Motion Planning for the On-orbit Grasping of a Non-cooperative Target Satellite with Collision Avoidance

R. Lampariello

Institute of Robotics and Mechatronics, DLR, Germany
e-mail: roberto.lampariello@dlr.de

Abstract

A method for grasping a tumbling non-cooperative target is presented, which is based on nonlinear optimization and collision avoidance. Motion constraints on the robot joints as well as on the end-effector forces are considered. Cost functions of interest address the robustness of the planned solutions during the tracking phase as well as actuation energy. The method is applied in simulation to different operational scenarios.

1 Introduction

The problem of grasping a target satellite by means of a free-floating robot requires particular attention, when the target is non-cooperative and uncontrolled in its attitude. Such grasping task may be solved within the context of optimal or nonlinear control [1][2][3], with use of the on-board sensor signals, or in that of tele-presence control, with a human operator in the loop [4]. However, both these approaches can be regarded as local methods, even though the operator may undertake training. More generally, due to the complexity of the problem at hand, a motion planning of the task which can provide a higher level of confidence for its completion, is of interest. The motion planner may be conceived to be part of a semi-autonomous operational mode, which is based on the following three steps: 1. target motion observation and prediction; 2. planning of the robot grasping motion; 3. execution of the planned grasping motion with support of on-board sensor-based control.

The target motion prediction problem (step 1) was addressed in [5], with the aim of covering very long time periods of circa 100 seconds. This was done to allow for the computational time of the motion planner and for the execution of the maneuver. In this paper, the motion planning task (step 2) is addressed. The paper presents an implementation of nonlinear optimization for solving the grasping task in an optimal way and with the subsequent application of the solutions in a realistic scenario. The testbed involves two simulation models, shown in Fig. 1.

A nonlinear optimization problem is formulated in the standard way. As such, the robot joint positions are parameterized in time with B-splines. Inequality constraints are defined for the joint positions and velocities, as well as for the end-effector forces, and for collision avoidance. The motion grasping task is divided into three parts: an approach phase, a tracking phase (with grasping) and a stabilization phase. The chaser satellite, which carries the manipulator, is assumed to be actuated only in the approach and stabilization phases. A first optimal criterion is defined such as to minimize the risk of failure during the tracking phase of the trajectory (step 3 above). The latter in fact, could deviate from the nominal input trajectory, due to errors arising from modeling uncertainties or other sources. Another costs of interest is the thruster energy consumption dutring the approach phase.

The novelty presented here is the planning of the whole grasping maneuver for typical target tumbling motions and for any suitable robot kinematics. The use of nonlinear optimization and collision avoidance allows to choose a suitable initial position of the grasping robot with respect to the target, thus eliminating the problem of it being out of reach. Furthermore, the grasping strategy avoids impacts altogether.

In the formulation of the optimization problem, based on direct single shooting, the first priority is given to satisfying the inequality constraints mentioned above. Such problems can generally be solved within the allocated operational time of 100 seconds. If however, a cost function is to be optimized, the run times become longer. Due to this fact, and to the fact that the problem presents many local minima, the use of a look-up-table is considered for real-time implementation of optimized solutions.

2 Bibliography

In [2] the authors emphasize that although there is a lot of literature on the subject of space robot control, there is no solution which can readily solve

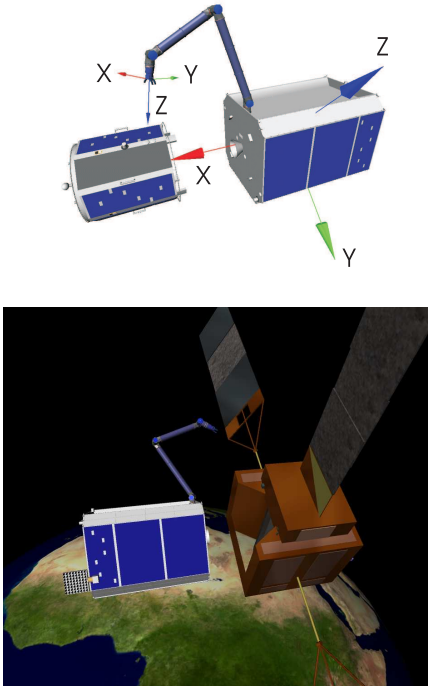

Figure 1. Two grasping scenarios

the task of capturing a tumbling object in space. It is also claimed that most works on the capturing of a tumbling target are divided into approach, impact and post-impact motion. Reference is made to work which focuses on minimizing the force impulse generated during the contact. These include, among others, the concept of impedance matching, to guarantee that the target will not be pushed away after an impact.

