CEAS AIDAA
3rd CEAS Air&Space Conference
21st AIDAA Congress

Autonomous Capture of Non-Cooperative Spacecraft with a Space Free-Flyer

Cesare Guariniello and Luigi Ansalone and Fabio Curti
DIAEE - Sapienza University of Rome, Italy

Keywords: space, robotics, capture, virtual manipulator

Abstract

This work deals with the problem of performing rendezvous and capture of a non-cooperative spacecraft by means of a space free-flyer, i.e. a satellite base equipped with a robotic manipulator. Though this kind of manoeuvres addresses the solving of relevant existing problems such as debris removal, satellite servicing, orbit changing, only few spaceborne experiments have been conducted, all keeping strong working hypothesis. In this work, a few techniques and strategies to obtain more realistic algorithms, taking into account relative motion and computational load, are presented. An orbit and attitude dynamics simulator has been developed to experiment the proposed strategies.

1 Introduction

Since the launch of the first artificial satellite, Sputnik 1, in 1957, more than 6000 satellites have been put in orbit, 3000 still orbiting Earth together with 12000 space debris.

Many issues may occur due to this enormous population of space waste objects. Deorbiting manoeuvres are not always provided for, or cannot be executed due to failures, thus causing hazards or expensive manoeuvres for incoming spacecraft. Furthermore, satellites may need simple servicing operations, like refueling or deployment of entangled structures, or recovery from failed orbit insertions, or orbit changes that require external help. The main idea that has inspired the present work is the possibility of having a robotic space vehicle equipped with manipulators, a free-flyer, to approach and grab a spacecraft for executing planned operations. While autonomous rendezvous and docking manoeuvres have been developed and realized many times, the use of robotic manipulators in space has always been limited to human-controlled handling and berthing of structures. In this paper a study on the autonomous capture of non-cooperative satellites is presented. A chaser, equipped with a robotic arm, must approach a tumbling non-cooperative target endowed with a grapple fixture and capture it, as shown in Fig. 1.

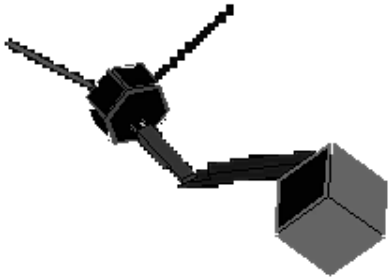

Figure 1: **The free-flyer captures the target (V-bar and R-bar shown)**

The dynamics of the manipulator is quite dif-

ferent from the dynamics of robotic arms with a fixed base; in fact in space the motion of a joint affects the attitude motion of the vehicle on which the manipulator is attached. According to the concept of the Virtual Manipulator and the Virtual Ground, the dynamics of the whole multi-body chaser has to be referred to the center of mass of the system, which moves when external forces are applied and when the capture is executed. Previous works, like [6] and [5], solved this problem using inertial wheels to keep the base attitude fixed with respect to an inertial reference frame, thus simulating a fixed base but wasting three degrees of freedom. The dynamics of a space free-flyer also involves heavy computational loads: an algorithm to reduce the load has been proposed and tested. The problem of the trajectory planning to guide a robotic manipulator to rendezvous and capture a non-cooperative target satellite is closely related to the application of the vision in the robotic capture of moving objects with known dynamics. In the study, the chaser is supposed to have a certain quantity of information about the target, i.e. an estimation of the rotational state and the geometry, the mass and the moment of inertia tensor. A fly-around strategy has been implemented, thus simulating the presence of a vision system apt to get and/or increase the quantity of information before starting the final capture manoeuvre. Capturing strategies for a free-floating space robot in grasping of a tumbling target with model uncertainty are presented. The proposed strategies take into account the relative orbital motion between the chaser and the target, thus dropping the flat-space assumption used in [5].

2 The free-flyer[4]

2.1 Historical Background

Although studied since the early nineties, space robotics has almost entirely been reduced to heavy robotic manipulators attached to much heavier spacecraft (e.g. the *RMS* onboard the Space Shuttle or the Canadarm2 on the *ISS*); this choice allows to avoid the difficulty concerning the kinematics and the dynamics of a space free flyer, because due the heaviness of the base, this can be regarded as fixed with reference to the center of mass of the system. Nevertheless this solution causes two problems: first of all, heavier payloads mean more expensive launches, while a free-flyer can be a medium sized satellite; in addition, large robotic manipulators in space are always human-operated, being the decisions left to astronauts and the computational duties put in the computers' charge.

During last twenty years, many articles have been published about different techniques to perform operations with a free-flyer, but such a wide field of study has never been deeply explored and tested: Japanese satellite ETS-VII ([5]) was the first and up to date the only space-borne experiment that performed rendezvous and capture operations with a space free-flyer. After that, many researchers suggested strategies based on a certain number of hypothesis to make the study easier, but sometimes losing the possible advantages of the free-flyer (e.g., the hypothesis of flat space i.e. lack of relative motion between the free-flyer and the target may lead not to use this relative motion to get a better position of the arm with respect to the grapple fixture).

