Cornel Sultan
United Technologies Research Center,
East Hartford, CT, 06108, USA
cornel_sultan@yahoo.com

Sanjeev Seereram
Raman K. Mehra
Scientific Systems Company Inc.,
Woburn, MA, 01801, USA
sanjeev@ssci.com and rkm@ssci.com

Deep Space Formation Flying Spacecraft Path Planning

Abstract

Efficient algorithms for collision-free energy sub-optimal path planning for formations of spacecraft flying in deep space are presented. The idea is to introduce a set of way-points through which the spacecraft are required to pass, combined with parameterizations of the trajectories which are energy-optimal for each spacecraft. The resulting constrained optimization problem is formulated as a quasi-quadratic parameter optimization problem in terms of the way-points parameters. The mathematical structure of the problem is further exploited to develop gradient-based algorithms in which the gradients are computed analytically. The collision avoidance constraints are approximated such that closed form solutions are generated. This combination results in fast and robust numerical algorithms which work very well for scenarios involving a large number of spacecraft (e.g. 20).

KEY WORDS—path planning for multiple mobile robot systems, formation flying spacecraft, trajectory generation

Notation

a_l	acceleration vector of spacecraft l
a_{l_k}	component of acceleration vector of spacecraft l on axis k ($k = x, y, z$)
A_{l_k}	upper limit on the absolute value of a_{l_k}
c_j, d_j	coefficients for the energy optimal trajectory of one spacecraft
d_{lm}	distance between spacecraft l and m
d_{lm*}	global minimum of the distance between spacecraft l and m with respect to ζ
$d_{\min}$	minimum distance
d_+	distance evaluated at x_+
g	direction of movement in the DIG and DJ algorithms
H	Hessian of the penalty function
J_l	energy of spacecraft l
J_μ	μ energy of the formation
n	direction of movement in the JG algorithm
N	number of spacecraft
$P(x)$, P	penalty function
r_l	position vector of spacecraft l
$r(t)$	position vector
r_{l_0}	initial position vector of spacecraft l
r_{l_T}	final position vector of spacecraft l
R	forbidden sphere radius
R_l	forbidden sphere radius of spacecraft l
s	step size in the JG algorithm
t	time
t_j	time of the jth way-point
T	duration of the maneuver
v_j	velocity vector at the j-th way-point
v_l	velocity vector of spacecraft l
v_{l_0}	initial velocity vector of spacecraft l
v_{l_j}	velocity vector of spacecraft l at the j-th way-point
v_{l_T}	final velocity vector of spacecraft l
w_j	position vector of the jth way-point
x	vector of optimization variables
x_t	tentative value of x
x_+	predicted value of x at the next step
α	parameter of the line search used in DIG and DJ algorithms
δx	maximum allowed variation in x
∇J_μ	gradient of J_μ
∇P	gradient of $P(x)$

The International Journal of Robotics Research
Vol. 26, No. 4, April 2007, pp. 405–430
DOI: 10.1177/0278364907076709

Figures 2–19 appear in color online: http://ijr.sagepub.com

μ_l	energy weight for spacecraft l
ς	dimensionless time
ς_c	dimensionless time at which $d_{\min}$ is attained
ς_{lm*}	dimensionless time at which d_{lm*} is attained
ς_{l_j}	dimensionless time of spacecraft l at the j-th waypoint

1. Introduction

1.1. History and Motivation

In the late 1960s data from the European Space Research Organization, USA, and USSR satellites were correlated to study how large solar flares interact with the Earth, thereby achieving the first contemporaneous spatial sampling by a group of separated spacecraft (Manno and Page 1969). About a decade later Labeyrie proposed forming a stellar interferometer from free-flying telescopes (Labeyrie 1978). Today there are several missions that use the formation flying spacecraft concept (Bristow et al. 2000), which refers to a set of spatially distributed spacecraft capable of interacting and cooperating with one another. A formation of flying spacecraft is aimed at achieving the functionality of a very large spacecraft with multiple small spacecraft or at carrying out missions which cannot be completed by a single spacecraft. The benefits of this new concept include flexible mission capabilities achieved through the reconfiguration of the formation, lower life-cycle cost, more adaptability to changing mission goals and less susceptibility to the loss of individual spacecraft: if a spacecraft is malfunctioning it can easily be replaced or the formation can be reconfigured to compensate for the loss (Inalhan et al. 2002).

Formation flying spacecraft missions can be categorized as planetary orbital environment flying missions, where spacecraft have significant orbital dynamics, and deep-space missions, where spacecraft dynamics usually reduces to double integrators (Scharf et al. 2002). The first category includes missions like TechSat21 (Das and Cobb 1998), Earth-Observing-1 (Folta and Quinn 1997), whereas the latter category, which is of interest to this article, includes long-baseline infrared interferometry missions like the Terrestrial Planet Finder (TPF; Beichman 1998), Darwin (Fridlund 2000; Li and Williams 2004), etc. These missions will use formation flying extensively for time-varying gravity field measurements, *in situ* magnetosphere and radiation measurements and 3-D mapping for planetary explorers, among others. For example, by combining the high sensitivity of space telescopes with novel imaging technologies, TPF will measure the size, temperature, and placement of planets as small as the Earth in distant solar systems. TPF will use formation flying spacecraft to synthesize a large baseline interferometer operating in the infrared. The present TPF concept assumes four 3.5 m diameter telescopes, each on its own spacecraft, and a central spacecraft that houses the beam combining apparatus and astronomical instrumentation (Beichman 1998). An alternative to TPF is the Darwin mission, which assumes that a larger number of observatory-spacecraft (six in the current scenario) are equidistantly distributed on the circumference of a circle which is centered at a seventh spacecraft, the combiner, in charge of collecting and processing the information from the observatory-spacecraft. An additional spacecraft, the master, will be positioned out of the plane of the seven spacecraft and will be in charge of communicating with the Earth as well as monitoring the relative position of the spacecraft (Li and Williams 2004).

Another exciting example of a formation flying spacecraft application is the next generation optical space telescope. It is well known that conventional space telescopes with a monolithic principal mirror (e.g. Hubble Space Telescope) are upper bounded in terms of the principal mirror diameter and control system authority, resulting in limitations on the light/information gathering capability (Zhu et al. 1995). By attaching mirrors to individual spacecraft a much larger aperture can be synthesized, hence considerably increasing the scientific return of a distributed space telescope. In this context formation flying spacecraft control becomes crucial and can be used to modify the optical characteristics of the space telescope (e.g. the focal distance, the diameter of the virtual mirror, its shape, etc.). By sharing the individual measurements, the resolution of the spacecraft formation is potentially much higher than the resolution of any individual spacecraft. Additionally the individual information can be fused into much more reliable, fault tolerant and redundant information than if from a single spacecraft. A distributed space telescope requires precise, coordinated control of a larger number of spacecraft than in the TPF (5) or Darwin (6–8 spacecraft) missions.

The new technology of formation flying spacecraft brings along numerous challenges into the field of guidance, navigation, control, and communication (Mesbahi and Hadaegh 2001; Tillerson et al. 2002). Formation flying guidance and control requires autonomous fleet reconfiguration for which a path planner is needed to compute spacecraft maneuvers. The most important requirement is that the path planner guarantees collision-free trajectories; eventually some performance index (fuel, energy, maneuver time, etc.) is optimized (see Scharf et al. 2002 for a comprehensive survey of formation flying guidance). Since the size of future formation flying missions is expected to increase continuously, path planners capable of handling large scale formations and a large number of conflicting trajectories are desired. Another requirement is that these planning algorithms should be computationally simple enough for onboard implementation and be able to compute collision-free trajectories very fast (ideally in real-time).

This article addresses these issues by presenting path planning algorithms which are fast and robust even in the case of large scale formations flying in deep space, as illustrated by the examples considered. In the following we present a quick overview of related work along with their advantages and deficiencies with respect to the above mentioned requirements.

1.2. Previous and Related Work

Collision avoidance and trajectory generation problems have been the subject of extensive research in formation flying spacecraft (Seereram et al. 2000; Singh and Hadaegh 2001; Beard and McLain 2001; Mesbahi and Hadaegh 2001; Richards et al. 2002; Prasanth et al. 2002; McQuade et al. 2002; Rathbun et al. 2002; Tillerson et al. 2002; Inalhan et al. 2002; Kim et al. 2003; Phillips et al. 2003), air traffic control (Tomlin et al. 1998; Faiz et al. 2001; Frazzoli et al. 2001; Hu et al. 2002; Clements 2002; Richards and How 2002), robotics (Kavraki et al. 1996; Barraquand et al. 1997; Lygeros et al. 1998; Kavraki et al. 1998; Hsu et al. 1999; Hsu et al. 2000; Foskey et al. 2001; Garber and Lin 2002; Dunbar and Murray 2002; Frazzoli et al. 2002; Saber et al. 2003; Cerven et al. 2003). These problems are in general difficult to solve because the set of feasible solutions is non-convex, possibly infinitely dimensional and defined using an infinite number of constraints. As a consequence several methods have been suggested to generate solvable approximations in which the trajectories are restricted to a set of basis functions and in which the constraints are imposed at a finite number of points in time (Singh and Hadaegh 2001; Faiz et al. 2001). These approaches generally result in very large feasibility problems, whose numerical solution is difficult, especially for large scale formation flying. For example Faiz et al. (2001) propose a solution based on differentially flat systems theory in which a number of nonlinear optimization problems have to be solved in order to compute an inner polytopic approximation of the feasible set defined by the constraints. Additionally a set of basis functions is used to represent the solution and a collocation grid in time is used to solve the problem.

Several authors (Richards et al. 2002; Prasanth et al. 2002; Richards and How 2002) propose the use of Mixed Integer Linear Programming (MILP) or Mixed Integer Linear Matrix Inequalities (MI/LMI) techniques for solution. Casting the problem as a MILP or MI/LMI requires many simplifications at the modeling and constraints formulation level (Richards et al. 2002). Because MILP techniques deal with linear problems, it is necessary to represent the system dynamics in linear form, as well as the constraints (e.g. convex polygons). A major drawback of solving a MILP is that the computation time increases at least polynomially with the number of variables and constraints. MILP problems are also known to be NP complete (Richards et al. 2002) and, in general, their solution requires branch and bound algorithms, whereas the size of the MI/LMI problems increases dramatically with the number of spacecraft (Prasanth et al. 2002).

Potential function approaches have been employed in the solution of trajectory generation and path planning problems with collision avoidance constraints (McQuade et al. 2002; Dunbar and Murray 2002; Saber et al. 2003). The potential function method is based on Lyapunov's second method and its principal advantage comes from its robustness and flexibility. McQuade et al. (2002) use a potential function to perform the deployment of a formation: when the potential function reaches a minimum the formation is correctly deployed. Dunbar and Murray (2002) combine model predictive control with potential functions techniques to solve a formation stabilization problem. However the resulting trajectories are not guaranteed to be free of collisions, illustrating one of the deficiencies such an approach might have.

Evolutionary algorithms have also been used in the context of path planning for formation flying (Seereram et al. 2000; Rathbun et al. 2002). Genetic algorithms are appealing because of their potential to find globally optimal solutions, however, due to their exhaustive nature and the high dimensionality of the multiple spacecraft control problem, these techniques suffer from an accelerated increase in computational complexity as the number of spacecraft increases.

Recently, multi-agent hybrid system optimization techniques have been proposed to solve multiple spacecraft formation reconfiguration (Yang et al. 2002), complex air traffic management (Tomlin et al. 1998), and motion planning problems (Lygeros et al. 1998). Multi-agent optimal control is quite different from the traditional optimal control for a single agent. In multi-agent hybrid systems conflicts which arise in the form of potential collisions are resolved locally by inter-agent coordination. This approach results in a decentralized architecture in which safety issues are resolved locally and central agencies, such as air traffic controllers, focus on global issues like efficiency and optimal output. The design issues associated with this approach are its complexity and the fact that a systematic investigation of switching between nonlinear control laws is still in its infancy.

Other approaches to trajectory generation problems include LMIs and graph theory (Mesbahi and Hadaegh 2001), variational calculus (Kim et al. 2003), convex optimization (Tillerson et al. 2002), semidefinite programming (Frazzoli et al. 2001). Frazzoli et al. (2001) solve a conflict resolution problem in air traffic control, which is based on convex programming and randomized searches, showing that a version of the planar case can be cast as a nonconvex, quadratically constrained quadratic program for which efficient numerical relaxations based on semidefinite programming can be used.

The last decade saw an increased interest in randomized path planners such as probabilistic roadmaps (PRM) and rapidly exploring random trees (RRT) which originated in the robotics community (Kavraki et al. 1996; Barraquand et al. 1997; Kavraki et al. 1998; Hsu et al. 1999; Hsu et al. 2000; Frazzoli et al. 2002; Phillips et al. 2003; Cerven et al. 2003). Most of the initial randomized planners dealt with static obstacles and they focused on collision avoidance, the optimization of some performance index being a secondary objective.

Several researchers (Hsu et al. 1999; Hsu et al. 2000) introduced kino-dynamic constraints and moving obstacles and indicated that the randomized planners can also handle these more complex problems. RRT-based approaches for path planning with time-varying constraints have been proposed for asteroid landing applications (Cerven et al. 2003). Phillips et al. (2003) used a probabilistic search approach (guided RRT) combined with gradient descent methods to solve a two-body, rendezvous and docking problem with kino-dynamic constraints for which classical PRM are not well equipped. Frazzoli et al. (2002) illustrate the use of RRT in computing trajectories in the case of moving obstacles; the example considered is that of a small autonomous helicopter flying through mobile obstacles. The RRT-based approaches are computationally simpler and faster than initial randomized planners but more research is needed to make them attractive for real-time applications of large scale problems. Other path planning methods from the robotics community include the use of Voronoi diagrams (Foskey et al. 2001; Garber and Lin 2002) in hybrid motion planning.

Hu et al. (2002) consider a problem which is, to some extent, similar to the one analyzed in this article, namely the minimization of an energy-related performance index (the integral of kinetic energy) subject to collision avoidance constraints for air traffic control. The aircraft are modeled as point masses moving in a gravity-free environment and the resulting problem is simplified by introducing way-points through which the aircraft must pass. The aircraft follow simple trajectories (line segments), which are proved to be optimal for the chosen performance index. The collision avoidance problem is reduced to a nonlinear optimization problem, which is further approximated by a finitely dimensional convex problem. The nonlinear collision avoidance constraints on the way-points locations are linearly approximated. Examples show that the approach works well for up to eight aircraft but there is no guarantee that the resulting trajectories are free of collisions.

Singh and Hadaegh (2001) tackle a problem which is very similar to the one in this article: energy optimal, collision-free, trajectory generation for deep space formation flying spacecraft. The energy is defined as the integral of the accelerations norm squared and the spacecraft are modeled as point masses in a gravity-free environment. The approach parameterizes the trajectories using polynomials of a variable degree in time. A parameter optimization problem in terms of the coefficients of the polynomials is then formulated and a heuristic numerical algorithm is proposed for the solution. Examples show that the method works well for up to five spacecraft, however, the heuristic nature of the algorithm and the fact that it requires numerical approximations for gradients calculation leads to increased computational time and numerical stability problems for large scale problems.

