

# TIME SUBOPTIMAL FORMATION FLYING MANOEUVRES THROUGH IMPROVED MAGNETIC CHARGED SYSTEM SEARCH

Andrea D'Ambrosio<sup>a,\*</sup>, Dario Spiller<sup>b</sup>, Fabio Curti<sup>c</sup>

<sup>a</sup>*PhD student, School of Aerospace Engineering, Sapienza University of Rome, Via Salaria 851, 00138 Rome*

<sup>b</sup>*PostDoc, School of Aerospace Engineering, Sapienza University of Rome, Via Salaria 851, 00138 Rome*

<sup>c</sup>*Associate Professor, School of Aerospace Engineering, Sapienza University of Rome, Via Salaria 851, 00138 Rome*

---

## Abstract

The development of fast and reliable optimization algorithms is required in order to obtain real-time optimal trajectory on-board spacecraft. In addition, the wide spread of small satellites, due to their low costs, is leading to a greater number of satellite formations in space. This paper presents an Improved version of the Magnetic Charged System Search (IMCSS) metaheuristic algorithm to compute time-suboptimal manoeuvres for satellite formation flying. The proposed algorithm exploits some strategies aimed at improving the convergence to the optimum, such as the chaotic local search and the boundary handling technique, and it is able to self-tune its internal parameters and coefficients. Moreover, the inverse dynamics technique and the differential flatness approach, through the B-splines curves, are used to approximate the trajectory. The optimization procedure is applied to the circular J2 relative model developed by Schweighart and Sedwick and to the elliptical relative motion model developed by Yamanaka and Ankersen. The results of this paper show that the convergence is better achieved by using the proposed tools, thus proving the efficiency and reliability of the algorithm in solving some space engineering problems.

---

\*Corresponding author

*Email addresses:* [andrea.dambrosio@uniroma1.it](mailto:andrea.dambrosio@uniroma1.it) (Andrea D'Ambrosio), [dario.spiller@uniroma1.it](mailto:dario.spiller@uniroma1.it) (Dario Spiller), [fabio.curti@uniroma1.it](mailto:fabio.curti@uniroma1.it) (Fabio Curti)

*Keywords:* Formation flying manoeuvres, optimization, metaheuristic algorithms, inverse dynamics.

---

## 1. Introduction

The interest in satellite formations has rapidly increased in the last years due to the wide dissemination of small satellites, which are cheaper than the traditional ones: this feature has also allowed small companies and universities to have their own access to space. Satellite formations can be exploited for many different fields, such as Earth and planetary science, astronomy and astrophysics, technology demonstration and exploration. They are also important to have the possibility to distribute scientific instruments over many satellites. Indeed, the effective observation baseline can be expanded and the sensitivity of scientific instruments can be increased consequently. Moreover, since multiple scientific instruments often present competing and conflicting requirements on one satellite design and its operation, using a formation flying made up of simpler satellites having one instrument each can accomplish the same complex missions without the added design and operational overhead, while risking only one payload at a time (Xiang & Jørgensen, 2005). Therefore, it is essential to know how to manage a satellite formation. In particular, the design of the manoeuvres is fundamental in order to reconfigure and replace the satellites. This must be done by studying very accurately the GNC and, in particular, the requested optimal manoeuvres in terms of the minimum time of flight and/or the minimum fuel consumption. Two different approaches can be exploited to solve these optimization problems: deterministic and stochastic algorithms. The main difference is represented by the final result; in fact, with the same initial condition, deterministic algorithms always provide the same result, whereas stochastic algorithms always follow different paths, because of their typical randomization process, which cause the results not to be certainly the same. Among the advantages of stochastic algorithms, are their simplicity and global search capability. Moreover, there is no need to provide an initial

guess to start the optimization, that is a drawback for classical deterministic optimization algorithms for which the results can be significantly affected by the choice of the initial guesses. Since it is not possible to search for all the possible solutions, the aim of stochastic algorithms is to have a practical, efficient and fast algorithms that work most of the time providing feasible and good quality solutions. Thus, it is expected that some of them are nearly optimal, although there is no guarantee for such optimality (Yang, 2018).

Within the framework of stochastic algorithms, metaheuristics represent a very widely spread category. Many metaheuristic algorithms, inspired both by physic processes and animal biology/behaviour, have been proposed in recent years (some of them are reported in (Bandaru & Deb, 2016)). Among the most famous and employed metaheuristics, there are: Genetic Algorithm (GA) (Holland, 1975; Goldberg, 1989), inspired by Darwin's theory of evolution; Differential Evolution (DE) (Storn & Price, 1997), based on the concepts of mutation/recombination/selection; ant colony optimization (Dorigo et al., 1996), inspired by the behaviour of ants; Particle Swarm Optimization (PSO) (Clerc, 2010), based both on the social behaviour of humankind and on swarms of birds; Harmony Search (Geem, 2009), inspired by music; Gravitational Search Algorithm (GSA) (Rashedi et al., 2009), based on the Newton's law of gravitation; Charged System Search (CSS) (Kaveh & Talatahari, 2010), which exploits the electric force and the uniformly accelerated motion laws to make the particles move through the domain and Magnetic Charged System Search (MCSS), which is an improved version of the CSS, where the electric force is combined with the magnetic force to obtain a more realistic and efficient algorithm.

Metaheuristic algorithms have already been demonstrated to succeed in facing optimization problems for space applications (Pontani & Conway, 2013; Kim & Spencer, 2002; Zhang et al., 2008; Wang & Zheng, 2012). Within the field of spacecraft formation flying, DE and PSO have already been used to study optimal manoeuvres (Parente et al., 2018)); PSO has been applied to reconfiguration manoeuvres considering perturbation forces (Spiller et al., 2017a), to passive formation reconfiguration using attitude control (Spiller et al., 2018a),

to energy-optimal trajectory planning for spacecraft formation reconfiguration with collision avoidance (Huang et al., 2012); Inverse PSO has been used to study the rendezvous of nanosatellites taking into account the differential drag (Spiller & Curti, 2015). In particular, in the latter, the inverse dynamics (Ross & Fahroo, 2002; Boyarko et al., 2011; Spiller et al., 2016, 2018b), along with the differential flatness approach (Fliess et al., 1995; Louembet, 2007), has been used to achieve a reasonable compromise between the computational time and the optimality of the results. Thanks to this method, the state and the control are expressed as a function of a minimum number of optimization parameters named flat outputs. The advantage of using this approach is that the dynamical constraints are a priori satisfied and the flat output can be chosen to automatically fulfil the boundary constraints. Moreover, the approximate and sub-optimal control policy is obtained in an analytical closed form. In particular, B-spline curves are employed to accomplish this task because they provide very good results using a minimal number of support functions (Boor, 1972).

In order to model the relative motion of spacecraft within a formation flying, the reference trajectory of the chief satellite is usually considered. The well-known Clohessy-Wiltshire (CW) equations (Clohessy, 1960) are very often employed to study spacecraft formations, although other more complex models can be used. Among these, there are: the circular  $J_2$  relative model developed by Schweighart and Sedwick (Schweighart & Sedwick, 2001, 2002) and all the models that take into account also the chief elliptical orbits. In particular, we can distinguish between two different approaches to deal with the relative elliptical motion: the first uses orbital elements to describe the trajectory (Cao & Misra, 2015; Schaub, 2004; Schaub & Alfriend, 2001; Hamel & Lafontaine, 2007; Kuiack & Ulrich, 2018; Sengupta & Vadali, 2007), whereas the second exploits the time as the independent variable (Melton, 2000; Broucke, 2003; Yamanaka & Ankersen, 2002). Since in this work B-spline curves are employed to approximate the trajectory (and thus the velocity and the acceleration), the second approach is considered because it allows the use of differential equations written directly in terms of position and velocity.

Thus, the goal of this paper is to compute and analyse time suboptimal formation flying manoeuvres, considering the circular and elliptical relative motion models, by means of a new improved version of MCSS metaheuristic algorithm, called Improved Magnetic Charged System Search (IMCSS), which is self-adaptive and autonomous in the determination of the internal parameter and coefficients. In addition, some useful tools, such as the chaotic local search, the boundary handling technique and the entropy of the system, are introduced to speed up and improve the convergence. Obviously, there are some parameters appearing in the formulation of IMCSS which are intended to be suggested numbers for non-expert users that work well for the problems analyzed in this paper. However, they could be further optimized and tuned.

This paper is organized as follows. In Sections 2 and 3, an overview of the original MCSS and the improvements of the proposed algorithm are provided; Section 4 presents the optimal control problem and the employed mathematical models; finally, Section 5 shows the numerical simulations and results.

## 2. MCSS Overview

In this section, an overview of the original Magnetic Charged System Search algorithm (Kaveh et al., 2013) is provided. MCSS is a physics-inspired algorithm that exploits the electro-magnetic force and the Newtonian mechanics laws to make the charged particles (CPs) move through the search space to look for the best optimal solution.

Some preliminary parameters should be introduced before starting the algorithm. Let  $\mathbf{X}_i$  be the position vector associated with each CP, i.e., the vector that contains the optimization variables, and  $D$  the dimension of the problem (equal to the length of the vector  $\mathbf{X}_i$ ). In addition, an objective function, also called the performance index or fitness function ( $J_i$ ), is computed for each CP. Throughout this paper the subscript  $i$  (and also  $j$ ) is used to indicate the  $i(j)$ -th CP and therefore it varies in the range  $[1, N_{CP}]$ , where  $N_{CP}$  is the number of CPs. Moreover, let  $G_k$  be the maximum iterations number. Once the itera-

tions numbers and the number of CPs have been defined, it is possible to start with the actual optimization procedure. At the beginning, each component of the position vector associated with each CP is initialized randomly within the range  $[LB_m, UB_m]$ , where the subscript  $m$  refers to the  $m$ -th component of the particle, with  $m \in [1, D]$ . The initial velocity of all the CPs ( $\mathbf{V}_i^0$ ) is set equal to the null vector. Afterwards, the performance index is assigned to each CP and the objective functions, along with the corresponding CPs, are sorted increasingly. The first best CPs are stored in the Charged Memory (CM), and they are updated at each iteration. The number of the CPs stored in CM is defined as CMS (throughout this paper CMS is set equal to  $N_{CP}/5$ ). Other information that must be stored (and updated accordingly during the whole algorithm) are the best and worst objective functions.

The charge  $q_i^k$  of each CP at the step  $k$  is computed by taking into account the current performance index of the CP ( $J_i$ ), the so far best ( $J_g$ ) and worst ( $J_{worst}$ ) fitnesses among all the CPs, i.e.

$$q_i^k = \frac{J_i^k - J_{worst}}{J_g - J_{worst}}. \quad (1)$$

The charges must be stored because they are subsequently used to obtain the electric current. The indices  $k$  and  $g$  are used to indicate respectively the current iteration and the best CP, i.e., the CP with the best performance index. As can be seen from Eq. (1), the values of the charges are normalized within the range  $[0; 1]$ , where 0 is associated with the worst CP and 1 with the best CP.

Supposing that CPs move in a virtual straight wire of radius  $R_w$ , based on physical laws, they create a magnetic field in space, which generates magnetic forces on the other CPs. According to (Kaveh et al., 2013), the average electric current ( $I_{avg}$ ) can be computed as

$$(I_{avg})_i^k = \text{sign}(J_i^k - J_i^{k-1}) \cdot \frac{df_i^k - df_{min}^k}{df_{max}^k - df_{min}^k}, \quad (2)$$

$$df_i^k = |J_i^k - J_i^{k-1}|. \quad (3)$$

In Eq. (2), the function  $df$  represents the absolute value of the objective function variation of the  $i$ -th CP in the  $k$ -th movement (iteration), the function  $\text{sign}(\cdot)$  is

the sign function, and  $df_{max}^k$  and  $df_{min}^k$  are respectively the maximum and the minimum values of  $df$ .

