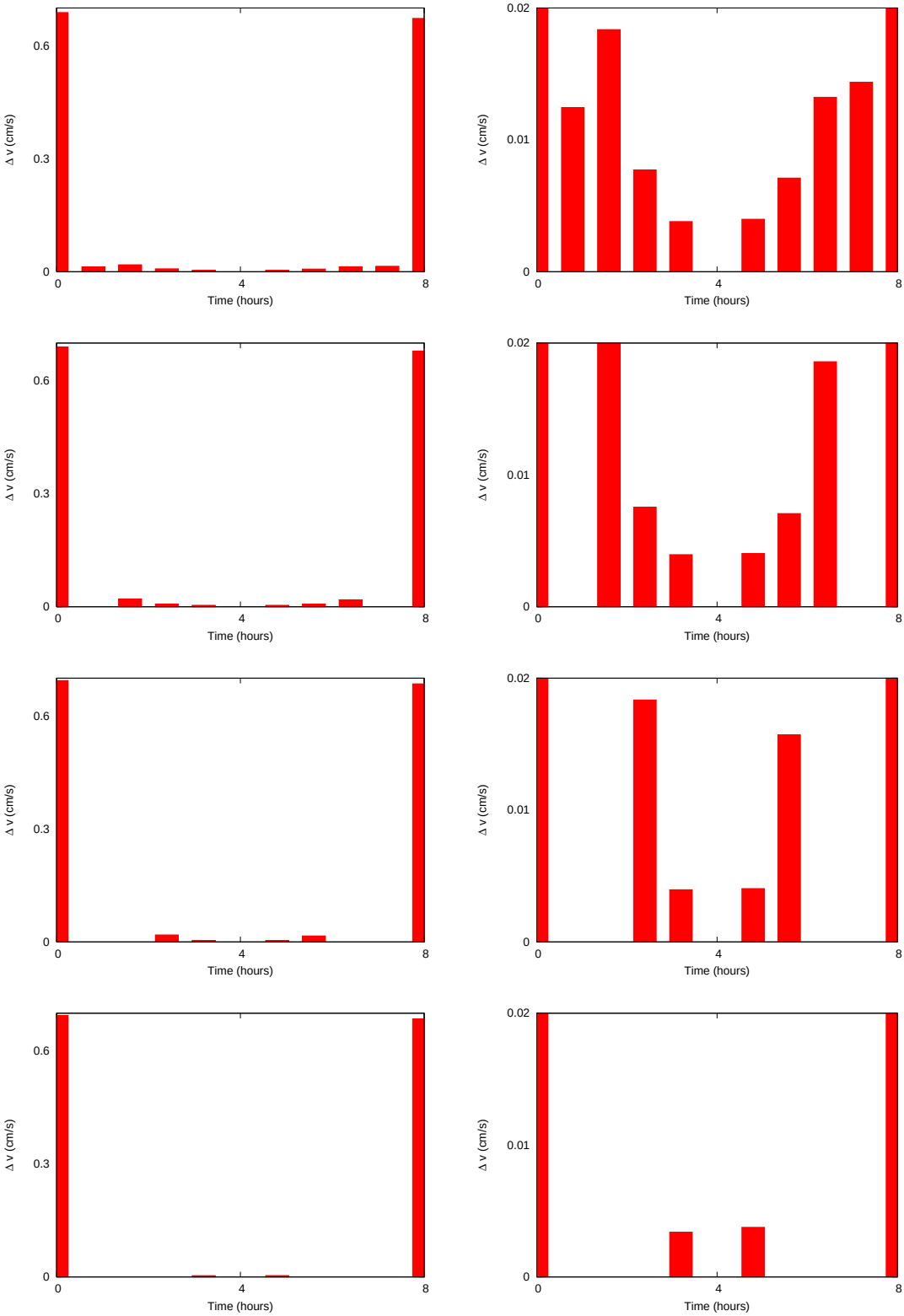

Figure 3.9: Profile of the delta-v obtained in the first four iterations of FEFF-DV in the case of 10 elements. On the left-hand plots, there are the profiles for all of the nodes, where we note that the profile is very close to a bang-bang control. On the right-hand plots, there is a zoom of the delta-v for the interior nodes. We note that on each iteration, two nodes are suppressed by procedure III.

Elements	10-15	16-19	20-26	27-34	35-42
Iterations	3	4	5	6	7
Elements	43-52	53-61	62-74	75-88	89-110
Iterations	8	9	10	11	12

Table 3.1: Number of iterations necessary to obtain the bang-bang solution depending on the number of nodes of the initial mesh (with elements of the same length), using $V_m = 10^{-3}$. Case example of parallel shift.

V_m	10^{-4}	5×10^{-3}	2×10^{-3}	10^{-3}	5×10^{-2}	2×10^{-2}	10^{-1}
It. (10 elem)	9	7	5	3	3	2	2
It. (20 elem)	18	12	8	5	5	4	2
It. (50 elem)	28	19	13	9	6	5	3

Table 3.2: Number of iterations necessary to obtain the bang-bang solution depending on the threshold V_m, using an initial mesh of 10, 20 and 50 elements. Case of parallel shift.

We have to say that the results obtained with this simple example are the same ones that we obtain using other examples ending in bang bang controls: at first iteration, procedure FEFF-DV suppress one node. Then, on each iteration, some nodes are removed via procedures I or III. After a few iterations, we obtain the bang-bang trajectory. In table 3.1 we present the number of iterations required to obtain the bang-bang solution for this problem, considering different initial meshes.

We can also study how the different thresholds affect to the number of iterations we need to obtain the optimal trajectory. In table 3.2 we show the number of iterations required to obtain the bang-bang solution with an initial mesh of 20 elements, for different V_m. As expected, we can see that when V_m is bigger, the number of iterations is lower, since there are more nodes suppressed by procedure I.

Solution using FEFF-A

Again we consider the problem we have treated using FEFF-DV: the transfer of a spacecraft from a position 100 m far from a Halo orbit to the symmetrical position with respect to the Halo orbit in 8 hours (see figure 3.7).

We know that the optimal solution is a bang-bang trajectory, with an initial and final delta-v of 0.69 cm/s, and we start with the solution that we obtain with procedure FEFF-DV2, which uses the penalty function (3.2),

ν	Elements it. 1	Iterations
0.0001	3008	Fail
0.001	301	Fail
0.002	149	25
0.005	61	16
0.01	31	14
0.02	15	10
0.03	11	6
0.04	7	4
0.05	6	4
0.06	4	2
0.07	4	Fail

Table 3.3: Number of iterations necessary to obtain the bang-bang solution depending on ν, using an initial mesh of 20 elements and the Li and Bettess remeshing strategy. We have indicated by "fail" the cases where the procedure does not converge.

and is a trajectory with the delta-v profile of figure 3.8.

Since we are minimizing the same functional, and we must deal with the same ill-conditioned problems, we expect to obtain similar results to the previous case. The difference is now the remeshing procedure.

We have some different parameters and criteria to obtain a new mesh in our iterative procedure. In the first scenario, we use the Li and Bettess remeshing strategy and consider different values for the parameter ν. We note that this parameter ν does not only appear in the acceptability criteria. It is also used to obtain the new mesh in the procedure of Li and Bettess. If we take a small value of ν, we can end up with a mesh with more nodes than wanted. In the other way around, if we use a big ν, we could end up accepting some meshes with big errors. In table 3.3, we have a summary of the results obtained with different values of parameter ν, the number of iterations needed to reach the bang-bang solution and the number of elements after the first iteration of the methodology.

We note that when ν is very small, there is no convergence. In this case, the number of nodes does not converge to 2. When ν is big, there is also no convergence: the final mesh contains more elements than expected, because it passes the acceptability criteria before converging to the bang-bang control.

The best values for ν are inside the range $[0.04, 0.06]$. With values larger than 0.06, the algorithm does not converge.

From now on we take $\nu = 0.05$ and study which one of the remeshing strategies converge faster. In table 3.4, we present the number of iterations

N	10	20	30	40	50	60	70	80	90	100
LB	3	4	6	10	17	21	32	41	54	63
ZZ	3	4	6	11	18	23	31	44	56	66

Table 3.4: Number of iterations necessary to obtain the bang-bang solution with Li and Bettess (LB) and Zienkiewicz Zhu (ZZ) depending on the initial number of elements.

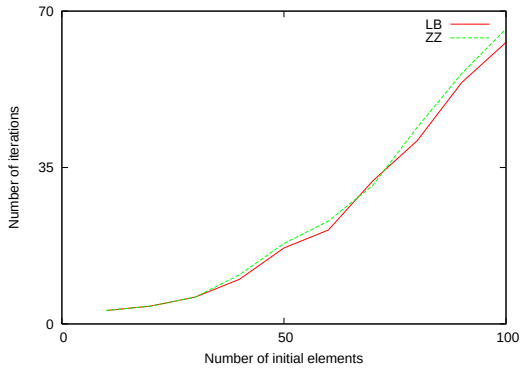

Figure 3.10: Number of iterations necessary to obtain the bang-bang solution depending on the initial number of elements, using the Li and Bettess (LB) and the Zienkiewicz Zhu (ZZ) strategies.

to obtain the bang-bang solution, depending on the initial number of elements and the strategy. We can see than the Li and Bettess strategy and Zienkiewicz and Zhu are very similar in the number of iterations, but Li and Bettess seems better. In both cases (see figure 3.10) the number of iterations to reach the bang-bang trajectory seems exponential depending on the initial number of elements.