This article presents a method for the generation of energy sub-optimal, collision-free trajectories for formations flying in deep space (gravity-free environment). In a previous note (Sultan et al. 2006) some of the algorithms presented herein have been reported in a preliminary form. The present article is an enlarged, integrated perspective on a *class of algorithms* of which several instantiations (including those presented in Sultan et al. 2006) are analyzed. The idea used in our method is to simplify the problem by introducing a set of way-points through which the spacecraft trajectories pass and by assuming that these trajectories are piecewise cubic polynomials of class C^1 in time. Under the assumption that spacecraft dynamics are modeled using double integrators, for a given set of way-points, these trajectories are *energy optimal for each individual spacecraft*. This is the first major advantage of the solution, because it guarantees that the chosen parameterization is appropriate for the energy optimal problem posed. Furthermore, the resulting constrained optimization problem is shown to have a *quasi-quadratic structure* in the way-points locations and velocities. Gradient based numerical algorithms which exploit this mathematical structure are developed and used to select the way-points parameters such that collisions are avoided and energy is further minimized. A second major advantage of the solution presented in this article, is that the collision avoidance constraints are approximated such that *closed form solutions* are generated. Another major advantage with respect to other numerical algorithms like the one presented by Singh and Hadaegh (2001) is that the *gradients are analytically computed*, making the application of these algorithms very efficient in terms of computational time. Furthermore, *inversion of large matrices is not required* by our algorithms. Combination of closed form solutions and analytically computed gradients results in very fast numerical algorithms. Examples show that these algorithms are also very robust and efficient even for large scale formations (e. g. 20 spacecraft).

2. Statement of the Problem

The spacecraft are modeled as *points of constant mass* in a *gravity-free* environment, acted upon *only* by internally generated forces (e.g. by thrusters) used to control their motion. We assume that the maneuver time is the same for all spacecraft (i.e. *synchronous* maneuver); let this be T. Let N be the number of spacecraft, and r_l, v_l, a_l, $l = 1,\dots, N$, denote the position, velocity, and acceleration vectors of spacecraft l with respect to an inertial reference frame. Then the equations of motion and the terminal conditions are:

$$\begin{aligned} \dot{r}_l(t) &= v_l(t), \quad \dot{v}_l(t) = a_l(t), \quad r_l(0) = r_{l_0}, \\ r_l(T) &= r_{l_T}, \quad v_l(0) = v_{l_0}, \quad v_l(T) = v_{l_T}, \\ l &= 1, ..., N, \end{aligned} \tag{1}$$

where r_{l_0}, v_{l_0}, r_{l_T}, v_{l_T}, are the initial and final conditions and t is the time.

We remark that a double integrator has been used to represent each spacecraft dynamics, ignoring the orbital forces. This simplification has been adopted in view of the following observation. Consider a deep space Earth-trailing formation flying mission and assume that the spacecraft are only a few kilometers apart and that the masses of the spacecraft are of the order of a few hundred kilograms. When the linearized Hill's equations are used to describe their motion, it can be shown that the differential orbital force between two spacecraft is of the order of 10^{-23} N (Kim et al. 2003). Because the reconfiguration scenarios that we are interested in occur on relatively short time scales, ignoring the orbital forces in this work is well justified. We also remark that the double integrator has become an accepted, standard modeling approximation in preliminary deep space formation flying spacecraft path planning research (see Scharf et al. 2002). The deep space formation flying spacecraft is the best suited application for the modeling assumptions used herein; other areas in which the same assumptions are used heavily (aircraft traffic control, robot path planning) are more questionable since these systems operate in relatively strong gravitational environments (e. g. the Earth). This is one of the reasons we focus on deep space formation flying spacecraft missions: even though the methods presented in this article can be extended to air traffic control, Earth or underwater robots path planning, etc., they are best justified in situations where the modeling assumptions are closest to reality, like formations of spacecraft flying in deep space. The constant mass approximation is also appropriate for deep space flying spacecraft because the maneuvers analyzed here occur on relatively short time scales and the propulsion systems used for these maneuvers are very efficient, hence the mass of the spacecraft does not change much during the maneuver.

The collision avoidance constraints are specified in terms of the forbidden spheres associated with the spacecraft: any two forbidden spheres do not intersect:

$$\begin{aligned} \|r_l(t) - r_m(t)\|^2 &\geq (R_l + R_m)^2, \quad l = 1, ..., N-1, \\ m &= l+1, ..., N, \quad t \in [0, T], \end{aligned} \tag{2}$$

where R_l is the radius of the forbidden sphere associated with spacecraft l.

We remark that the diameter of the collision avoidance region (forbidden sphere) is a reflection of how far the actual spacecraft is away from being a point mass. Making the point mass approximation is especially useful when there is no interest in the orientation of the spacecraft. If orientation is important during the maneuver, 6 DOF models, which include attitude dynamics, have to be employed.

In addition limitations on the absolute value of the accelerations components need to be observed:

$$\begin{aligned} \left|a_{l_k}(t)\right| &\leq A_{l_k}, \quad l = 1, ..., N; \quad k = x, y, z, \\ t &\in [0, T], \end{aligned} \tag{3}$$

where $a_{l_k}(t)$ is the acceleration component of spacecraft l on axis k and $A_{l_k}$is the corresponding limit. The objective is to find $a_l(t)$, $t \in [0, T]$, $l = 1,\ldots, N$, such that the μ energy consumed,

$$J_\mu = \sum_{l=1}^{N} \mu_l \int_0^T a_l^T(t) a_l(t) dt, \tag{4}$$

is minimized, collisions are avoided, and saturation limits on the accelerations are observed. The weights $\mu_l > 0$, which add up to 1, $\sum_{l=1}^{N} \mu_l = 1$, are introduced to allow for the prioritization of the spacecraft within a formation; for example if all weights are equal all spacecraft energy consumptions are equally important, however if the weight associated with one spacecraft is dominant, that spacecraft's energy consumption is more important and drastically penalized. Optimization of the total energy of the formation is an objective for many missions (Kim et al. 2003; Singh and Hadaegh 2001; Hu et al. 2002). In addition, such an objective is justified because since in general the energy is directly related to the fuel consumption, minimization of energy consumption leads to less fuel being consumed.

3. Solution Approach

In the following we present an important result, which is the basis of our approach of energy optimal path planning problems.

Lemma: Energy Optimal Trajectories for Given Waypoints. Consider the case of one spacecraft ($N = 1$, $\mu_1 = 1$). Let $\{(t_j, w_j, v_j), j = 1,\ldots, M+2\}$ be a sequence of way-points specifying time, position, and spacecraft velocity, with $t_j < t_{j+1}$. Let $r(t)$ denote C^1 trajectories going through these way-points. Then the unique trajectory that minimizes the energy of the spacecraft given by (4) is given by:

$$\begin{aligned} r(t) &= \frac{1}{6}c_j(t^3 - t_j^3) + \frac{1}{2}d_j(t^2 - t_j^2) \\ &- (\frac{1}{2}c_j t_j^2 + d_j t_j - v_j)(t - t_j) + w_j, \\ t_j &\leq t \leq t_{j+1} \end{aligned} \tag{5}$$

for $j = 1,\ldots, M+1$, where

$$c_j = \frac{-12(w_{j+1} - w_j) + 6(v_{j+1} + v_j)(t_{j+1} - t_j)}{(t_{j+1} - t_j)^3} \tag{6}$$

$$\begin{aligned} d_j &= \frac{v_{j+1} - v_j}{t_{j+1} - t_j} + \frac{t_{j+1} + t_j}{(t_{j+1} - t_j)^3}(6(w_{j+1} - w_j) \\ &- 3(v_{j+1} + v_j)(t_{j+1} - t_j)). \end{aligned} \tag{7}$$

Proof: see Appendix A.

Hence the energy optimal trajectory is a C^1 piecewise cubic polynomial in time, a rational function in the way-points times, and a linear function in the way-points locations and velocities. These facts will lead to tremendous simplification of the problem and will be exploited in the development of efficient numerical algorithms.

We remark that if the only way-points are the end (initial and final) points of the trajectory and the initial and final conditions are such that the corresponding velocities are zero ("rest to rest" maneuvers) the energy optimal trajectory is a straight line segment cubic parameterized by time:

$$r(t) = r_0(1 - 3(\frac{t}{T})^2 + 2(\frac{t}{T})^3) + r_T(3(\frac{t}{T})^2 - 2(\frac{t}{T})^3) \quad (8)$$

where r_0 and r_T are the initial and final position vectors of the spacecraft, respectively.

We also note that the above result is valid only for the standard double integrator dynamics model. Hence if different models are used for the spacecraft dynamics, the C^1 piecewise cubic polynomial in time is no longer the energy optimal trajectory.

3.1. A Quasi-quadratic Optimization Problem

In order to simplify the problem we introduce the dimensionless time $\varsigma = t/T$, $\varsigma \in [0, 1]$. The equations of motion and the terminal conditions become:

$$\begin{aligned} r_l'(\varsigma) &= Tv_l(\varsigma), \quad v_l'(\varsigma) = Ta_l(\varsigma), \quad r_l(0) = r_{l_0}, \\ v_l(0) &= v_{l_0}, \quad r_l(1) = r_{l_T}, \quad v_l(1) = v_{l_T}, \end{aligned} \quad (9)$$

where $l = 1,\ldots, N$, and $'$ denotes differentiation with respect to ς. The μ energy becomes:

$$J_\mu = T\sum_{l=1}^{N}\mu_l \int_0^1 a_l^T(\varsigma)a_l(\varsigma)d\varsigma. \quad (10)$$

We shall treat the reconfiguration duration T as a parameter to be used *a posteriori* to enforce (3). We ignore (3) for the time being and later choose T so that (3) are satisfied. In order to do so we shall exploit the following relationship:

$$a_l(\varsigma) = \frac{1}{T^2}r_l''(\varsigma). \quad (11)$$

Consider now a formation of N spacecraft and assume that for each spacecraft, l, $M_l + 2$ way-points are introduced. Our method requires that each spacecraft follows a trajectory described by (5). Hence the resulting trajectories are energy optimal for each individual spacecraft.

Let ς_{l_j} and w_{l_j} denote the dimensionless time and location of the jth way-point of the lth spacecraft. The spacecraft velocity at this point is $v_{l_j} = u_{l_j}/T$, where $u_{l_j} = \partial r_l/\partial\varsigma(\varsigma_{l_j})$. Let ς be such that $\varsigma_{l_j} \leq \varsigma \leq \varsigma_{l_{j+1}}$.

Then using (5), the position vector of the lth spacecraft, r_l, can be expressed in terms of ς as follows:

$$r_l(\varsigma) = Z_{l_j}U_{l_j}, \quad (12)$$

where

$$Z_{l_j} = [M_1 \quad M_2 \quad M_3 \quad M_4], \quad (13)$$

and

$$M_1 = \frac{(\varsigma_{l_{j+1}} - \varsigma_{l_j})^3 + 2(\varsigma^3 - \varsigma_{l_j}^3) - 3(\varsigma_{l_{j+1}} + \varsigma_{l_j})(\varsigma^2 - \varsigma_{l_j}^2) - 6\varsigma_{l_j}\varsigma_{l_{j+1}}(\varsigma_{l_j} - \varsigma)}{(\varsigma_{l_{j+1}} - \varsigma_{l_j})^3}$$

$$\times I_3, \quad (14)$$

$$M_2 = \frac{\varsigma^3 - \varsigma_{l_j}^3 - (2\varsigma_{l_{j+1}} + \varsigma_{l_j})(\varsigma^2 - \varsigma_{l_j}^2) - (2\varsigma_{l_{j+1}}\varsigma_{l_j} + \varsigma_{l_{j+1}}^2)(\varsigma_{l_j} - \varsigma)}{(\varsigma_{l_{j+1}} - \varsigma_{l_j})^2}$$

$$\times I_3, \quad (15)$$

$$M_3 = \frac{-2(\varsigma^3 - \varsigma_{l_j}^3) + 3(\varsigma_{l_{j+1}} + \varsigma_{l_j})(\varsigma^2 - \varsigma_{l_j}^2) - 6\varsigma_{l_j}\varsigma_{l_{j+1}}(\varsigma - \varsigma_{l_j})}{(\varsigma_{l_{j+1}} - \varsigma_{l_j})^3} I_3, \quad (16)$$

$$M_4 = \frac{\varsigma^3 - \varsigma_{l_j}^3 - (\varsigma_{l_{j+1}} + 2\varsigma_{l_j})(\varsigma^2 - \varsigma_{l_j}^2) + \varsigma_{l_j}(2\varsigma_{l_{j+1}} + \varsigma_{l_j})(\varsigma - \varsigma_{l_j})}{(\varsigma_{l_{j+1}} - \varsigma_{l_j})^2} I_3, \quad (17)$$

$$U_{l_j} = [w_{l_j}^T \quad u_{l_j}^T \quad w_{l_{j+1}}^T \quad u_{l_{j+1}}^T]^T. \quad (18)$$

Consider now two spacecraft, l and m. Let ς be the current dimensionless time such that $\varsigma_{l_j} \leq \varsigma \leq \varsigma_{l_{j+1}}$ and $\varsigma_{m_k} \leq \varsigma \leq \varsigma_{m_{k+1}}$. Then, using (12), the distance square between spacecraft l and m, d_{lm}^2, can be expressed as

$$d_{lm}^2(\varsigma) = \|r_l(\varsigma) - r_m(\varsigma)\|^2 = U^T Z_{lm}(\varsigma)U, \quad (19)$$

where

$$\begin{aligned} Z_{lm}(\varsigma) &= (Z_l - Z_m)^T(Z_l - Z_m), & (20) \\ Z_l &= [0 \ \ldots \ 0 \ \ Z_{l_j} \ \ 0 \ \ldots \ 0], & \\ Z_m &= [0 \ \ldots \ 0 \ \ Z_{m_k} \ \ 0 \ \ldots \ 0], & (21) \\ U &= \Big[w_{1_1}^T \quad u_{1_1}^T \quad \ldots \quad w_{1_{M_1+2}}^T & \\ &\qquad u_{1_{M_1+2}}^T \quad \ldots \quad w_{l_j}^T \quad u_{l_j}^T \quad \ldots \quad w_{m_k}^T & \\ &\qquad u_{m_k}^T \quad \ldots \quad w_{N_{M_N+2}}^T \quad u_{N_{M_N+2}}^T\Big]^T. & (22) \end{aligned}$$

Hence d_{lm}^2 is a *time-varying quadratic form* in the way-points locations and velocities, a *piecewise polynomial* of degree 6 in ς, and a *rational* function in ς_{l_j} and $\varsigma_{l_{j+1}}$.

Using (4) and (12) J_μ can be expressed as a *quadratic form* in U:

$$\begin{aligned} J_\mu &= \sum_{l=1}^{N} \mu_l J_l = \frac{1}{T^3} U^T B U, \\ B &= diag(\mu_1 B_1 \quad \ldots \quad \mu_N B_N), \end{aligned} \tag{23}$$

where B_l is associated with the energy of the l-th spacecraft, J_l:

$$J_l = \sum_{j=1}^{M_l+1} U_{l_j}^T B_{l_j} U_{l_j} = U_l^T B_l U_l, \tag{24}$$

$$\begin{aligned} B_{l_j} &= \frac{4}{T^3}(\varsigma_{l_{j+1}} - \varsigma_{l_j})(C^T C \\ &+ 3(\varsigma_{l_{j+1}}^2 + \varsigma_{l_{j+1}}\varsigma_{l_j} + \varsigma_{l_j}^2) D^T D \\ &+ \frac{3}{2}(\varsigma_{l_{j+1}} + \varsigma_{l_j})(D^T C + C^T D)), \end{aligned} \tag{25}$$

$$\begin{aligned} C &= \left[\frac{-3(\varsigma_{l_{j+1}} + \varsigma_{l_j})}{(\varsigma_{l_{j+1}} - \varsigma_{l_j})^3} I_3 \quad \frac{-2\varsigma_{l_{j+1}} - \varsigma_{l_j}}{(\varsigma_{l_{j+1}} - \varsigma_{l_j})^2} I_3 \right. \\ &\left. \frac{3(\varsigma_{l_{j+1}} + \varsigma_{l_j})}{(\varsigma_{l_{j+1}} - \varsigma_{l_j})^3} I_3 \quad \frac{-\varsigma_{l_{j+1}} - 2\varsigma_{l_j}}{(\varsigma_{l_{j+1}} - \varsigma_{l_j})^2} I_3 \right], \end{aligned} \tag{26}$$

$$\begin{aligned} D &= \left[\frac{2}{(\varsigma_{l_{j+1}} - \varsigma_{l_j})^3} I_3 \quad \frac{1}{(\varsigma_{l_{j+1}} - \varsigma_{l_j})^2} I_3 \right. \\ &\left. \frac{-2}{(\varsigma_{l_{j+1}} - \varsigma_{l_j})^3} I_3 \quad \frac{1}{(\varsigma_{l_{j+1}} - \varsigma_{l_j})^2} I_3 \right], \end{aligned} \tag{27}$$

$$U_l = \left[w_{l_1}^T \quad u_{l_1}^T \quad \ldots \quad w_{l_{M_l+2}}^T \quad u_{l_{M_l+2}}^T\right]^T. \tag{28}$$

Assume now that the end points parameters, w_{l_1}, u_{l_1}, $w_{l_{M_l+2}}$, $u_{l_{M_l+2}}$, and the way-points times, ς_{l_j}, $l = 1, \ldots, N$, $j = 1, \ldots, M_l + 2$, are given. We introduce the vector of optimization variables, given by the *intermediate* way-points parameters,

$$x = \left[w_{1_2}^T \quad u_{1_2}^T \quad \ldots \quad w_{N_{M_N+1}}^T \quad u_{N_{M_N+1}}^T\right]^T, \tag{29}$$

such that J_μ and d_{lm}^2 are expressed as

$$J_\mu = x^T Q x + b^T x + c, \tag{30}$$

$$d_{lm}^2(\varsigma) = x^T Q_{lm}(\varsigma) x + b_{lm}^T(\varsigma) x + c_{lm}(\varsigma), \tag{31}$$

with $Q > 0$, b, c, $Q_{lm}(\varsigma)$, $b_{lm}(\varsigma)$, $c_{lm}(\varsigma)$ easy to compute from B, $Z_{lm}(\varsigma)$ and the given parameters. Note: the proof that $Q > 0$ is given in Appendix B.