The same authors then tackle the hard problem of grasping the tumbling target with zero attitude change of the chaser satellite. They suggest in [2][6][7] preloading bias angular momentum in the chaser manipulator and reaction wheels, which is of equal magnitude and opposite sign to that of the target, such that after the capture the compound will have zero angular momentum. The task so formulated, is found here to be generally hard to accomplish, due to the limits of the robot kinematics and dynamics.

In [8] the chaser satellite is set into rotational motion with a constant angular velocity, in order to minimize the relative velocity between the robot end-effector and the target grasping point. It is assumed that the target is nearly axis-symmetric. The robot motion control is based on a Jacobian transpose scheme with proportional gain.

The autonomous grasping problem is also addressed in [9], based on an inverse kinematics algorithm, which however only applies to Puma-like robots. In [10] the target is assumed to rotate slowly, such that the assumption can be made that the robot end-effector trajectory should follow a circular path, or a spiral ascending trajectory. Trigonometric spline functions are used to plan the trajectory of the manipulator in Cartesian space. A strong assumption is made that the orientation of the end-effector matches that of the target.

In [3] the optimal control of the pre and post-capturing of a tumbling satellite is addresses. The optimal control for the capturing maneuver is solved as a rendezvous problem between two rigid bodies. For the detubling motion, it is also assumed that the chaser satellite attitude is unperturbed, with aid of the attitude control system. The problem is solved by considering the spacecraft dynamics, to which an external moment is applied.

More generally, with regards to motion planning in the context of nonlinear optimization, in [11] nonlinear optimization is applied to solve point-to-point motion planning problems for a free-floating robot. Cost functions considered include minimum chaser satellite attitude deviation, minimum time and minimum distance in Cartesian space. A main result is the so-called "V-maneuver", where the end-effector moves along radii into and out of the center of mass of the system, to minimize the chaser satellite attitude deviation.

Nonlinear optimization applied to a fixed-based 6 d.o.f. manipulator executing point-to-point maneuvers in configuration space, was already treated in [12], where collision avoidance is also included, and in [13]. The first made use of direct single shooting, while the second made use of collocation. Cost functions addressed were time, mechanical energy and power.

There are many papers on motion planning for free-floating and free-flying robots (see review collections [14], [16]). However, few are found which treat the collision avoidance issue.

Robot collision avoidance was already treated within optimization in [17], where the method of growth distance is used (this same method is also used in [12]). For a general class of robotic systems, an optimization problem is formulated over a family of continuous paths which satisfy the specified end conditions and possess robot-obstacle collisions. The cost to be minimized depends on the penetration growth distance, a measure of the depth of intersection between a pair of object models. The method is seemingly robust to non-convex problems, due to the shrinking of the obstacles. In [18] the authors use

strict-convexity bounding volumes, close to polyhedral convex hulls, to represent the obstacles, in order to guarantee the continuity of the proximity distance gradient.

3 Problem formulation

The addressed problem is to develop a motion planner for grasping and stabilizing a non-cooperative free-floating target satellite in orbit, by means of a robot manipulator mounted on a spacecraft. The spacecraft is actuated, although actuation is only used for the first part of the approach phase and for the final stabilization phase of the compound. The manipulator has seven rotational joints and rigid links, while joint friction and flexibility are ignored. The initial configuration of the robot is predetermined and fixed. The target trajectory is assumed to be determined by a vision system (e.g., [5]). Its geometry is assumed to be outlined by an operator (convex hull, composed of boxes and cylinders), if it is not provided by the target satellite owner. The operator also defines the position and orientation of the grasping point.