In conclusion, we have seen than with a problem without collision risk, the adaptive remeshing converges to the bang-bang solution. The value of ν must be in the range $[0.4, 0.6]$, and the best remeshing strategy is Li and Bettess.

3.4.2 Cases converging to low-thrust solutions

In reconfigurations where bang-bang trajectories for all spacecraft end up with collisions, FEFF obtains trajectories which of course are different from bang-bang. The objective of this section is to study whether these trajectories could tend to low thrust arcs.

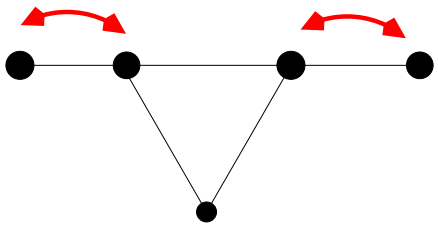

Figure 3.11: Example of reconfiguration with collision risk: the switch of two pairs of spacecraft of the TPF formation.

We start applying the procedures FEFF-DV and FEFF-A to a particular reconfiguration, to see the general behavior of the methodologies.

For this case we assume that the satellites are initially contained in the local plane $Z = 0$, with the interferometry baseline aligned on the X axis. 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 (this is inner satellites are maneuvered to attain outer positions and vice-versa as shown in figure 3.11). Again we consider 8 hours for the reconfiguration. The process of switching positions has a collision risk and simple bang-bang controls are no longer valid.

Solution by means of procedure FEFF-DV

We start applying the methodology FEFF-DV using an initial mesh of 10 elements. The delta-v profile after a first iteration is the one plotted in figure 3.12.

Now, procedures I and III do not suppress any nodes, and procedure II does not add more nodes either. So, the methodology FEFF-DV gives this trajectory as the optimal one.

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.

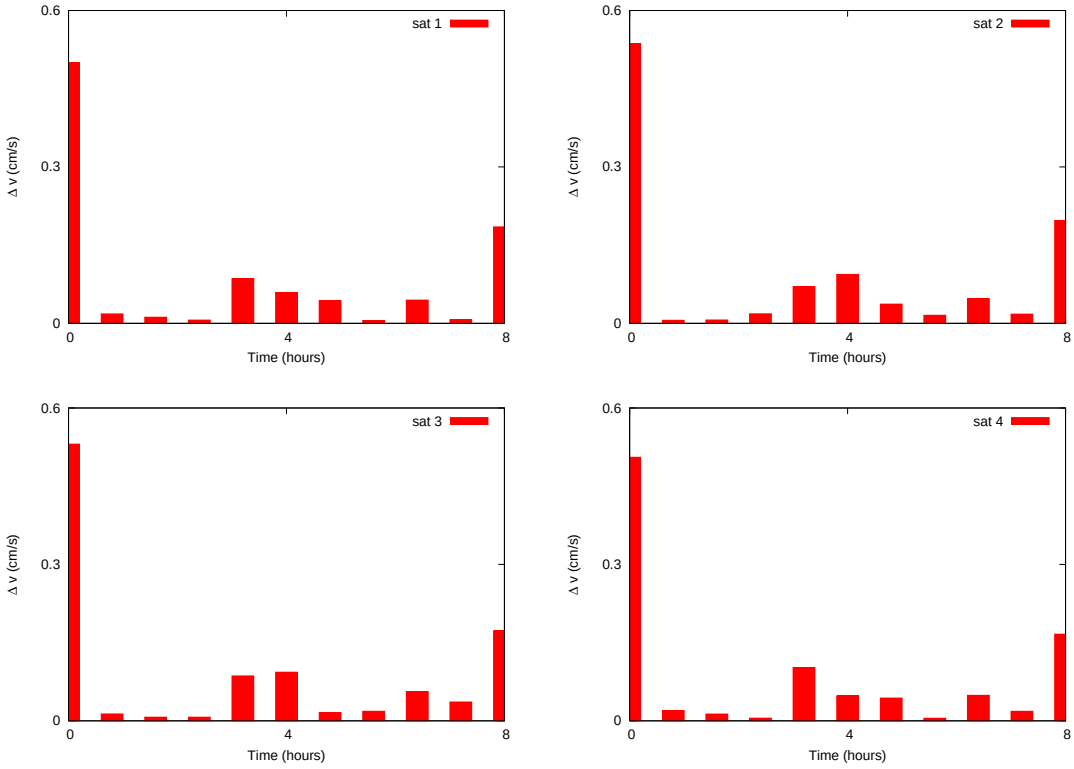

Figure 3.12: Delta-v (in cm/s) of the switch between inner and outer satellites of the TPF formation in 8 hours. FEFF-DV has been used starting with 10 elements. In each row, there are the delta-v profiles of a pair of inner-outer spacecraft.

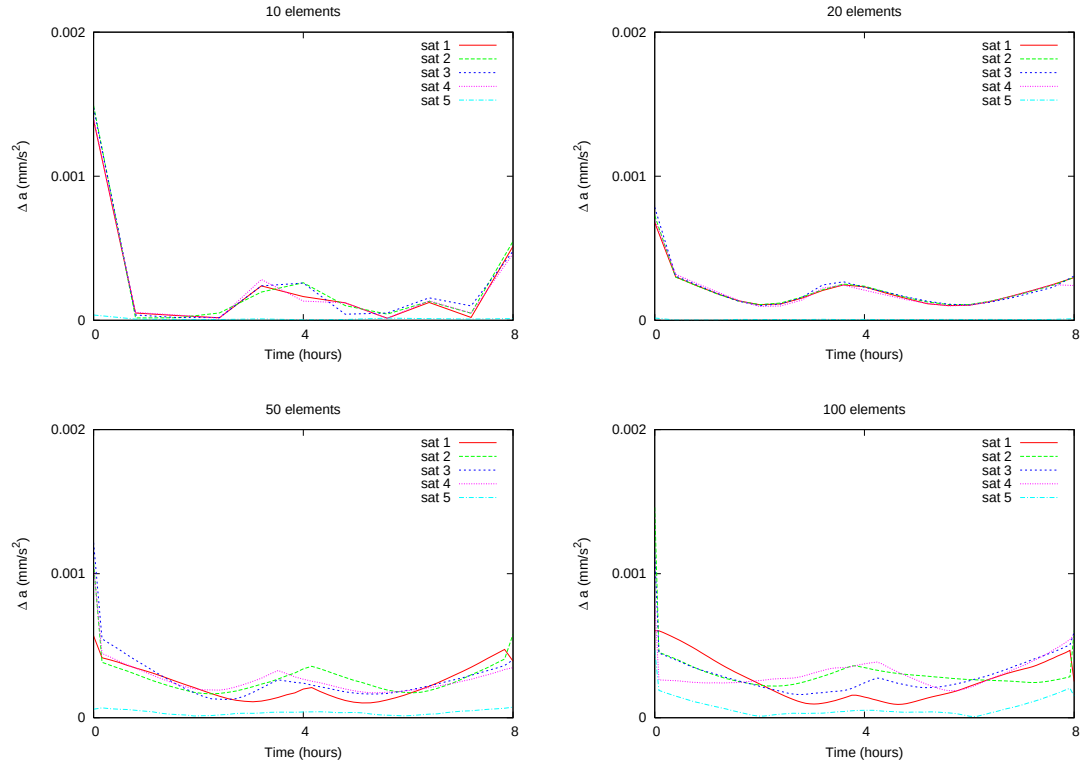

Figure 3.13: Delta-v divided by the element length for the case example II, using 10, 20, 50 and 100 elements. The results tend to a low-thrust profile.

In this example, we do the computations with 10, 20, 50 and 100 elements (also it is much more efficient in term of CPU time to increase gradually the number of elements than computing at once the low thrust control solution with a big number of them). In figure 3.13 we plot the delta-v divided by the element length (acceleration thrust) to see the convergence towards the low-thrust optimal profile. Additionally, the accumulated delta-v is shown in figure 3.14. We note that a pair of spacecraft involved in a switch do not have the same delta-v profile, but at the end, the amount of delta-v is similar. In figure 3.14 we also plot the trajectories for the spacecraft using 100 elements. We observe that switchings are performed each one in a different plane. Due to the symmetry of the problem, the switching plane changes with small changes in the initial position since in an ideal situation all the planes have associated the same cost.