In order to obtain the electric and magnetic forces acting on all CPs for each iteration, the separation distance between each pair of CPs and the separation distance between the  $i$ -th wire and the  $j$ -th CP must be computed. In this work, these two characteristic distances are supposed to be equal and are indicated with  $r_{ij}$ . The explicit expression of  $r_{ij}$  is given by (Kaveh & Talatahari, 2010; Kaveh et al., 2013)

$$r_{ij} = \frac{\|\mathbf{X}_i - \mathbf{X}_j\|}{\|(\mathbf{X}_i + \mathbf{X}_j)/2 - \mathbf{X}_g\| + \epsilon} \quad (4)$$

As already stated,  $\mathbf{X}_{i/j}$  represents the position of the generic CP (i.e., the vector of optimized variables) and  $\epsilon$  is a small positive constant to prevent singularity. One should note that Eq. (4) does not represent a real physical distance; thus, it is conceptually different from the actual distance between the position vectors of the CPs in the optimization procedure, computed as  $\|\mathbf{X}_i - \mathbf{X}_j\|$ . The particular expression of the separation distance is employed to adapt the physical equations of the electric and magnetic forces, described later in the current section, to the physics-inspired metaheuristic algorithms equations. Indeed, the separation distance is important since it takes into account not only the actual distance between particles, but also the position of the best particle, which is a crucial for the exploitation phase of metaheuristic algorithms.

Let us consider now an insulating solid spheres of radius  $R_a$  with a uniform volume charge density, each of which carries a total charge  $q_i$ . As already well-known from physical laws, the electric force acting on the  $j$ -th CP and produced by multiple CPs is

$$\mathbf{F}_{E,j} = k_e q_j \sum_{i,i \neq j} \left( \frac{q_i}{R_a^3} r_{ij} w_1 + \frac{q_i}{r_{ij}^2} w_2 \right) \frac{\mathbf{r}_i - \mathbf{r}_j}{\|\mathbf{r}_i - \mathbf{r}_j\|} \quad (5)$$

where

$$\begin{cases} w_1 = 1, w_2 = 0 & \iff r_{ij} < a \\ w_1 = 0, w_2 = 1 & \iff r_{ij} \geq a \end{cases}$$

and  $\mathbf{r}_{i,j}$  are the positions of the charges. There are two contributions within the brackets of the right-handed side of Eq. (5): the first one is valid inside

the sphere and it is a linear function of  $r_{ij}$ , the second one is valid outside the sphere and it is proportional to the inverse of the square of  $r_{ij}$ . In analogy with the real electric force, the equation is adapted for the case of this metaheuristic algorithm in the following way

$$\mathbf{F}_{E,j} = q_j \sum_{i,i \neq j} \left( \frac{q_i}{R_a^3} r_{ij} w_1 + \frac{q_i}{r_{ij}^2} w_2 \right) \cdot p_{ij} \cdot (\mathbf{X}_i - \mathbf{X}_j) \quad (6)$$

where

$$\begin{cases} w_1 = 1, w_2 = 0 & \iff r_{ij} < R_a \\ w_1 = 0, w_2 = 1 & \iff r_{ij} \geq R_a \end{cases}$$

and

$$p_{ij} = \begin{cases} 1 & \text{if } \frac{J_i - J_g}{J_j - J_i} > r \text{ or } J_j > J_i \\ 0 & \text{otherwise} \end{cases}$$

Here,  $p_{ij}$  indicates the probability of attraction of the  $i$ -th CP by the  $j$ -th CP and  $r$  represents a real uniformly-distributed random number within the range  $[0,1]$ . Throughout this work, the radius of the insulating sphere  $R_a$  is set equal to 1. The magnetic force acting on the  $j$ th CP is

$$\mathbf{F}_{B,j} = q_j \sum_{i,i \neq j} \left( \frac{I_i}{R_w^3} r_{ij} z_1 + \frac{I_i}{r_{ij}} z_2 \right) \cdot pm_{ij} \cdot (\mathbf{X}_i - \mathbf{X}_j) \quad (7)$$

where

$$\begin{cases} z_1 = 1, z_2 = 0 & \iff r_{ij} < R_w \\ z_1 = 0, z_2 = 1 & \iff r_{ij} \geq R_w \end{cases}$$

and

$$pm_{ij} = \begin{cases} 1 & \text{if } J_j > J_i \\ 0 & \text{otherwise} \end{cases}$$

In Eq. (7),  $R_w$  is set equal to 1 and  $pm_{ij}$  is the probability of the magnetic influence (attracting or repelling) of the  $i$ -th wire on the  $j$ -th CP. This factor is very important because only a good CP can influence a bad CP. Instead, this does not happen for the electric force, where good and bad CPs can influence each other. In addition, in order to emphasize the best CPs and give them much more importance in the algorithm, the number of ‘‘active’’ CPs (the CPs



which actually attract/repel) is not assumed to be equal to  $N_{CP}$ , but only to a fraction of the whole population. This fraction can be a free parameter decided by the user and it can affect significantly the optimization procedure; thus, it must be set carefully. A reasonable choice for the active CPs, found during this work, is  $N_{CP}/10$ , which is the value used throughout this paper. The total force acting on the  $j$ -th CP is then

$$\mathbf{F}_j = p_r \mathbf{F}_{E,j} + \mathbf{F}_{B,j} \quad (8)$$

$$p_r = \begin{cases} 1 & \text{if } r > k_{ar} \cdot (1 - k/G_k) \\ -1 & \text{otherwise} \end{cases} \quad (9)$$

where  $p_r$  is the probability that an electrical force is a repelling force. The expression  $k_{ar} \cdot (1 - k/G_k)$  decreases with the iterations: this means that the repelling forces ensure the exploration at the beginning of the algorithm, whereas, at the end, the electric forces become more and more attractive, so that the exploitation is increased. Once the forces are computed, the motion of the CPs has to be considered. Even in this case, the equations of motion are inspired by the physics laws, in particular by the uniformly accelerated motion laws.

$$\mathbf{X}_j^{k+1} = r_{j1} \cdot k_a \cdot \frac{\mathbf{F}_j}{m_j} \cdot \Delta t^2 + r_{j2} \cdot k_v \cdot \mathbf{V}_j^k \cdot \Delta t + \mathbf{X}_j^k \quad (10)$$

$$\mathbf{V}_j^{k+1} = \frac{\mathbf{X}_j^{k+1} - \mathbf{X}_j^k}{\Delta t} \quad (11)$$

In Eqs. (10) and (11),  $r_{j1}$  and  $r_{j2}$  are two random numbers that are uniformly distributed in the range  $[0,1]$  and they replace the real physics coefficients determined by the double integration of the acceleration (in particular,  $r_{j1}$  replaces  $1/2$  and  $r_{j2}$  replaces  $1$ ). In addition, the time interval  $\Delta t$  is set equal to the unity and the mass of each CP ( $m_j$ ) is indeed substituted by its charge ( $q_j$ ). This last choice makes the simplification of a fraction ( $\mathbf{F}_j/q_j$ ) occurs and allows to take into account also the electric and magnetic accelerations (Eqs. (6),(7)) acting on the worst CP (for which  $q_j = 0$ ); otherwise, a null force would be applied on it. The remaining coefficients are respectively the coefficient of the acceleration ( $k_a$ ) and the coefficient of the velocity ( $k_v$ ). In the original MCSS,

$k_{ar}$  is set equal to 0.1, whereas  $k_a$  and  $k_v$  start from a magnitude of 0.5 and are linearly increased and decreased, respectively.

### 3. Improved Magnetic Charged System Search

One of the efforts computed in this work consists of trying to make MCSS autonomous in the choice of its internal parameters and coefficients. This results in a useful tool to optimize problems also for users who are not expert within the field of metaheuristic algorithms. Thus, this section focuses on the proposed new version (IMCSS) of the original MCSS. The global flow chart of IMCSS is illustrated in Fig. 1, where the white blocks indicate the common parts between IMCSS and the original MCSS, whereas the light and dark grey blocks indicate, respectively, the novelties and the modifications introduced. Basically, IMCSS inherits the core of MCSS, described in section 2, which involve the computation of the electric and magnetic forces, along with the equations for the electric charge and current, and the particles position and velocity updates.

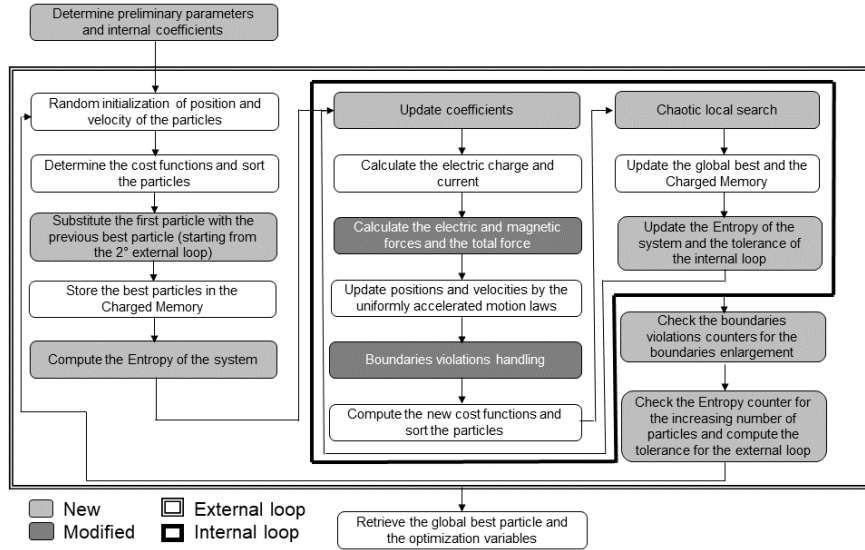


Figure 1: IMCSS flow chart.

All the novelties and modifications with respect to the original MCSS are

described in detail in the rest of this section. In particular, some strategies for the selection of equations coefficients and hyperparameters of the algorithm, such as the initial number of CPs and the maximum iterations number, are proposed. Furthermore, the introduction of the chaotic local search, the boundary handling technique and the concept of entropy of the system are presented.

### 3.1. Novel Strategies for Hyperparameters and Coefficients Selection

Let  $W$  be the maximum order of magnitude of the problem, computed as the order of magnitude of the maximum value among the components of the difference vector between the upper bound (**UB**) and the lower bound (**LB**) vectors of the optimization variables, i.e.

$$W = \lfloor \log_{10}(\max(\mathbf{UB} - \mathbf{LB})) \rfloor, \quad (12)$$

where  $\lfloor \cdot \rfloor$  computes the nearest integer towards minus infinity and  $|\cdot|$  the absolute value. The idea of IMCSS is to link the number of CPs and the maximum iterations number to the only parameters defined by the (non-expert) user, i.e., **UB** and **LB**. The number of CPs is set equal to

$$N_{CP} = 10 \cdot \{W + r_{in}[2, 3\lceil \ln(D + 1) \rceil]\} \quad (13)$$

In Eq. (13),  $\lceil \cdot \rceil$  and  $r_{in}$  represent respectively the nearest integer towards infinity and a random integer number between the two in brackets. Since it is not appropriate to have a too large number of CPs because of the slow computational time, it is necessary to set a maximum number of CPs ( $N_{CP,max}$ ). Thus, in this work,  $N_{CP,max}$  is reasonably set equal to 50, which represents a good compromise between the computational time and the effectiveness of the algorithm. The choices behind Eq. (13) are related to the fact that if the dimension increases, also the complexity of the problem increases and thus a greater number of CPs could be required for the success of the algorithm.