Trajectories should be found which first bring the end-effector into an orientation suitable for grasping, i.e. such that the target grasping point is at some predefined angle and small distance to it. The point in time along the target trajectory when the robot meets the target should be determined by the motion planner. The end-effector should then track the grasping point for some seconds, to allow for position mismatches to be corrected, and to allow for the initially imposed relative distance to be reduced to zero and the gap to be closed (minimum impact grasp). Finally, the relative motion between the chaser and the target, as well as that of the compound (relative to the orbital frame), should be brought to zero.

The chaser and target satellites are assumed to be in the same orbit (orbital relative dynamics is not considered). The rotational velocity of the target about each of its body-fixed axes may have any value between +/- 4 deg./sec.. A particular case is also of interest, in which the target is affected by energy dissipation (arising from flexible appendages like solar panels, or from sloshing). For this case, it is assumed that the target tumbling motion has decayed to a state of flat spin, i.e. a pure rotation about the major principal axis of inertia. This axis is inertially fixed and is assumed here to lie in the orbital plane (x-z plane in Fig. 1). This is because the resulting necessary motion prediction times can be contained to the allocated 100 seconds (see also Sec. 5).

Furthermore, one single grasping point on the satellite structure is considered. Note that the inertial properties of the target must not be known, for the purpose of the motion prediction [5]. However, in order to satisfy force constraints on the end-effector or attitude constraints on the chaser satellite, it is assumed that the error margin between the real and the assumed values can be dealt with by the on-board control system.

4 Method

The optimization problem formulated in section 3 is solved here as a series of two single-shooting subproblems. The first addresses the approach and tracking phases, while the second addresses the stabilization phase.

4.1 Mathematical formulation of the optimization problem

We begin by postulating that the mathematical problem at hand contains a world W, a known obstacle region O composed by the tumbling target satellite, and a configuration space C of dimensions $C(\boldsymbol{\theta}) \subseteq \Re^n$, with $n = 6 + 7 = 13$. The dimension of n derives from the 6 generalized coordinates of the free-flying chaser satellite and the 7 robot joint positions, denoted by $\boldsymbol{\theta} = [\boldsymbol{\theta}_m \quad \boldsymbol{\theta}_b]$. The orbital frame coordinate system is shown in Fig. 1. Note that this is regarded here as an inertial frame and equal to the chaser satellite reference frame at the initial time. The time interval is unbounded: $t = [0, \infty)$. The chaser satellite-robot system is subject to a bounded action $\boldsymbol{\tau} \subseteq \Re^m$, where $m = 6 + 7 = 13$, of which the relation to the system state $\Theta = [\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}]$ is given by the state transition equations $\dot{\boldsymbol{\Theta}} = \mathbf{f}(\boldsymbol{\Theta}, \boldsymbol{\tau})$, defined for every $\Theta \cup C$ and $\tau \cup \boldsymbol{\tau}$. This is generally written as follows [14]:

$$\begin{bmatrix} \boldsymbol{H}_b & \boldsymbol{H}_{bm} \\ \boldsymbol{H}_{bm}^T & \boldsymbol{H}_m \end{bmatrix} \begin{bmatrix} \ddot{\boldsymbol{\theta}}_b \\ \ddot{\boldsymbol{\theta}}_m \end{bmatrix} + \begin{bmatrix} \boldsymbol{c}_b \\ \boldsymbol{c}_m \end{bmatrix} = \begin{bmatrix} \mathcal{F}_b \\ \boldsymbol{\tau} \end{bmatrix} \quad (1)$$

where the inertia matrix $\boldsymbol{H}$ and the non-linear velocity dependent term $\boldsymbol{c}$ are written to include the components which relate to the chaser spacecraft and to the robot manipulator respectively. Also note that $\mathcal{F}_b$ and $\boldsymbol{\tau}$ refer to the actions on the chaser satellite and the torques on the robot joints respectively. Note that in particular phases of the maneuver, for which the robot is free-floating, $\mathcal{F}_b = 0$.

The nonlinear optimization problem can then be formulated by the following two sub-problems:

4.1.1 Approach and Tracking with grasping

The approach starts from an Observation distance and brings the chaser satellite to an optimal

grasping pose, which precedes the tracking phase. The Observation distance is function of the sphere of safety around the target satellite, which is in turn function of its rotational velocity and geometry.