2.2 Kinematics

To study the motion and the geometry of a space system, is advisable decoupling the equations describing the orbital motion, i.e. the motion of the Center of Mass (CM) of the system, and the equations describing the attitude motion, i.e. the motion around the CM. A space free-flyer, however, is a multi-body system with moving link: therefore, its CM is not a physical point of the system; nevertheless, the position of the CM can be determined as a function of the base attitude and of the joint variables, i.e. the angles of the revolute joints and the shift of the prismatic joints.

2.2.1 The Virtual Manipulator

The free-flyer robotic system is represented by a base whose center of mass CM_0 is identified by means of the vector ${}^c\vec{r}_{c_0}$ with reference to the CM of the whole system. Similarly, ${}^c\vec{r}_{c_i}$ is

the position vector of the center of mass CM_i of the i-th link of the manipulator. These vectors are defined with respect to an in-orbit inertial reference frame RF_c centered in CM.

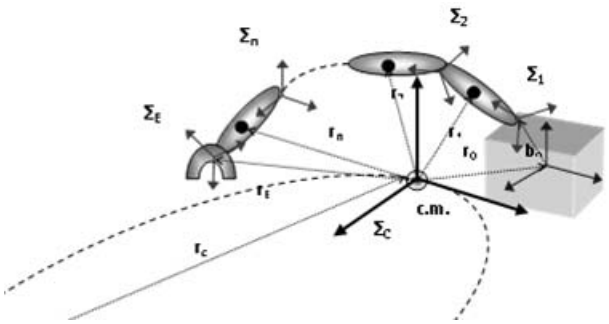

Figure 2: **The free-flyer kinematics**

For each link, in addition to the Denavit-Hartenberg parameters, two more vectors are introduced: ${}^c\vec{a}_i$ is the position of CM_i with respect to the origin of the i-th reference frame RF_i, expressed in RF_c; ${}^c\vec{b}_i$ is the position of the origin of the i+1-th reference frame RF_{i+1} with respect to CM_i, expressed in RF_c.

As for the platform, RF_0 is the reference frame centered in CM_0 and having its axes along the principal axes of inertia of the platform.

According to the given definitions of the position vectors, the following relations between the ${}^c\vec{r}_{c_i}$ can be written:

$$\begin{cases} {}^c\vec{r}_{c_1} = {}^c\vec{r}_{c_0} + {}^c\vec{b}_0 + {}^c\vec{a}_1 \\ \vdots \\ {}^c\vec{r}_{c_i} = {}^c\vec{r}_{c_{i-1}} + {}^c\vec{b}_{i-1} + {}^c\vec{a}_i \\ \vdots \\ {}^c\vec{r}_{c_n} = {}^c\vec{r}_{c_{n-1}} + {}^c\vec{b}_{n-1} + {}^c\vec{a}_n \end{cases} \quad (1)$$

Such equations are recursive, ${}^c\vec{r}_{c_0}$ being the only unknown quantity.

Let m_0 be the base mass and m_i the mass of the i-th link; being the ${}^c\vec{r}_{c_i}$ referred to the CM, the static moment of the whole system is equal to zero:

$$m_0\ {}^c\vec{r}_{c_0} + \ldots + m_i\ {}^c\vec{r}_{c_i} + \ldots + m_n\ {}^c\vec{r}_{c_n} = 0 \quad (2)$$

Given the Eq. (1),

$${}^c\vec{r}_{c_0} = \sum_{i=0}^{n} c_i^{(0)} \left({}^c\vec{b}_i + {}^c\vec{a}_{i+1} \right) \quad (3)$$

being

$$M = \sum_{i=0}^{n} m_i \quad (4)$$

$$c_i^{(0)} = \frac{\sum_{k=0}^{i} m_k}{M} - 1 \quad (5)$$

Calling

$$c_i = \frac{\sum_{k=0}^{i} m_k}{M} \quad (6)$$

the position of each CM can be calculated as

$${}^c\vec{r}_{c_k} = \sum_{i=0}^{n} c_i^{(k)} \left({}^c\vec{b}_i + {}^c\vec{a}_{i+1} \right) \quad (7)$$

being

$$c_i^{(k)} = \begin{cases} c_i & i < k \\ c_i^{(0)} = c_i - 1 & i \geq k \end{cases} \quad (8)$$

Since the position of the end effector (EE) is ${}^c\vec{r}_{EE} = {}^c\vec{r}_{c_n} + {}^c\vec{b}_n$, the expression of the *direct kinematics* of the space free-flyer is

$${}^c\vec{r}_{EE} = \sum_{i=0} nc_i \left({}^c\vec{b}_i + {}^c\vec{a}_{i+1} \right) \quad (9)$$

Comparing the expression of the direct kinematics with the geometric path given by ${}^c\vec{r}_{EE} = {}^c\vec{r}_0 + {}^c\vec{b}_0 + {}^c\vec{a}_1 + {}^c\vec{b}_1 + \ldots + {}^c\vec{a}_n + {}^c\vec{b}_n$, it can be noticed that the position of the EE with respect to the CM of the whole system is equivalent to that of a virtual manipulator with the base located in CM and links parallel to that of the real manipulator, but having the lengths ${}^c\vec{a}_i$ and ${}^c\vec{b}_i$ scaled by the coefficients c_i. In this virtual description, the base of the real manipulator is represented by means of a spherical joint.