Another novelty with respect to the original MCSS is that the proposed algorithm is divided into external and internal loops. The internal loops exploit the fast convergence (even for a low number of iterations) of the original MCSS,

whereas the external loops emphasize the exploration ability, reducing at the same time the possibility to obtain a premature stall of the algorithm. In order to ensure a continuity in the search of the optimized variables and preserve the results of the previous iterations, starting from the second external loop, the first particle (the current re-initialized best particle) is replaced with the last best particle of the whole previous external loop. The maximum internal iterations number ( $G_k$ ) is set to

$$G_k = 600 - 3N_{CP} \quad (14)$$

Also in this case, a maximum number of internal iterations is implicitly set for the same aforementioned reason. The choice of the decreasing law in Eq. (14), instead of having a constant value, provides some guarantees for the exploration of the search space even if the number of CPs chosen by the algorithm is too small. Indeed, if  $N_{CP}$  is low,  $G_k$  is high to compensate for the reduced number of particles with the possibility to have more iterations to explore the search space. On the contrary, if  $N_{CP}$  is high, the search space is better explored from the beginning of each internal loop, because the particles are more widely distributed. The external iterations number ( $G_K$ ) is

$$G_K = \max(\lceil 12 - G_k/10^{\lfloor \log_{10}(G_k) \rfloor} \rceil ; 3), \quad (15)$$

where the function  $10^{\lfloor \log_{10}(G_k) \rfloor}$  extracts the order of magnitude of  $G_k$ . In order to improve the convergence at the end of the algorithm, the iterations number of the last internal loop ( $G_{kf}$ ) is

$$G_{kf} = 5G_k. \quad (16)$$

New expressions are also introduced for some coefficients appearing in the equations of the previous section. First, the value of  $k_{ar}$  in Eq. (9) is now defined as

$$k_{ar} = \min(G_k/1000 + 10/N_{CP} ; 0.5). \quad (17)$$

The reason behind the form of Eq. (17) is due to the fact that when  $G_k$  is high, there is more "time" (iterations) to explore the search space, thus it is possible

to have a great repulsion (high value of  $k_{ar}$ ). On the contrary, when  $G_k$  is not so high, the exploration phase is reduced in favor of a faster exploitation, in order to ensure the convergence. For what concerns the parameter  $N_{CP}$ , if its value is high, the exploration is already ensured at the beginning by the wide diffusion of the CPs in the search space; thus, the exploitation is more encouraged (this is the reason why an inverse proportionality is chosen). The maximum admitted value for  $k_{ar}$  is equal to the probability of 50% between attraction and repulsion.

The expressions of the acceleration and velocity coefficients, appearing in Eqs. (10) and (11), are chosen to be

$$k_a = k_{a,0} + (k_{a,f} - k_{a,0}) \cdot k/G_k \quad (18)$$

$$k_v = k_{v,f} + (k_{v,0} - k_{v,f}) \cdot k/G_k \quad (19)$$

As it can be seen from Eqs. (18) and (19), considering  $k_{a/v,f} > k_{a/v,0}$ ,  $k_a$  is a linearly increasing function, while  $k_v$  is linearly decreasing. The initial and final values of  $k_a$  and  $k_f$  are set equal to

$$\left\{ \begin{array}{l} k_{v,f} = 1 + r \cdot \left( \frac{1}{N_{CP}} + \frac{G_k}{10^{\lceil \log_{10}(G_k) \rceil}} \right) \\ k_{v,0} = 0.8 \\ k_{a,0} = \lceil k_{v,f} \rceil - k_{v,f} \\ k_{a,f} = 2 \cdot k_{a,0} \end{array} \right. \quad (20)$$

The reasons behind the increasing and decreasing laws of  $k_a$  and  $k_v$  needs to be searched in Eq. (10), which shows that the new position of each CPs is determined by three main contributions: the old position and the terms related to both the velocity and the acceleration. The term related to the velocity can be thought of as the inertia term which appears in the velocity equation of the PSO (Clerc, 2010); thus  $k_v$  plays here the role that  $w$  plays in the PSO. It's important that the inertia (velocity) term is more relevant at the beginning of the algorithm to ensure the exploration, but then it should be decreased in favor of the exploitation. On the other hand, the acceleration term should be more relevant at the end of the iterations to improve the exploitation in the neighborhood of the best CP: this idea can also be clearly seen from Eqs.

(8) and (9), where the probability of attraction increases with the iteration number. The inverse and direct proportionalities, with respect to  $N_{CP}$  and  $G_k$ , appearing in the first of Eq. (20), have basically the same explanation provided for Eq. (17). As already stated in the introduction, some parameters appear in the previous equations. These are suggested parameters that work well for the analyzed problems and they can be helpful for non-expert users. However, they could eventually be modified by the user in the attempt to obtain better performances.

### 3.2. Chaotic Local Search

To enhance the performance of the IMCSS, a chaotic local search (CLS) is introduced. Chaotic search strategy is inspired by the chaos phenomenon in nature, modelled as a classic nonlinear dynamical system with the properties of ergodicity, randomness, and sensitivity to its initial conditions (Jia et al., 2011). Thanks to these features, a chaotic system can randomly generate a long-time sequence that is able to traverse through every state of the system without repetitions in certain ranges (Guo et al., 2015). In order to achieve a better balance between exploration and exploitation abilities of the IMCSS and improve the convergence of the whole algorithm, more heuristic information is integrated, such as the search direction provided by the best particles in the current population. The chaotic local search employed for the proposed algorithm is inspired by the work of (Junkai et al., 2017). Thus, the main features of their chaotic local search are employed, but with some differences. The chaotic local search is exploited in order to improve the best CP within the current iteration by searching in its neighborhood. This strategy could help to reach faster the global best. Therefore, the following criterion is applied

$$\mathbf{X}_{g1}^k = \mathbf{X}_g^k + (Z - 0.5) \cdot (\mathbf{X}_{CM_{r_{i1}}}^k - \mathbf{X}_{CM_{r_{i2}}}^k) \iff r_1^2 < r_2 \quad (21)$$

In Eq. (21),  $\mathbf{X}_{g1}^k$  is a new possible global best,  $\mathbf{X}_{CM_{r_{i1/2}}}^k$  are two random CPs chosen among the Charged Memory, and  $r_{1/2}$  are two real random numbers. The condition  $r_1^2 < r_2$  makes the local search occur with enough probability,

since  $r_{1/2}$  varies within the range  $[0; 1]$  and therefore  $r_1^2$  is generally smaller than  $r_2$ . The chaotic local search is not performed at each iteration in order to achieve a good compromise between the effectiveness of the search and the required computational effort; in fact, since the success of the local search is not guaranteed, it is more reasonable to make it happen randomly to save computational time.  $Z$  is also a real number which is initialized randomly in the range  $[0; 1]$  and it is updated with the rule

$$Z_{t+1} = 4Z_t \cdot (1 - Z_t). \quad (22)$$

The value of  $Z$  is updated only if the chaotic local search succeeds in finding a new global best, i.e.,  $J_{g1}^k < J_g^k$  (for a minimization problem). In this case, the index  $t$  is increased by 1 and the new best CP ( $\mathbf{X}_{g1}^k$ ) replaces the old one ( $\mathbf{X}_g^k$ ).

### 3.3. Boundary Constraints Handling

The update of the position vectors of the CPs, expressed by Eq. (10), can cause the optimization variables to exceed the user-defined boundaries, represented by the vectors **UB** and **LB**. This can lead the algorithm to diverge or to explore a region which is not of interest for the user. Thus, the components of the position vectors of each CP that exceed their respective boundaries must be collocated again inside the feasible search space. These boundary violations can be managed in very different ways. For example, the variables exceeding the boundaries could be directly re-initialized or substituted by the lower or upper bound values. More complex strategies could be adopted, such as the harmony search-based algorithm (Kaveh & Talatahari, 2010; Kaveh et al., 2013; Lee & Geem, 2004; Kaveh & Talatahari, 2009), also employed in the original MCSS. In this case, when a variable exceeds the lower/upper bound defined by the user, it is either re-initialized or substituted with the corresponding value belonging to a CP chosen among the CM (or among its neighborhood). This strategy is ruled by two parameters: the CMCR, which is the Charged Memory Considering Rate, and the PAR, that is the Pitching Adjusting Rate. However, two further values should be determined. In order to make the algorithm automat-

ically choose the best way to handle these violations, the following strategy is exploited instead,

$$\begin{cases} X_{i,m} = X_{CM_{r_i},m} & \text{if } r_1^2 < r_2, \\ X_{i,m} = LB_m + r \cdot (UB_m - LB_m) & \text{otherwise.} \end{cases} \quad (23)$$

The index  $m$  is used to indicate the dimension associated with the outbound variable, while  $r$ ,  $r_1$  and  $r_2$  are three real random numbers. Every time a variable violates its corresponding boundary, a counter is updated ( $c_{LB,m}$  in the case of a lower bound violation and  $c_{UB,m}$  for an upper bound violation). At the end of an internal loop (i.e. before starting the successive external loop), these counters are used to make the algorithm understand if the boundaries could have been badly defined by the user. If these counters exceed a maximum value  $C_{max}$ , the corresponding boundary is enlarged (Eq. (24)).

$$\begin{cases} UB_m = UB_m \cdot 10 + \epsilon & \iff c_{UB,m} > C_{max} \wedge UB_m \geq 0 \\ LB_m = LB_m/10 - \epsilon & \iff c_{LB,m} > C_{max} \wedge LB_m \geq 0 \\ UB_m = UB_m/10 & \iff c_{UB,m} > C_{max} \wedge UB_m \leq 0 \\ LB_m = LB_m \cdot 10 & \iff c_{LB,m} > C_{max} \wedge LB_m \leq 0 \end{cases} \quad (24)$$

Note that  $C_{max} = 0.1 \cdot k_f \cdot N_{CP}$ , where  $k_f$  represents the index of the last internal iteration. This value could be different from  $G_k$  ( $k_f \leq G_k$ ) because two exit strategies are considered, i.e., when the maximum iterations number ( $G_k$ ) is reached or when a pre-defined tolerance is satisfied. In Eq. (24),  $\epsilon$  represents a small positive constant. This tool is very useful and could help the algorithm to find a global minimum when it is located on the boundaries or when it is not inside the search space (in these cases the CPs tend to move much more towards the boundaries and this could be an indication of a bad-defined boundary). In addition, this approach could be essential in some cases in which the optimization variable do not have a direct physical meaning, as it is for example for the coefficients of polynomials that approximate the trajectory. Certainly, if the objective function has local minima inside the search space, it



could be more difficult for the IMCSS to realize that the boundaries have been ill defined.

### 3.4. Entropy of the System

In order to have an idea of how the optimization is being performed by the IMCSS, a parameter, called “entropy” ( $S$ ), can be introduced. As in physics, the idea of entropy is associated to the disorder of the system. Within IMCSS framework, this quantity is supposed to measure the disorder of the whole population of CPs with respect to the best one. This is strictly connected to the stall of the algorithm, i.e. when the objective functions of the CPs are no more significantly improved through the iterations, creating convergence problems. In the  $D$ -dimensional search space, the stall problem corresponds to have the CPs wandering around without finding better solutions. The concept of entropy here defined aims at overcoming stall problems, thus leading to better results. The entropy can be defined as

$$S = \text{median}(\mathbf{J}^k) \quad (25)$$

Eq. (25) associates the entropy of the population with the median element of the objective functions vector. Obviously, if  $S$  decreases, it means that the CPs are closer to the best CP; thus, the algorithm is going to converge. It is important to note that, in order to state that the algorithm is performing well, the entropy needs to be decreasing. However, this is in contrast with the concept of entropy in physics, which is always a non-decreasing function for natural transformations (the analogy with the physical entropy is just related to the concept of disorder).

At each iteration, if  $S$  is non-decreasing, a counter ( $c_S$ ) is updated. At the end of an internal loop, the  $N_{CP}$  is increased if  $c_S$  exceeds a defined maximum value, as follows:

$$N_{CP}^{K+1} = \min[N_{CP}^K + r_{in}[W ; 3[\log(D+1)]] ; [N_{CP,max} \cdot (1+0.1)]] \text{ if } c_S/k_f > 0.5 \quad (26)$$

In Eq. (26),  $r_{in}$  represents a random integer number between the two in brackets. It is important to notice that this strategy based on the entropy is not directly essential and determinant in finding the optimum, but it could be a helpful tool for the algorithm to avoid possible stalls. Indeed, a greater number of CPs can support the following loops of the algorithm to be more successful with respect to the previous ones.