The approach maneuver then consists of the spacecraft maneuver to the optimal grasping pose, followed by a robot maneuver to bring the end-effector into its initial position for the tracking phase. The end position of the spacecraft maneuver is function of the following robot approach and tracking phases only. Also note that during the robot maneuver, the chaser spacecraft is not actuated. The tracking phase is intended to minimize any residual relative velocity between the robot end-effector and the target grasping point. Its duration should dependent on the tracking controller performance, such that this may have sufficient time to bring any positioning error to zero.

Following is the robot approach phase addressed. This can be formulated mathematically as follows: find $\boldsymbol{\theta}_m{}^1(t), \boldsymbol{\theta}_b(0)$ minimizing $\Gamma(\boldsymbol{\theta}_m{}^1(t), \boldsymbol{\theta}_b(0))$ subject to

$$\mathbf{M}(\boldsymbol{\theta}^1)\,\ddot{\boldsymbol{\theta}}^{\mathbf{1}}(t) \;+\; \boldsymbol{c}(\boldsymbol{\theta}^1, \dot{\boldsymbol{\theta}}^1)\,\dot{\boldsymbol{\theta}}^{\mathbf{1}}(t) \;=\; \boldsymbol{\tau}^1 \tag{2}$$

$$\mathbf{h}^1(\boldsymbol{\theta}^1(t)) \leq 0 \tag{3}$$

$$\mathbf{h}^1{}_{coll}(\boldsymbol{\theta}^1(t)) \leq 0 \tag{4}$$

$$\mathbf{g}^1(\mathbf{r}^e(t_f{}^1)) = 0 \tag{5}$$

$$\boldsymbol{\theta}^1(0) = \boldsymbol{\theta}^1_{in}, \dot{\boldsymbol{\theta}}^1(0) = 0, \dot{\boldsymbol{\theta}}^1(t_f{}^1) = 0 \tag{6}$$

for $0 \leq t \leq t_f{}^1$ and where $t_f{}^1$ is a predefined final time, Γ^1 is a predefined cost function, $\mathbf{h}^1$ are inequality box constraints of type $\mathbf{x}_{min} \leq \mathbf{x}(t) \leq \mathbf{x}_{max}$, for $\mathbf{x} = \{\boldsymbol{\theta}^1, \dot{\boldsymbol{\theta}}^1\}$ and $\mathbf{h}^1{}_{coll}$ are collision avoidance constraints. Functions $\mathbf{g}^1(\mathbf{r}^e(t_f{}^1))$ are equality constraints on the final end-effector state $\mathbf{r}^e(t_f{}^1)$, which is clearly function of the given target trajectory (see Sec. 4.4). Finally Eq.s (6) express boundary conditions on position, where $\boldsymbol{\theta}_{in}$ is the given initial configuration, and on velocity. More will be said about boundary conditions on acceleration and jerk in section 4.6. Furthermore, in order to simplify the optimization problem, the value of $t_f{}^1$ is chosen to be that which minimizes the distance between the end-effector and the target.

The spacecraft maneuver preceding the robot approach maneuver is determined by the found condition for $\boldsymbol{\theta}_b(0)$. In order to limit this to a V-bar and/or R-bar maneuver (in the orbital plane), only two components of $\boldsymbol{\theta}_b(0)$ are optimized and the third is kept constant. In this phase of the maneuver, the equations of motion are reduced to mere single rigid body dynamic equations, where the robot states are locked and can be ignored. This is trivially solved, and as such is omitted here. In the robot approach maneuver instead, the spacecraft states are not actuated, but rather obey the law of conservation of linear and angular momentum.

The tracking phase is solved by means of an inverse kinematic algorithm. The algorithm is based on the pseudo-inverse kinematic solution for the redundant 7 d.o.f. robot:

$$\dot{\boldsymbol{\theta}}^2{}_{rob} = \mathbf{J}^\dagger \mathbf{r}^e{}_{des} + \left(\mathbf{E}_7 - \mathbf{J}^\dagger \mathbf{J}\right) \dot{\boldsymbol{\theta}}_{rob\ null}, \tag{7}$$

where $\mathbf{E}_7$ is the (7×7) identity matrix, $\dot{\boldsymbol{\theta}}^2{}_{rob}$ is the joint velocity vector and $\dot{\boldsymbol{\theta}}_{rob\ null}$ acts in the null space of the robot generalized jacobian matrix $\mathbf{J}$ [14], therefore not affecting the end-effector motion. This well known solution of the inverse kinematics minimizes the joint velocities, while the vector $\dot{\boldsymbol{\theta}}_{rob\ null}$ can be expressed by a potential function to improve quantities such as the manipulability of the robot [15].