2.3 Differential Kinematics

The linear velocity of the EE, ${}^c\vec{v}_E$ can be expressed as a function of the linear and angular velocities of the base, ${}^c\vec{v}_{c_0}$ and $\vec{\omega}_0$, and of the velocity of each joint $\dot{q}_i$ as follows:

$${}^c\vec{v}_E = {}^c\vec{v}_{c_0} + \vec{\omega}_0 \times ({}^c\vec{r}_{EE} - {}^c\vec{r}_{c_0}) + \sum_{i=1}^{n} J_{L_i} \dot{q}_i \quad (10)$$

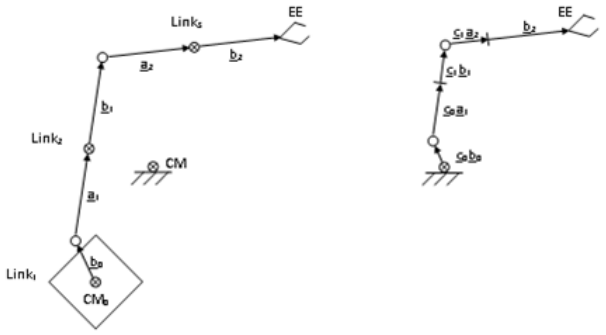

Figure 3: **The virtual manipulator**

being J_{L_i} column arrays based on the joint type:

$$J_{L_i} = \begin{cases} {}^c\hat{z}_i & Pris \\ {}^c\hat{z}_i \times ({}^c\vec{r}_{EE} - {}^c\vec{r}_i) & Rev \end{cases} \quad (11)$$

where ${}^c\vec{r}_i$ is the position of the origin or RF_i with respect to the origin of RF_c, i.e. ${}^c\vec{r}_i = {}^c\vec{r}_{c_{i-1}} + {}^c\vec{b}_{i-1}$.
Let be ${}^c\vec{r}_{OE} = {}^c\vec{r}_{EE} - {}^c\vec{r}_{c_0}$ and let also be

$$\vec{v} = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} \Rightarrow \tilde{v} = \vec{v}\times = \begin{bmatrix} 0 & -v_3 & v_2 \\ v_3 & 0 & -v_1 \\ -v_2 & v_1 & 0 \end{bmatrix} \quad (12)$$

Then the linear velocity of the *EE* can be written as

$${}^c\vec{v}_E = {}^c\vec{v}_{c_0} - {}^c\tilde{r}_{OE}\vec{\omega}_0 + J_L\dot{\vec{q}} \quad (13)$$

being

$$J_L = \begin{bmatrix} J_{L_1} & \dots & J_{L_n} \end{bmatrix} \qquad \dot{\vec{q}} = \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \vdots \\ \dot{q}_n \end{bmatrix} \quad (14)$$

To calculate ${}^c\vec{r}_{c_0}$, assuming that *no external force is acting on the free-flyer*, the conservation of momentum gives, starting with initial momentum equal to zero, $m_0\,{}^c\vec{v}_{c_0} + m_1\,{}^c\vec{v}_{c_1} + \dots + m_n\,{}^c\vec{v}_{c_n} = 0$.

Since for the i-th link

$${}^c\vec{v}_{c_i} = {}^c\vec{v}_{c_0} + \vec{\omega}_0 \times ({}^c\vec{r}_{c_i} - {}^c\vec{r}_{c_0}) + J^{(i)}_{L_c}\dot{\vec{q}} \quad (15)$$

where the columns of $J^{(i)}_{L_c}$ are

$$J^{(i)}_{L_{c_j}} = \begin{cases} {}^c\hat{z}_j & j \le i \quad and\ prism \\ {}^c\hat{z}_j \times ({}^c\vec{r}_{c_i} - {}^c\vec{r}_j) & j \le i \quad and\ rev \\ \vec{0} & j > i \end{cases} \quad (16)$$