Solution by means of procedure FEFF-A

We consider again the problem of switching two pairs of spacecraft of the

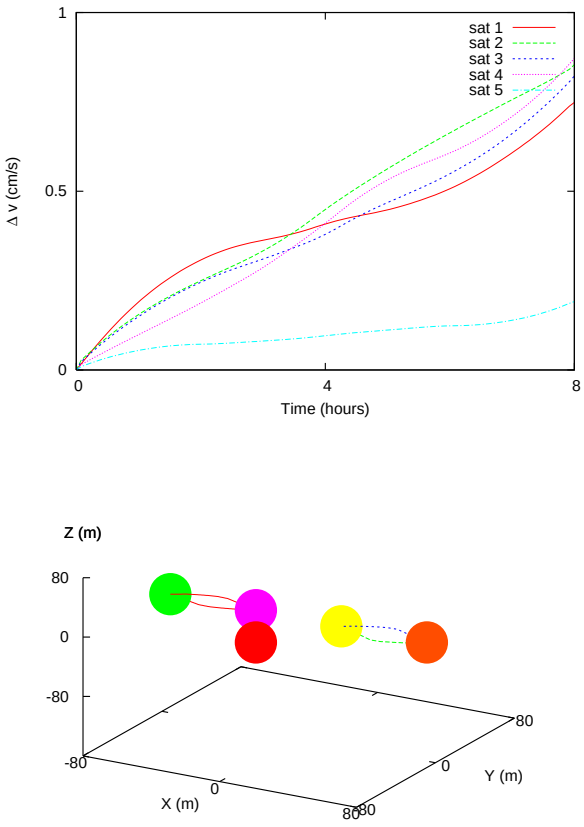

Figure 3.14: In the left-hand plot, delta-v accumulated (cm/s) for each satellite in the TPF reconfiguration of example II. Computations have been done using 100 elements for each trajectory. In the right-hand plot, the trajectories for the spacecraft.

TPF formation, but now applying the adaptive remeshing methodology. As in the case of the bang-bang solution, we want to study which parameter ν is better and the best remeshing strategy.

As in the previous case, we start with the Li and Bettess remeshing procedure and we consider different values for the parameter ν. Similar reasoning as in the previous section is valid here: using a small ν, we can end up with a mesh with many elements. For example, taking $\nu = 0.0005$, in the first iteration we have around 1000 elements. We do not only have the problem of having very small elements; a minimization problem with 1000 elements may not be not desirable. And again, if we take a big ν, we can end up with a mesh with big errors, or a mesh with only a few elements.

In table 3.5, we display a summary of the results obtained for different values of the parameter ν, the number of iterations until the methodology converges (Iter), the number of elements in the first iteration (N_1) and the number of elements in the last iterate (N_F).

As in the previous case, when ν is small, the number of elements is big, and the computation of the optimum is very expensive. Also, taking ν big, the number of elements may not be enough.

We note that the best values are in the range $[0.005, 0.05]$. With values larger than 0.05, the number of elements is very small and with values smaller than 0.005 the number of elements makes the computation much more expensive.

Since the value $\nu = 0.05$ is appropriated for the two cases, we use this ν for our computations.

So, from now on we fix $\nu = 0.05$ and study which of the remeshing strategies converge faster. In table 3.6 we present the number of iterations to obtain the final solution, depending on the initial number of nodes and the strategy. In this case, the two strategies give the same number of iterations.

In conclusion, we have seen than with a problem with collision risk, the adaptive remeshing converges to a solution, depending on the value of ν. Essentially, this value must be in the range $[0.005, 0.05]$, and the remeshing strategies of Li and Bettess and Zienkiewicz Zhu give similar results.

Considerations about the value of ν

With the example of the previous section, we have estimated that the value of ν must be in the range $[0.005, 0.05]$. This range gives us an idea of the value of ν we must choose.

We have applied the procedure to a test bench of 25 reconfigurations which include switches between spacecraft located at opposite vertices of polygons (6), switches in the TPF formation (9) and parallel shifts (10) of different size with a number of spacecraft from 3 to 10. 10 of the reconfigu-

ν	N_1	Iter	N_F
0.0001	3504	Fail	
0.001	350	10	232
0.002	175	8	202
0.005	70	8	171
0.01	34	7	89
0.02	18	6	45
0.03	12	4	33
0.04	9	3	27
0.05	6	3	15
0.06	6	3	9
0.07	5	2	7

Table 3.5: Number of iterations and elements obtained with the switching example of TPF, depending on ν. E_1 are the number of elements at first iteration, and E_F the number of elements in the last iteration.

N	10	20	30	40	50	60	70	80	90	100
LB	3	3	4	4	5	5	5	5	5	6
ZZ	3	3	4	4	5	5	5	5	5	6

Table 3.6: Number of iterations necessary to obtain the switching trajectory depending on the initial number of elements and the remeshing strategy (with $\nu = 0.05$).

ν	0.005	0.01	0.02	0.03	0.04	0.05	0.055	0.06
Iterations	10.2	8.4	7.1	4.2	3.7	3.2	4.3	5.2

Table 3.7: Mean of the number of iterations as a function of ν for the 25 test bench reconfigurations.

rations are converging to a bang-bang solution, the other 15 reconfigurations are converging to low-thrust. We have applied the methodology using different values of ν and we have computed the mean of the number of iterations of the the adaptive process necessary to converge. The results can be seen in table 3.7.

Chapter 4

Reconfigurations in the RTBP and JPL-ephemeris models

In previous chapters, we have developed a methodology to reconfigure spacecraft formations. This methodology gives us a trajectory for each spacecraft in the linearized model about the nonlinear orbit, and the maneuvers necessary on each node to achieve prescribed states.

Since we work with small formations in size when compared to the Halo orbit, the linearized equations give a good approximation for the nonlinear model. The object of this chapter is to study how the truncated nonlinear terms as well as other perturbations affect to the obtained trajectory, and the corrections that must be done to the computed and executed maneuvers in order to reach the same goal. The study of the influence of these new nonlinearities is done in two steps: the first one, taking into account the full RTBP equations, and the second one using JPL-ephemeris.

4.1 Nonlinear models and delta-v corrections

With the FEFF-DV and FEFF-A methodologies, we obtain some trajectories, characterized by the spacecraft positions and the delta-v that must be applied on each node of the mesh, in order to follow the given paths. Now we are going to take into account also that the spacecraft will not follow accurately the nominal trajectory we have obtained, due to several errors. We will be dealing and considering two kind of errors that need correction:

- **Truncation error**: using the procedures FEFF-DV or FEFF-A, we have obtained the delta-v necessary to follow a nominal trajectory computed by means of linearized equations about a nonlinear orbit. When adding the truncated nonlinear terms, and considering the output of

the procedures, we have a reminder error due to the influence of these terms.

- **Execution error**: this error is due to the accuracy in the execution of maneuvers. The thrusters that produce the control of the spacecraft can introduce deviations with respect to the nominal delta-v. This error can be also considered in two parts, taking into account the direction and the modulus.

A main objective is then to correct both truncation and execution errors. For this purpose, we consider first the RTBP and we study the truncation error produced by the nonlinear part of the RTBP that was previously skipped. In a second step, we use the JPL-ephemeris to study the influence of the truncation error. Finally, we introduce a randomized deviation to the nominal maneuvers, to simulate the execution errors, and to study the influence of both truncation and execution errors at the same time.

4.1.1 Corrective maneuvers inside the elements

Let us assume that we have computed a reconfiguration trajectory using FEFF-DV or FEFF-A methodologies. These methodologies give us nodal positions for each spacecraft and the delta-v that must be applied at these positions using the linearized model.

Using the linear equations of motion about the Halo orbit, and without any execution error assumption, the result of applying the corresponding maneuver on a given node is that the spacecraft reaches the desired state (position and velocity) in the following one. However, we consider now some truncation and execution errors that were not previously taken into account, and the position and velocity we are going to obtain in the following node is no longer the one assigned by FEFF-DV or FEFF-A.

The corrective maneuvers will be computed using a strategy similar to [13]. Our nominal path, the one that the spacecraft must follow, is the trajectory obtained with the finite element methodology. When we use the full RTBP or the JPL models, or when we add the execution error, we obtain a trajectory differing from the nominal path, we call it the true state. In each element, the difference between the nominal path and the true state will be corrected by the addition of some corrective maneuvers (see figure 4.1).

The correction of the trajectory that we consider uses a fixed number of small maneuvers (the *corrective maneuvers*) inside the element Ω^k, $\Delta\hat{v}_k^0, \Delta\hat{v}_k^1, \ldots, \Delta\hat{v}_k^n$, which will be applied at some given times $\hat{t}_0, \hat{t}_1, \ldots, \hat{t}_n$. Eventually, these maneuvers should be applied as soon as possible in order to avoid the inherent exponential grow of errors with respect to time.

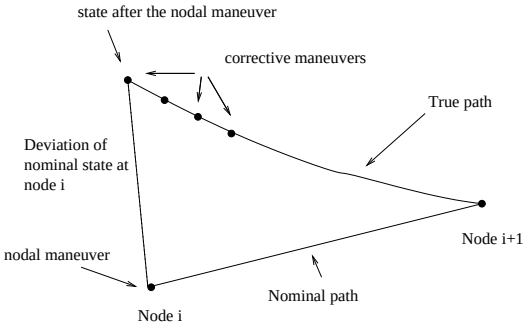

Figure 4.1: Corrective maneuvers inside the element Ω^k (from node k to node $k+1$).