The inequality constraints in Eq.s(3)-(4) are also applied to this phase of the motion:

$$\mathbf{h}^2(\boldsymbol{\theta}^2(t)) \leq 0 \tag{8}$$

$$\mathbf{h}^2{}_{coll}(\boldsymbol{\theta}^2(t)) \leq 0 \tag{9}$$

Simultaneously to the tracking motion, the imposed relative distance between the end-effector and the target grasping point is reduced to zero and the grasp is closed. It is assumed here that no significant impact occurs. Therefore, from this point on, the two satellites can be considered as one system, where the inertial parameters of the last robot link are updated accordingly (with any knowledge or estimate of their value). Furthermore, at the end of this phase, the robot joints are not at rest, nor is the system at rest with respect to the orbital frame. The stabilization of this residual motion is the task of the next phase.

4.1.2 Stabilization

The modeling of the dynamics for the stabilization phase ideally follows that of a single multibody system, for which the end-effector now includes the target, with known non-zero initial conditions. These, representing the positions and velocities of all the system degrees of freedom, are given from the preceding tracking phase.

In this phase, the relative motion between the chaser and the target is brought to zero by slowly reducing the manipulator joint velocities. This can be done in an optimal way, in order to minimize a specified cost function and to satisfy motion constraints, like the forces at the grasping point or the attitude disturbance on the chaser or target satellites.

The optimization problem formulation is simply as follows: find $\boldsymbol{\theta}_m{}^3(t)$ minimizing $\Gamma(\boldsymbol{\theta}_m{}^3(t))$ subject

to

$$\boldsymbol{\theta}(0) = \boldsymbol{\theta}({t_f}^2), \dot{\boldsymbol{\theta}}(0) = \dot{\boldsymbol{\theta}}({t_f}^2) \tag{10}$$

$$\dot{\boldsymbol{\theta}}_{rob}({t_f}^3) = 0, \tag{11}$$

$$\mathbf{h}^3(\boldsymbol{\theta}^3(t)) \leq 0 \tag{12}$$

$$\mathbf{h}^3{}_{coll}(\boldsymbol{\theta}^3(t)) \leq 0 \tag{13}$$

where Γ_3 is for example the chaser satellite or target attitude disturbance. The inequality in Eqn.(12) may also express limits on the forces on the robot gripper. The initial conditions in Eqn.(10) express the dependency on the final conditions of the previous tracking phase. Again, the value of ${t_f}^3$ is predefined here, for simplicity.

The following detumbling of the system can again be handled as a single rigid body control problem and is not addressed here.

4.2 Method of solution of the optimization problem

The optimization problems above are solved as a nonlinear programming problem (NPL), by satisfying the equality and inequality constraints at a finite number k of via points. The system independent states are parameterized in time, i.e., $\boldsymbol{\theta} = \boldsymbol{\theta}(t, \mathbf{p})$ with $\mathbf{p} \subseteq \Re^N$, for N optimization parameters, as described in subsection 4.6. The NPL is solved with an SQP algorithm from the MOPS library [19]. The equations of motion are modelled with the dynamics library described in [20].

Note that to solve the above NPL problems, the equations of motion are integrated in function of the initial conditions and optimization parameters. This is necessary to determine all quantities appearing in the inequality constraints expressed above. Furthermore, in the first sub-problem, each integration of the equations of motion is followed by a computation of the inverse kinematics for the tracking phase.

4.3 Cost functions

The cost functions of interest are those which minimize the risk of failure during the tracking phase of the trajectory. The latter in fact, could deviate from the nominal input trajectory during the tracking phase, due to errors arising from modeling uncertainties, camera occlusions, or other sources.