Due to the conservation of momentum and setting ${}^c\vec{r}_m = \frac{1}{M}\sum_{i=1}^n m_i\vec{\omega}_0 \times ({}^c\vec{r}_{c_i} - {}^c\vec{r}_{c_0})$, the final result is

$${}^c\vec{v}_E = \left(J_L - \frac{1}{M}\sum_{i=1}^n m_i J^{(i)}_{L_c}\right)\dot{\vec{q}} + ({}^c\tilde{r}_m - {}^c\tilde{r}_{OE})\,\vec{\omega}_0 \quad (17)$$

As for the angular velocity of the *EE*,

$${}^c\vec{\omega}_E = {}^c\vec{\omega}_0 + \sum_{i=1}^n J_{A_i}\dot{q}_i = {}^c\vec{\omega}_0 + J_A\dot{\vec{q}} \quad (18)$$

where the columns of J_A are defined as follows, basing on the joint type:

$$J_{A_i} = \begin{cases} \vec{0} & Prism \\ {}^c\hat{z}_i & Rev \end{cases} \quad (19)$$

Thus, the final expression for the *differential kinematics* of the free-flyer is

$$\begin{bmatrix} {}^c\vec{v}_E \\ {}^c\vec{\omega}_E \end{bmatrix} = J_{FF}\begin{bmatrix} \dot{\vec{q}} \\ \vec{\omega}_0 \end{bmatrix} \quad (20)$$

being J_{FF} the Jacobian of the free-flyer:

$$J_{FF} = \left[\begin{array}{c:c} \underbrace{J_L - \frac{1}{M}\sum_{i=1}^n m_i J^{(i)}_{L_c}}_{J_A} & \underbrace{{}^c\tilde{r}_m - {}^c\tilde{r}_{OE}}_{I_{3x3}} \end{array}\right] \quad (21)$$

2.4 Dynamics

The equations of the free-flyer dynamics can be derived using an approach based on the lagrangian mechanics. Assuming that there is not any conservative forces acting on the spacecraft (i.e., neglecting the effects of gravity gradient), the lagrangian term is only the kinetic energy T of the free-flyer:

$$T = \frac{1}{2}\sum_{i=0}^{n}\left({}^{c}\vec{\omega}_i^T \; {}^{c}I_i \; {}^{c}\vec{\omega}_i + m_i \; {}^{c}\vec{v}_{c_i}^T \; {}^{c}\vec{v}_{c_i}\right) \quad (22)$$

The expression for ${}^{c}\vec{v}_{c_i}$ is described in Eq. (15). The angular velocity of the i-th link is ${}^{c}\vec{\omega}_i = {}^{c}\vec{\omega}_0 + J_A^{(i)}\dot{\vec{q}}$, where the columns of $J_A^{(i)}$ are defined as follows, basing on the joint type:

$$J_{A_j}^{(i)} = \begin{cases} \vec{0} & j \leq i \quad \textit{and prismatic} \\ {}^{c}\hat{z}_j & j \leq i \quad \textit{and revolute} \\ \vec{0} & j > i \end{cases} \quad (23)$$

Taking into account the equations of differential kinematics,

$$T = \frac{1}{2}\begin{bmatrix} \vec{\omega}_0^T & \dot{\vec{q}}^T \end{bmatrix}\begin{bmatrix} H_{\omega\omega} & H_{\omega q} \\ H_{q\omega} & H_{qq} \end{bmatrix}\begin{bmatrix} \vec{\omega}_0 \\ \dot{\vec{q}} \end{bmatrix} \quad (24)$$

being H_{FF} the *generalized inertia matrix* of the space free-flyer.

The blocks $H_{\omega\omega}$, $H_{\omega q}$, $H_{q\omega}$ and H_{qq} can be calculated using the equations of the kinematics and differential kinematics.

The classical lagrangian approach then allows to calculate the evolution of the dynamics, given the torques $\vec{\tau}$ acting on the spacecraft:

$$\frac{d}{dt}\left[\frac{\partial T}{\partial \dot{\vec{q}}}\right] - \frac{\partial T}{\partial \vec{q}} = \vec{\tau} \quad (25)$$

However, in the free-flyer dynamics equations the base angular velocity $\vec{\omega}_0$ constitutes a non-holonomic constraint (causing the possibility to get to a previous configuration without reaching the previous state), thus not integrable.

A different approach, based on the so-called *quasi-coordinates* ([3]) allows to solve the problem of the derivation of the lagrangian term.

2.4.1 Quasi-Coordinates Lagrangian Approach

When the equations of motion are not restricted to *true coordinates* (e.g., the derivatives of the joint variables), they use non integrable variables, called *quasi-coordinates*, that can be expressed as

$$\omega_s = \alpha_{1s}\dot{q}_1 + \alpha_{2s}\dot{q}_2 + \ldots + \alpha_{ns}\dot{q}_n = \sum_{i=1}^{n}\alpha_{is}\dot{q}_i \quad (26)$$

where the coefficients α_{rs} are function of the generalized coordinates q. In the matrix form, $\vec{\omega_{ql}} = \alpha^T\dot{\vec{q}}$. Assuming that the matrix α is not singular, $\dot{\vec{q}} = \beta\vec{\omega_{ql}}$ with $\beta\alpha^T = I$.