The corrective maneuvers are computed in order to satisfy that the state of spacecraft at node $k+1$ is the one of the nominal path.

The maneuvers $\Delta\hat{\mathbf{v}}_k^j$ are obtained solving an equation that in the case of two maneuvers is,

$$\phi_{(1-\alpha)\Delta t}\left[\phi_{\alpha\Delta t}\left(\mathbf{x}_k+\begin{pmatrix}0\\ \Delta\hat{\mathbf{v}}_k^1\end{pmatrix}\right)+\begin{pmatrix}0\\ \Delta\hat{\mathbf{v}}_k^2\end{pmatrix}\right]=\mathbf{x}_{k+1},$$

where $\mathbf{x}_i$ is the initial state, $\mathbf{x}_{i+1}$ the final state and ϕ_t is the time-t flow of the RTBP. In case of doing more maneuvers, the equation is constructed in a similar way.

The delta-v are also chosen to minimize the functional:

$$\sum_{j=0}^{n_i} 2^{-j}||\Delta\hat{v}_k^j||^2,$$

where the weights 2^{-j} grant someway that the corrective delta-v decay at each step approximately by a factor of two.

4.2 Correction of truncation errors using full RTBP

In previous chapters we considered a Halo orbit, and the linearized equations about this non-linear orbit. Now we are going to consider trajectories in the full RTBP equations for each spacecraft in the formation.

Let us focus on a given reconfiguration problem. Methodologies FEFF-DV and FEFF-A work with a system of coordinates centered in a Halo orbit. In order to apply the usual RTBP, we change to the RTBP coordinates (see

section 2.1.1). The change of coordinates consists essentially on a translation from the Halo orbit to the origin of the synodic system.

After performing computations, we observe that the corrective maneuvers in this case are very small when compared with the nominal maneuvers obtained by the finite element methodologies. In the following tables, we present some results obtained by means of this methodology. For the simulations, we take into account different parameters and indicators:

- Δv_L: the total delta-v (the total amount considering all the maneuvers for all the spacecraft in the formation) obtained using the finite element method. This is the delta-v that we obtained with the methodology of previous chapters and measures the nominal cost of the reconfiguration.
- n: the number of corrective maneuvers we perform in each element.
- $\Delta\hat{v}_{LRmax}$: the maximum of the modulus of the corrective maneuvers.
- $(\Delta\hat{v}/l)_{LRmax}$: the maximum of the modulus of the corrective maneuvers divided by the length of the element.
- $\Delta\hat{v}_{LR}$: the sum of all the corrective maneuvers performed in the RTBP model.
- %: percentage of $\Delta\hat{v}_{LR}$ with respect to $\Delta\hat{v}_L$.

We first compute some examples using reconfigurations that end up in a bang-bang solution. The results are displayed in tables 4.1, 4.2 and 4.3. The different results in the tables are obtained by changing the time span where the corrective maneuvers are performed. In table 4.1, they are done in half the length of the element; in table 4.2, the corrective maneuvers are done in the first third of the time span; and in table 4.3 they are done in the first quarter of the time span. We note that in this case, since the solution of the finite element method only has two nodes, n is the total number of corrective maneuvers in all the reconfiguration process. As in previous chapters, for this kind of scenarios we use a single spacecraft and we transfer it from its current position to the symmetrical one with respect to the Halo orbit (figure 3.7). In the first example, the initial position and the final one are 200 meters apart and the reconfiguration time considered is 8 hours. The total delta-v needed in the linear model is 0.69 cm/s. In the second example, the distance is still 200 meters, but the reconfiguration time is 24 hours. In the third example, the distance is 400 meters and the reconfiguration time is 8 hours.

In tables 4.4, 4.5 and 4.6 we display some results for low thrust examples. We note that in the case of low thrust, the length of the elements is very

Δv_L	n	$\Delta\hat{v}_{LRmax}$	$\Delta\hat{v}_{LR}$	%
0.69	3	2.8×10^{-3}	6.1×10^{-3}	0.88
0.69	4	2.2×10^{-3}	5.8×10^{-3}	0.84
0.69	5	2.2×10^{-3}	5.6×10^{-3}	0.80
0.69	6	1.9×10^{-3}	5.6×10^{-3}	0.80
0.23	3	3.3×10^{-3}	6.9×10^{-3}	3.01
0.23	4	2.5×10^{-3}	6.4×10^{-3}	2.77
0.23	5	2.2×10^{-3}	6.1×10^{-3}	2.65
0.23	6	1.9×10^{-3}	5.8×10^{-3}	2.53
2.8	3	4.7×10^{-3}	9.2×10^{-3}	0.33
2.8	4	3.6×10^{-3}	8.3×10^{-3}	0.30
2.8	5	2.8×10^{-3}	7.5×10^{-3}	0.27
2.8	6	2.5×10^{-3}	7.2×10^{-3}	0.26

Table 4.1: Corrective maneuvers in the RTBP for the bang-bang control scenario (delta-v in cm/s). The first group corresponds to a parallel shift of 200 meters in 8 hours of reconfiguration time. The second one, 200 meters in 24 hours and the third one, 400 meters in 8 hours time. Corrective maneuvers are performed in the first half of the elements.

Δv_L	n	$\Delta\hat{v}_{LRmax}$	$\Delta\hat{v}_{LR}$	%
0.69	3	2.7×10^{-3}	6.3×10^{-3}	0.91
0.69	4	2.4×10^{-3}	5.9×10^{-3}	0.86
0.69	5	2.3×10^{-3}	5.5×10^{-3}	0.80
0.69	6	1.8×10^{-3}	5.4×10^{-3}	0.78
0.23	3	3.5×10^{-3}	6.7×10^{-3}	2.91
0.23	4	2.4×10^{-3}	6.5×10^{-3}	2.83
0.23	5	2.1×10^{-3}	6.2×10^{-3}	2.70
0.23	6	1.9×10^{-3}	5.7×10^{-3}	2.48
2.8	3	4.6×10^{-3}	9.3×10^{-3}	0.32
2.8	4	3.6×10^{-3}	8.4×10^{-3}	0.30
2.8	5	2.9×10^{-3}	7.7×10^{-3}	0.28
2.8	6	2.7×10^{-3}	7.3×10^{-3}	0.26

Table 4.2: Corrective maneuvers in the RTBP for the same bang-bang control scenario as in table 4.1 (delta-v in cm/s). Here corrective maneuvers are performed in the first third of the elements.

Δv_L	n	$\Delta\hat{v}_{LRmax}$	$\Delta\hat{v}_{LR}$	%
0.69	3	2.7×10^{-3}	6.1×10^{-3}	0.88
0.69	4	2.3×10^{-3}	5.9×10^{-3}	0.86
0.69	5	2.1×10^{-3}	5.6×10^{-3}	0.81
0.69	6	1.8×10^{-3}	5.5×10^{-3}	0.80
0.23	3	3.2×10^{-3}	6.8×10^{-3}	2.96
0.23	4	2.7×10^{-3}	6.5×10^{-3}	2.83
0.23	5	2.2×10^{-3}	6.3×10^{-3}	2.74
0.23	6	1.9×10^{-3}	5.8×10^{-3}	2.52
2.8	3	4.9×10^{-3}	9.2×10^{-3}	0.33
2.8	4	3.6×10^{-3}	8.4×10^{-3}	0.30
2.8	5	2.5×10^{-3}	7.6×10^{-3}	0.27
2.8	6	2.4×10^{-3}	7.1×10^{-3}	0.25

Table 4.3: Corrective maneuvers in the RTBP for the bang-bang control scenario as in table 4.1 (delta-v in cm/s). Here corrective maneuvers are performed in the first quarter of the elements.

small, and thus the corrective maneuvers are also very small. In this case we do not display the maximum of these delta-v (that is very small and tends to zero with the size of the element); we display instead the maximum of the delta-v divided by the length of the element (i. e., the acceleration).

As in the case of bang-bang trajectories we also note that the total cost of corrective maneuvers is always of the same order and essentially does not depend on the length of the corrective maneuvers time span within certain ranges. In the following studies we fix this parameter equal to half the length of the element.

4.3 Corrective maneuvers for truncation errors using JPL ephemeris

Once we have computed the delta-v corrections necessary to follow the nominal path in the full RTBP, we apply the same methodology to obtain the corrections in a more complete model: the JPL ephemeris. As a result of the previous section, we have all the positions and delta-v in RTBP synodical coordinates. Now, before performing the computations with the JPL ephemeris, we must change the system of coordinates to the inertial ones of these ephemeris.

The change of coordinates consists on changing time scale and distance.