In this framework the μ energy optimal problem with collision avoidance constraints becomes:

$$\begin{aligned} &\min_x (x^T Q x + b^T x + c) \quad subject \quad to \\ &x^T Q_{lm}(\varsigma) x + b_{lm}^T(\varsigma) x + c_{lm}(\varsigma) \ \geq \ (R_l + R_m)^2, \end{aligned} \tag{32}$$

where $l = 1, \ldots, N-1, \quad m = l+1, \ldots, N, \quad \varsigma \in [0, 1]$.

Hence a quadratic performance index minimum is sought subject to time-varying quadratic inequality constraints. Moreover the time dependency of the constraints is simple, being given by piecewise polynomials of degree 6. This mathematical structure of the problem facilitates the development of very efficient numerical algorithms, as shown next.

We remark that the inputs to the algorithms are the given end points of the trajectories and the intermediate way-points times, the outputs being the intermediate way-points locations and velocities. If the way-points times are introduced in the vector of optimization variables, then the previously mentioned nice structure of the problem (e.g. quadratic performance index minimization subject to time-varying quadratic inequality constraints) is destroyed and other algorithms have to be developed.

We also remark that the unconstrained problem solution is very simple:

$$x = -\frac{1}{2} Q^{-1} b. \tag{33}$$

This solution provides the best possible result in terms of the μ energy achievable and will be used as a starting point for our algorithms.

3.2. Numerical Solution

In the following we propose a *sequential* algorithm to solve (32). There are several motivations behind such an approach, the most important being that the *collision avoidance constraints are critical*. In fact, some path planning algorithms are only interested in generating collision-free trajectories, minimization of a performance index like energy or fuel consumption not even being explicitly considered (e.g. Faiz et al. 2001). In our sequential approach we first solve the collision avoidance problem and then minimize J_μ while making sure that the collision avoidance constraints are satisfied.

Another important motivation is our intention to develop *fast, numerically efficient* algorithms for *large scale* applications. In such a pursuit the mathematical structure of the problem has to be kept relatively simple and exploited in order to eliminate the need for computationally complex algorithms. The classical approach to nonlinear optimization under inequality constraints leads to complicated first-order necessary conditions (i.e. Kuhn–Tucker conditions) whose solutions are computationally intensive, involving complex numerical algorithms (see Luenberger 1984). If we consider the use of classical nonlinear programming approaches, these are plagued by various issues which manifest themselves especially for large scale problems: feasible directions methods result in algorithmic maps which are not closed being affected by jamming, penalty and barrier functions methods result in ill-conditioned numerical problems, which require the inversion of large matrices, etc.

In order to avoid these problems we present ingenious numerical solutions which use concepts from classical nonlinear programming (active sets, gradients) but avoid complicated computations by exploiting the structure of the problem. Closed form solutions and analytically computable gradients are used, resulting in very fast numerical algorithms which have proven to work well in all our experiments.

3.2.1. Collision Avoidance Problem Solution

The collision avoidance problem is to find x such that

$$\begin{aligned} d_{lm}^2(\varsigma) &\geq (R_l + R_m)^2, \quad l = 1, ..., N-1, \\ m &= l+1, ..., N, \quad \varsigma \in [0, 1]. \end{aligned} \tag{34}$$

For this problem's solution we use a methodology inspired by the *active set method.* The idea underlying active set methods is to partition inequality constraints into two groups: those that are to be treated as active and those that are to be treated as inactive. The constraints treated as inactive are essentially ignored, decreasing the number of computations, which is especially useful for large scale problems (see Luenberger 1984 for details on active set methods). Experience with active set methods has shown that they are very efficient for large scale problems (see Luenberger 1984).

Our methodology proceeds as follows: at the current iteration step, for a known value of x, for each pair of spacecraft, (l, m), we calculate the global minimum of $d_{lm}^2(\varsigma)$ with respect to ς, $0 \leq \varsigma \leq 1$. Since $d_{lm}^2(\varsigma)$ is a piecewise polynomial of degree 6 in ς, this reduces to the simple task of polynomials minimization, for which fast algorithms are available. Let d_{lm*}^2 denote those global minima which violate the constraints ($d_{lm*}^2 < (R_l + R_m)^2$) and ς_{lm*} be the corresponding dimensionless times.

A natural choice of the active constraints is given by the violating constraints. Hence we build a penalty function, $P(x)$, based only on these constraints:

$$\begin{aligned} P(x) &= \sum_{l,m}((R_l + R_m)^2 - d_{lm*}^2) \\ &= \sum_{l,m}((R_l + R_m)^2 - x^T Q_{lm}(\varsigma_{lm*})x \\ &- b_{lm}^T(\varsigma_{lm*})x - c_{lm}(\varsigma_{lm*})) > 0 \end{aligned} \tag{35}$$

where only the violating pairs (l, m), appear in the sum.

The penalty function proposed here deserves some discussion. We stress that this particular choice of penalty function is extremely advantageous because, as shown next, it facilitates closed form approximate solutions. If classical penalty functions are used (see Luenberger 1984), the simple mathematical structure of the problem is destroyed and closed form solutions are not possible. Moreover, as is well known, the use of classical penalty functions for nonlinear optimization problems solution is plagued by many disadvantages. For example the solution is generally approached from outside the feasible region, hence the intermediate values of the constraints are not satisfied. In addition the numerical structure of the problem becomes increasingly ill-conditioned as the solution is approached, which leads to slower convergence. Computations require inversions of matrices because, in general, gradient or Newton based methods are used for solution, which for large scale problems are time consuming (see Luenberger 1984 for more details). In contrast, our approach *does not involve inversions of matrices* and *uses simple computations.* If convergence is obtained it guarantees collision-free trajectories which are generally obtained very rapidly even for large scale applications.

Our goal is to drive the penalty function to zero at the next iteration step. For this we use a line search approach and assume that a change in x is made along a direction $g \neq 0$:

$$x_+ = x + \alpha g. \tag{36}$$

Then $P(x_+)$ can be expressed as a *quadratic* function in α (this is where the particular choice of the penalty function is extremely advantageous):

$$P(x_+) = P + \alpha^2 g^T H g + \alpha g^T \nabla P, \tag{37}$$

where $P = P(x)$, $H \leq 0$ is half of the Hessian of $P(x)$, and ∇P is the gradient of $P(x)$.

Since our goal is to drive the penalty function to zero, next we solve $P(x_+) = 0$ for α, yielding

$$\begin{aligned} \alpha &= \frac{-g^T \nabla P + / - \sqrt{(g^T \nabla P)^2 - 4 g^T H g P}}{2 g^T H g}, \\ if \quad g^T H g &\neq 0, \end{aligned} \tag{38}$$

$$\begin{aligned} \alpha &= -\frac{P}{g^T \nabla P}, \\ if \quad g^T H g &= 0, \end{aligned} \tag{39}$$

and select α with minimum absolute value. Note: if $g^T \nabla P = g^T H g = 0$ (for example if $g = 0$) then we slightly perturb x such that (37) has at least a solution for α.

At the next step we update the penalty function based on the *current* violating constraints and iterate until convergence is obtained (constraints are not violated), the number of iterations allowed is exceeded, or the norm of x variation between two consecutive steps is smaller than the allowed tolerance.

We remark that this procedure implicitly assumes that the time dependency of the constraints is negligible for small variations of x. This is one of the reasons for choosing the solution of minimum absolute value for α of $P(x_+)$=0. By doing so, we expect the number of iterations to be small.

As can be ascertained, unlike classical penalty function based approaches, this one has the advantage of being extremely simple in computational complexity (few constraints are taken into account and the required computations are very simple). Experience showed that the resulting algorithms have very fast convergence.

This methodology results in a *class of algorithms* that can be developed based on the selection of the direction g. In the following we show two options based on gradients, which result in efficient algorithms. However, other choices for g can be devised, resulting in algorithms having high performance.

The DIG Algorithm

The first idea is to choose $g = \nabla P$, the gradient of $P(x)$, because it provides the fastest variation in $P(x)$. The resulting algorithm is coined DIG (from DIstance and Gradient) and it is described next.

DIG Algorithm Description

1. **Initialization**: Set

$$x = -\frac{1}{2}Q^{-1}b \tag{40}$$

which is the solution of the unconstrained μ energy optimal problem.

2. **Trajectories assessment**: For all (l, m) pairs compute the global minima of $d^2_{lm}(\varsigma)$, $\varsigma \in [0, 1]$, which violate the constraints and the corresponding ς, let them be called d^2_{lm*} and ς_{lm*}, respectively. If none of the constraints are violated, exit.

3. **Calculation of the direction of movement, *g***: compute the gradient of the penalty function:

$$g = \nabla P = -\sum_{l,m}(2Q_{lm}(\varsigma_{lm*})x + b_{lm}(\varsigma_{lm*})) \tag{41}$$

where only the violating pairs appear in the sum.
If g = 0 then slightly perturb x randomly and go back to step 2, else calculate H and P:

$$\begin{aligned} H &= -\sum_{l,m} Q_{lm}(\varsigma_{lm*}), \\ P &= \sum_{l,m}((R_l + R_m)^2 - d^2_{lm*}). \end{aligned} \tag{42}$$

4. **Prediction**: Predict the next value of x:

$$x_+ = x + \alpha g \tag{43}$$

where

$$\alpha = \frac{-\|g\|^2 + \sqrt{\|g\|^4 - 4g^T HgP}}{2g^T Hg}, \quad if \quad g^T Hg \neq 0, \tag{44}$$

$$\alpha = -\frac{P}{\|g\|^2}, \quad if \quad g^T Hg = 0. \tag{45}$$

Note: $\alpha < 0$ given in (44) and (45) is the solution of minimum absolute value of $P(x_+) = 0$.

5. **Return**: If $\|\alpha g\| < \delta x$, where δx is the minimum allowed variation of x, or the number of iterations is greater than the maximum allowed, exit, else set

$$x = x_+ \tag{46}$$

and return to step 2.

The process terminates when all spacecraft are separated (successful termination, through step 2), the variation in x is too small, or the number of iterations allowed is exceeded.

The DJ Algorithm

We remark that in the choice of g for DIG we do not take into account the objective of minimizing J_μ. In the following we propose a different choice for g, which takes into account this objective.

A good direction, g, in which x should change is one which would decrease as fast as possible the penalty function and would not increase J_μ too much. Since J_μ is quadratic in x, we have

$$J_\mu(x_+) = J_\mu(x) + \alpha^2 g^T Qg + \alpha g^T \nabla J_\mu \tag{47}$$

where $\nabla J_\mu = 2Qx + b$ is the gradient of J_μ evaluated at x.

Ideally, for fastest variation in $P(x)$ we would like $g = \nabla P$. Obviously, if $\nabla P^T \nabla J_\mu \geq 0$ we can choose $g = \nabla P$ and $\alpha < 0$ given by the solution of minimum absolute value of $P(x_+) = 0$ for α:

$$\alpha = \frac{-\|g\|^2 + \sqrt{\|g\|^4 - 4g^T HgP}}{2g^T Hg}, \quad if \quad g^T Hg \neq 0, \tag{48}$$

$$\alpha = -\frac{P}{\|g\|^2}, \quad if \quad g^T Hg = 0. \tag{49}$$

For $\nabla P^T \nabla J_\mu < 0$ such a choice ($g = \nabla P$ and $\alpha < 0$) would lead to an increase in J_μ given both by the linear and quadratic terms in α in (47). One idea is to choose $g = \nabla P$and

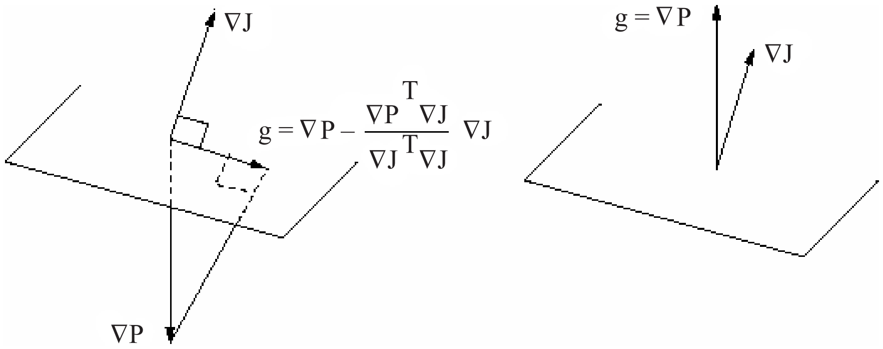
∇J
$g = \nabla P - \frac{\nabla P^T \nabla J}{\nabla J^T \nabla J} \nabla J$
∇P
g = ∇P
∇J

Fig. 1. Direction of movement, g, in the DJ algorithm.

the other solution for α of the quadratic equation, $P(x_+) = 0$, namely:

$$\alpha = \frac{-\|g\|^2 - \sqrt{\|g\|^4 - 4g^T HgP}}{2g^T Hg}, \quad if \quad g^T Hg \neq 0, \tag{50}$$

$$\alpha = -\frac{P}{\|g\|^2}, \quad if \quad g^T Hg = 0. \tag{51}$$

However this is not the minimum absolute value solution and such a choice might actually result in a large increase in J_μ due to its quadratic component in α (see (47)). Hence we choose g such that $g^T \nabla J_\mu = 0$. Such a direction is the projection of ∇P onto the plane perpendicular to ∇J_μ, as shown in Figure 1:

$$g = \nabla P - \frac{\nabla P^T \nabla J_\mu}{\nabla J_\mu^T \nabla J_\mu} \nabla J_\mu. \tag{52}$$

This choice fails when ∇P and ∇J_μ are collinear (opposite), a case which we shall analyze later.

Note: it is easy to show that $g^T \nabla P = 0 \Leftrightarrow g = 0$, hence (37) always has at least one solution (since we consider $g \neq 0$).