### 3.5. End of the Algorithm

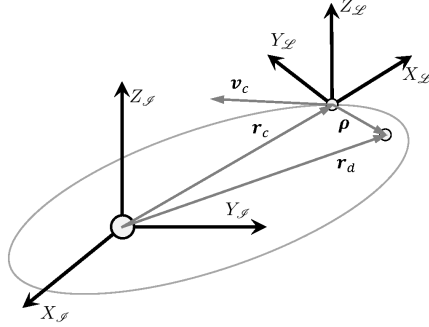
As already mentioned, the internal loop ends when the maximum iterations number is reached or when a pre-defined tolerance ( $\delta_{user}$ ) is achieved. The same occurs with the external loop. The tolerances can be defined in many different ways by taking into account the objective functions of the CPs, such as by computing the absolute error, the relative error, the standard deviation among a set of CPs ( $N_\delta$ ), etc.. Throughout this paper, the tolerance ( $\delta$ ) must be lower than  $10^{-10}$  and it is defined as the standard deviation of the last  $N_\delta$  objective functions.

$$\delta = \sqrt{\frac{\sum_{k=1}^{N_\delta} (J_g^k - J_g^{mean})^2}{N_\delta}} \quad (27)$$

where  $J_g^{mean}$  is the mean value of the last  $N_\delta$  objective functions. Furthermore, if  $N_{CP} < N_{CP,max}$ , the last external loop is performed with an increased number of CPs, which is updated with the same law as the one in Eq. (26), in order to ensure a better convergence.

## 4. Optimal Control Problem Statement

The description of the optimal control problem dealing with satellite formation time-optimal manoeuvres is provided in this Section. Hereafter, a formation made up of two satellites is considered: the main spacecraft is called chief ( $c$ ) and the other one deputy ( $d$ ). Two different models are presented. The first one is the circular  $J_2$  model developed by Schweighart and Sedwick, referred to as the SS model (Schweighart & Sedwick, 2001, 2002), which takes into account the  $J_2$  perturbation due to the non-spherical shape of the Earth. This perturbation



**Figure 2:** ECI coordinate system  $\mathcal{S}$  and LVLH coordinate system  $\mathcal{L}$ .

is significant when low-Earth orbits are considered, for which the orbital period is low. In this framework, the time of flight of the manoeuvres could represent a significant fraction compared to the entire orbital period. The second model deals with elliptical relative motion with arbitrary eccentricity, referred to as the Yamanaka-Ankersen model (YA) (Yamanaka & Ankersen, 2002).

#### 4.1. Circular Model Including the $J_2$ Perturbation

The equations of motion of the system, referred to an Earth-centered inertial (ECI) coordinate system  $\mathcal{S} = \{X_{\mathcal{S}}, Y_{\mathcal{S}}, Z_{\mathcal{S}}\}$  and shown in Fig. 2, are

$$\ddot{\mathbf{r}} = -\frac{\mu}{r^3}\mathbf{r} + \mathbf{u} + \mathbf{f} \quad (28)$$

where  $\mu = 3.986 \cdot 10^5 \text{ km}^3/\text{s}^2$  is the Earth gravitational parameter,  $\mathbf{r}$  is the position vector of the satellite, and  $r$  is its magnitude,  $\mathbf{u}$  is the external acceleration (provided by the engines), and  $\mathbf{f}$  is the perturbation acceleration (depending on the environmental forces).

The relative motion is usually described in the local-vertical/local-horizontal (LVLH) reference frame  $\mathcal{L} = \{X_{\mathcal{L}}, Y_{\mathcal{L}}, Z_{\mathcal{L}}\}$ , shown in Fig. 2, with  $X_{\mathcal{L}}$  pointing from the center of the Earth to the origin of  $\mathcal{L}$ ,  $Z_{\mathcal{L}}$  perpendicular to the orbital plane and  $Y_{\mathcal{L}}$  completing the right-hand Cartesian coordinate system. This reference system is centered on the chief satellite. Let  $\boldsymbol{\rho} = \mathbf{r}_d - \mathbf{r}_c = [\rho_x, \rho_y, \rho_z]^T$  be the relative distance between the chief and

the deputy. Considering the chief satellite on a circular orbit (which implies  $\omega = \sqrt{\mu/r^3}$ ), including the  $J_2$  gravitational effect in  $\mathbf{f}$  and imposing  $\mathbf{u}_c = 0$  (the chief is not controlled), the relative motion is described by the SS model

$$\begin{cases} \ddot{\rho}_x - 2\bar{m}\dot{\rho}_y - (4\bar{m}^2 - \bar{n}^2)\rho_x = u_x \\ \ddot{\rho}_y + 2\bar{m}\dot{\rho}_x = u_y \\ \ddot{\rho}_z + (2\bar{m}^2 - \bar{n}^2)\rho_z = u_z \end{cases} \quad (29)$$

where  $\bar{m} = \omega\sqrt{1 + k_{J_2}}$ ,  $\bar{n} = \omega\sqrt{1 - k_{J_2}}$  and  $k_{J_2} = (3J_2R_c^2/8r_c^2)[1 + 3\cos(2i_c)]$ .

It is possible to obtain closed relative trajectories within the SS model by setting the no along-track shift condition  $\dot{\rho}_y(t_0) = -2\bar{m}\rho_x(t_0)$ . In particular, three characteristic types of closed trajectory can be derived: along-track formation (ATF), general circular formation (GCF), and projected circular formation (PCF). In these cases, the general satellite trajectory can be written in magnitude-phase form as

$$\begin{cases} \rho_x(t) = \frac{1}{2}R \sin(\bar{n}t + \alpha_0) \\ \rho_y(t) = \rho_{y,0} + R \cos(\bar{n}t + \alpha_0) \\ \rho_z(t) = \frac{\chi_{zx}}{2}R \sin(\bar{n}t + \alpha_0) \end{cases} \quad (30)$$

where  $R$  dictates the dimension the trajectory,  $\alpha_0$  is the initial phase angle, and  $\rho_{y,0}$  is the center of the close trajectory along the y axis. For the PCF,  $\chi_{zx} = 2$ ; for the GCF,  $\chi_{zx} = \sqrt{3}$ ; and for the ATF,  $R = 0$  and  $\rho_{y,0} \neq 0$ .

#### 4.2. Elliptical Relative Motion

In this section, the elliptical relative motion model is introduced considering the work of Yamanaka and Ankersen (Yamanaka & Ankersen, 2002). The equation of motion are given here in the LVLH rotating reference frame shown in Fig. 2, whereas in the original work the same equations are reported for a different reference frame, where the  $Z$  axis points the nadir direction (in the opposite direction of the current  $X_{\mathcal{L}}$  axis), the  $Y$  axis is normal to the orbital plane and opposite to the angular momentum vector (in the opposite direction of the current  $Z_{\mathcal{L}}$  axis), and the  $X$  axis completes the right-hand system (in the same direction of the current  $Y_{\mathcal{L}}$  axis). However, only a simple rotation

is required to pass from one frame to the other. Supposing that all the perturbation forces are the same for the chief and the deputy and considering to have thrusters only on-board the deputy satellite, the linearized equations for the elliptical relative motion are

$$\begin{cases} \ddot{\rho}_x - 2\frac{\mu}{h^{3/2}}\omega^{3/2}\rho_x - 2\omega\dot{\rho}_y - \dot{\omega}\rho_y - \omega^2\rho_x = u_x \\ \ddot{\rho}_y + \frac{\mu}{h^{3/2}}\omega^{3/2}\rho_y + 2\omega\dot{\rho}_x + \dot{\omega}\rho_x - \omega^2\rho_y = u_y \\ \ddot{\rho}_z + \frac{\mu}{h^{3/2}}\omega^{3/2}\rho_z = u_z \end{cases} \quad (31)$$

where  $h$  is the angular momentum,  $\omega$  and  $\dot{\omega}$  are in this case the angular velocity and its variation, whose expressions are provided in Eq. (32)

$$\begin{cases} \omega = \frac{\mu^2}{h^3}(1 + e \cos \theta)^2 \\ \dot{\omega} = -2\frac{\mu^2}{h^3}e\omega \sin \theta(1 + e \cos \theta) \end{cases} \quad (32)$$

Note that  $\omega$  and  $\dot{\omega}$  are both function of the true anomaly ( $\theta$ ); thus, in order to retrieve those value at each time instant, the true anomaly must be computed from the time by using the well-known Kepler equation. This is accomplished by means of the Newton-Raphson method. Moreover, the only assumption made in Eq.(31) is that the distance between the deputy and the chief is small compared to the distance between the chief and the center of the Earth. Therefore, the equations are applicable to orbits of arbitrary eccentricity ( $e$ ).

A state transition matrix ( $\Phi(\theta, \rho_0, \dot{\rho}_0)$ ) can be obtained for Eq. (31) under the assumption of free motion (both the chief and the deputy are not controlled). This matrix is not reported here for the sake of conciseness, but it can be found in (Yamanaka & Ankersen, 2002). In the case of elliptical relative motion, it is still possible to find closed relative trajectories if the orbital periods (and thus the semi-major axes) of the chief and the deputies are the same. Assuming that the relative orbit radius is small compared to the chief orbit radius, the condition for closed periodic relative trajectory, written in the frame represented

in Fig. 2, is (Sengupta & Vadali, 2007)

$$(2 + e \cos \theta)(1 + e \cos \theta)^2 \left( \frac{\rho_x}{p} \right) + e \sin \theta \left( \dot{\rho}_x \sqrt{\frac{p}{\mu}} \right) - e \sin \theta (1 + e \cos \theta)^2 \left( \frac{\rho_y}{p} \right) + (1 + e \cos \theta) \left( \dot{\rho}_y \sqrt{\frac{p}{\mu}} \right) = 0 \quad (33)$$

In Eq. (33),  $p$  represent the semi-latus rectum of the reference orbit. In order to have a closed relative trajectory, the condition expressed by Eq. (33) must be fulfilled at each point of the orbit. If the previous equation is computed at the perigee of the chief orbit, the condition becomes easier:

$$\frac{\dot{\rho}_{y,0}}{\rho_{x,0}} = \frac{-n(2 + e)}{\sqrt{(1 + e)(1 - e)^3}} \quad (34)$$

where  $n$  is the mean motion of the reference orbit. It is important to note that Eq. (33) is satisfied for an infinite combination of initial conditions, except when  $\theta_0 = 0$  or  $\theta_0 = \pi$ . Furthermore, when  $e = 0$ , this reduces to the well-known Hill's condition for periodicity.

#### 4.3. Optimal Control Problem Definition

It is easy to see that Eqs. (29) and (31) can also be written in the state-space form in the following form

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \quad (35)$$

where matrices  $\mathbf{A}$  and  $\mathbf{B}$  can be easily computed from Eqs. (29) and (31),  $\mathbf{x} = [\boldsymbol{\rho}^T, \dot{\boldsymbol{\rho}}^T]^T$  and  $\mathbf{u} = [u_x, u_y, u_z]^T$ . Let  $t_0$  and  $t_f$  be the initial and the desired final relative state; the time-optimal problem considered in this paper is defined as follows

$$\left\{ \begin{array}{l} \text{Find } (\mathbf{x}, \mathbf{u}, t_f)^* \\ \text{minimizing } J = t_f - t_0 \\ \text{subject to, } \forall t \in [t_0, t_f], \\ \text{dynamical constraints : } \dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \\ \text{boundary conditions : } \mathbf{x}(t_0) = \mathbf{x}_0, \mathbf{x}(t_f) = \mathbf{x}_f, \\ \text{control constraint : } \|\mathbf{u}(t)\|_\infty - u_{max} \leq 0 \end{array} \right. \quad (36)$$

where the infinity norm is defined as  $\|\mathbf{u}(t)\|_\infty = \max\{|u_x(t)|, |u_y(t)|, |u_z(t)|\}$ . The reconfiguration manoeuvre must be accomplished in the shortest final time  $t_f^*$ . The reason why the time-optimal control problem is addressed in this paper, rather than the energy or fuel-optimal problems, is that the minimum time requires and exploits much faster dynamics. Hence, to the best of the authors' knowledge, who have already faced the three optimal problems (Spiller et al., 2017b) with metaheuristic algorithms, time-optimal problem is more complex to deal with for this kind of optimization approaches. However, energy and fuel-optimal problems could be faced as well with the proposed methodology by simply changing the cost function.