Δv_L	n	$(\Delta\hat{v}/l)_{LRmax}$	$\Delta\hat{v}_{LR}$	%
0.63	3	6.5×10^{-3}	6.9×10^{-3}	1.10
0.63	4	6.2×10^{-3}	6.7×10^{-3}	1.06
0.63	5	4.7×10^{-3}	6.7×10^{-3}	1.06
0.63	6	4.3×10^{-3}	6.1×10^{-3}	0.97
2.34	3	9.6×10^{-3}	1.19×10^{-2}	0.51
2.34	4	4.2×10^{-3}	1.03×10^{-2}	0.44
2.34	5	3.3×10^{-3}	9.7×10^{-3}	0.42
2.34	6	3.0×10^{-3}	8.6×10^{-3}	0.37
1.26	3	6.7×10^{-3}	9.7×10^{-3}	0.77
1.26	4	5.3×10^{-3}	9.2×10^{-3}	0.73
1.26	5	4.9×10^{-3}	8.3×10^{-3}	0.66
1.26	6	3.1×10^{-3}	7.5×10^{-3}	0.60

Table 4.4: Corrective maneuvers in the RTBP for the low thrust scenarios (delta-v in cm/s). Corrective maneuvers are performed in the first half of the element.

Δv_L	n	$(\Delta\hat{v}/l)_{LRmax}$	$\Delta\hat{v}_{LR}$	%
0.63	3	6.3×10^{-3}	6.9×10^{-3}	1.10
0.63	4	6.2×10^{-3}	6.8×10^{-3}	1.08
0.63	5	4.8×10^{-3}	6.7×10^{-3}	1.06
0.63	6	4.1×10^{-3}	6.2×10^{-3}	0.98
2.34	3	9.5×10^{-3}	1.21×10^{-2}	0.52
2.34	4	4.2×10^{-3}	1.01×10^{-2}	0.43
2.34	5	3.2×10^{-3}	9.5×10^{-3}	0.41
2.34	6	2.9×10^{-3}	8.6×10^{-3}	0.37
1.26	3	6.5×10^{-3}	9.8×10^{-3}	0.78
1.26	4	5.4×10^{-3}	9.1×10^{-3}	0.72
1.26	5	5.0×10^{-3}	8.5×10^{-3}	0.67
1.26	6	3.1×10^{-3}	7.2×10^{-3}	0.57

Table 4.5: Corrective maneuvers in the RTBP for the low thrust scenarios as in table 4.4 (delta-v in cm/s). Here corrective maneuvers are performed in the first third of the element.

Δv_L	n	$(\Delta\hat{v}/l)_{LRmax}$	$\Delta\hat{v}_{LR}$	%
0.63	3	6.5×10^{-3}	7.1×10^{-3}	1.13
0.63	4	6.4×10^{-3}	6.9×10^{-3}	1.10
0.63	5	5.2×10^{-3}	6.8×10^{-3}	1.08
0.63	6	4.6×10^{-3}	6.3×10^{-3}	1.00
2.34	3	9.4×10^{-3}	1.23×10^{-2}	0.53
2.34	4	4.0×10^{-3}	1.11×10^{-2}	0.47
2.34	5	3.5×10^{-3}	9.8×10^{-3}	0.42
2.34	6	3.2×10^{-3}	8.7×10^{-3}	0.37
1.26	3	6.9×10^{-3}	9.9×10^{-3}	0.79
1.26	4	5.6×10^{-3}	9.3×10^{-3}	0.74
1.26	5	4.9×10^{-3}	8.5×10^{-3}	0.67
1.26	6	3.4×10^{-3}	7.7×10^{-3}	0.61

Table 4.6: Corrective maneuvers in the RTBP for the low thrust scenarios as in table 4.4 (delta-v in cm/s). Here corrective maneuvers are performed in the first quarter of the element.

In the RTBP, the period of the primaries is equal to 2π, and corresponds to the sideral period of the chosen primaries. There is also the scaling of the distance: in the RTBP, the distance between primaries is equal to 1, and now it must be the physical distance in kilometers.

This change of coordinates can be implemented by means of the transformation defined by (see [12])

$$\mathbf{R} = \mathbf{B} + kC\mathbf{r},$$

where $\mathbf{r}$ are RTBP synodical coordinates and $\mathbf{R}$ the JPL ones.

In this change of coordinates, $\mathbf{B}$ is a translation vector to put the centroid of the primaries at the origin in the RTBP reference frame. This vector can be obtained by means of

$$\mathbf{B} = \frac{m_1\mathbf{R}_1 + m_2\mathbf{R}_2}{m_1 + m_2},$$

where m_i are the masses of the primaries.

C is an orthogonal matrix that performs a rotation to keep the primaries on the x-axis, and k is a scaling factor to make the distance between primaries equal to 1.

We use the model given by the JPL ephemeris (DE403) to obtain the trajectories of the spacecraft. Since we have done all the computations of

FEFF-DV and FEFF-A considering the linearized equations, we must correct for the truncation error. This error can be corrected in two different ways:

- Correcting directly from FEFF-DV or FEFF-A results to JPL ephemeris.
- First making the corrections in the RTBP model, and then correcting again to obtain the trajectory in JPL ephemeris.

As in the previous section, we present some tables of results obtained with these procedures. First considering bang-bang scenarios and secondly in low-thrust scenarios. In all the tables, added to the previous items, we consider also the following ones:

- $\Delta\hat{v}_{LJmax}$: the maximum of the modulus of the corrective maneuvers applied, when correcting directly in the JPL model from the solution of the linearized equations.
- $(\Delta\hat{v}/l)_{LJmax}$: the maximum of the modulus of the corrective maneuvers applied divided by the length of the element where we are applying the corrective maneuver, when correcting directly in the JPL model.
- $\Delta\hat{v}_{LJ}$: the total amount of the corrective maneuvers, when correcting directly in the JPL model.
- $\Delta\hat{v}_{RJmax}$: the maximum of the modulus of the corrective maneuvers, when correcting from RTBP equations of motion to JPL ephemeris.
- $(\Delta\hat{v}/l)_{RJmax}$: the maximum of the modulus of the corrective maneuvers divided by the length of the element, when correcting from RTBP equations of motion to JPL ephemeris.
- $\Delta\hat{v}_{RJ}$: the total amount of the corrective maneuvers, when correcting from RTBP to JPL ephemeris.

The test examples considered are the same ones of the previous section. In the case of bang-bang scenarios, the results for the corrective maneuvers obtained when correcting directly with the JPL ephemeris are presented in table 4.7. We see that these corrections are slightly bigger than the ones obtained for the RTBP case although the order of magnitude is the same.

In table 4.8 we present the corrective maneuvers using the two ways considered: the first delta-v correspond to the corrective maneuvers obtained when applying the corrections directly to the JPL ephemeris. The second ones correspond to the case of the correction using first the RTBP and

Δv_L	n	$\Delta\hat{v}_{LJmax}$	$\Delta\hat{v}_{LJ}$	%
0.69	3	3.3×10^{-3}	6.7×10^{-3}	0.96
0.69	4	2.5×10^{-3}	6.1×10^{-3}	0.88
0.69	5	2.2×10^{-3}	5.8×10^{-3}	0.84
0.69	6	2.2×10^{-3}	5.6×10^{-3}	0.80
0.23	3	3.6×10^{-3}	7.5×10^{-3}	3.25
0.23	4	2.7×10^{-3}	6.7×10^{-3}	2.89
0.23	5	2.2×10^{-3}	6.4×10^{-3}	2.77
0.23	6	2.2×10^{-3}	6.1×10^{-3}	2.65
2.8	3	5.3×10^{-3}	9.7×10^{-3}	0.35
2.8	4	3.9×10^{-3}	9.2×10^{-3}	0.33
2.8	5	3.1×10^{-3}	8.1×10^{-3}	0.29
2.8	6	2.7×10^{-3}	7.5×10^{-3}	0.27

Table 4.7: Corrective maneuvers in JPL ephemeris for the bang-bang control scenarios considered in table 4.1

then correcting the delta-v obtained with the RTBP model using the JPL ephemeris. We can see that the corrections in this second case are slightly bigger than when doing the JPL correction directly, although again the order of magnitude is essentially the same.

We have performed similar studies for low-thrust cases, and again we encounter with the same result: performing directly the JPL corrections (table 4.9) we obtain somewhat better results than making first a RTBP correction and then the JPL correction (table 4.10).

4.4 Corrective maneuvers for the execution error

Finally, we consider errors produced by the thrusters. We introduce them inside the maneuvers, scaling by a factor which depends on the percentage of the error p, and a random variable η, which follows a normal distribution $N(0, 1)$. The nodal values of delta-v we apply to the spacecraft are then,

$$\Delta\bar{v} = \Delta v(1 + \eta p),$$

where Δv are the nominal maneuvers obtained by FEFF-DV or FEFF-A.

In table 4.11, we present some results for a low thrust example, changing the error of the thrust. For each value of the parameters (p and n), we calculate the mean for $\Delta\hat{v}_{LRmax}$, $\Delta\hat{v}_E$ and % in $\Delta\hat{v}_E$ with 500 simulations.