Calculating the relations between the kinetic energy as function of the true coordinates and the kinetic energy T as function of the quasi-coordinates allows to write the equations of dynamics in the integrable form

$$\begin{aligned}\alpha\tfrac{d}{dt}\left[\tfrac{\partial T}{\partial \vec{\omega}_{ql}}\right] + \left[\vec{\omega}_{ql}^T\beta^T\left\{\tfrac{\partial \alpha}{\partial \vec{q}}\right\}\right]\tfrac{\partial T}{\partial \vec{\omega}_{ql}} - \tfrac{\partial T}{\partial \vec{q}} + \\ - \left[\vec{\omega}_{ql}^T\beta^T\left[\tfrac{\partial \alpha}{\partial \vec{q}}\right]\right]\tfrac{\partial T}{\partial \vec{\omega}_{ql}} = \vec{\tau}\end{aligned} \quad (27)$$

where $\left\{\frac{\partial \alpha}{\partial \vec{q}}\right\}$ does not involve summation over the indices of α while $\left[\frac{\partial \alpha}{\partial \vec{q}}\right]$ does.

The computational load of such a calculation is huge, involving inversion of 6-by-6 matrices.

2.5 Inverse Kinematics

When designing a robotic manipulator control system, there is a fundamental choice between control in the joint space and control in the task space; nevertheless, even when choosing a centralized control based on the whole dynamics of the system, i.e. in the task space, the desired trajectory is converted in joint positions, velocities and accelerations.

The inverse kinematics allow to calculate the joint variables, given a desired pose of the *EE*. While the inverse differential kinematics is based on a simple inversion of a matrix, the inverse kinematics is not always easily derivable in an analytical form; thus, a numeric algorithm has to be implemented to realize the inverse kinematics.

The numeric algorithms are based on the analytical Jacobian matrix of the direct kinematics map $\vec{r} = f_r(\vec{q})$:

$$J_r(\vec{q}) = \frac{\partial f_r(\vec{q})}{\partial \vec{q}} \quad (28)$$

The easiest way to calculate the analytical Jacobian matrix, however, is through the geometric Jacobian J_G, which is the matrix mapping the derivatives of the joint variables into the linear and angular velocities of the *EE*:

$$\begin{bmatrix} \dot{\vec{r}}_{EE} \\ \vec{\omega}_E \end{bmatrix} = J_G \left[\dot{\vec{q}}\right] \tag{29}$$

The Jacobian Matrix J_{FF} in Eq. (21) maps the derivatives of the joints variables and the angular velocity of the base into the linear and angular velocities of the *EE*. If we choose a parametric representation of the attitude of the base, e.g. by means of a set of Euler angles $\vec{\phi}$, the relation between angular velocity and derivatives of the Euler angles can be expressed as $\vec{\omega}_0 = T(\vec{\phi})\dot{\vec{\phi}}$ and for the *EE* $\vec{\omega}_E = T(\vec{\phi}_{EE})\dot{\vec{\phi}}_{EE}$. Thus a relation between J_{FF} and J_G is given by

$$\begin{bmatrix} \dot{\vec{r}}_{EE} \\ \vec{\omega}_E \end{bmatrix} = J_{FF} \begin{bmatrix} \dot{\vec{q}} \\ \vec{\omega}_0 \end{bmatrix} = J_{FF} \begin{bmatrix} I & 0 \\ 0 & T(\vec{\phi}) \end{bmatrix} \begin{bmatrix} \dot{\vec{q}} \\ \dot{\vec{\phi}} \end{bmatrix} \tag{30}$$

Thus, $J_G = J_{FF} \begin{bmatrix} I & 0 \\ 0 & T(\vec{\phi}) \end{bmatrix}$.

Similarly, the relation between J_G and J_r is function of $T(\vec{\phi}_{EE})$. In conclusion

$$J_r = \begin{bmatrix} I & 0 \\ 0 & T^{-1}(\vec{\phi}_{EE}) \end{bmatrix} J_{FF} \begin{bmatrix} I & 0 \\ 0 & T(\vec{\phi}) \end{bmatrix} \tag{31}$$

2.5.1 Newton Method

The first numeric method to calculate the inverse kinematics is the Newton method, based on the linearization of the evolution of the direct kinematics $\vec{r} = f_r(\vec{q}) = f_r(\vec{q}^k) + J_r(\vec{q}^k)(\vec{q} - \vec{q}^k) + \circ(\|\vec{q} - \vec{q}^k\|^2)$. The step of the algorithm thus is

$$\vec{q}^{k+1} = \vec{q}^k + J_r^{-1}(\vec{q}^k)\left[r - f_r(\vec{q}^k)\right] \tag{32}$$

The convergence can be reached only if the initial guess is close enough to the real solution; in this case the algorithm has quadratic convergence rate.