#### 4.4. Inverse Dynamics Technique

The direct dynamics formulation is usually employed for most of the applications; it consists on obtaining the trajectory via an integration process under an applied control. Instead, the inverse dynamics technique is exploited in this paper. This method can be used only if the external control ( $\mathbf{u}(t)$ ) can be written as a function of the state. In this case, the state vector  $\mathbf{x}$  is approximated by some curves and the optimization algorithm searches for the optimal values of the  $N_P$  coefficients, which describe the curve, of each state component. The inverse dynamics approach overcomes some direct-dynamics issues such as low computational speeds and integration numerical errors and has a reduced number of optimization parameters with respect to traditional methods (collocation and pseudospectral). All these features permit to have a very good compromise between the computational time and the optimality of the results. It is important to note that, because of the approximation of the position vector and its first and second derivatives with the inverse dynamics technique, the solution is defined to be suboptimal. Within the frame of the inverse dynamics technique, a particular subclass is the differential flatness (DF) formulation (Fliess et al., 1995), which is based on the so-called flat output  $\mathbf{y}(t)$  (with the same dimension of the control  $\mathbf{u}(t)$ ). If such a flat output exists, then the state and the control

can be written as functions of  $\mathbf{y}$ ,

$$\mathbf{x} = \mathbf{a}(\mathbf{y}, \dot{\mathbf{y}}, \dots, \mathbf{y}^{(\beta)}), \quad \mathbf{u} = \mathbf{b}(\mathbf{y}, \dot{\mathbf{y}}, \dots, \mathbf{y}^{(\beta+1)}) \quad (37)$$

For our optimization problem, let  $\mathbf{y} = \boldsymbol{\rho}$ . According to this choice,  $\beta = 1$  and the full state  $\mathbf{x}$  is obtained via time differentiation of  $\boldsymbol{\rho}$ , whereas  $\mathbf{u} = \mathbf{b}(\boldsymbol{\rho}, \dot{\boldsymbol{\rho}}, \ddot{\boldsymbol{\rho}})$ . Therefore, the optimization problem, illustrated in Eq. (36), can be rewritten as a function of the flat output as

$$\left\{ \begin{array}{l} \text{Find } (\boldsymbol{\rho}, t_f)^* \\ \text{minimizing } J = t_f - t_0 \\ \text{subject to, } \forall t \in [t_0, t_f], \\ \text{boundary conditions : } \boldsymbol{\rho}(t_0) = \boldsymbol{\rho}_0, \boldsymbol{\rho}(t_f) = \boldsymbol{\rho}_f, \\ \quad \dot{\boldsymbol{\rho}}(t_0) = \dot{\boldsymbol{\rho}}_0, \dot{\boldsymbol{\rho}}(t_f) = \dot{\boldsymbol{\rho}}_f, \\ \text{control constraint : } \|\mathbf{b}(\boldsymbol{\rho}, \dot{\boldsymbol{\rho}}, \ddot{\boldsymbol{\rho}})\|_\infty - u_{max} \leq 0 \end{array} \right. \quad (38)$$

As can be seen, the dynamics constraint no longer appears in the new formulation of the optimization problem, because it is satisfied a priori. In order to make the problem as simple as possible, the approximation curves can be chosen also to directly fulfill the boundary conditions, which are linked to a clever choice of the approximating polynomials coefficients. Thus, calling with  $D_B$  the set of approximating function  $\boldsymbol{\rho}_N(t)$  satisfying the boundary constraints, the problem is reformulated as follows

$$\left\{ \begin{array}{l} \text{Find } (\boldsymbol{\rho}_N, t_f)^* \text{ such that } \boldsymbol{\rho}_N \in D_B \\ \text{minimizing } J = t_f - t_0 \\ \text{subject to, } \forall t \in [t_0, t_f], \\ \text{control constraint : } \|\mathbf{b}(\boldsymbol{\rho}_N, \dot{\boldsymbol{\rho}}_N, \ddot{\boldsymbol{\rho}}_N)\|_\infty - u_{max} \leq 0 \end{array} \right. \quad (39)$$

By employing the DF formulation, three main advantages can be highlighted: 1) the minimum number of optimization parameters is employed, 2) the control policy is obtained in an analytical closed form avoiding the integration of dynamics equations (thus saving also computational time), and 3) the initial and final conditions are automatically respected since they are imposed a priori in the polynomial approximation of  $\boldsymbol{\rho}$ .



#### 4.5. B-spline Curves Approximation

Many interpolating functions can be chosen to approximate the trajectory of the deputy satellite: this choice can strongly affect the results of the optimization problem. Among the most used approximation, there are the Chebyshev polynomials and the B-spline curves. In this paper, B-spline curves are employed, instead of Chebyshev polynomials, because it has already been demonstrated that B-spline curves perform better and result in more accurate results in this kind of optimization problem (Parente et al., 2018). The B-spline  $\mathcal{B}(\lambda; \mathbf{a}, \mathbf{K})$  is characterised by a strictly increasing independent variable  $0 \leq \lambda \leq 1$ , the coefficient vector  $\mathbf{a} = [a_0, a_1, \dots, a_{N_P-1}]$  and the Knot vector  $\mathbf{K}$ , made up of  $m + 1$  components,

$$\mathbf{K} = \{k_n, n = 0, \dots, m | k_0 = 0, k_m = 1, k_n \leq k_{n+1}\}. \quad (40)$$

Here,  $N_P$  is the number of interpolating parameters and  $m = N_P + \mathcal{D}$ , where  $\mathcal{D}$  is the degree of the polynomials. Hence, the approximated relative displacement  $\rho_{N;l}$  and its first and second time derivatives ( $\dot{\rho}_{N;l}$  and  $\ddot{\rho}_{N;l}$ ) can be expressed as

$$\rho_{N;l}(\lambda) = \mathcal{B}(\lambda; \mathbf{a}, \mathbf{K}) = \sum_{n=0}^{N_P-1} a_n \mathcal{N}_{n,\mathcal{D}}(\lambda; \mathbf{K}), \quad (41)$$

$$\dot{\rho}_{N;l}(\lambda) = \dot{\mathcal{B}} = \Delta_\lambda \mathcal{B}' = \Delta_\lambda \sum_{n=0}^{N_P-2} a'_n \mathcal{N}_{n+1,\mathcal{D}-1}(\lambda; \mathbf{K}), \quad (42)$$

$$\ddot{\rho}_{N;l}(\lambda) = \ddot{\mathcal{B}} = \Delta_\lambda^2 \mathcal{B}'' = \Delta_\lambda^2 \sum_{n=0}^{N_P-3} a''_n \mathcal{N}_{n+2,\mathcal{D}-2}(\lambda; \mathbf{K}), \quad (43)$$

In Eqs. (41)-(43),  $\mathcal{N}_{n,\mathcal{D}}(\lambda; \mathbf{K})$  represent the basis functions and are defined through the Cox-de Boor recursion formula (Boor, 1972),  $\Delta_\lambda = t_f^{-1}$ ,  $(\cdot)'$  is the derivative with respect to  $\lambda$  and  $a'_n$  and  $a''_n$  are given by the following finite differences:

$$a'_n = \mathcal{D} \frac{a_{n+1} - a_n}{k_{n+\mathcal{D}} - k_n}, \quad a''_n = (\mathcal{D} - 1) \frac{a'_{n+1} - a'_n}{k_{n+\mathcal{D}-1} - k_n}. \quad (44)$$

To build a B-spline, a polyline defined over the control points  $\mathbf{A}_j = [jt_f/(N_P - 1), a_j]$ ,  $j = 0, \dots, N_P - 1$ , is introduced. In the sequel, *clamped* B-spline will

be employed: in this case, the curve passes through  $\mathbf{A}_0$  and  $\mathbf{A}_{N_P-1}$  and it is obtained using *non-uniform knot points* given by

$$\begin{aligned} k_n &= 0 & \text{if } 0 \leq n \leq \mathcal{D}, \\ k_n &= \frac{n - \mathcal{D}}{N_P - \mathcal{D}} & \text{if } \mathcal{D} < n \leq N_P - 1, \\ k_n &= 1 & \text{if } N_P - 1 < n \leq m. \end{aligned} \quad (45)$$

Initial and final conditions on the  $l^{\text{th}}$  component of the state  $x_l(t)$  are imposed by setting

$$\begin{aligned} a_0 &= x_l(t_0), & a_1 &= a_0 + \frac{t_f}{N_P - 1} \dot{x}_l(t_0), \\ a_{N_P-1} &= x_l(t_f), & a_{N_P-2} &= a_{N_P-1} - \frac{t_f}{N_P - 1} \dot{x}_l(t_f). \end{aligned} \quad (46)$$

Finally, the time can be defined from  $\lambda$  as  $t = t_f \lambda$ .

## 5. Numerical Results

The optimal control problem explained before is applied to the case of a controlled deputy and a non-cooperative chief spacecraft. This section is divided as follows: first, an analysis about the SS circular  $J_2$  relative motion model is performed; secondly, the YA elliptical relative motion model is considered and finally the effectiveness of some techniques introduced in the proposed algorithm is tested through some simulations concerning with the two models.

For all the cases, the initial time instant ( $t_0$ ) is set equal to 0. Moreover, non-dimensional units are used; consequently, distances are divided by  $K_x$ , that is the initial relative distance between the two satellites, and the control is divided by  $K_u = u_{max}$ . The manoeuvre time is normalized by  $K_t = \sqrt{K_x/u_{max}}$ , and velocities are divided by  $K_v = K_x/K_t$ . The maximum value of the thrust has been set to  $u_{max} = 5 \cdot 10^{-4}$  m/s<sup>2</sup>. For what concerns with the B-spline,  $N_P$  is set equal to 8; whereas the trajectory is discretized over  $N_t = 100$  points. The tolerance is computed by means of Eq. (27) with  $N_\delta = 3$  for both the internal and external loops. The objective function is defined as

$$J = \bar{t}_f + \sum_{l=1}^3 \sum_{k=0}^{N_t} \eta_{l,k}(\mathbf{a}_l, t_k) + 100 \cdot N_{viol}$$

$$\eta_{l,k}(\mathbf{a}_l, t_k) = \begin{cases} 0 & \text{if } \frac{|u_{l,k}(\mathbf{a}_l, t_k)|}{u_{max}} \leq 1, \\ \frac{|u_{l,k}(\mathbf{a}_l, t_k)|}{u_{max}} & \text{otherwise.} \end{cases} \quad (47)$$

Note that  $\bar{t}_f$  is the normalized manoeuvre time and  $N_{viol}$  represent the number of violated constraints, which makes the algorithm prefer those CPs that violate a small number of constraints than the others. In particular,  $N_{viol}$  is set equal to 1 if  $u_x$  or  $u_y$  or  $u_z$  exceeds the maximum admitted value  $u_{max}$ , otherwise it is set equal to 0. Using basic B-splines, the vector  $\mathbf{X}_i$ , associated to each CP of IMCSS, is made up of 13 variables. Indeed,  $3(N_P - 4)$  are the parameters needed for the B-spline approximation of the trajectory (along the three dimensions), and the last variable represents the time of the manoeuvre.