Δv_L	n	$\Delta\hat{v}_{LRmax}$	$\Delta\hat{v}_{LR}$	$\Delta\hat{v}_{RJmax}$	$\Delta\hat{v}_{RJ}$	%
0.69	3	2.8×10^{-3}	6.1×10^{-3}	2.6×10^{-4}	8.6×10^{-4}	1.00
0.69	4	2.2×10^{-3}	5.8×10^{-3}	2.2×10^{-4}	6.7×10^{-4}	0.92
0.69	5	2.2×10^{-3}	5.6×10^{-3}	2.0×10^{-4}	6.1×10^{-4}	0.92
0.69	6	1.9×10^{-3}	5.6×10^{-3}	1.9×10^{-4}	3.5×10^{-4}	0.84
0.23	3	3.3×10^{-3}	6.9×10^{-3}	3.2×10^{-4}	1.4×10^{-3}	3.61
0.23	4	2.5×10^{-3}	6.4×10^{-3}	2.7×10^{-4}	1.1×10^{-3}	3.25
0.23	5	2.2×10^{-3}	6.1×10^{-3}	2.4×10^{-4}	8.7×10^{-4}	3.01
0.23	6	1.9×10^{-3}	5.8×10^{-3}	2.2×10^{-4}	8.3×10^{-4}	2.89
2.8	3	4.7×10^{-3}	9.2×10^{-3}	3.5×10^{-4}	1.7×10^{-4}	0.39
2.8	4	3.6×10^{-3}	8.3×10^{-3}	3.2×10^{-4}	1.1×10^{-3}	0.34
2.8	5	2.8×10^{-3}	7.5×10^{-3}	2.5×10^{-4}	8.2×10^{-4}	0.30
2.8	6	2.5×10^{-3}	7.2×10^{-3}	2.4×10^{-4}	6.7×10^{-4}	0.28

Table 4.8: Corrective maneuvers in RTBP and in JPL ephemeris for bang-bang control scenarios considered in table 4.1

Δv_L	n	$(\Delta\hat{v}/l)_{LJmax}$	$\Delta\hat{v}_{LJ}$	%
0.63	3	9.1×10^{-3}	7.5×10^{-3}	1.19
0.63	4	8.5×10^{-3}	6.9×10^{-3}	1.10
0.63	5	7.7×10^{-3}	6.7×10^{-3}	1.06
0.63	6	5.9×10^{-3}	6.4×10^{-3}	1.01
2.34	3	9.9×10^{-3}	1.3×10^{-2}	0.51
2.34	4	8.2×10^{-3}	1.1×10^{-2}	0.44
2.34	5	6.4×10^{-3}	1.0×10^{-2}	0.42
2.34	6	2.6×10^{-3}	9.2×10^{-3}	0.37
1.26	3	8.3×10^{-3}	1.0×10^{-2}	0.77
1.26	4	8.0×10^{-3}	9.4×10^{-3}	0.73
1.26	5	5.9×10^{-3}	8.9×10^{-3}	0.66
1.26	6	5.5×10^{-3}	8.1×10^{-3}	0.60

Table 4.9: Corrective maneuvers in the JPL model for low thrust scenarios considered in table 4.4

Δv_L	n	$(\Delta\hat{v}/l)_{LRmax}$	$\Delta\hat{v}_{LR}$	$(\Delta\hat{v}/l)_{RJmax}$	$\Delta\hat{v}_{RJ}$	%
0.63	3	6.5×10^{-3}	6.9×10^{-3}	2.1×10^{-3}	5.6×10^{-4}	1.10
0.63	4	6.2×10^{-3}	6.7×10^{-3}	1.3×10^{-3}	4.7×10^{-4}	1.06
0.63	5	4.7×10^{-3}	6.7×10^{-3}	1.1×10^{-3}	4.2×10^{-4}	1.06
0.63	6	4.3×10^{-3}	6.1×10^{-3}	1.0×10^{-3}	3.7×10^{-4}	0.97
2.34	3	9.6×10^{-3}	1.19×10^{-2}	1.7×10^{-3}	6.5×10^{-4}	0.51
2.34	4	4.2×10^{-3}	1.03×10^{-2}	1.5×10^{-3}	5.7×10^{-4}	0.44
2.34	5	3.3×10^{-3}	9.7×10^{-3}	1.3×10^{-3}	5.1×10^{-4}	0.42
2.34	6	3.0×10^{-3}	8.6×10^{-3}	1.1×10^{-3}	4.4×10^{-4}	0.37
1.26	3	6.7×10^{-3}	9.7×10^{-3}	2.6×10^{-3}	6.5×10^{-4}	0.77
1.26	4	5.3×10^{-3}	9.2×10^{-3}	2.3×10^{-3}	6.2×10^{-4}	0.73
1.26	5	4.9×10^{-3}	8.3×10^{-3}	1.9×10^{-3}	6.0×10^{-4}	0.66
1.26	6	3.1×10^{-3}	7.5×10^{-3}	1.5×10^{-3}	5.7×10^{-4}	0.60

Table 4.10: Corrective maneuvers in the RTBP and then in the JPL ephemeris for low thrust scenarios in table 4.4

We have not continued after 20% of execution error, because the corrections would be considered very big and the accuracy of the thruster is too poor. However, we have performed computations up to this big execution error to prove the robustness of the methodology.

In figure 4.2 we present the result of a simulation using 25 different examples of reconfigurations, making 500 simulations for each of the examples. We fix $n = 4$, and we study how the percentage of error grows depending on the parameter p. Essentially we find a linear behavior with respect to this variable.

4.5 An application to transfers between Halo orbits

The methodology presented in this work can be applied to many other situations, that we want to study in the future. For instance, it can be applied to the reconfiguration of spacecraft in some other scenarios, not only on free space or about Halo orbits. It particularly may work reconfiguring formations about the Earth. Also, with appropriated changes, it can be useful for other kind of problems not involving spacecraft formations, such as monitoring submarine robots or other kind of vehicles.

The application we consider in this section is the solution of a problem slightly different from the ones we have considered before. The objective is to

Δv_L	% in p	n	$\Delta\hat{v}_{LRmax}$	$\Delta\hat{v}_E$	% in $\Delta\hat{v}_E$
0.63	0	4	8.7×10^{-5}	0.007	1.06
0.63	2	4	2.5×10^{-4}	0.011	1.80
0.63	4	4	3.2×10^{-4}	0.019	3.08
0.63	6	4	6.5×10^{-4}	0.029	4.62
0.63	8	4	9.4×10^{-4}	0.034	5.41
0.63	10	4	1.3×10^{-3}	0.042	6.64
0.63	12	4	2.0×10^{-3}	0.056	8.84
0.63	14	4	2.5×10^{-3}	0.064	10.16
0.63	16	4	2.8×10^{-3}	0.070	11.13
0.63	18	4	3.1×10^{-3}	0.075	11.92
0.63	20	4	3.6×10^{-5}	0.085	13.42

Table 4.11: Corrective maneuvers for execution error in the JPL model.

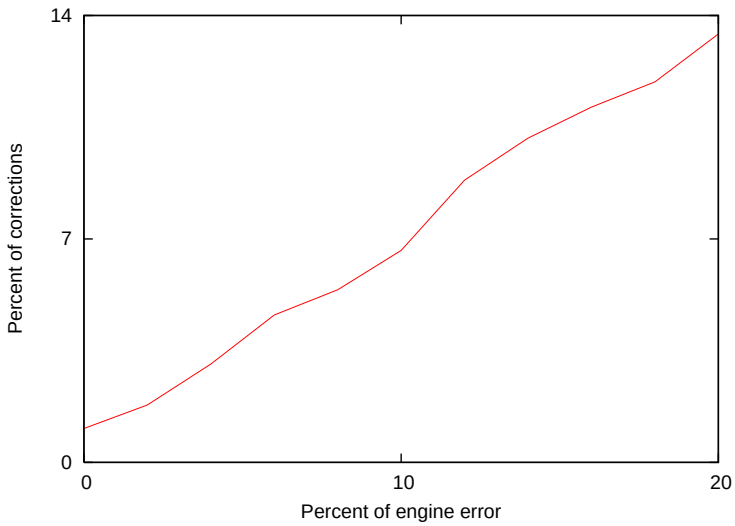

Figure 4.2: Percentage of the corrective maneuvers with respect to the total amount of nominal delta-v of the reconfiguration, as a function of p, using a test bench of 25 reconfigurations, with $n = 4$.

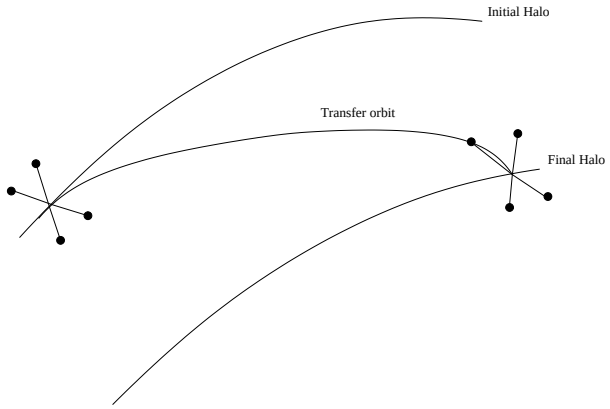

Figure 4.3: Schema of the transfer of a formation of 2 pairs of spacecraft from an initial Halo orbit to a neighboring one.

transfer a formation of spacecraft from a Halo orbit to the neighborhood of another Halo orbit about the same libration point. Under the specifications in project [4], the objective is to transfer the cluster from a initial Halo orbit to a final Halo orbit in 210 days.