In mathematical terms, these are defined as follows. To allow for deviations of the end-effector position and orientation from the nominal trajectory during execution of the task, the manipulability should be maximized:

$$\min_{\mathbf{p}} \ \Gamma_{manip} = \sqrt{\det(\mathbf{J}\,\mathbf{J}^T)({t_f}^1)}\,, \tag{14}$$

where $\mathbf{J}$ is the Generalized Jacobian of the free-floating robot. This will minimize the risk of meeting a singularity during the tracking phase. The outcome will be an optimal spacecraft state and robot configuration for the grasping task. Note that this will depend on the target motion and geometry (i.e., position of grasping point).

In order to minimize the risk of collisions deriving from path deviations, the minimum distance from collision D along the trajectory should be maximized:

$$\max_{\mathbf{p}} \ \Gamma_{dist} = min(D)\,. \tag{15}$$

A simple cost function may also be defined to minimize the translation of the chaser spacecraft (and thus the energy) during the spacecraft approach phase:

$$\min_{\mathbf{p}} \ \Gamma_{approach} = \|\boldsymbol{\theta}_b(0)\|^2 \tag{16}$$

4.4 Inequality constraints

The inequality constraints, which are satisfied on the k via points, are firstly the box constraints on the joint positions and velocities:

$$\boldsymbol{\theta}_{min} \leq \boldsymbol{\theta}(t, \mathbf{p}) \leq \boldsymbol{\theta}_{max} \tag{17}$$

$$\dot{\boldsymbol{\theta}}_{min} \leq \dot{\boldsymbol{\theta}}(t, \mathbf{p}) \leq \dot{\boldsymbol{\theta}}_{max} \tag{18}$$

The constraint bounds are given by the robot design specifications.

Further constraints arise from the collision avoidance. To compute collision detections and to formulate the collision avoidance problem within an NLP context, bodies in the scene are represented here as convex polytopes. For our purpose these include boxes, cylinders and elongations of these.

The collision avoidance problem can be simply formulated as an inequality constraint in the optimization problem:

$$D(i) > 0.0, \quad 1 < i < m, \tag{19}$$

where the function $D(i)$ constitutes a minimum distance between two bodies or a penetration depth, if the two bodies intersect. The scalar m is the number of body pairs in the given problem.

To compute the distance between two bodies, or their penetration depth, the ODE library was implemented [21]. The library allows to represent objects as boxes or capsules (a capsule is like a normal cylinder except it has half-sphere caps at its ends. This feature makes the internal collision detection code particularly fast and accurate). For these, it is possible to compute, in case of collision, the penetration depth as the minimal length of translation needed to separate them. If the cost function (15) is to be minimized, the size of the robot is overscaled and the resulting penetration depth minimized.

4.5 Equality constraints

In the robot approach phase, an extra equality constraint is required on the final end-effector position and orientation, in order for it to meet the target at some point on the trajectory:

$$\mathbf{r}^e(t_f^1, \mathbf{p}) - \mathbf{r}^{target}(t_f^1) = 0 \tag{20}$$

$$\boldsymbol{\phi}^e(t_f^1, \mathbf{p}) - \boldsymbol{\phi}^{target}(t_f^1)) = 0 \tag{21}$$

where $\mathbf{r}^e$ is the end-effector position vector, computed at the final time t_f^1, $\mathbf{r}^{\,target}$ is the given target trajectory at the final time, $\boldsymbol{\phi}^e$ are the quaternion parameters which describe the end-effector orientation and $\boldsymbol{\phi}^{\,target}$ are the quaternion parameters which describe the target orientation, also computed at the final time. The latter can be relaxed in one direction, if the grasping point may be grasped at different angles (e.g., a handle or a solar panel boom).

4.6 Parameterization

An order-4 B-spline was chosen for the joint states, in order to allow for smoothness up to the third derivative. Comparisons were made for different numbers of nodes N, with respect to the cost function.