Table 1 and Table 2 show respectively the boundaries chosen for the optimization variables and the orbital elements of the chief satellite for both the test cases. For the YA test case, the orbital parameters of the chief satellite are chosen to be equal to those of the Coronagraph spacecraft of the ESA mission Proba 3. This mission consists on a formation of two spacecraft, the Coronagraph and the Occulter, flying in a close proximity in the range [25,250] m. This close distance allows the Coronagraph to study the inner solar corona while the Sun is occulted by the Occulter (Kramer, 2014).

**Table 1:** Boundaries of the optimization variables.

Model	Variables	Boundaries
SS	Coefficients for the displacements	$[-5, 5]$
SS	Manoeuvre time	$[0.25, 4] \cdot T_{orb}/K_t$
YA	Coefficients for the displacements	$[-10, 10]$
YA	Manoeuvre time	$[0.01, 4] \cdot T_{orb}/K_t$

**Table 2:** Orbital parameters of the chief satellites.

Model	$a$ (km)	$e$	$i$ (deg)	$\Omega$ (deg)	$\omega$ (deg)	$\theta_0$ (deg)	$T_{orb}$ (s)
SS	7000	0	45	0	0	0	5828.52
YA	36943	0.8111	59	84	188	170	70665.79

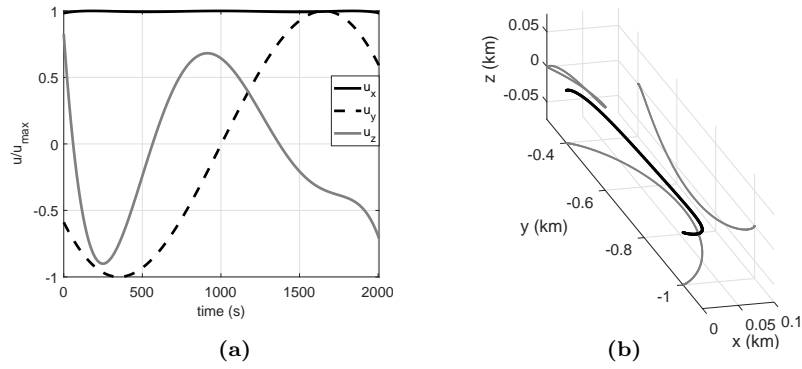
### 5.1. Application to SS Model

Two different reconfiguration manoeuvres are analysed for the case of the SS model.

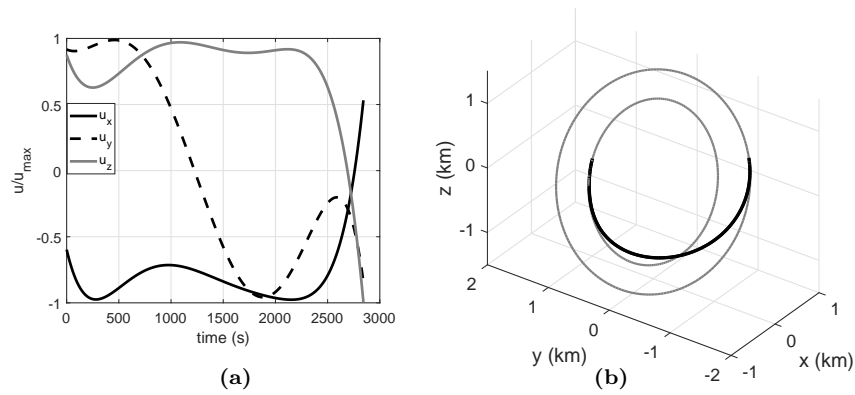
1. ATF-ATF: the deputy starts from  $-0.4$  km and arrives at  $-1$  km.
2. GCF-PCF: the deputy satellite starts from a GCF with  $R = 1.5$  km and an initial phase angle  $\alpha_0$  equal to 150 deg, and arrives at a PCF with  $R = 1$  km. For both the trajectories  $\rho_{y,0} = 0$ .

It is important to notice that if a GCF or a PCF are considered as arrival trajectories, the final condition does not correspond to one point, but to an infinite set of points fulfilling Eq. (30). In that case, another optimization parameter, which is the phase angle  $\alpha_0$  varying within the range  $[0, 2\pi]$ , must be introduced. The results of the ATF-ATF manoeuvre are shown in Fig. 3. The normalization parameters for this first case are:  $K_x = 0.4$  km,  $K_t = 894.43$  s and  $K_v = 4.4721 \cdot 10^{-4}$  km/s. As it can be seen, the dimensionless control  $u_x$  is quite constant and equal to  $+1$ , whereas the control  $u_y$  basically switches from  $-1$  to  $+1$ . This behaviour can be thought as it was an ideal two impulsive manoeuvre: a first impulse, which makes the deputy leave its starting position, and a second one, which is needed to stop the deputy when it reaches the final position. The value of the computed optimal manoeuvre time, which is equal to 2005.92 s, is consistent with the value found in literature (about 2000 s in (Parente et al., 2018)), thus proving the reliability of the proposed algorithm. Regarding the GCF-PCF manoeuvre, the normalization parameters are:  $K_x = 1.3521$  km,  $K_t = 1644.43$  s and  $K_v = 8.2222 \cdot 10^{-4}$  km/s. The results of the optimization are reported in Fig. 4. In this case,  $u_x$  and  $u_z$  present quite a constant control

close to  $-1$  and  $+1$  respectively (for most of the manoeuvre duration), while  $u_y$  basically switches from  $+1$  to  $-1$ . The computed optimal manoeuvre time is  $2842.85$  s. It has to be noticed that extremal controls are a good evidence about the effectiveness of the optimization procedure, since the control is a good approximation of the bang-bang manoeuvre, which is shown to be the analytical optimum by the Pontryagin Maximum Principle (PMP) applied to the Hamiltonian written for the SS model (Kirk, 2012).



**Figure 3:** Normalized control policy (a) and trajectory (b) for ATF-ATF manoeuvre.

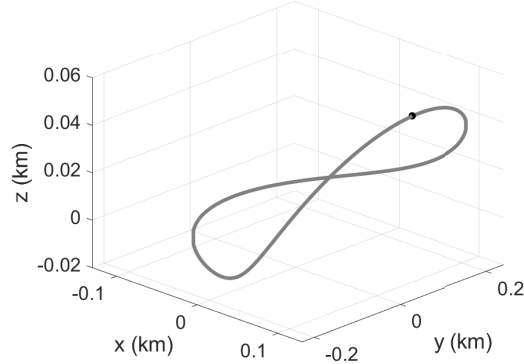


**Figure 4:** Normalized control policy (a) and trajectory (b) for GCF-PCF manoeuvre.

## 5.2. Application to YA Model

In this Section, the elliptical relative motion is considered. In order to perform the tests for useful and interesting orbits, the orbit of the chief satellite, chosen as the reference orbit for the formation, is assumed to have the orbital parameters of the Coronagraph spacecraft of the Proba 3 mission. As it can be seen from Table 2, the reference orbit is eccentric with a high apogee. This choice allows to minimize the propellant consumption of the formation flying manoeuvres. The orbit has a period of 19.63 hour and the formation flying will take place over a 12 hours arc centered near the apogee, allowing a minimum of 6 hours/day of coronal observations (Kramer, 2014). For this reason, the initial true anomaly is arbitrarily set equal to 170 deg. This indicates that the formation is flying towards the apogee, which is where most of the operations will be performed.

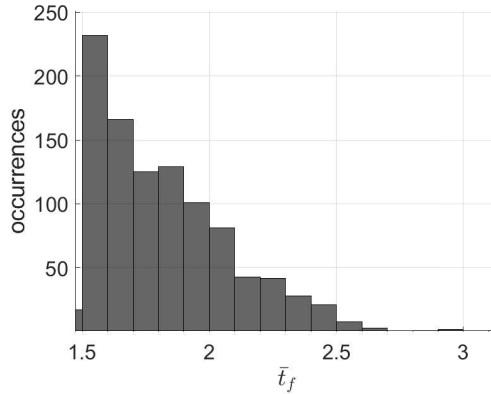
The goal of this optimization is to reach a closed relative trajectory, generated with the condition expressed by Eq. (33). Thanks to the closed relative trajectory, the deputy can orbit around the chief and the operations for the occultation can be prepared accurately. As already mentioned, since the two satellites have a relative distance varying within the range [25,250] m, the arrival closed trajectory is chosen to be generated by the following relative state:  $\mathbf{x}_{0,ref} = [0.1 \text{ km}, 0.1 \text{ km}, 0.05 \text{ km}, 0 \text{ km/s}, -1.0149 \cdot 10^{-5} \text{ km/s}, 0 \text{ km/s}]^T$ , where the norm of the relative position vector ( $\rho$ ) is set equal to 150 m and  $\dot{\rho}_y$  is computed by means of Eq. (33). The reference closed relative trajectory, shown in Fig. 5, is then obtained by propagating  $\mathbf{x}_{0,ref}$  with the free motion state transition matrix  $\Phi(\theta, \mathbf{x}_{0,ref})$ . This allows to determine the relative position and velocity that the deputy satellite should have at each time instant to remain along this orbit without using any further control. Therefore, the deputy satellite must match the proper final relative state ( $\mathbf{x}_f$ ) during the optimization, taking into account the time required for the manoeuvre ( $t_f$ ). Thus, the required final condition can be expressed as a function of the manoeuvre time through  $\mathbf{x}_f = \Phi(\theta_f, \mathbf{x}_{0,ref}) \cdot \mathbf{x}_{0,ref}$ , where  $\theta_f$  is the true anomaly of the chief corresponding to  $t_f$ .



**Figure 5:** Reference closed relative trajectory.

The first analysis performed on the YA model is presented below. The initial relative state ( $\mathbf{x}_0$ ) is chosen to be equal to  $\mathbf{x}_0 = [0.13826 \text{ km}, 0.43803 \text{ km}, 0.46379 \text{ km}, -4.6002 \cdot 10^{-5} \text{ km/s}, -8.7233 \cdot 10^{-5} \text{ km/s}, -8.8844 \cdot 10^{-5} \text{ km/s}]$ . The normalization parameters are:  $K_x = 0.65276 \text{ km}$ ,  $K_t = 1142.59 \text{ s}$  and  $K_v = 5.71295 \cdot 10^{-4} \text{ km/s}$ . In order to test the proposed optimization algorithm, a Monte Carlo analysis consisting of 1000 simulations has been carried out by keeping the initial relative state fixed. The results are shown in the histogram of Fig. 6, which displays the occurrences of the computed optimal normalized manoeuvre time ( $\bar{t}_f$ ). As can be noticed, most of the simulations is able to reach low manoeuvre times; in particular, 77% of cases provides a value lower than 2 (corresponding to 2285.18 s). Taking into account that the bin edges are about 0.1 wide (which corresponds to 114.26 s), the difference in time between the best and the worst case is about 30 minutes. However, the histogram shows that the highest times only occur with a very low frequency, thus proving the reliability of IMCSS. Among all the simulations, the results for the best case are illustrated and commented.