The formations considered for this project have an even number of spacecraft, all of them flying in the initial orbit. The only restriction in the position of the spacecraft is that they must have the positions grouped in pairs: for each pair of spacecraft, the position of one of them is free, and the position of the other must be symmetrical to the first one which respect to the Halo orbit (by this we mean that the line joining the satellites intersects the Halo orbit in its middle point). In figure 4.3, we present a sketch of such a formation of four spacecraft symmetrical in pairs with respect to the Halo orbit.

This problem could also be considered using the Cucker-Smale control law [26], but including the collision avoidance. In our case, we use the same methodology we have used in the previous chapters.

We consider the spacecraft subjected to two types of constraints: they must avoid collisions (restriction that we introduce as in all the previous computations) and we also impose that during the reconfiguration time the spacecraft must keep a minimum security distance between them and the transfer orbit (i.e., we also impose that the spacecraft must be confined inside a sphere centered on the transfer orbit).

We note that our reconfiguration methodology FEFF-DV2 finds an optimal trajectory in terms of delta-v square expenditure to perform a reconfiguration with fixed initial and final states and with prescribed transfer time, while FEFF-DV and FEFF-A give us the optimal trajectory in terms of the

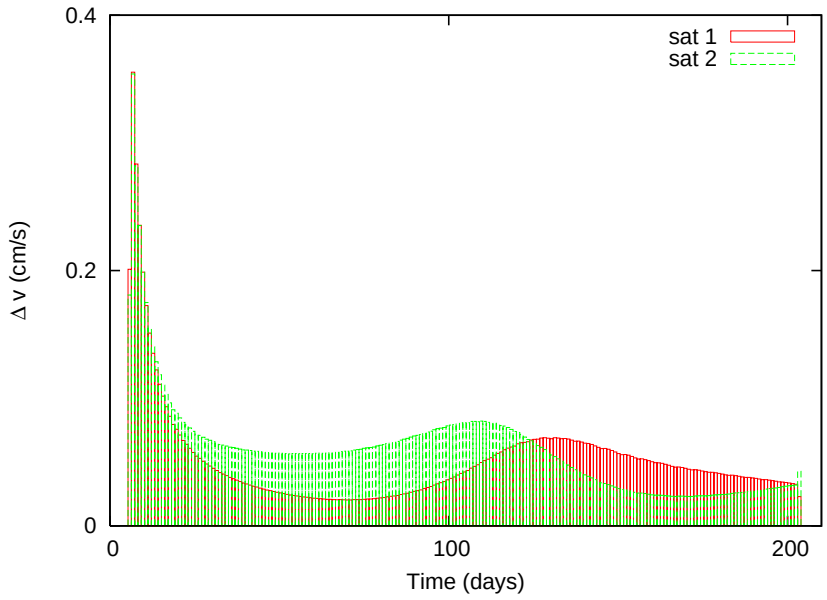

Figure 4.4: Delta-v obtained to make a transfer, using 200 elements and with a random initial and final position.

delta-v. With the transfers we are studying, the transfer time is fixed, but the initial and final positions are free for half of the spacecraft (except for the usual requirements of security distances at the initial and final positions).

Initial and final positions for the spacecraft

Since the initial and final positions for the spacecraft are free, our objective is to find if there are some initial and final positions where the total amount of delta-v is smaller. For this purpose, we consider a formation of two spacecraft (in symmetrical positions with respect to the Halo orbit at initial and final epochs), and we fix an initial (and final) distance of 100 meters between the spacecraft and the transfer orbit, and a sphere of 300 meters centered in the orbit, which is the sphere where all the spacecraft must be confined.

Taking 200 elements and random initial and final positions (but always 100 meters away from the orbit), we obtain profiles like the one on figure 4.4. The mean of total delta-v using 100 different initial and final positions is of the order of 19 cm/s.

As is stated in [13] privileged locations for this problem can be initial and final states with zero radial acceleration in the linearized equations about the Halo orbit.

These relative states are such that with $\dot{\mathbf{r}} = 0$ (relative radial velocity zero) the relative radial acceleration is also zero. They can be obtained using the linear equations about the orbit:

$$\begin{pmatrix} \dot{\mathbf{r}} \\ \ddot{\mathbf{r}} \end{pmatrix} = \left(\begin{array}{c|c} 0 & I \\ \hline F & J \end{array}\right) \begin{pmatrix} \mathbf{r} \\ \dot{\mathbf{r}} \end{pmatrix}, \tag{4.1}$$

where $(\mathbf{r}, \dot{\mathbf{r}})$ are the states and

$$\left(\begin{array}{c|c} 0 & I \\ \hline F & J \end{array}\right) = \left(\begin{array}{ccc|ccc} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ \hline f_x^4 & f_y^4 & f_z^4 & 0 & 2 & 0 \\ f_x^5 & f_y^5 & f_z^5 & -2 & 0 & 0 \\ f_x^6 & f_y^6 & f_z^6 & 0 & 0 & 0 \end{array}\right),$$

where f_k^n is the derivative with respect to k of the n-th component of the vector field.

Taking $\dot{\mathbf{r}} = 0$, we see that the points with zero acceleration are the ones which

$$\ddot{\mathbf{r}} = F\mathbf{r}.$$

Then the radial component of the acceleration will be zero when $\ddot{\mathbf{r}}.\mathbf{r} = 0$, this is, for the points which $\mathbf{r}^T F \mathbf{r} = 0$. The solution of this equation is a quadric, that in a Halo orbit represents a set of cones, as can be seen in figure 4.5.

Let us now consider consider that the initial and final positions are random, but constrained to be in a cone. We compute again the cost of the transfer considering 100 configurations. The mean of the reconfiguration cost obtained is of 13 cm/s, which is better than the result using pure random configurations.

We have checked initial and final positions at random places and in the cones of zero radial acceleration. Computations give the result that the best position in terms of fuel consumption for the initial and final states is to chose them on the cones of the initial and final orbit. In figure 4.6 we can see the delta-v profile of this example, and we see that the profile is similar to the ones of the random position, but the cost is lower.

Once we have confirmed that the best place to put the spacecraft is on the cones of zero radial acceleration, we want to know if some positions of the cones have a smaller fuel consumption than others. We have computed the total amount of maneuvers depending on the initial position of the cone. As we can see in figure 4.7, the initial position inside the cone is not important in terms of the delta-v cost.

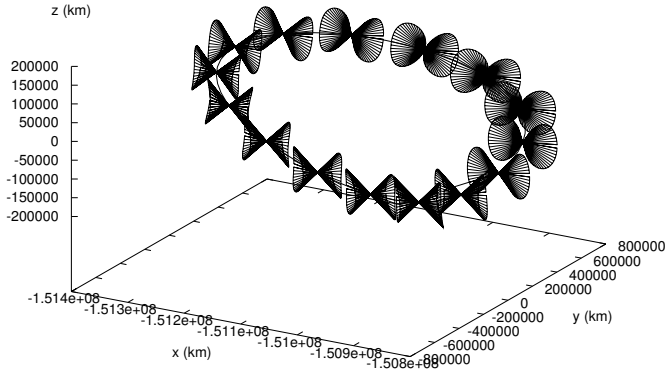

Figure 4.5: Zero relative radial acceleration cones along a Halo orbit.

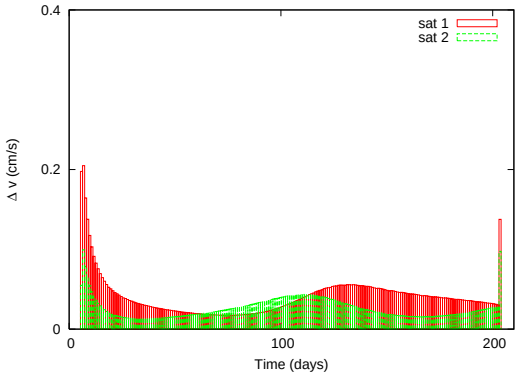

Figure 4.6: Delta-v for the transfer of the formation between Halo orbits (in cm/s) depending on time (days), with the initial and final position located in a cone of zero radial acceleration.

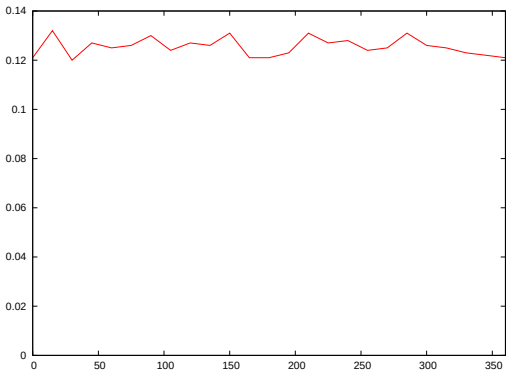

Figure 4.7: Delta-v for the transfer to another Halo orbit, changing the initial position of the spacecraft. The initial position is the angle on the cone and the delta-v are in m/s.

Conclusions and future work

The aim of this work has been to obtain a methodology to perform the reconfigurations of a formation of spacecraft, consisting mainly on changing the relative position of the spacecraft in the formation.