Next, we choose α as the solution of minimum absolute value of $P(x_+)$=0, yielding:

$$\alpha = \frac{-g\nabla P + \sqrt{(g^T \nabla P)^2 - 4g^T HgP}}{2g^T Hg}, \quad if \quad g^T Hg \neq 0, \tag{53}$$

$$\alpha = -\frac{P}{g^T \nabla P}, \quad if \quad g^T Hg = 0. \tag{54}$$

When $\nabla P$and ∇J_μ are collinear, since our main goal is to get $P(x_+)=0$ we choose $g = \nabla P$.As before α is the solution of minimum absolute value of $P(x_+) = 0$, given by (48) and (49).

Following is a description of the algorithm, which we shall coin DJ, an acronym of Distance and J the symbol for the μ energy, J_μ.

DJ Algorithm Description

The first two steps are the same as in the DIG algorithm. The calculation of the direction of movement is replaced by the following sequence:

3. **Penalty function gradient calculation,** ∇P:

$$\nabla P = -\sum_{l,m} (2Q_{lm}(\varsigma_{lm*})x + b_{lm*}(\varsigma_{lm*})) \tag{55}$$

where only the violating pairs appear in the sum. If $\nabla P = 0$, then perturb x randomly (small perturbations) and return to step 2.

4. **Penalty function Hessian and μ energy gradient calculation**: compute half of the Hessian of the penalty function, H, the gradient of J_μ , and the penalty function value, P:

$$H = -\sum_{l,m} Q_{lm}(\varsigma_{lm*}), \quad \nabla J_\mu = 2Qx + b,$$

$$P = \sum_{l,m} ((R_l + R_m)^2 - d_{lm*}^2) \tag{56}$$

where (l, m) is any pair which violates the collision avoidance constraints.

5. **Direction of movement calculation**: If $\nabla P^T \nabla J_\mu \geq 0$ or $\hat{\nabla} P^T \hat{\nabla} J_\mu = -1$ the direction is

$$g = \nabla P \tag{57}$$

and α is

$$\alpha = \frac{-\|g\|^2 + \sqrt{\|g\|^4 - 4g^T HgP}}{2g^T Hg}, \quad if \quad g^T Hg \neq 0, \tag{58}$$

$$\alpha = -\frac{P}{\|g\|^2}, \quad if \quad g^T Hg = 0. \tag{59}$$

Else (if $\nabla P^T \nabla J_\mu < 0$ and $\hat{\nabla} P^T \hat{\nabla} J_\mu \neq -1$) the direction is

$$g = \nabla P - \frac{\nabla P^T \nabla J_\mu}{\nabla J_\mu^T \nabla J_\mu} \nabla J_\mu \tag{60}$$

and α is given by

$$\alpha = \frac{-g^T \nabla P + \sqrt{(g^T \nabla P)^2 - 4 g^T H g P}}{2 g^T H g}, \quad if \quad g^T H g \neq 0, \tag{61}$$

$$\alpha = -\frac{P}{g^T \nabla P}, \quad if \quad g^T H g = 0. \tag{62}$$

Note: in the above $\hat{\nabla} P$ and $\hat{\nabla} J_\mu$ represent the unit vectors along $\nabla P$and ∇J_μ, respectively.

6. **Prediction**: Predict the next value of the unknowns:

$$x_+ = x + \alpha g \tag{63}$$

7. **Return**: If $\|\alpha g\| < \delta x$, where δx is the minimum allowed variation in x, or the number of iterations is greater than the maximum allowed, exit, else set

$$x_+ = x \tag{64}$$

and return to step 2.

The process terminates when all spacecraft are separated (successful termination, through step 2), the variation in x is too small, or the number of allowed iterations is exceeded.

3.2.2. Minimization of J_μ

After the spacecraft have been separated through successful application of the separation algorithm (DIG or DJ) we can further minimize J_μ taking care that the collision avoidance constraints are satisfied. Because J_μ is quadratic in x, a gradient based procedure, in which the next value of x is determined by

$$x_+ = x + sn, \quad n = -\frac{\|\nabla J_\mu\|^2}{2 \nabla J_\mu^T Q \nabla J_\mu} \nabla J_\mu, \tag{65}$$

is indicated for this purpose. An important issue in this process is the determination of the step size, s, based on the collision avoidance constraints. The procedure described next has proven very efficient in our applications.

Consider an arbitrary pair of spacecraft, l and m, and determine the corresponding minimum distance between them, let it be called $d_{\min}$. Let ς_c be the corresponding dimensionless time. Let $x_+ = x + sn$ and approximate the corresponding minimum distance square by:

$$d_+^2 = x_+^T \tilde{Q} x_+ + \tilde{b}^T x_+ + \tilde{c} \tag{66}$$

where $\tilde{Q} = Q_{lm}(\varsigma_c)$, $\tilde{b} = b_{lm}(\varsigma_c)$, $\tilde{c} = c_{lm}(\varsigma_c)$ (this is where the approximation takes place). Next we set d_+^2 equal to $(R_l + R_m)^2$ to obtain

$$s^2 n^T \tilde{Q} n + s n^T \nabla d_{lm}^2 + d_{\min}^2 - (R_l + R_m)^2 = 0 \tag{67}$$

yielding

$$s_1 = \frac{-n^T \nabla d_{lm}^2 - \sqrt{\Delta}}{2 n^T \tilde{Q} n}, \quad s_2 = \frac{-n^T \nabla d_{lm}^2 + \sqrt{\Delta}}{2 n^T \tilde{Q} n}, \quad if \quad n^T \tilde{Q} n \neq 0 \tag{68}$$

$$s_1 = \frac{(R_m + R_l)^2 - d_{\min}^2}{n^T \nabla d_{lm}^2}, \quad if \quad n^T \tilde{Q} n = 0 \quad and \quad n^T \nabla d_{lm}^2 \neq 0 \tag{69}$$

where

$$\Delta = (n^T \nabla d_{lm}^2)^2 - 4(d_{\min}^2 - (R_l + R_m)^2) n^T \tilde{Q} n, \quad and \quad \nabla d_{lm}^2 = 2 \tilde{Q} x + \tilde{b}. \tag{70}$$

The decision process is as follows: if $0 < s_2 < 1$ set $s_{lm} = s_2$ else if $0 < s_1 < 1$ set $s_{lm} = s_1$. In any other case (including when there are no solutions of (67)) set $s_{lm} = 1$. This process is performed for all spacecraft pairs. Finally s is selected as $s = \min_{l,m} s_{lm}$. The collision avoidance constraints are tested at s and if they are violated we set $s = s/2$ and iterate until no constraints are violated. Note: in the rare situation in which the minimum distance is attained for multiple, different, values of ς_c, the same calculations are performed for each ς_c and corresponding s_{lm} values are computed. The important thing is that minimization of s_{lm} is performed for the selection of s and that the collision avoidance constraints are tested to validate the selection of s.

The resulting algorithm, coined JG, is described below:

JG Algorithm

1. **Direction of movement (n) calculation:**

$$n = -\frac{\|\nabla J_\mu\|^2}{2 \nabla J_\mu^T Q \nabla J_\mu} \nabla J_\mu, \quad \nabla J_\mu = 2 Q x + b. \tag{71}$$

2. **Calculation of the tentative step size, s:** For each pair of spacecraft, (l, m), compute the minimum distance, $d_{\min}$, the corresponding minimum dimensionless time, ς_c, and the step size, s_{lm}:

$$
\begin{aligned}
s_1 &= \frac{-n^T \nabla d_{lm}^2 - \sqrt{\Delta}}{2n^T \tilde{Q} n}, \\
s_2 &= \frac{-n^T \nabla d_{lm}^2 + \sqrt{\Delta}}{2n^T \tilde{Q} n}, \\
if \quad n^T \tilde{Q} n &\neq 0 \qquad (72) \\
s_1 &= \frac{(R_m + R_l)^2 - d_{\min}^2}{n^T \nabla d_{lm}^2}, \\
if \quad n^T \tilde{Q} n &= 0 \quad and \quad n^T \nabla d_{lm}^2 \neq 0 \qquad (73)
\end{aligned}
$$

where

$$
\begin{aligned}
\Delta &= (n^T \nabla d_{lm}^2)^2 \\
&- 4(d_{\min}^2 - (R_l + R_m)^2) n^T \tilde{Q} n, \\
and \quad \nabla d_{lm}^2 &= 2\tilde{Q} x + \tilde{b}. \qquad (74)
\end{aligned}
$$

If $0 < s_2 < 1$, set $s_{lm} = s_2$, else if $0 < s_1 < 1$set $s_{lm} = s_1$. In any other cases set $s_{lm} = 1$. Finally s is selected as $s = \min_{l,m} s_{lm}$.

3. **Tentative step:**

$$x_t = x + sn. \qquad (75)$$

4. **Return:** If $\|x_t - x\| < \delta x$, where δx is the minimum allowed variation in x, or the number of iterations is greater than the maximum allowed, exit, else compute the distances between spacecraft at x_t and determine if the collision avoidance constraints are violated or not. If the collision avoidance constraints are satisfied set $x = x_t$ and go back to step 1. If there is at least one violation set $s = s/2$ and go back to step 3.

We remark that all algorithms presented herein (DIG, DJ, JG) exploit the fact that their objective functions (P and J_μ respectively) are quadratic in x, making the application of gradient based algorithms extremely efficient. Also they do not involve complex computations since, for example, the gradients are computed analytically and are not numerically approximated. Hence the computation time per iteration is very small even for large scale formations.

Once a solution has been obtained we turn our attention to the satisfaction of (3). An estimate of a maximum of $|a_{k_i}(\varsigma)|, \quad i = x, y, z$ is easy to obtain since the accelerations are piecewise linear functions of time. Accelerations vary as $1/T^2$, hence it is trivial to choose T such that none of the acceleration components exceeds A_{k_i}.

4. Examples

4.1. Swapping Circle

The first example involves 16 spacecraft equidistantly placed on a circle of 10 m radius. We consider a rest-to-rest reconfiguration maneuver, coined swapping circle, in which each spacecraft must swap its place with the one opposite to it with respect to the center of the circle. Such an example is inspired by the Darwin mission in which several spacecraft (in the current proposed design six but the number might increase in subsequent designs) will be placed on the circumference of a circle. Our intention is also to generate a complex collision avoidance problem in order to test our algorithms; if the unconstrained energy optimal trajectories are used, the spacecraft collide at the center of the circle, all of the 120 collision avoidance constraints being simultaneously violated.

The number of intermediate way-points per spacecraft is chosen using the following heuristic method. Because the energy optimal unconstrained trajectories reach the critical point, the center of the circle, at the same time, $\varsigma_c = 0.5$, we introduce one intermediate way-point per spacecraft, $M_l = 1$, with $\varsigma_{l_2} = 0.5$, $l = 1, ..., 16$. We remark that, for most of the missions currently under consideration, the introduction of one way-point per spacecraft may be sufficient.

For this example the collision avoidance spheres have the same radius, R_l =1 m, the energy weights are equal, $\mu_l = 1/16$, $l = 1, ..., 16$ and the duration of the maneuver is T = 20 s. Application of our methodology, in which DIG is followed by JG, yielded a solution in 39 iterations. The resulting trajectories are shown in Figure 2 and the time histories of the distances in Figure 3, indicating that the minimum allowed distance between any two spacecraft (2 m) is not violated. The acceleration time histories are given in Figure 4.

The ratio between the final value of the μ energy and the value of the μ energy along the energy optimal unconstrained trajectories, which we shall call the *energy index*, is 2.108, indicating that 108% more energy is required to avoid collisions. If DJ and JG are used, 126 iterations are required for convergence and the energy index is 2.003, indicating a slightly better solution. The number of iterations required for convergence is higher than when DIG and JG are used. This is expected because of the way DJ operates: it does not always follow the gradient of the penalty function in an attempt to keep the energy low at each iteration. Due to the simplicity of the computations involved, convergence is still achieved very rapidly.

The method used to evaluate our algorithms with respect to the μ energy performance index deserves some discussion. We have chosen to compare this parameter with the μ energy of the energy optimal unconstrained trajectories for several reasons. The μ energy of the energy optimal unconstrained trajectories is the *global minimum* of the μ energy in the absence of constraints, hence the best possible result. It is also the value of the performance index of the starting point in our algorithms. By comparing the final value of the performance

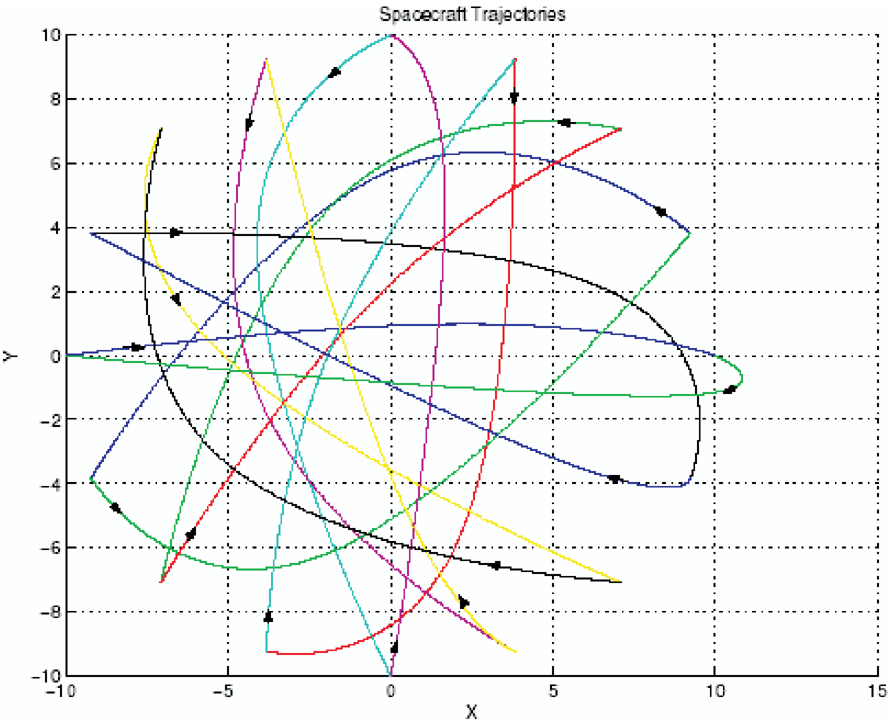

Fig. 2. Trajectories of the swapping circle maneuver generated by the DIG-JG algorithms.

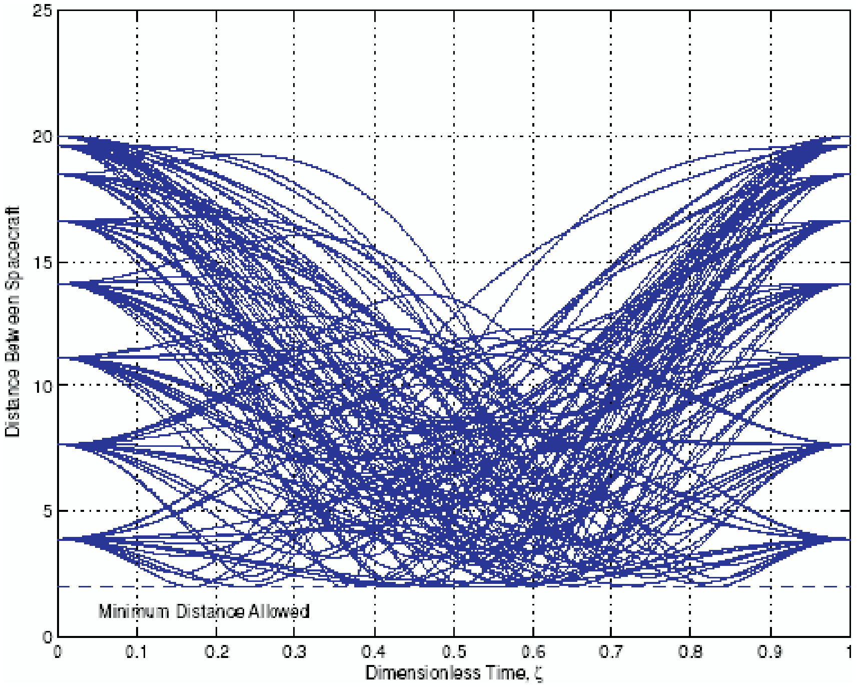

Fig. 3. Distance time histories for the swapping circle maneuver.