Indeed, Fig. 7 shows the optimal trajectory to reach the arrival closed trajectory and the corresponding control policy for the best case. As can be seen, the dimensionless control  $u_y$  and  $u_z$  switch from -1 to +1 after a short

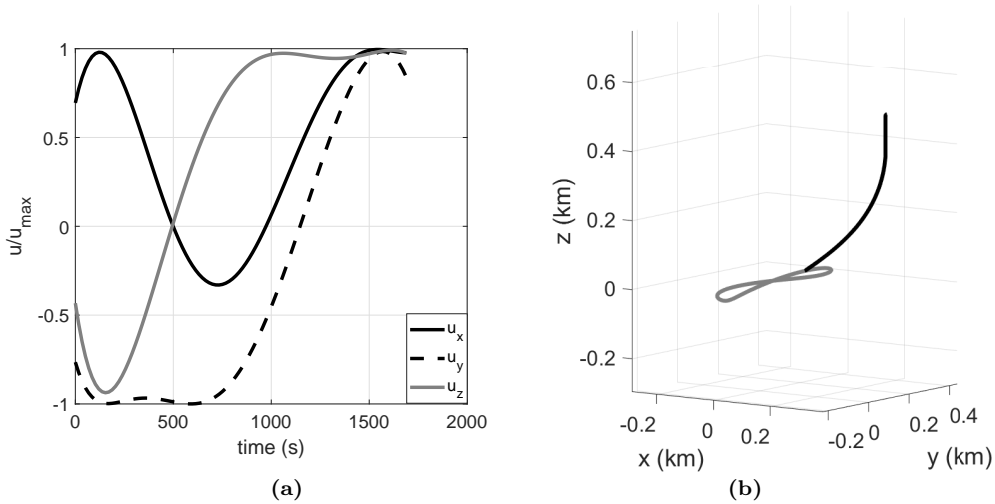


**Figure 6:** Monte Carlo over 1000 simulations by fixing the initial relative state.

transient needed to reach the value  $-1$ . For what concerns with  $u_x$ , it basically represents a bang-off-bang manoeuvre, since it switches from  $+1$  to a little negative value and then comes back to  $+1$ . The point where the reference closed relative trajectory is reached is depicted as a black point in Fig. 5. The computed optimal manoeuvre time is equal to  $1689.07$  s. During this time, the chief satellite has traveled about  $1.71$  deg in true anomaly. Thus, the final configuration is obtained before reaching the apogee and the formation would ideally be ready to perform the operations close to the apogee, as it is planned for the Proba 3 mission. Even for this analysis, the presence of switching between extremals controls is a good evidence of the good reliability of the algorithm.

In addition, a new Monte Carlo simulation, made up of 2000 tests, is performed by varying the initial condition. The initial relative state ( $\mathbf{x}_0$ ) is chosen such that the relative position is obtained randomly among all the points lying on a sphere centered at the origin of the reference frame and having a radius varying from  $0.05$  to  $0.5$  km,  $\dot{\rho}_{x,0}$  and  $\dot{\rho}_{z,0}$  are determined randomly within the range  $[-0.1, 0.1]$  m/s, whereas  $\dot{\rho}_{y,0}$  is computed in order to satisfy Eq. (33). The results of the analysis are summarized in Fig. 8, where the normalized optimal manoeuvre times are plotted as functions of the initial position and velocity norms. In this analysis, the algorithm has been able to find a feasi-





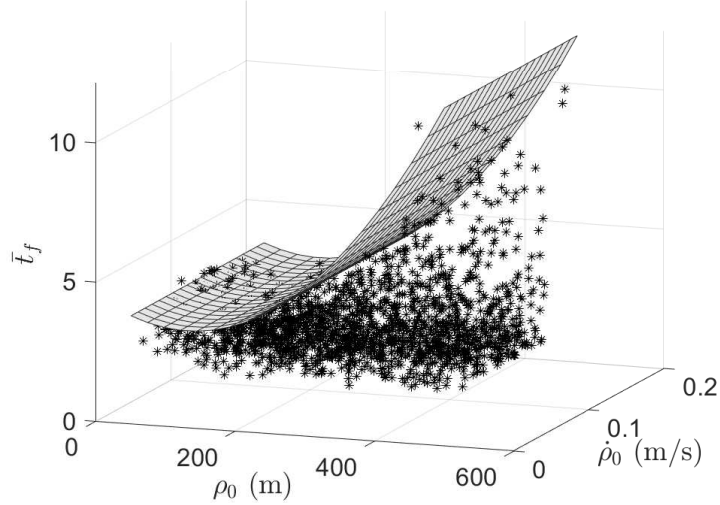
**Figure 7:** Normalized control policy (a) and trajectory (b) for the case of elliptical motion.

ble optimal solution for each test. As can be seen from Fig. 8, a surface has been found to well represent the upper boundary of the distribution of the cost functions. This surface has been generated by considering the following 2D parabola:  $y = 63.20x^2 - 16.38x + 4.54$ . It is found that only 1.61% of the cost functions lie above that surface. Moreover, taking into account this feature, the initial positions seem to play a major role with respect to the velocities with the imposed defined ranges. In fact, one can assume that for the upper boundary of this analysis  $\frac{\partial \bar{\ell}_f}{\partial \rho_0}$  is 0 and that  $\frac{\partial \bar{\ell}_f}{\partial \rho_0}$  increases quite linearly, especially for greater initial distances. This last analysis shows the ability of the algorithm to find solutions also if the initial conditions vary.

### 5.3. IMCSS Techniques Demonstration

This Section aims at investigating the effectiveness of some tools introduced in the proposed algorithm, such as the boundary handling technique and the chaotic local search.

To demonstrate the boundary handling technique, the YA model is considered (see Table 2 for the reference orbit parameters). In particular, the final reference closed trajectory to reach is the same as the one proposed in



**Figure 8:** Monte Carlo over 2000 simulations by varying the initial relative state.

the previous Section, whereas the new initial condition is chosen to be the following:  $\mathbf{x}_0 = [0.26087 \text{ km}, 0.11367 \text{ km}, 0.52075 \text{ km}, 8.26752 \cdot 10^{-5} \text{ km/s}, 2.64718 \cdot 10^{-5} \text{ km/s}, -8.04919 \cdot 10^{-5} \text{ km/s}]$ . The corresponding normalization parameters are:  $K_x = 0.593423 \text{ km}$ ,  $K_t = 1089.42424 \text{ s}$  and  $K_v = 5.44712 \cdot 10^{-4} \text{ km/s}$ . In order to test the boundary handling technique, the boundaries of the coefficients for the displacements are shortened with respect to those of Table 1; thus, let the lower and upper bounds of the coefficients be equal to  $[-1,1]$ . If the boundary handling technique is not employed, the optimization fails to find an optimal value for the manoeuvre since the constraints are always violated; in this case, the final value of the cost function is 113.632. On the other hand, if the boundary handling technique is used, the optimization process can be successful. Indeed, Table 3 shows the results of the optimization, in which only the counters for the boundaries of the displacements coefficients that are actually enlarged ( $c_{UB;2,9}$  and  $c_{LB;2,9}$ ) are reported, whereas the other boundaries remain the same as those imposed by the user at the beginning of the procedure and do not change during the optimization. As can be seen from the results, in the first external loop the objective function does not satisfy the control con-

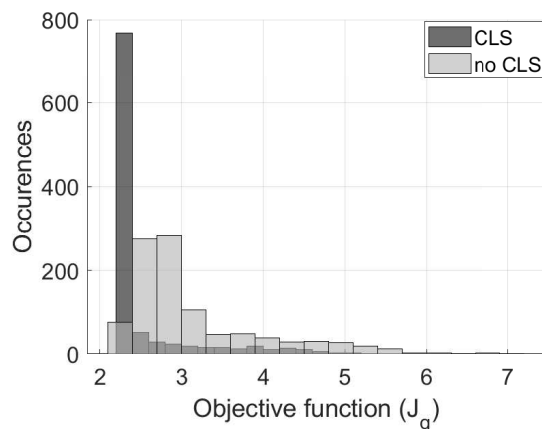
straints, but the maximum counter  $C_{max}$  is already exceeded by  $c_{LB;2}$ . This causes the corresponding lower boundary to be enlarged. At this point, the algorithm can continue to improve the objective function, but the control constraints are still not fulfilled in the second and third external loops (in fact, the objective function does not improve significantly). At the same time, there is the trend of  $c_{LB;9}$  to increase until it exceeds  $C_{max}$  in the fourth loop and the corresponding lower bound is enlarged. From this point on, the algorithm is able to find an optimum value fulfilling the control constraints. The objective function at the end of the optimization is 3.31247, which corresponds to a manoeuvre time of 3608.69 s. As can be noticed, this tool allows the algorithm to find an optimum even if the user has badly defined the boundaries. This technique is particularly useful when the optimization variables do not directly convey an evident physical meaning, as it is for the present case in which the coefficients of the B-splines, chosen as the optimization variables, do not allow the user to have immediately an idea of the physics behind the problem. Thus, in these cases, it is more likely for the user to badly define the boundaries.

**Table 3:** Results of the boundary handling technique for the YA model.

$G_K$	$G_{k,f}$	$J_g$	$\delta$	$C_{max}$	$c_{UB;2,9}$	$c_{LB;2,9}$	$UB_{2,9}$	$LB_{2,9}$
1	450	116.00166	1.4644E-05	<b>2250</b>	[141,112]	<b>[2353,1979]</b>	[1,1]	[-1,-1]
2	450	111.89505	1.07269E-03	2250	[90,102]	[87,1567]	[1,1]	<b>[-10,-1]</b>
3	450	110.10763	1.34945E-05	2250	[159,130]	[162,1907]	[1,1]	[-10,-1]
4	450	110.08565	1.27783E-05	<b>2250</b>	[123,112]	[66, <b>2950]</b>	[1,1]	[-10,-1]
5	450	3.61685	5.89808E-04	2250	[115,123]	[143,60]	[1,1]	[-10,- <b>10]</b>
6	450	3.50710	2.38434E-04	2250	[99,99]	[143,80]	[1,1]	[-10,-10]
7	450	3.46292	5.53854E-04	2250	[91,116]	[141,73]	[1,1]	[-10,-10]
8	2250	3.31247	1.21952E-03	2250	[51,67]	[110,38]	[1,1]	[-10,-10]

A second analysis is carried out to test the effectiveness of the chaotic local search introduced in the proposed algorithm. In this case, the same ATF-ATF manoeuvre reported in Section 4.1 (with the same initial and final conditions) is considered. In particular, two Monte Carlo analysis, based both on 1000 runs, are performed considering first the presence of the CLS and then its ab-

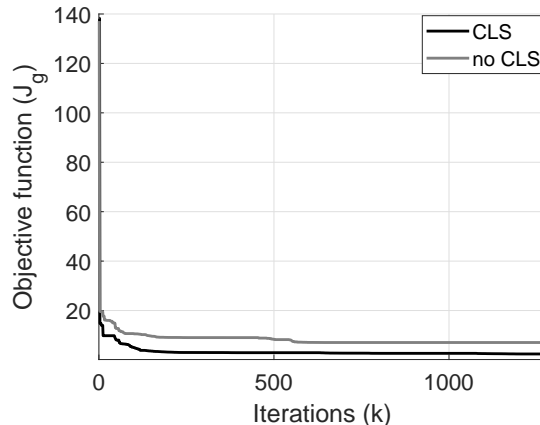
sence. The same random numbers are generated so that the tests can actually be compared. The analysis show that for most of the cases (86.3 % ) the objective functions obtained with the CLS are better than the corresponding values obtained without using the CLS. Indeed, this result can easily be noticed in Figure 9, which illustrates the occurrences of the dimensionless objective functions reached at the end of each optimization. It is clear that the algorithm with CLS can achieve better performances, since better objective functions are reached for most of the cases. In particular, the history of the objective functions are reported for one of the analyzed cases in Figure 10. It can be seen that at the end of the optimization the algorithm without CLS reaches a value equal to 7.06, whereas the introduction of the CLS leads the algorithm to reach a value equal to 2.25. Therefore, it can be concluded that the chaotic local search can actually speed up and improve the convergence of the algorithm.