The reconfigurations are considered to be on an Halo orbit about the Lagrangian Sun-Earth+Moon system, or in free space. They are usually small formations (hundreds of meters of diameter), and the reconfiguration time is usually small when compared with the period of the Halo (reconfiguration time ranging from 8 to 24 hours). Due to this fact, the linearized equations of motion about the orbit are a good approximation of the model, and have been the primary focus of interest.

In our methodology, the reconfiguration time span of the formation is divided into elements, suitable to apply the finite element method. The procedure to obtain the trajectories is via an optimal control problem, where the variables of the optimization are the position and velocity of the spacecraft on the nodes, and the functional to be minimized is related to the sum of modulus of the delta-v. Collision avoidance enters in the optimization problem as restrictions.

The procedure to obtain the trajectory in the linearized model is an iterative process. The first step of the process is to obtain a trajectory which minimizes the square of the modulus of delta-v (procedure FEFF-DV2). This functional is necessary to avoid ill-conditioned problems that we can find in the process. Once we have this first approach, we use it as an initial seed to find the trajectory which minimize the sum of delta-v. The optimal solution is found again by an iterative process, where the nodes of the finite element method mesh are changing depending on the trajectory obtained. We have implemented two methodologies to change the mesh, which have a similar behavior. The first one (FEFF-DV) is a methodology based on the obtainment of the mesh avoiding ill-conditioned problems and big maneuvers. The second one (FEFF-A) is the adaptive remeshing.

Once we have the optimal trajectories for the functional which is more related to fuel consumption, we consider these trajectories for a more real-

istic model, such as full RTBP or JPL ephemeris. We consider also some errors in the execution of the maneuvers that must be corrected. We have concluded that the linear approximation is good to obtain nominal reconfiguration trajectories, since the corrections due to the truncation of the equations of motion are small for the cases under consideration.

In the future, we will try to adapt this method to formations of spacecraft flying in orbits about Earth, and the influence of the J_2 perturbation on the motion of the formation. We note that when we take the equations about the Earth, the perturbations due to J_2 are bigger than the ones that we obtain in the case of Halo orbits, and the linearized equations may not be good enough to obtain a first guess of the solution.

Bibliography

[1] R. W. Beard, T. W. McLain, F. Y. Hadaegh. *Fuel Optimization for Constrained Rotation of Spacecraft Formations*. AIAA Journal of Guidance, Control, and Dynamics 23, pp. 339-346, 2000.

[2] C. Beichman, G. Gómez, M.W. Lo, J.J. Masdemont, L. Romans. *Searching for Life with the Terrestrial Planet Finder: Lagrange Point Options for a Formation Flying Interferometer*. Advances in Space Reseach, 34, 637–644, 2004.

[3] E. Canalias, J.J. Masdemont. *Eclipse Avoidance for Lissajous Orbits Using Invariant Manifolds*. IAC Electronic Library, 1-11, 2004.

[4] *DARWIN System Assessment Study Orbit and Deployment Analysis*. Projecte amb Deimos Space S. A. num DW-SAS-ASP-TN-446. Octubre-Desembre, 2006.

[5] P. Díez, A. Huerta. *A unified approach to remeshing strategies for finite element h-adaptivity*. Computer methods in applied mechanics and engineering 176, pp. 215-229, 1999.

[6] D. Dunham, R. Farquhar. *Libration Point Missions, 1978-2002*. World Scientific Pub., pp. 45-73, 2002.

[7] R. W. Farquhar, D. P. Muhonen, C. Newman, H. Heuberger. *Trajectories and Orbital Maneuvers for the First Libration-Point Satellite*. Journal of Guidance and Control, Vol. 3, No. 6, pp. 549-554, 1980.

[8] M. Farrar, M. Thein and D. C. Folta *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.

[9] R. Fletcher. *Practical Methods of Optimization*. Volume 2: Constrained Optimization. John Wiley and Sons, Chichester and New York, 1981.

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

[11] P. E. Gill, W. Murray and M. H. Wright. *Practical Optimization*. Academic Press, London and New York, 1981.

[12] G. Gómez, J. Llibre, R. Martínez, C. Simó. *Dynamics and Mission Design near Libration Points. Vol 1, Fundamentals: The Case of Collinear Libration Points*, World Scientific Pub., xi+443, 2001.

[13] G. Gómez, M. Marcote, J. J. Masdemont, J. M. Mondelo. *Zero Relative Radial Acceleration Cones and Controlled Motions Suitable for Formation Flying*, The Journal of the Astronautical Sciences, Vol. 53, no. 4, 2005.

[14] G. Gómez, M.W. Lo, J.J. Masdemont, K. Museth. *Simulation on Formation flight near Lagrange Points for the TPF mission*. Advances in the Astronautical Sciences, 109, 61–75, 2002.

[15] F.Y. Hadaegh, D.P. Scharf, S.R. Ploen. *Initialization of distributed spacecraft for precision formation flying*. Control Applications, 2003. Proceedings of 2003 IEEE Conference on 2003 IEEE Conference on Volume 2, 1463- 1468 vol.2, 2003.

[16] M. Hechler, J. Cobos. *Herschel, Planck and GAIA Orbit Design*. In Procedings of the Libration Point Orbits and Applications, pp 115-135, 2002.

[17] G. T. Huntington, D. Benson, A. V. Rao. *Optimal Configuration of Tetrahedral Spacecraft Formations*. The Journal of Astronautical Sciences, Vol. 55, no.2, 141-169, 2007.

[18] O. Junge, S. Ober-Blobaum. *Optimal Reconfiguration of Formation Flying Satellites*. Decision and Control, 2005 European Control Conference. CDC-ECC '05., 66- 71, 2005.

[19] L. Y. Li, P. Bettess, J.W. Bull, T. Bond, I. Applegarth. *Theoretical formulation for adaptive finite element computations*. Communications in Numerical Methods in Engineering 11, 858-868, 1995.

[20] B. G. Marchand and K. C. Howell, *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.

[21] J. Masdemont. *Curs d'elements finits amb aplicacions*. Edicions UPC, 2002.

[22] L.M. Mailhe, J.J. Guzman. *Initialization and resizing of formation flying using global and local optimization methods*. Aerospace Conference, 2004. Proceedings. 2004 IEEE, - 556 Vol.1, 2004.

[23] C. R. McInnes. *Autonomous proximity manoeuvring using artificial potential functions*. ESA Journal, vol. 17, no. 2, pp. 159-169, 1993.

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

[25] J. T. Oden, J. N. Reddy. *Variational methods in theoretical mechanics*. Springer Verlag, Berlin, 1983.

[26] L. Perea, G. Gómez, P. Elosegui. *Extension of the Cucker-Smale Control Law to Space Flight Formations*. Journal of Guidance, Control and Dynamics, Vol. 32, no. 2, 2009.

[27] J. N. Reddy. *An Introduction to the Finite Element Method*. McGraw-Hill, New York, 1993.

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

[29] G. Singh, F. Hadaegh. *Autonomous Path-Planning for Formation Flying Applications*. In International Symposium on Space Flight Dynamics, 2001.

[30] V. Szebehely. *Theory of Orbits*. Academic Press, 1967.

[31] 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 12, iii+218, July 2000.

[32] The TPF Science Working Group. *The Terrestrial Planet Finder: A NASA Origins Program to Search for Habitable Planets.* JPL Publication 99-3, 1999.

[33] J. B. Thevenet, R. Epenoy. *Minimum-Fuel Deployment for Spacecraft Formations via Optimal Control.* Journal of Guidance, Control and Dynamics, Vol. 31, no. 1, 2008.

[34] R. A. Volpe. *Real and Artificial Forces in the Control of Manipulators: Theory and Experiments.* Ph.D. Thesis, Dept. of Physics, Carnegie Mellon University, September, 1990.

[35] P.K.C. Wang, F. Y. Hadaegh. *Minimum-fuel Formation Reconfiguration of Multiple Free-flying Spacecraft* Journal of the Astronautical Sciences, vol. 47, no. 1,2, pp. 77-102, 1999.

[36] O.C.Zienkiewicz, R.L.Taylor. *The finite element method*, McGraw-Hill, 1994.

[37] O. C. Zienkiewicz, J. Z. Zhu. *A simple error estimator and adaptive procedure for practical engineering analysis* International Journal for Numerical Methods in Engeneering., 24, pp. 337-357, 1987.

[38] http://lisa.nasa.gov/. *Laser Interferometer Space Antenna – LISA.*

[39] http://www.rssd.esa.int/index.php?project=xeus. *XEUS - The X-Ray Evolving Universe Spectroscopy Mission.*

[40] http://maxim.gsfc.nasa.gov/. *Micro-Arcsecond X-ray Imaging Mission - MAXIM.*

[41] http://www.asdc.asi.it/simbol-x/. *Simbol-x.*

[42] http://constellation.gsfc.nasa.gov/. *Official NASA Constellation-X Home Page.*

[43] http://cloudsat.atmos.colostate.edu/home. *CloudSat.*

[44] http://chandra.harvard.edu. *The Chandra X-ray Observatoy Center.*

[45] http://sohowww.nascom.nasa.gov. *Solar and Heliospheric Observatory Homepage.*