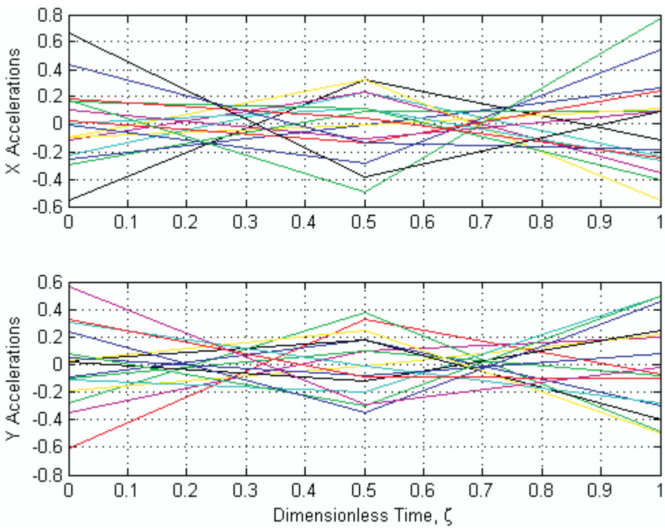

Fig. 4. Accelerations time histories for the swapping circle maneuver.

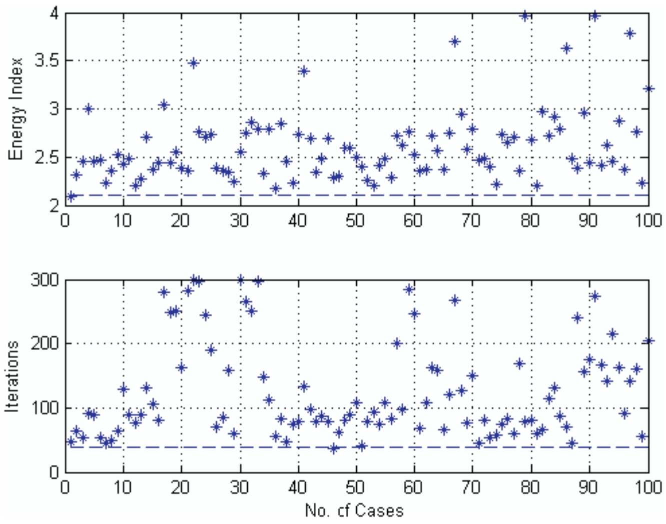

Fig. 5. Starting point influence for the swapping circle maneuver and the DIG-JG algorithms.

index with its initial value we ascertain how far away from the initial point the algorithms have departed. A very large energy index indicates that the solution obtained is poor in terms of the μ energy performance index. Even though our solution strategy was constructed such that we stay close to the μ energy global minimum (obtained when the constraints are discarded) in the hope that nearly optimal solutions are found, there is no guarantee of that, especially for the complicated problems we consider.

4.2. The Starting Point Influence

A natural question arises regarding the choice of the starting point for our algorithms: would it be better to select a different one? In order to address this question we tested the algorithms for cases in which the initial guess was different from the one corresponding to the μ energy global minimum $\left(x = -\frac{1}{2}Q^{-1}b\right)$. In most of our numerical experiments we obtained solutions which were worse than the ones obtained when the μ energy global minimum was used as a starting point. For example, Figure 5 shows the distribution of the results obtained for the swapping circle maneuver, characterized by the same parameters as in 4.1. (i.e. circle radius 10 m, T = 20 s, R_l = 1 m, M_l = 1, $\varsigma_{l_2} = 0.5$, $\mu_l = 1/16$, $l =$ 1, ..., 16.) when the DIG-JG algorithms were used for 100 cases in which the starting point was *randomly* generated. The initial values for the *intermediate* way-points locations and velocities were randomly generated within +/–30% of the values corresponding to the μ energy global minimum (note: in general, larger perturbations resulted in worse results). In Figure 5 the dashed lines correspond to the μ energy and number of iterations for the solution obtained when the starting point was the one corresponding to the μ energy global minimum. We remark that in all cases the trajectories thus generated obey the collision avoidance constraints and, *except for one case*, all have final values of μ energy larger than the μ energy of the solution obtained when the starting point corresponded to the μ energy global minimum. The only case which resulted in slightly better energy consumption, yielded an energy index of 2.09, obtained after 47 iterations (recall that the solution corresponding to the μ energy global minimum starting point had an energy index of 2.108 obtained after 39 iterations). Similar results were obtained for the DJ-JG combination, increasing the level of confidence in the choice of the starting point as the μ energy global minimum.

4.3. Comparison with Other Feasible Solutions

Another question one might ask is the following: since the proposed procedure is sub-optimal, how much confidence do we have that the solutions thus generated are good ones in terms of the μ energy consumed? In order to address this question we use the following comparison scenario: consider the set of piecewise cubic C^1 polynomials (in time) trajectories satisfying the given end points constraints (i.e. initial and final spacecraft locations and velocities). We want to see how the solutions obtained using the proposed methodology compare to other feasible solutions (i.e. which obey the collision avoidance constraints) within this set. In order to generate piecewise cubic C^1 polynomials trajectories that obey the collision avoidance constraints, we proceed as follows. First, we *randomly select* the intermediate way-points times, their locations, and

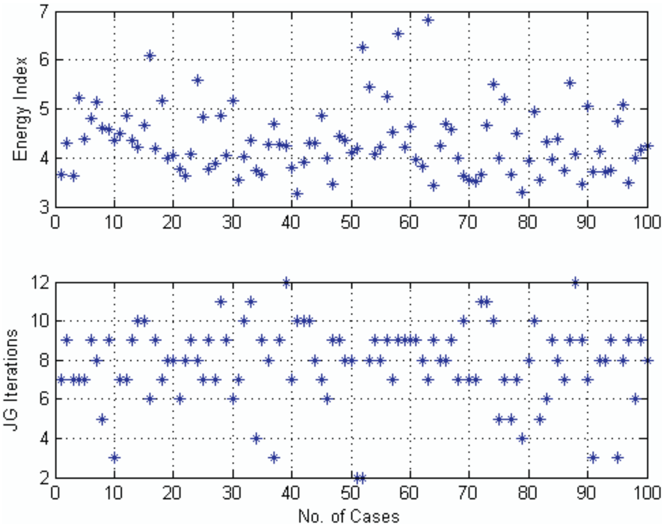

Fig. 6. Performance of other piecewise cubic C^1 polynomials feasible trajectories for the swapping circle maneuver.

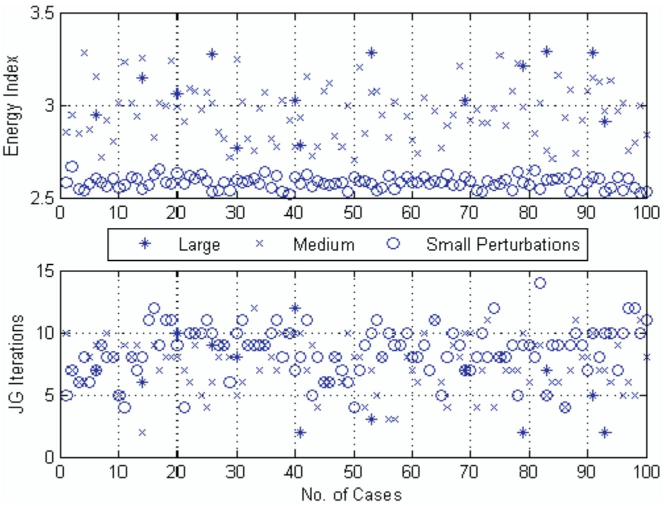

Fig. 7. Performance of neighboring feasible solutions for the swapping circle maneuver.

velocities. Then we test the resulting piecewise cubic C^1 polynomials trajectories passing through these way-points (given by (5)) against the collision avoidance constraints (given by (2)). If the collision avoidance constraints are not violated then we have found a feasible solution. However, if the resulting trajectories do not satisfy the collision avoidance constraints, we first run the DIG (or DJ) algorithm with *larger than the nominal* collision avoidance radii in order to generate feasible solutions. The μ energies of the feasible solutions thus generated are further improved by running JG on them, using the *nominal* collision avoidance radii. Note: obtaining feasible solutions for the complicated problems we consider (many spacecraft, large number of collision avoidance constraints) is very difficult if we just randomly generate way-points parameters. This is why we have to de-conflict the conflicting trajectories using, for example, DIG and DJ with larger collision avoidance radii.

Figure 6 shows the results obtained on the swapping circle maneuver, defined by the same parameters as before. In order to generate feasible trajectories we used the DIG algorithm with the “larger radii” all equal to $1.2R_l = 1.2$ m. One intermediate way-point per spacecraft was introduced, whose time was randomly generated within +/–20% of the value corresponding to the nominal one (fixed at 0.5). Likewise, the intermediate way-points velocities and locations were randomly generated within +/–20% of the values corresponding to the trajectories of the μ energy global minimum (unconstrained maneuver) of the swapping circle maneuver. In Figure 6 we plot the energy index and the number of JG iterations for 100 test cases. It can easily be ascertained that the energy index (defined with respect to the μ energy global minimum of the swapping circle maneuver analyzed in 4.1.) is, in all cases, much higher than the one obtained when DIG-JG or DJ-JG algorithms were applied (recall that these energy indices were 2.1 for the DIG-JG and 2 for the DJ-JG cases). These results indicate that the procedure might even find *near-optimal* solutions, at least within the class of C^1 piecewise polynomials.

4.4. Exploring the Neighborhood of the Solution

Next we consider another set of experiments as follows: we randomly perturb the *solution* obtained using DIG-JG or DJ-JG. However we keep the values of the way-points times *unchanged*. Then we use the intermediate way-points thus generated in an optimization scheme as follows: if the corresponding piecewise cubic C^1 polynomials trajectories obey the collision avoidance constraints, we run JG on them in order to decrease the μ energy consumed and then compute the corresponding energy index. If the random perturbation of the solution results in conflicting trajectories we disregard them. The motivation for such a comparison scenario is to investigate the *neighborhood of the solution* for feasible trajectories with lower μ energy consumption within the class of piecewise cubic C^1 polynomials.

Figure 7 shows the results obtained for the swapping circle maneuver (defined by the same parameters as in 4.1.). We considered the solution obtained using the DIG-JG combination and three levels of perturbation: large (30%), medium (15%), and small (5%), where the percentages correspond to the absolute value of the maximum possible deviation from the nominal values (i.e. 30% means that the intermediate way-points locations and velocities were randomly perturbed within +/–30% of their values corresponding to the *solution*). In each case we ran 100 test cases. All the small perturbations

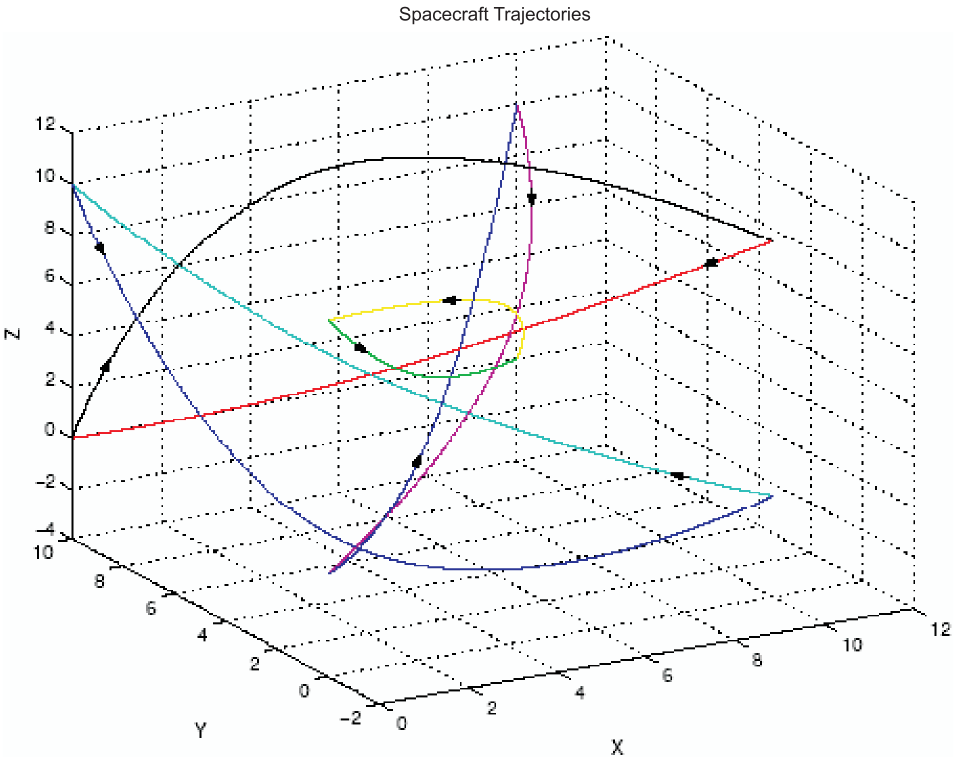

Fig. 8. Trajectories for the swapping cube maneuver obtained using the DIG-JG algorithms.

resulted in trajectories which obeyed the collision avoidance constraints and whose further improvement using JG led to incremental decrease in the μ energy (circles in Figure 7). The large size perturbations resulted mostly in conflicting trajectories, and application of JG on the few feasible trajectories did not improve the energy index by much (stars in Figure 7). In all cases the energy index is higher than the ones for the solutions obtained using DIG-JG (energy index = 2.1) or DJ-JG (energy index = 2). These numerical experiments further increase confidence in the fact that the procedure proposed herein might even find *nearly optimal* solutions, at least within the class of piecewise cubic C^1 polynomials trajectories.

4.5. Swapping Cube

The next example is of a three-dimensional reconfiguration maneuver, coined the swapping cube. In this scenario eight spacecraft placed at the corners of a cube of 10 m side length must swap places simultaneously. The maneuver is rest-to-rest, with T = 11 s, R_l = 1 m, and $\mu_l = 1/8, \quad l = 1, ..., 8$. The unconstrained energy optimal trajectories are lines cubic parameterized by time. If these are used all 28 collision avoidance constraints are violated simultaneously. We selected intermediate way-points heuristically, with M_l = 1, and $\varsigma_{l_2} = 0.5, \quad l = 1, ..., 8$.

Application of the DIG-JG algorithms yielded a collision-free solution very rapidly, in only 19 iterations. The energy index is 1.306, showing that 30.6% more energy is required to avoid collisions. If the DJ-JG algorithms are applied, a solution is obtained after 22 iterations and the energy index is 2.05. Hence 105% more energy and three more iterations are required to avoid collisions. This example shows that there are cases in which the DJ-JG algorithms yield worse results than DIG-JG.

The trajectories obtained when the DIG-JG algorithms are used for solution are shown in Figure 8, and the time histories of the distances and accelerations in Figure 9 and Figure 10 respectively, indicating that the collision avoidance constraints are satisfied.