**Figure 9:** Performance comparison of IMCSS using or not the CLS.

## 6. Conclusions

A self-adaptive and improved version of the metaheuristic Magnetic Charged System Search algorithm (called IMCSS) is proposed in this work. It presents some advantages with respect to the original algorithm in order to better ensure the convergence to the optimum. Indeed, the proposed algorithm is able



**Figure 10:** Performance comparison of IMCSS using or not the CLS for a test case.

to automatically (and dynamically) determine its internal parameters and coefficients, making it user-friendly. Moreover, the chaotic local search and the boundary handling technique have been shown to succeed in improving and speeding up the convergence. IMCSS is applied to optimize two different cases about formation flying reconfiguration manoeuvres, considering the inverse dynamics technique and the B-spline curves, . First, the case of a reference circular trajectory under the  $J_2$  perturbation of the Earth is considered. The computed optimal manoeuvre times and trajectories show very good results, thanks to the presence of extremals controls, which are very close to the ideal optimal manoeuvres (bang-bang) computed analytically. Moreover, a comparison with the literature is carried out, showing that the manoeuvre time is consistent with the value obtained in another work. Secondly, the case of an elliptical reference trajectory is studied with the aim for the deputy to reach a closed relative orbit around the chief. The parameter related to the formation of the Proba 3 mission are taken into account as an example. In particular, two Monte Carlo simulations have been performed to show the reliability of the algorithm, one with fixed initial condition (to compare the good convergence of IMCSS with the best optimal solutions) and another one with varying initial conditions (to test the reliability of IMCSS for general cases). Even for this analysis, the

optimization procedure provides good results. All the tests performed in this work show the effectiveness and reliability of the proposed algorithm to solve space guidance problems with particular regard to formation flying reconfiguration manoeuvres. The adopted optimization strategy also represents a very good compromise between the computational efforts and the optimality of the results. In the future, this algorithm will be tested for other space guidance and control problems, such as for real time trajectory planning and autonomous landing missions.

## References

- Bandaru, S., & Deb, K. (2016). *Metaheuristic Techniques*. Published in Decision Sciences: Theory and Practice, 693-750, CRC Press, Taylor & Francis Group.
- Boor, C. D. (1972). On calculating with b-splines. *Journal of approximation theory*, 6, 50–62. doi:[10.1016/0021-9045\(72\)90080-9](https://doi.org/10.1016/0021-9045(72)90080-9).
- Boyarko, G., Yakimenko, O., & Romano, M. (2011). Optimal rendezvous trajectories of a controlled spacecraft and a tumbling object. *Journal of Guidance, Control, and Dynamics*, 34, 1239–1252. doi:[10.2514/1.47645](https://doi.org/10.2514/1.47645).
- Broucke, R. A. (2003). Solution of the elliptic rendezvous problem with the time as independent variable. *Journal of Guidance, Control, and Dynamics*, 26, 615–621.
- Cao, L., & Misra, A. K. (2015). Linearized j2 and atmospheric drag model for satellite relative motion with small eccentricity. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 229, 2718–2736.
- Clerc, M. (2010). *Particle swarm optimization*. London, UK: ISTE Ltd.
- Clohessy, W. (1960). Terminal guidance system for satellite rendezvous. *Journal of the Aerospace Sciences*, 27, 653–658.

- Dorigo, M., Maniezzo, V., & Colorni, A. (1996). Ant system: optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, *26*, 29–41.
- Fliess, M., Lévine, J., Martin, P., & Rouchon, P. (1995). Flatness and defect of non-linear systems: Introductory theory and examples. *International Journal of Control*, *61*, 1327–1361. doi:[10.1080/00207179508921959](https://doi.org/10.1080/00207179508921959).
- Geem, Z. W. (2009). *Music-inspired harmony search algorithm: theory and applications*. Berlin Heidelberg: Springer.
- Goldberg, D. (1989). *Genetic algorithms in search, optimization, and machine learning*. Reading, Massachusetts, USA: Addison-Wesley.
- Guo, Z., Huang, H., Deng, C., Yue, X., & Wu, Z. (2015). An enhanced differential evolution with elite chaotic local search. *Computational intelligence and neuroscience*, *2015*, 6.
- Hamel, J.-F., & Lafontaine, J. D. (2007). Linearized dynamics of formation flying spacecraft on a  $J_2$ -perturbed elliptical orbit. *Journal of Guidance, Control, and Dynamics*, *30*, 1649–1658.
- Holland, J. (1975). *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor, MI.
- Huang, H., Ma, G., Zhuang, Y., & Lv, Y. (2012). Optimal spacecraft formation reconfiguration with collision avoidance using particle swarm optimization. *Information Technology and Control*, *41*, 143–150.
- Jia, D., Zheng, G., & Khan, M. K. (2011). An effective memetic differential evolution algorithm based on chaotic local search. *Information Sciences*, *181*, 3175–3187.
- Junkai, J., Shangce, G., Shuaiqun, W., Yajiao, T., Y, H., & Yuki, T. (2017). Self-adaptive gravitational search algorithm with a modified chaotic local search. *IEEE Access*, *5*, 17881–17895.

- Kaveh, A., Share, M. A. M., & Moslehi, M. (2013). Magnetic charged system search: a new meta-heuristic algorithm for optimization. *Acta Mechanica*, *224*, 85–107.
- Kaveh, A., & Talatahari, S. (2009). Particle swarm optimizer, ant colony strategy and harmony search scheme hybridized for optimization of truss structures. *Computers and Structures*, *87*, 267–283. doi:[10.1016/j.compstruc.2009.01.003](https://doi.org/10.1016/j.compstruc.2009.01.003).
- Kaveh, A., & Talatahari, S. (2010). A novel heuristic optimization method: charged system search. *Acta Mechanica*, *213*, 267–289.
- Kim, Y., & Spencer, D. (2002). Optimal spacecraft rendezvous using genetic algorithms. *Journal of Spacecrafts and Rockets*, *39*, 859–865. doi:[10.2514/2.3908](https://doi.org/10.2514/2.3908).
- Kirk, D. E. (2012). *Optimal Control Theory: An Introduction*. Dover Books on Electrical Engineering. Dover Publications.
- Kramer, H. J. (2014). Proba-3 (project for on-board autonomy-3). <https://earth.esa.int/web/eoportal/satellite-missions/p/proba-3>. Accessed: 2019-05-14.
- Kuiack, B., & Ulrich, S. (2018). Nonlinear analytical equations of relative motion on  $j$  2-perturbed eccentric orbits. *Journal of Guidance, Control, and Dynamics*, *41*, 2666–2677.
- Lee, K., & Geem, Z. (2004). A new structural optimization method based on the harmony search algorithm. *Computers and Structures*, *82*, 781–798. doi:[10.1016/j.compstruc.2004.01.002](https://doi.org/10.1016/j.compstruc.2004.01.002).
- Louembet, C. (2007). Design of algorithms for satellite slew manoeuver by flatness and collocation. *Proceedings of the American Control Conference, IEEE Publ., Piscataway, NJ*, (pp. 3168–3173). doi:[10.1109/ACC.2007.4282459](https://doi.org/10.1109/ACC.2007.4282459).



- Melton, R. G. (2000). Time-explicit representation of relative motion between elliptical orbits. *Journal of Guidance, Control, and Dynamics*, *23*, 604–610.
- Parente, D., Spiller, D., & Curti, F. (2018). Time-suboptimal satellite formation maneuvers using inverse dynamics and differential evolution. *Journal of Guidance, Control, and Dynamics*, *41*, 1108–1121.
- Pontani, M., & Conway, B. (2013). Optimal finite-thrust rendezvous trajectories found via particle swarm algorithm. *Journal of Spacecrafts and Rockets*, *50*, 1222–1234. doi:[10.2514/1.A32402](https://doi.org/10.2514/1.A32402).
- Rashedi, E., Nezamabadi-Pour, H., & Saryazdi, S. (2009). Gsa: A gravitational search algorithm. *Information Science*, *179*, 2232–2248.
- Ross, I., & Fahroo, F. (2002). A perspective on methods for trajectory optimization. *Proceedings of the AIAA/AAS Astrodynamics Specialist Conference and Exhibit, Guidance, Navigation, and Control and Co-Located Conferences, AIAA Paper 2002-4727*, . doi:[10.2514/6.2002-4727](https://doi.org/10.2514/6.2002-4727).
- Schaub, H. (2004). Relative orbit geometry through classical orbit element differences. *Journal of Guidance, Control, and Dynamics*, *27*, 839–848.
- Schaub, H., & Alfriend, K. T. (2001). Impulsive feedback control to establish specific mean orbit elements of spacecraft formations. *Journal of Guidance, Control, and Dynamics*, *24*, 739–745.
- Schweighart, S., & Sedwick, R. (2001). A perturbative analysis of geopotential disturbances for satellite cluster formation flying. *Proceedings of the 2001 IEEE Aerospace Conference*, *2*, 1001–1019. doi:[10.1109/AERO.2001.931281](https://doi.org/10.1109/AERO.2001.931281).
- Schweighart, S., & Sedwick, R. (2002). High-fidelity linearized j2 model for satellite formation flight. *Journal of Guidance, Control, and Dynamics*, *25*, 1073–1080. doi:[10.2514/2.4986](https://doi.org/10.2514/2.4986).
- Sengupta, P., & Vadali, S. R. (2007). Relative motion and the geometry of formations in keplerian elliptic orbits with arbitrary eccentricity. *Journal of guidance, control, and dynamics*, *30*, 953–964.

- Spiller, D., Ansalone, L., & Curti, F. (2016). Particle swarm optimization for time-optimal spacecraft reorientation with keep-out cones. *Journal of Guidance, Control, and Dynamics*, *39*, 312–325. doi:[10.2514/1.G001228](https://doi.org/10.2514/1.G001228).
- Spiller, D., Basu, K., Curti, F., & Circi, C. (2018a). On the optimal passive formation reconfiguration by using attitude control. *Acta Astronautica*, *153*, 259–273.
- Spiller, D., & Curti, F. (2015). Inverse dynamics particle swarm optimization for nanosatellites rendezvous via differential drag. *3rd IAA Conference on University Satellite Missions and CubeSat Workshop and International Workshop on Lean Satellite Standardization, International Academy of Astronautics (IAA) Paper IAA-CU-15-01-58*.
- Spiller, D., Curti, F., & Circi, C. (2017a). Minimum-time reconfiguration maneuvers of satellite formations using perturbation forces. *Journal of Guidance, Control, and Dynamics*, *40*, 1130–1143.
- Spiller, D., Melton, R. G., & Curti, F. (2017b). Inverse dynamics particle swarm optimization applied to bolza problems. In *AAS/AIAA Astrodynamics Specialist Conference, Stevenson, Washington, US*.
- Spiller, D., Melton, R. G., & Curti, F. (2018b). Inverse dynamics particle swarm optimization applied to constrained minimum-time maneuvers using reaction wheels. *Aerospace Science and Technology*, *75*, 1–12. doi:[10.1016/j.ast.2017.12.038](https://doi.org/10.1016/j.ast.2017.12.038).
- Storn, R., & Price, K. (1997). Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, *11*, 341–359.
- Wang, S., & Zheng, C. (2012). A hierarchical evolutionary trajectory planner for spacecraft formation reconfiguration. *IEEE Aerospace and Electronic Systems Magazine*, *48*, 279–289. doi:[10.1109/TAES.2012.6129635](https://doi.org/10.1109/TAES.2012.6129635).

- Xiang, W., & Jørgensen, J. (2005). Formation flying: a subject being fast unfolding in space. In *5th IAA Symposium on Small Satellites for Earth Observation*.
- Yamanaka, K., & Ankersen, F. (2002). New state transition matrix for relative motion on an arbitrary elliptical orbit. *Journal of guidance, control, and dynamics*, *25*, 60–66.
- Yang, X.-S. (2018). *Optimization techniques and applications with examples*. John Wiley & Sons.
- Zhang, D., Song, S., & Duan, G. (2008). Fuel and time optimal transfer of spacecrafts rendezvous using lamberts theorem and improved genetic algorithm. *Proceedings of the 2008 2nd International Symposium on Systems and Control in Aerospace and Astronautics, ISSCAA 2008, IEEE Publ., Piscataway, NJ*, (pp. 1–6). doi:[10.1109/ISSCAA.2008.4776390](https://doi.org/10.1109/ISSCAA.2008.4776390).