2.5.2 Gradient Method

Another method is based on the minimization of the quadratic error and the step is

$$\vec{q}^{k+1} = \vec{q}^k + \alpha J_r^T(\vec{q}^k)\left[r - f_r(\vec{q}^k)\right] \tag{33}$$

The scalar parameter α has to be chosen so as to guarantee a decrease of the error function at each iteration: too large values may lead the algorithm to miss the minimum; on the other side, if α is too small, the convergence is extremely slow. This algorithm never diverges and is computational simpler than the other.

An efficient iterative scheme has been devised by combining initial iterations with Gradient method, sure but slow, then switch to Newton method (quadratic final convergence rate). Choices to be made concern the initial guess, the step size in gradient method and the stopping criteria (cartesian error lower than a certain limit or algorithm increment lower than a certain limit).

3 Proposed strategies and results

The proposed mission scenario involves a target and a chaser orbiting on circular LEO orbits on the same plane at different height; the state of the chaser free-flyer is supposed to be known. The free-flyer is equipped with a 3-links robotic arm.

3.1 Rendezvous Strategy

Faraway operations are similar to those of an automated rendezvous and docking mission. Therefore, standard phasing manoeuvres are used: orbital plane corrections, Hohmann transfers, R-bar transfers ([2]). Nevertheless, the proposed strategy includes two fundamental characteristics that make it quite different from a standard rendezvous manoeuvre:

- while the first phase of manoeuvres is executed using *AGPS*, standard homing manoeuvres are not suitable to get closer to a non-cooperative spacecraft, because there is no possibility to use relative positioning. Such manoeuvres need to be performed

by means of absolute positioning (knowing orbital parameters about the object to be chased) and, when getting closer to the target, by visual systems;

- the trajectory planner needs to know not only relative position and velocity (in order to calculate relative orbital motion), but also the state of motion of the target, in addition to the position of the grapple fixture, when present aboard the target. To achieve the goal of getting information about the state of motion, the chaser will perform a fly-around manoeuvre.

The proposed rendezvous strategy can therefore be describes as follows:

1. the chaser performs orbital correction manoeuvres to get the orbital plane of the target and drifts in order to phase with the target;
2. the chaser performs height change transfer manoeuvres in order to reach a station keeping point a few hundreds of meters behind the target;
3. the chaser performs an R-bar transfer or a V-bar transfer in order to reach a station keeping point a few meters behind the target, at a distance suitable to use the motion estimation visual system;
4. the chaser performs a complete fly-around manoeuvre in order to get starting information about the state of motion of the target. When come back to the starting point of the manoeuvre, capture strategies can be put into execution.

Performing a closing transfer while neglecting the perturbation effects, may lead to a quite large error, as shown in Fig. 4.

3.2 Capture Strategies

Due to the complexity of the many variables involved in an autonomous capture mission, a great number of strategies could be implemented and experimented; nevertheless, a few cases of interest have been identified and studied during the development of this thesis:

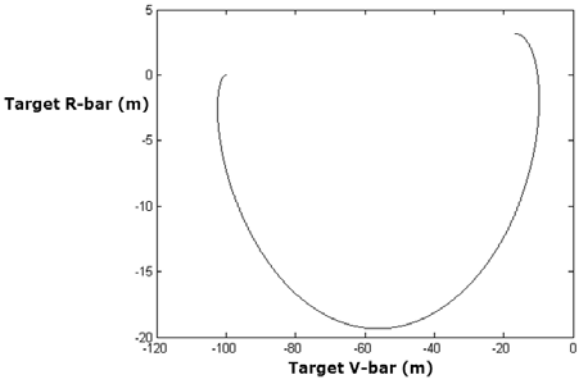

Figure 4: **Closing manoeuvre trajectory with perturbations**

3.2.1 Free-flyer *CM* and Grapple Fixture at Similar Heights

When the state of motion of target causes the grapple fixture to be in such a position that exists a point whence the robotic arm could reach the grapple fixture without moving the free-flyer *CM* out of the target orbit during the *final capture*, the best choice is to manoeuvre the free-flyer as to perform the final capture leaving unaltered the *CM* position with respect to the target.

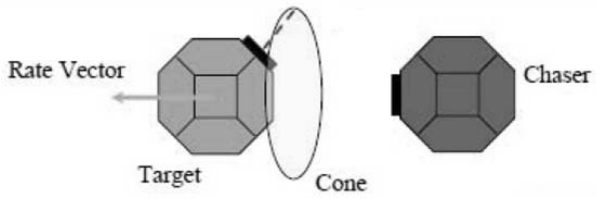

Figure 5: **Direct final capture**

3.2.2 Free-flyer *CM* and Grapple Fixture at a Different Height

When the state of motion of the target causes the grapple fixture to be in such a position that the robotic arm cannot reach it without taking the free-flyerCM out of the target orbit (e.g., the grapple fixture is on the rotation axis or coning near it and far from the H-bar - V-bar plane), the suggested strategy involves a transfer manoeuvre able to exploit the relative motion to

carry the free-flyer *CM* to a point whence the grapple fixture can be reached. The trajectory planner must take into account the relative motion to calculate the required pose and velocity of the *EE* with respect to the grapple fixture; during the execution of the relative motion, the manipulator is controlled in order to reach the desired goal.