4.6. Non-rest-to-rest Swapping Cube Maneuver

The next example shows how the methodology developed performs on a maneuver which is not rest-to-rest. We consider the same formation of eight spacecraft, as in the swapping cube maneuver. In the initial configuration the spacecraft are placed at the corners of the same cube, of side length 10 m, which was used in the swapping cube maneuver, but their velocity vectors are non-zero as follows: for the four spacecraft placed in the

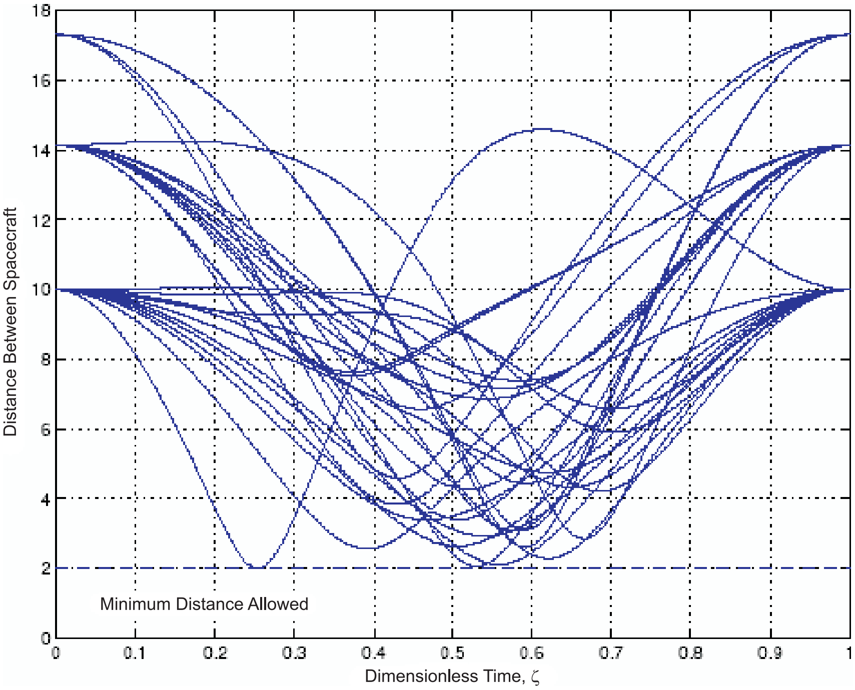

Fig. 9: Distances Time Histories for the Swapping Cube Maneuver

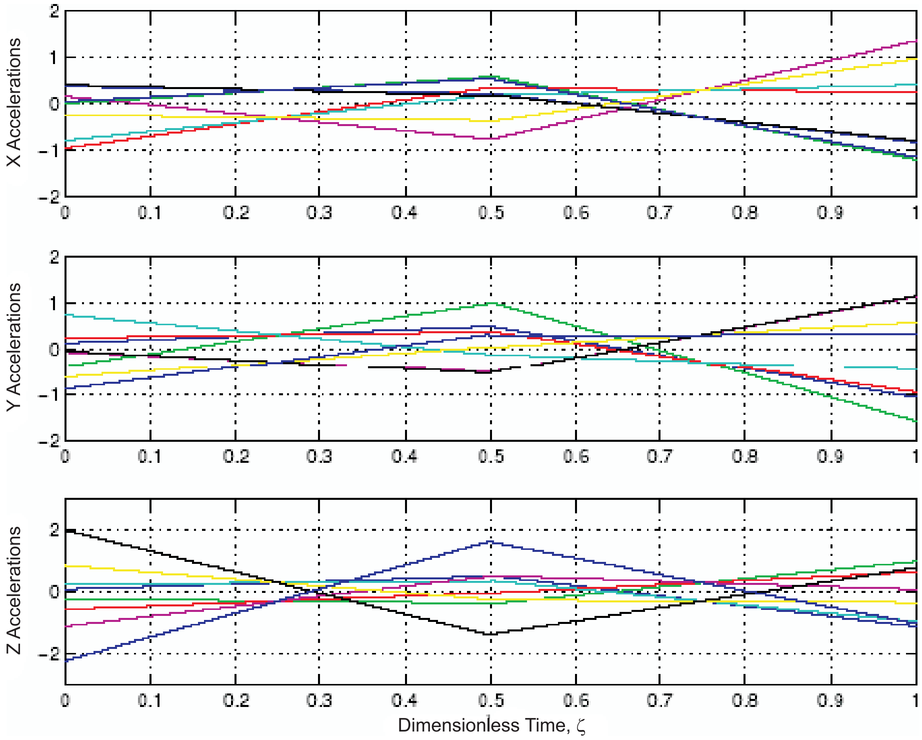

Fig. 10. Accelerations time histories for the swapping cube maneuver.

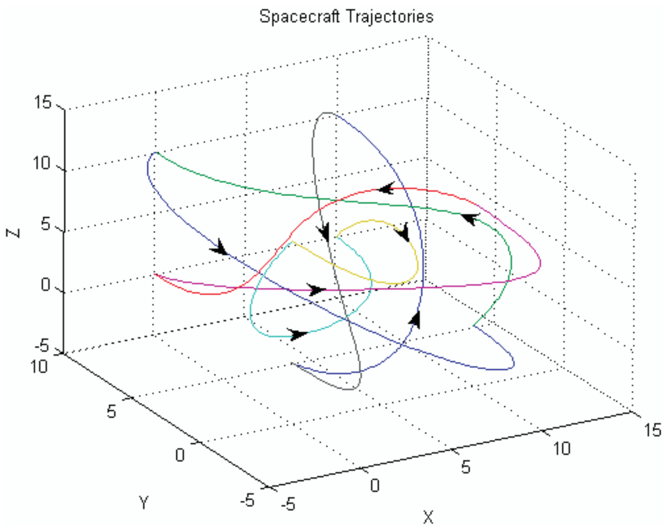

Fig. 11. Trajectories for the non-rest-to-rest swapping cube maneuver obtained using DIG-JG.

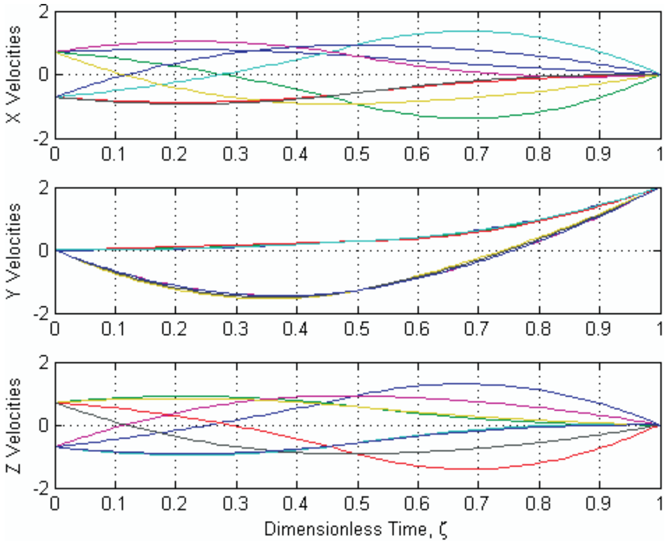

Fig. 12. Velocities time histories for the non-rest-to-rest swapping cube maneuver.

face of a cube (e.g. the xz face), the velocity vectors are all of the same magnitude, equal to 1 m/s and tangent to the circle circumscribed to that face of the cube (i.e. the spacecraft describe the same circle, at the same speed, and in the same direction). The other four spacecraft velocity vectors are parallel and equal to their counterparts (i.e. they are obtained through a virtual translation of the four spacecraft placed in the xz plane along the y axis). In a practical application this configuration corresponds to an "observational mode" in which the velocity of the mass center of the formation is zero. Let the origin of the inertial reference frame coincide with one spacecraft's (which we label as spacecraft number 1) initial position and the inertial reference frame be such that the initial cube formed by the spacecraft is in the first quadrant. Then the initial positions of the spacecraft are:

$$\begin{aligned}
r_1(0) &= [0 \quad 0 \quad 0]^T, \quad r_2(0) = [10 \quad 0 \quad 0]^T, \\
r_3(0) &= [10 \quad 0 \quad 10]^T, \quad r_4(0) = [0 \quad 0 \quad 10]^T, \\
r_5(0) &= [0 \quad 10 \quad 0]^T, \quad r_6(0) = [10 \quad 10 \quad 0]^T, \\
r_7(0) &= [10 \quad 10 \quad 10]^T, \quad r_8(0) = [0 \quad 10 \quad 10]^T. \quad (76)
\end{aligned}$$

The maneuver involves swapping the spacecraft positions with respect to the center of the cube; the following pairs of spacecraft have to change places: 1–7, 2–8, 3–5, 4–6. In addition it is desired that in the final configuration all spacecraft have equal velocity vectors, parallel to the y axis and of 2 m/s magnitude. This configuration corresponds to a "translational" mode in which the center of mass velocity is non-zero and the "spacecraft cube" translates along the y axis. The final positions of the spacecraft are:

$$\begin{aligned}
r_1(T) &= [10 \quad 10 \quad 10]^T, \quad r_2(T) = [0 \quad 10 \quad 10]^T, \\
r_3(T) &= [0 \quad 10 \quad 0]^T, \quad r_4(T) = [10 \quad 10 \quad 0]^T, \\
r_5(T) &= [10 \quad 0 \quad 10]^T, \quad r_6(T) = [0 \quad 0 \quad 10]^T, \\
r_7(T) &= [0 \quad 0 \quad 0]^T, \quad r_8(T) = [10 \quad 0 \quad 0]^T. \quad (77)
\end{aligned}$$

The following parameters were considered:

$$\begin{aligned}
T &= 20s, \quad R_l = 2 \text{ m}, \quad \varsigma_{l_2} = 0.5, \\
\mu_l &= 1/8, \quad l = 1, ..., 8.
\end{aligned}$$

Application of the DIG-JG combination resulted in very fast convergence, obtained after only 19 iterations. The solution is depicted in Figures 11, 12, 13 and 14 (trajectories, velocities, accelerations, distances time histories, respectively). The μ energy required to avoid collisions is 16% greater than the μ energy of the unconstrained trajectories. This shows that our methodology performs very well on maneuvers which are not rest-to-rest.

4.7. A TPF Maneuver

In the following we compare the performance of our methodology with the performance of the one used by Singh and Hadaegh (2001) on a TPF maneuver that involves a formation of five spacecraft, four of which are coplanar. The maneuver consists in changing the orientation of the plane of the four coplanar spacecraft while the center of mass of the formation remains unchanged. The maneuver is rest-to-rest with

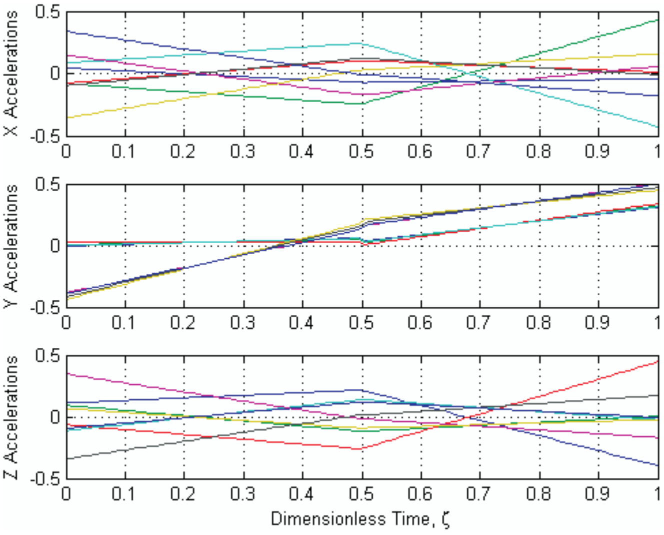

Fig. 13. Accelerations time histories for the non-rest-to-rest swapping cube maneuver.

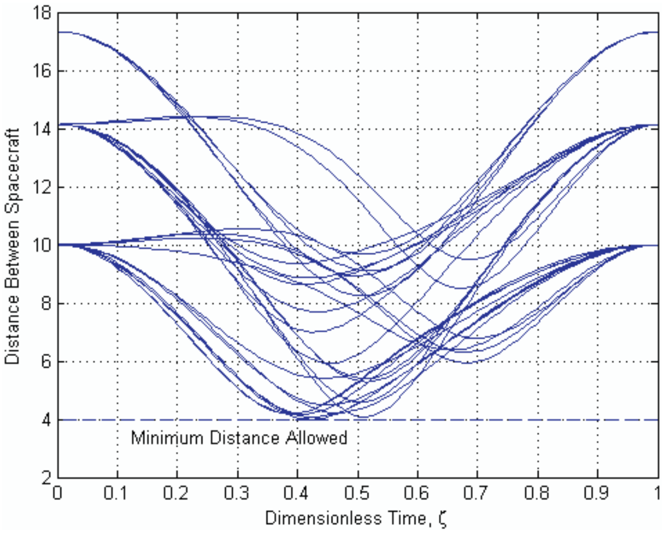

Fig. 14. Distances time histories for the non-rest-to-rest swapping cube maneuver.

$R_l = 10$ m, $\quad l = 1, ..., 5$. We consider that the inertial reference frame's origin coincides with the center of mass of the formation. Then the initial and final inertial coordinates of the spacecraft (adapted from Singh and Hadaegh 2001) are:

$$\begin{aligned}
r_1(0) &= [4.45 \quad -33.97 \quad 21.16]^T, \\
r_2(0) &= [4.45 \quad -11.78 \quad 6.37]^T, \\
r_3(0) &= [4.45 \quad 10.41 \quad -8.42]^T, \\
r_4(0) &= [4.45 \quad 32.60 \quad -23.21]^T, \\
r_5(0) &= [-17.80 \quad 2.74 \quad 4.11]^T \\
r_1(T) &= [-23.12 \quad -23.34 \quad -23.27]^T, \\
r_2(T) &= [-10.28 \quad -6.10 \quad -7.50]^T \\
r_3(T) &= [2.57 \quad 11.13 \quad 8.28]^T, \\
r_4(T) &= [15.42 \quad 28.37 \quad 24.05]^T, \\
r_5(T) &= [15.42 \quad -10.06 \quad -1.57]^T. \qquad (78)
\end{aligned}$$

The following limits on the accelerations have to be observed (from Singh and Hadaegh 2001):

$$\begin{aligned}
A_{l_x} &= 0.005, \quad A_{l_y} = 0.004, \\
A_{l_z} &= 0.003, \quad l = 1, ..., 4, \\
A_{5_x} &= 0.004, \quad A_{5_y} = 0.003, \quad A_{5_z} = 0.005. \qquad (79)
\end{aligned}$$

If the unconstrained energy optimal trajectories are used (line segments, cubic parameterized by time) the collision avoidance constraints are violated, with the maximum violation occurring at half time during the maneuver. For the application of our methodology we choose $M_l = 1$, $\varsigma_{l_2} = 0.5$, $\mu_l = 1/5$, $l = 1, ..., 5$.

Application of the DIG-JG combination resulted in very fast convergence, obtained after only eight iterations. The solution is depicted in Figures 15–17 (trajectories, accelerations, distances time histories, respectively). The maneuver time, obtained *a posteriori* to satisfy the limits on the accelerations, is T=316 s, like that obtained in Singh and Hadaegh (2001), however, only 52% more energy with respect to the energy of the unconstrained trajectories is required to avoid collisions. For comparison, Singh and Hadaegh's algorithm, in which the trajectories are parameterized using polynomials of degree 4 in time, yielded a solution which requires 80% more energy to avoid collisions, indicating that our methodology provides a better solution.

4.8. Monte Carlo Simulations for the Sphere Scenario

In order to evaluate the performance and robustness properties of the algorithms, we tested them using Monte Carlo simulations for the following scenario. A formation of 20 spacecraft is considered, whose initial and final positions are distributed on the surface of a sphere of radius 10 m. Both the initial and final positions are resting conditions and are randomly generated. The formation has to be synchronously reconfigured in a prescribed time, T = 10 s. One way-point per spacecraft is introduced, placed at half time during the maneuver. It is also

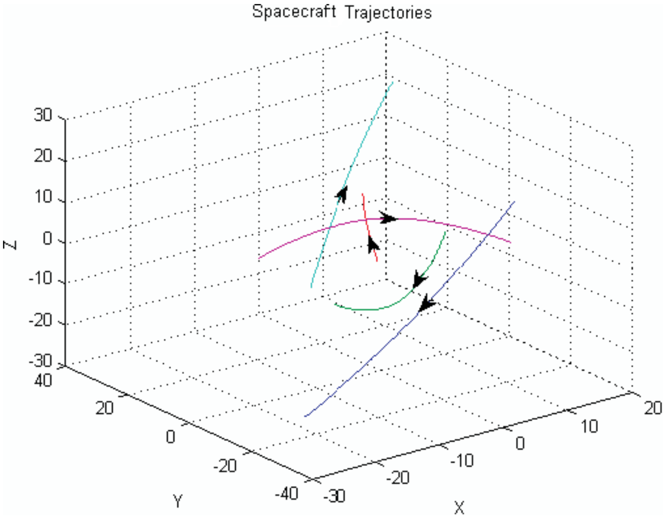

Fig. 15. Trajectories for the TPF maneuver obtained using DIG-JG.

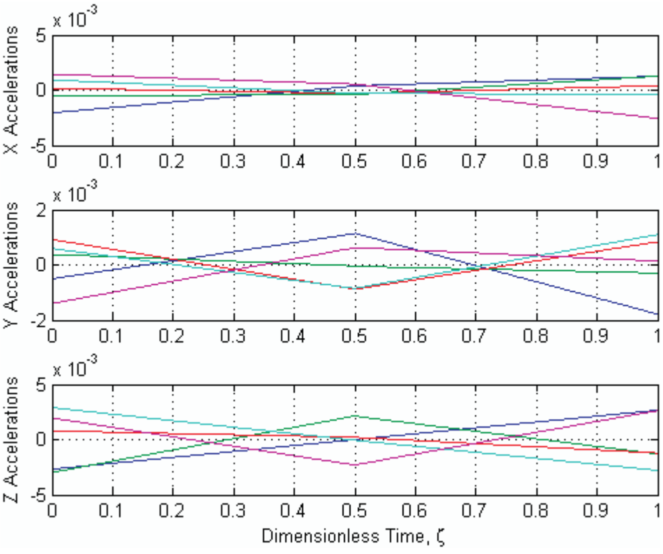

Fig. 16. Accelerations time histories for the TPF maneuver.

assumed that $\mu_l = 0.05$ and that the forbidden sphere radius is the same for all spacecraft, $R_l = R$, $l = 1, \ldots, 20$.

The DIG and DJ algorithms followed by JG have been applied to the resulting reconfiguration problems. In order to evaluate the performance of these algorithms we used the following measures: the energy index, defined here as the ratio between the μ energy of the solution provided by the algorithms and the μ energy of the μ energy optimal reconfiguration problem solution in the absence of constraints, and the number of iterations required for convergence.

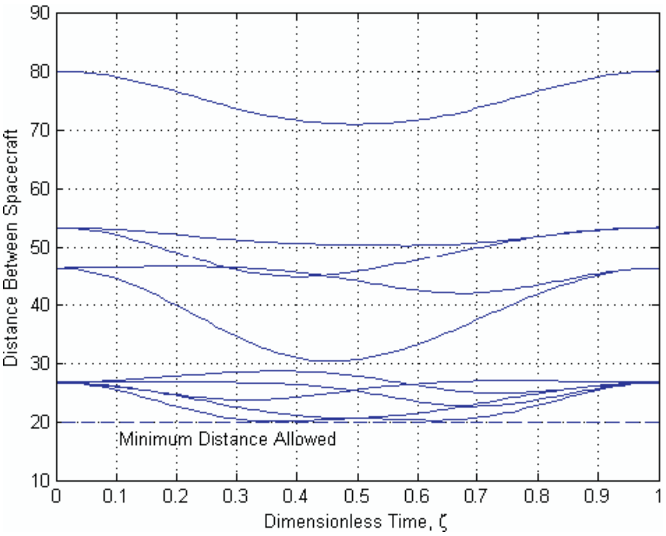

Fig. 17. Distances time histories for the TPF maneuver.

Figures 18 and 19 show the results obtained for 100 cases and three different values of the forbidden sphere radius, R = 1, 2, 4 m. The number of iterations required varied with the forbidden sphere radius, however convergence was always obtained relatively quickly. The energy indices also indicate that in general good solutions were obtained, in which the μ energy spent to avoid collisions is not dramatically greater than that consumed along the hypothetical μ energy optimal trajectories obtained in the absence of constraints. Figures 18 and 19 also indicate that the algorithms are very robust (i.e. there are no big variations in the performance indices).

In general the DIG-JG combination converges faster than the DJ-JG combination, but better energy indices are obtained in the later case. However, as clearly indicated by the swapping cube example, this is not always the case: there are situations when the DJ-JG combination might be worse both in terms of energy expenditure and in terms of the number of iterations required for convergence.

5. Conclusions

The path planning methodology proposed herein can be applied directly to problems in which modeling the dynamics of the vehicles as double integrators is accurate enough and in which the only constraints are control saturation and collision avoidance ones, represented as forbidden spheres. The model used to develop the methodology in this article requires two important assumptions: first that the vehicle can be approximated as a point of constant mass, and second, that no other forces (e.g. gravity) act on the vehicle except for the control forces generated by the onboard thrusters. There are many problems in which the double integrator model has been

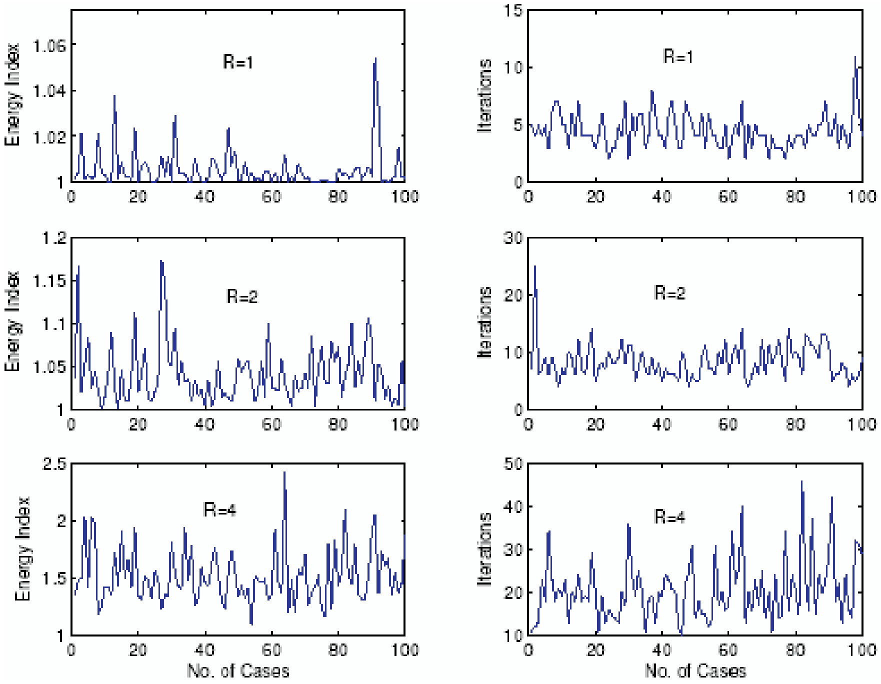

Fig. 18. Evaluation of the DIG-JG algorithms through Monte Carlo simulations.

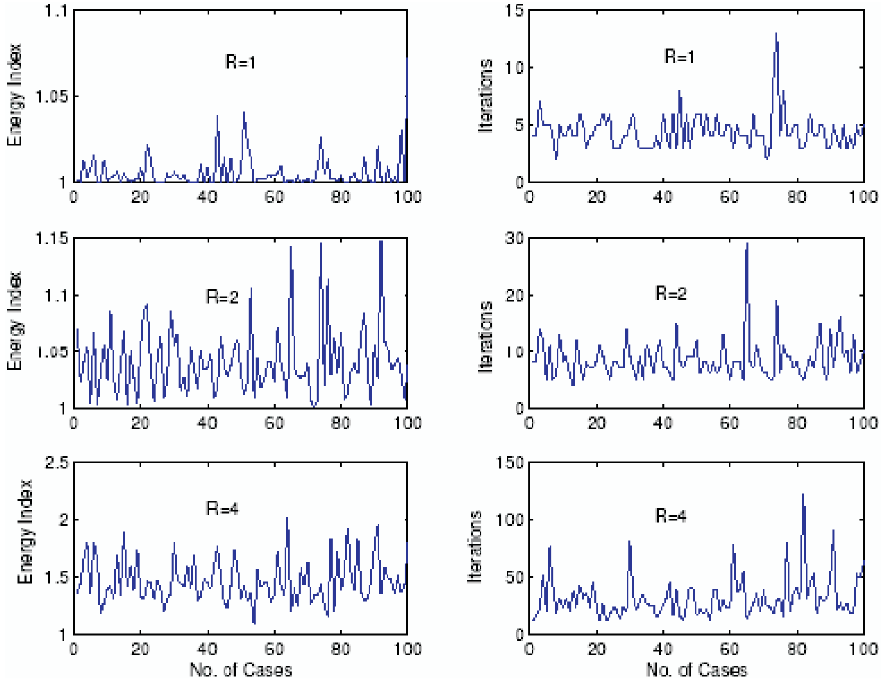

Fig. 19. Evaluation of the DJ-JG algorithms through Monte Carlo simulations.

used, including, but not limited to, air traffic control and mobile robots path planning. It is best suited for formations flying in deep space, where the gravitational influences can be neglected and the point mass approximation is largely accepted.

The path planning solution proposed in this article is an energy sub-optimal, possibly nearly optimal, collision avoidance methodology characterized by several important features. First, the spacecraft trajectories are parameterized using C^1 piecewise cubic polynomials, which are energy optimal for each individual spacecraft for a given sequence of way-points. This is a major advantage because it guarantees that the chosen parameterization is appropriate for the energy optimal problem posed while also facilitating the development of efficient numerical algorithms. Second, the resulting constrained optimization problem is cast as a quasi-quadratic optimization problem in the way-points locations and velocities. This structure results in the development of gradient based algorithms which are used to select the way-points parameters in order to avoid collisions and further minimize the energy. Third, the collision avoidance constraints are approximated such that closed form solutions for the collision avoidance problem are obtained. Fourth, the gradients used in the algorithms are computed analytically, making the application of these algorithms very efficient in terms of computational time. Fifth, inversion of large matrices, which can be computationally expensive, is not required. Due to these characteristics the resulting methodology yields fast algorithms, which generate solutions after relatively few iterations, even when applied to large scale, complicated, collision avoidance problems, as illustrated by the examples presented. Moreover, Monte Carlo simulations on examples involving up to 20 spacecraft indicate that the methodology and the algorithms are very robust and reliable.

Several comments are in order regarding the limitations and possible extensions through further research of the methodology described in this article. The most stringent limitations are related to the assumption that vehicle dynamics can be approximated using double integrators. One important aspect future research has to consider is that, if the model of a vehicle's dynamics is more complex, the energy optimal trajectory for a given sequence of way-points is no longer given by the C^1 piecewise cubic polynomial described in this article.

Most importantly, the methodology proposed herein cannot be applied to more sophisticated path planning problems which include *angular constraints* like, for example, *communication or observational constraints*. Examples include changing the orientation of the spacecraft and of the onboard antennas such that communication with other spacecraft is maintained or pointing an on board instrument (e.g. telescope) such that observation of a certain object is performed. These requirements lead to constraints on the *orientation* of the spacecraft, which cannot be captured by the point mass models used in this article. For solution of these problems future research must exploit the combination of complex, nonlinear, dynamic models (at least rigid body models) with the introduction of way-points and the development of easily computable (ideally analytically computable) gradients. This is a difficult problem because the dimension of the state space increases dramatically.

Other limitations relate to the constant mass approximation. This assumption can be used for short term maneuvers and when the fuel consumption rate of the propulsion system is negligible. When the constant mass approximation cannot be applied, the mass variation must be included in the dynamical model, hence, even in the simplest case of the point mass approximation, this will lead to an increase on the state space dimension. In this situation future research should investigate if extension of methodologies similar to the one described herein is possible.

Another limitation which is easier to handle is when gravitational forces cannot be neglected (e.g. for vehicles flying in the proximity of a planet). However, if the vehicles can be approximated as points of constant mass, the dynamics of the spacecraft will not increase the dimension of the state space. Future research should lead to a similar methodology based on the introduction of way-points and the use of analytically computable gradients.

Lastly this article analyzes only energy optimal trajectories. Similar research should be carried out in the area of fuel optimal problems. Even though directly related – in general lower energy consumption leads to less fuel being consumed – optimal energy and optimal fuel problems have different solutions.

Appendix A

Proof of the Lemma

Consider a generic interval, $I_j = [t_j, \quad t_{j+1}]$. Let us find the acceleration $a(t)$ which minimizes the energy defined as:

$$J = \int_{t_j}^{t_{j+1}} a(t)^T a(t) dt, \tag{80}$$

subject to the dynamics equations and the given boundary conditions:

$$\begin{aligned} \dot{r}(t) &= v(t), \quad \dot{v}(t) = a(t), \\ r(t_j) &= w_j, \quad r(t_{j+1}) = w_{j+1}, \\ v(t_j) &= v_j, \quad v(t_{j+1}) = v_{j+1}. \end{aligned} \tag{81}$$

This is a classical problem in the calculus of variations. The state and control vectors are

$$y = [r(t)^T \quad v(t)^T]^T, \quad u = a(t), \tag{82}$$

respectively and the Hamiltonian of the problem is:

$$\begin{aligned} H &= a^T(t)a(t) + \lambda_v^T v(t) + \lambda_a^T a(t) \\ &= u^T u + \lambda_v^T y(4:6) + \lambda_a^T u \\ \text{where} \quad \lambda &= [\lambda_v^T \quad \lambda_a^T]^T. \end{aligned} \tag{83}$$

The necessary conditions for *stationary* values of J are

$$\begin{aligned} \frac{\partial H}{\partial u} &= 0 \Rightarrow 2a(t) + \lambda_a = 0 \\ \frac{\partial H}{\partial y} &= -\dot{\lambda} \Rightarrow \dot{\lambda}_v = 0 \quad \text{and} \quad \dot{\lambda}_a = -\lambda_v, \end{aligned} \tag{84}$$

which yield $a(t) = c_j t + d_j$ where c_j and d_j are constants. Applying the given boundary conditions we obtain c_j, d_j and $r(t)$ given in the Lemma.

In order to prove that this is a *minimizing* solution we have to show that the second variation of J is positive definite for any nonzero variations of u. Indeed, the second variation is:

$$\begin{aligned} \delta^2 J &= 0.5 \int_{t_j}^{t_{j+1}} [\delta x^T \quad \delta u^T] \begin{bmatrix} H_{uu} & H_{xu} \\ H_{ux} & H_{uu} \end{bmatrix} \begin{bmatrix} \delta x \\ \delta u \end{bmatrix} dt \\ &= \int_{t_j}^{t_{j+1}} (\delta u^T \delta u) dt > 0 \\ &\quad \text{for any nonzero variation } \delta\text{u}. \end{aligned} \tag{85}$$

The energy minimizing control is then obtained by applying this linear acceleration on each time interval.

Appendix B

Proof of the Fact that $Q > 0$

It is sufficient to prove the fact for one spacecraft, since the matrix Q for N spacecraft is a diagonal one in which the diagonal entries are Q matrices corresponding to each spacecraft.

In the following, for simplicity, we drop the index representing the spacecraft number (l, in the body of the article). Then, using (11), the fact that $M + 2$ way points are introduced, and (12), the energy of one spacecraft can be expressed as:

$$\begin{aligned} J &= T \int_0^1 a^T(\varsigma)a(\varsigma)d\varsigma = \frac{1}{T^3} \int_0^1 r''^T(\varsigma)r''(\varsigma)d\varsigma \\ &= \frac{1}{T^3} \sum_{j=1}^{M+1} \int_{\varsigma_j}^{\varsigma_{j+1}} r''^T(\varsigma)r''(\varsigma)d\varsigma \\ &= \frac{1}{T^3} \sum_{j=1}^{M+1} \int_{\varsigma_j}^{\varsigma_{j+1}} U_j^T Z_j''^T Z_j'' U_j^T d\varsigma \end{aligned} \tag{86}$$

with obvious meaning for the notations involved. Here

$$\begin{aligned} U_j &= \left[w_j^T \quad u_j^T \quad w_{j+1}^T \quad u_{j+1}^T\right]^T \\ \text{and} \quad Z_j'' &= [M_1^{''''} \quad M_2'' \quad M_3'' \quad M_4''], \end{aligned} \tag{87}$$

with M_{1-4}'' easy to derive from (14)–(17).