Figure 6: **Final capture with relative motion**

3.3 Trajectory Planner

The trajectory planner is activated before the beginning of the final capture manoeuvre and whenever the trajectory needs to be recalculated. The planner needs three inputs:

- current state of the joint variables and of the attitude of the base: determinable by means of proprioceptive sensors;
- manoeuvre execution time: function of the manipulator physical limitations, it must be long enough to guarantee the possibility of replanning. It also can be calculated in function of optimization criteria, taking into account the grapple fixture motion and the relative motion between the chaser and the target;
- final state of the joint variables, of the attitude of the base and of the joint velocities required to execute the capture; they are determined as follows: the relative motion of the spacecraft and the manoeuvre execution time allows to calculate the position and velocity of the *EE* with respect to the free-flyer *CM* required to execute the capture; required velocity of the joints can be obtained by a simple inversion of the geometric Jacobian matrix, while the required joint variables can be calculated through kinematic inversion.

3.3.1 Kinematic Inversion

As stated in section 2.5, an important stress must be given to the choice of the numeric algorithm to use to calculate the inverse kinematics by means of the analytical jacobian: the experiments have in fact shown that the parameter for the gradient method has to be chosen little enough as to guarantee convergence of the algorithm, but this entails very slow convergence, sometimes more than 100 iteration steps and various minutes of computation, as shown in Fig. 7

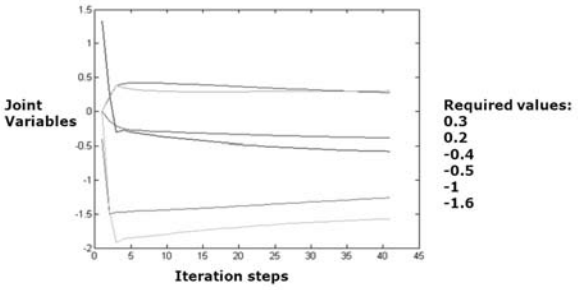

Figure 7: **Gradient Method**

on the other side, the Newton method can sometimes reach the solution with an error lower that 10^{-15} m in less than 10 steps, but can also lead to divergence, as shown in Fig. 8

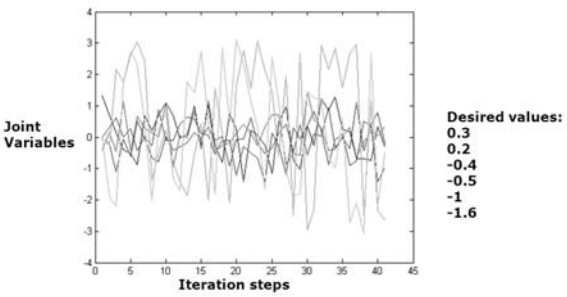

Figure 8: **Newton Method**

thus, an hybrid algorithm has been used: the gradient method, slow but assuring convergence,

is initially used to shift from the current variables, used as the first guess, to variables closer to the required solution; at this point, the Newton method starting from the last step of the gradient algorithm, guarantees convergence at quadratic rate in a few steps (Fig. 9).

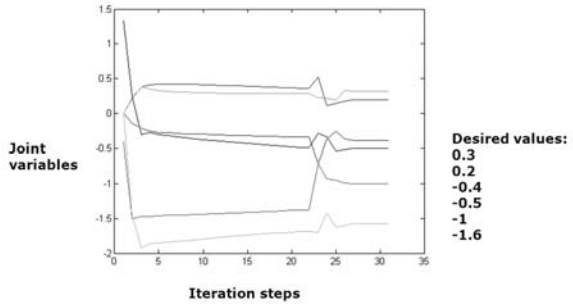

Figure 9: **Hybrid Method**

3.3.2 Planning

Knowing the initial position and velocity and the required final position and velocity for each joint variable, a cubic polynomial represents a solution for this problem.

3.4 Control Laws

After having calculated the desired trajectory for the joints, the free-flyer needs a controller to determine the torques to apply on the base and on the joints in order to execute the desired movement and to perform the capture. Two main families of controllers have been used in this work.