If we introduce the vector of unknowns for one spacecraft:

$$x = \left[w_2^T \quad u_2^T \quad \ldots \quad w_{M+1}^T \quad u_{M+1}^T\right]^T, \tag{88}$$

J is expressed as

$$J = x^T Q x + b^T x + c. \tag{89}$$

The contribution of the integral terms in (86) to the quadratic term in x in the J formula, $x^T Q x$, is easy to ascertain: all the intermediate segments (that is from j = 2 to j = M) fully contribute whereas the first (j = 1) and last (j = M + 1) segments contribute partially. In order to determine these terms contributions we evaluate the corresponding integrals as follows. The first segment leads to:

$$\begin{aligned} \int_{\varsigma_1}^{\varsigma_2} U_1^T Z_1''^T Z_1'' U_1^T d\varsigma &= [I^T \quad x_1^T] \left(\int_0^{\varsigma_2} Z_1''^T Z_1'' d\varsigma \right) [I^T \quad x_1^T]^T \\ &= x_1^T S_1 x_1 + q_1^T x_1 + p_1 \end{aligned} \tag{90}$$

where we used the following notation: $I = [w_1^T \quad u_1^T]^T$, $x_1 = [w_2^T \quad u_2^T]^T$ and

$$\begin{aligned} S_1 &= \int_0^{\varsigma_2} [M_3''^T \quad M_4''^T]^T [M_3'' \quad M_4''] d\varsigma \\ \text{with} \quad M_3'' &= \frac{-12\varsigma + 6\varsigma_2}{\varsigma_2^3} I_3, \\ M_4'' &= \frac{6\varsigma - 2\varsigma_2}{\varsigma_2^2} I_3. \end{aligned} \tag{91}$$

It is evident that $S_1 \geq 0$. We shall prove that $S_1 > 0$. Assume that S_1 is only semi-positive definite. Then there should exist $z = [z_1^T \quad z_2^T] \neq 0$ such that $z^T S_1 z = 0$, which leads to

$$\begin{aligned} [M_3'' \quad M_4'']z &= 0 \\ \Leftrightarrow \quad & \left[\frac{-12\varsigma + 6\varsigma_2}{\varsigma_2^3} I_3 \quad \frac{6\varsigma - 2\varsigma_2}{\varsigma_2^2} I_3 \right] \\ \times \left[z_1^T \quad z_2^T\right]^T &= 0 \quad \text{for} \quad \varsigma \in [0, \quad \varsigma_2]. \end{aligned} \tag{92}$$

But this actually leads to $z = 0$ (for example let $\varsigma = 0.5\varsigma_2$ to get $z_2 = 0$ and $\varsigma = \frac{\varsigma_2}{3}$ to get $z_1 = 0$), in contradiction with the assumption that $z \neq 0$. This proves that $S_1 > 0$.

In a similar manner, the last segment integral leads to:

$$\begin{aligned}
&\int_{\varsigma_{M+1}}^{\varsigma_{M+2}} U_{M+1}^T {Z''_{M+1}}^T Z''_{M+1} U_{M+1}^T d\varsigma \\
&= [x_{M+1}^T \quad F^T] \left(\int_{\varsigma_{M+1}}^{1} {Z''_{M+1}}^T Z''_{M+1} d\varsigma \right) \left[x_{M+1}^T \quad F^T\right]^T \\
&= x_{M+1}^T S_{M+1} x_{M+1} + q_{M+1}^T x_{M+1} + p_{M+1} \qquad (93)
\end{aligned}$$

where $F = [w_{M+2}^T \quad u_{M+2}^T]^T$, $x_{M+1} = [w_{M+1}^T \quad u_{M+1}^T]^T$ and

$$S_{M+1} = \int_{\varsigma_{M+1}}^{1} [M_1''^T \quad M_2''^T]^T [M_1'' \quad M_2''] d\varsigma \qquad (94)$$

with

$$\begin{aligned}
M_1'' &= \frac{12\varsigma - 6(1 + \varsigma_{M+1})}{(1 - \varsigma_{M+1})^3} I_3, \\
M_2'' &= \frac{6\varsigma - 2(\varsigma_{M+1} + 2)}{(1 - \varsigma_{M+1})^2} I_3.
\end{aligned}$$

Obviously, $S_{M+1} \geq 0$. Assume that S_{M+1} is only semi-positive definite. Then there should exist $z = [z_1^T \quad z_2^T] \neq 0$ such that $z^T S_{M+1} z = 0$, which leads to

$$\begin{aligned}
&\left[M_1'' \quad M_2''\right] z = 0 \\
&\Leftrightarrow \left[\frac{12\varsigma - 6(1 + \varsigma_{M+1})}{(1 - \varsigma_{M+1})^3} I_3 \quad \frac{6\varsigma - 2(\varsigma_{M+1} + 2)}{(1 - \varsigma_{M+1})^2} I_3 \right] \\
&\times \ [z_1^T \quad z_2^T]^T = 0 \quad \text{for} \quad \varsigma \in [\varsigma_{M+1}, \quad 1].
\end{aligned}$$

But this actually leads to $z = 0$, in contradiction with the assumption that $z \neq 0$. This proves that $S_{M+1} > 0$.

Summarizing the above discussion we can write:

$$\begin{aligned}
x^T Q x &= \frac{1}{T^3} \sum_{j=2}^{M} \int_{\varsigma_j}^{\varsigma_{j+1}} r''^T(\varsigma) r''(\varsigma) d\varsigma \\
&+ \ x_1^T S_1 x_1 + x_{M+1}^T S_{M+1} x_{M+1} \\
\text{with } S_1 &> 0 \text{ and } S_{M+1} > 0. \qquad (95)
\end{aligned}$$

It is evident that $Q \geq 0$. If we assume that Q is only positive semi-definite, there should exist $x \neq 0$ such that $x^T Q x = 0$. Using (95) this leads to

$$\begin{aligned}
0 &= \frac{1}{T^3} \sum_{j=2}^{M} \int_{\varsigma_j}^{\varsigma_{j+1}} r''^T(\varsigma) r''(\varsigma) d\varsigma \\
&+ \ x_1^T S_1 x_1 + x_{M+1}^T S_{M+1} x_{M+1} \qquad (96)
\end{aligned}$$

which holds only if on all segments $j = 2$ to M the acceleration $(r''(\varsigma))$ is zero and $x_1 = x_{M+1} = 0$. This leads to: $u_2 = ... = u_{M+1} = 0$ and $w_2 = ... = w_{M+1} = 0$ or $x = 0$ in contradiction with the initial assumption. Hence $Q > 0$.

Acknowledgements

This work was supported by NASA JPL under Contract No. NAS3-02180. Fred Hadaegh, Scott Ploen, and Daniel Scharf served as the technical monitors and their valuable support is gratefully acknowledged.

References

Barraquand, J., Kavraki, L. E., Latombe, J., Li. T., Motwani, R., and Raghavan, P. (1997). A random sampling scheme for path planning. *International Journal of Robotics Research*, **16**(6): 759–774.

Beard, R. and McLain, T. (2001). Fuel optimization for constrained rotation of spacecraft formations. *AIAA Journal of Guidance, Control, and Dynamics*, **23**(2): 339–346.

Beichman, C. A. (1998). The Terrestrial Planet Finder – the search for life-bearing planets around other stars. *Proceedings of Astronomical Interferometry Meeting*, Kona, HI. In *SPIE Proc.* 3350, Bellingham, WA, pp. 719–723.

Bristow, J., Folta, D., and Hartman, K. (2000). A formation flying technology vision. *AIAA Space Conference*, Long Beach, CA, AIAA Paper No. 2000-5194.

Cerven, W. T., Bullo, F., and Coverstone, V. L. (2003). Vehicle motion planning with time-varying constraints. *AIAA Journal of Guidance, Control, and Dynamics*, **27**(3): 506–509.

Clements, J. (2002). Optimal simultaneous pairwise conflict resolution maneuvers in air traffic management. *AIAA Journal of Guidance, Control, and Dynamics*, **25**(4): 815–818.

Das, A. and Cobb, R. (1998). TechSat21-space missions using collaborating constellations of satellites. *Proceedings of the 12th AIAA/USU Annual Conference on Small Satellites*, Utah State University, Logan, UT. A99-10826 01-20 AIAA, Reston, VA.

Dunbar, W. and Murray, R. (2002). Model predictive control of coordinated multi-vehicle formations. *Proceedings of the Conference on Decision and Control*, Las Vegas, NV.

Faiz, N., Agrawal, S., and Murray, R. (2001). Trajectory planning of differentially flat systems with dynamics and inequalities. *AIAA Journal of Guidance, Control, and Dynamics*, **24**(2): 219–227.

Folta, D. and Quinn, D. (1997). Enhanced formation flying for the Earth Observing-1 (EO-1) new millenium mission. *Proceedings of the Flight Mechanics Symposium*, NASA Goddard Space Flight Center, Greenbelt, MD, pp. 405–406.

Foskey, M., Garber, M., Lin, M., and Manocha, D. (2001). A Voronoi-based hybrid motion planner. *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems*, Wailea, HI.

Frazzoli, E., Mao, Z., Oh, J., and Feron, E. (2001). Resolution of conflicts involving many aircraft via semidefinite programming. *AIAA Journal of Guidance, Control, and Dynamics*, 24(1): 79–86.

Frazzoli, E., Dahleh, M. A., and Feron, E. (2002). Real-time motion planning for agile autonomous vehicles. *AIAA Journal of Guidance, Control, and Dynamics*, **25**(1): 116–129.

Fridlund, C. V. M. (2000). Darwin – the infrared space interferometry mission. *ESA Bulletin*, **103**: 20–63.

Garber, M. and Lin, M. (2002). Constraint based motion planning using Voronoi diagrams. *Proceedings of 5th International Workshop on Algorithmic Foundations of Robotics* (*WAFR*), Nice, France.

Hsu, D., Latombe, J., and Motwani, R. (1999). Path planning in expansive configuration spaces. *International Journal of Computational Geometry and Applications*, **9**(4-5): 495–512.

Hsu, D., Kindel, R., Latombe, J., and Rock, S. (2000). Randomized kinodynamic motion planning with moving obstacles. *International Journal of Robotics Research*, **21**(3): 233–255.

Hu, J., Prandini, M., and Sastry, S. (2002). Optimal coordinated maneuvers for three-dimensional aircraft conflict resolution. *AIAA Journal of Guidance, Control, and Dynamics*, **25**(5): 888–900.

Inalhan, G., Tillerson, M., and How, J. (2002). Relative dynamics and control of spacecraft formations in eccentric orbits. *AIAA Journal of Guidance, Control, and Dynamics*, **25**(1): 48–59.

Kavraki, L.E., Svestka, P., Latombe, J. C., and Overmars, M. H. (1996). Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE Transactions on Robotics and Automation*, **12**(4): 566–580.

Kavraki, L., Kolountzakis, M., and Latombe, J. (1998). Analysis of probabilistic roadmaps for path planning. *IEEE Transactions on Robotics and Automation*, **14**(1): 166–171.

Kim, Y., Mesbahi, M., and Hadaegh, F. (2003). Dual-spacecraft formation flying in deep space: optimal collision-free reconfiguration. *AIAA Journal of Guidance, Control, and Dynamics*, **26**(2): 375–379.

Labeyrie, A. (1978). Stellar interferometry methods. *Annual Revue of Astronomy and Astrophysics*, **16**: 77–102.

Li, H. and Williams, T. (2004). Uses of solar radiation pressure for reconfiguration of Sun-Earth libration point formations. *AIAA Guidance, Navigation, and Control Conference and Exhibition*, Providence, RI, AIAA Paper No. 2004-4785.

Lygeros, J., Godbole, D., and Sastry, S. (1998). A verified hybrid controller for automated vehicles. *IEEE Transactions on Automatic Control*, **43**(4): 522–539.

Luenberger, D. G. (1984). *Linear and Nonlinear Programming*. Addison-Wesley, Reading, MA.

Manno, V. and Page, D. E. (eds) (1969). *Intercorrelated Satellite Observations Related to Solar Events*. Springer Verlag/D. Reidel, New York/Dordrecht.

McQuade, F., Ward, R., and McInnes, C. (2002). The autonomous configuration of satellite formations using generic potential functions. *Proceedings of the International Symposium Formation Flying*, Toulouse, France.

Mesbahi, M. and Hadaegh, F. (2001). Formation flying control of multiple spacecraft via graphs, matrix inequalities, and switching. *AIAA Journal of Guidance, Control, and Dynamics*, **24**(2):369-377.

Phillips, J. M., Kavraki, L. E., and Bedrosian, N. (2003). Spacecraft rendezvous and docking with real-time randomized optimization. *Proceedings of AIAA Guidance, Navigation, and Control Conference*, Austin, TX.

Prasanth, R., Boskovic, J., and Mehra, R. K. (2002). Mixed Integer/LMI Programs for low-level path planning. *Proceedings of the American Control Conference*, Anchorage, AL.

Rathbun, D., Kragelund, S., and Pongpunwattana, A., (2002). An evolution based path planning algorithm for autonomous motion of a UAV through uncertain environments. *Proceedings of the Inernational. Symposium Formation Flying*, Toulouse, France.

Richards, A. and How, J. (2002). Aircraft trajectory planning with collision avoidance using mixed integer linear programming. *Proceedings of the American Control Conference*, Anchorage, AK.

Richards, A., Schouwenaars, T., How, J., and Feron, E. (2002). Spacecraft trajectory planning with avoidance constraints using mixed integer linear programming. *AIAA Journal of Guidance, Control, and Dynamics*, **25**(4): 755–764.

Saber, R., Dunbar, W., and Murray, R. (2003). Cooperative control of multi-vehicle systems using cost graphs and optimization. *Proceedings of the Automatic Control Conference*.

Scharf, D., Hadaegh, F., and Kang, B. (2002). A survey of spacecraft formation flying guidance. *Proceedings of the Intl. Symposium Formation Flying,* Toulouse, France.

Seereram, S., Li, E., Ravichandran, B., Mehra, R., K., Smith, R., and Beard, R. (2000). Multispacecraft formation initialization using genetic algorithms techniques. *23^{rd} Annual AAS Guidance and Control Conference*, Breckenridge, CO.

Singh, G. and Hadaegh, F. (2001). Collision avoidance guidance for formation flying applications. *AIAA Guidance, Navigation, and Control Conference and Exhibition*, Montreal, Canada.

Sultan, C., Seereram, S., and Mehra, R. K. (2006). Energy sub-optimal collision-free path planning for spacecraft formation flying. *AIAA Journal of Guidance, Control, and Dynamics*, **29**(1): 190–192.

Tillerson, M., Inalhan, G., and How, J. (2002). Coordination and control of distributed spacecraft systems using convex optimization techniques. *International Journal of Robust and Nonlinear Control*, **12**(2): 207–242.

Tomlin, C., Pappas, G., and Sastry, S. (1998). Conflict resolution for air traffic management: a study in multi-agent hybrid systems. *IEEE Transactions on Automatic Control*, **43**(4): 509–521.

Yang, G., Yang, Q., Kapila, V., Palmer, D., and Vaidyanathan, R. (2002). Fuel optimal maneuvers for multiple spacecraft formation reconfiguration using multi-agent optimization. *International Journal of Robust and Nonlinear Control*, **12**: 243–283.

Zhu, G., Grigoriadis, K. M., and Skelton, R.E. (1995). Covariance control design for the Hubble Space Telescope. *AIAA Journal of Guidance, Control, and Dynamics*, **18**(2): 230–236.