3.4.1 Joint Space Decentralized Control

The use of quasi-coordinates to calculate the dynamics of the manipulator leads to an expression having the form $H(q)\ddot{q} + C(q, \dot{q}, \omega)\dot{q} = \tau$. A controller in the joint space will use only the linear part of the dynamics equations, thus considering the matrix C, i.e. the contribution due to the configuration, as a disturbance. Each joint is controlled as a Single Input Single Output (*SISO*) system; the lack of the gravity term allows to use a simple proportional controller, with position and velocity feedback. The performances of such controllers decay when the desired trajectory is too fast; derivative of the desired motion to compensate the disturbance with a feedforward predictor may lead to improvement of this kind of controller. A delay in the expected trajectory is always present.

3.4.2 Centralized Control

This kind of controller takes into account the non-linear coupling between the joints, considering the free-flyer as a *MIMO* system. The so-called inverse dynamics control allows to decouple and linearize the original system $H(q)\ddot{q} + C(q, \dot{q}, \omega)\dot{q} = u$, with input $u = H(q)y + C(q, \dot{q}, \omega)\dot{q}$ depending on the manipulator state. Choosing a new input $y = -K_P q - K_D \dot{q} + r$, the resulting system is asymptotically stable and decoupled if the matrices K_P and K_D are symmetric and positive-definite. The desired trajectory following is guaranteed by the choice of $r = \ddot{q}_d + K_D \dot{q}_d + K_P q_d$.

Such kind of controllers are suitable to be made robust or adaptive.

3.5 Execution of Manoeuvres

Since the complete dynamics of the free-flyer manipulator is computationally hard to handle, in this work a decoupling between the base attitude motion and the joint motion has been proposed: when the trajectory planner calculates the desired trajectory, the joint motion is simulated in absence of platform control; this technique allows to calculate the base rotation due to the joint movement. If only the joints would rotate, the base would get an attitude R_x. The system then computes the *over-rotation* between the current attitude of the base and R_x and the base is actuated with such torques as to get to the desired final attitude minus the calculated over-rotation. Having the free-flyer moved as a whole body (the joints were not actuated during the base motion), the relative position of the joints with respect to the base has not changed: thus, realizing the previously simulated joint movement will add the over-rotation, taking the base to the desired attitude. The to-

tal manoeuvre time of the joints must be scaled down to take into account the separate base rotation and it has been experimented that this scaling does not affect the final attitude of the base.

3.6 Results

As an example of the obtained results, two tables containing the position error (centimeters) of the *EE* at the end of the capture manoeuvres are shown.

Table 1: Decentralized control

	Base case	Perturb.	Fast dyn
Fixed target	0.3 (delay)	1.2 (delay)	0.8 (delay)
V-bar rot	2.2 (delay)	2.5 (delay)	7.1 (delay)
H-bar rot	2.1 (delay)	3.8 (delay)	8.4 (delay)

Table 2: Centralized control with 2 steps

	Base case	Perturb. no feedback	Exstim. errs. no feedback
Fixed target	0.1	4.6	0.8
V-bar rot	1.2	5.0	2.0
H-bar rot	1.2	6.1	5.5

3.7 Comparison

Experimental comparison of various algorithms provides that:

- decentralized algorithms is computationally the lightest, but requires a sampling dynamics faster than the system dynamics and provides delay;
- centralized algorithms are stable due to feedback, but the inversion of the complete dynamics equations (involving inversion of 6-by-6 matrices) are impossible to be used in real-time.
- centralized algorithms with the proposed 2-step manoeuvre are computationally lighter than the complete dynamics inversion (involving two 3-by-3 matrices inversion) and achieve good results; such kind of manoeuvres are more sensible to rapid dynamics than the previous algorithm, since they reduce the capture manoeuvre time.

4 Conclusion

In conclusion, a rendezvous and capture mission with a free-flyer has been studied, designed and tested with a newly implemented orbit and attitude dynamics multisatellite simulator. A few problems of previous studies and mission have been addressed to and solved, such as kinematics inversion, use of relative motion, dynamics computational load. Future studies will involve optimization criteria for the strategies, robust and adaptive control and multiple-arms free-flyers, e.g. as proposed in [1] and [4].

References

[1] Dimitar Dimitrov. *Dynamics and Control of Space Manipulators During a Satellite Capturing Operation*. PhD Dissertation Thesis - Tohoku University, 2005.

[2] Wigbert Fehse. *Automated Rendezvous and Docking of Spacecraft*. Cambridge University Press, 2003.

[3] Leonard Meirovitch. *Methods of Analytical Dynamics*. McGraw-Hill, 1970.

[4] Evangelos Papadopoulos e S. Ali A. Moosavian. *Dynamics and Control of Space Free-Flyers with Multiple Manipulators*. Department of Mechanical Engineering and Centre for Intelligent Machines, McGill University.

[5] Kazuya Yoshida. *Space Robot Dynamics and Control: To Orbit, From Orbit, and Future*. Robotics Research: the 9th International Symposium, 2000.

[6] Kazuya Yoshida, Dimitar Dimitrov, e Hiroki Nakanishi. *On the Capture of Tumbling Satellite by a Space Robot*. Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2006.