4.6.1 Order 4 B-spline for robot states

We choose here periodic uniform B-splines for their particularly compact matrix form. For N vertices, $n_{seg} = N - 3$ segments of length $t_{seg} = \frac{t_f}{N-3}$ result. It follows that for the internal time of the i^{th} segment $u(t) = \frac{t}{t_{seg}} - (i-1)t_{seg}$, such that $0 \leq u < 1$, the computation of the uniform B-spline and derivatives is given by [22]:

$$\begin{pmatrix} s_i(u) \\ \dot{s}_i(u) \\ \ddot{s}_i(u) \\ \dddot{s}_i(u) \end{pmatrix} = \frac{1}{6} C(u)\, A \begin{pmatrix} B_i \\ B_{i+1} \\ B_{i+2} \\ B_{i+3} \end{pmatrix}, \quad 1 \leq i \leq n_{seg} \tag{22}$$

where B_i represents the i^{th} vertex, A is a constant matrix and $C(u)$ the matrix of basis functions. Furthermore, these matrices are invertible, so that they can be used to satisfy the boundary conditions, defined below.

1. Approach phase:

$$\begin{pmatrix} s_0(0) \\ \dot{s}_0(0) \\ \ddot{s}_0(0) \\ \dddot{s}_0(0) \end{pmatrix} = \begin{pmatrix} \boldsymbol{\theta}_{in} \\ 0 \\ 0 \\ \mathbf{p}_{\mathbf{j}_0} \end{pmatrix}, \begin{pmatrix} s_{N-3}(1) \\ \dot{s}_{N-3}(1) \\ \ddot{s}_{N-3}(1) \\ \dddot{s}_{N-3}(1) \end{pmatrix} = \begin{pmatrix} \mathbf{p}_{\boldsymbol{\theta} tf} \\ 0 \\ 0 \\ \mathbf{p}_{\mathbf{j}_{tf}} \end{pmatrix} \tag{23}$$

where $\mathbf{p}_{\mathbf{j}_0}$ are the parameters for the jerk at time $t = 0$, $\mathbf{p}_{\boldsymbol{\theta} tf}$ are the parameters for the joint positions and $\mathbf{p}_{\mathbf{j}_{tf}}$ are the parameters for the jerk at time $t = t_f$. Note that of the 8 boundary conditions, 5 are predefined, including zero velocities and accelerations. The latter ensure that there are no jumps on the accelerations at the boundaries, thus avoiding high levels of jerk.

2. Stabilization phase:

$$\begin{pmatrix} s_0(0) \\ \dot{s}_0(0) \\ \ddot{s}_0(0) \\ \dddot{s}_0(0) \end{pmatrix} = \begin{pmatrix} \boldsymbol{\theta}_{in} \\ \dot{\boldsymbol{\theta}}_{in} \\ 0 \\ \mathbf{p}_{\mathbf{j}_0} \end{pmatrix}, \begin{pmatrix} s_{N-3}(1) \\ \dot{s}_{N-3}(1) \\ \ddot{s}_{N-3}(1) \\ \dddot{s}_{N-3}(1) \end{pmatrix} = \begin{pmatrix} \mathbf{p}_{\boldsymbol{\theta} tf} \\ 0 \\ 0 \\ \mathbf{p}_{\mathbf{j}_{tf}} \end{pmatrix} \tag{24}$$

where $\mathbf{p}_{\mathbf{j}_0}$ are the parameters for the jerk at time $t = 0$ and $\mathbf{p}_{\boldsymbol{\theta} tf}$ and $\mathbf{p}_{\mathbf{j}_{tf}}$ are the parameters for the joint positions and for the jerk at time $t = t_f$ respectively.

5 Analysis of results

For the following examples, the number of B-spline parameters was initially set to N=42. The end times were defined as $t_f^1 = 10$, $t_f^2 = 5$ and $t_f^3 = 45$ sec.. The number of via points was chosen to be $k = 20$.

A solution for the flat spin case was first computed for the first scenario, with a target rotation about the y-axis, i.e. ω_{target}=[0 4 0] deg/sec. The resulting chaser satellite rotational motion is shown in Fig. 2. At first, no cost function was minimized. The computational time was 30 seconds. After setting the cost function to the energy consumption in the spacecraft approach maneuver defined in Eqn (16), the resulting initial chaser satellite position was $\boldsymbol{\theta}_b(0)$=(1.0e-4, 0, 1.0e-4), as expected. The number of collision pairs results to be m=29, for the 8 (chaser) + 1 (target) bodies in the scene.

In order to increase the distance from singularities and collision, as an element of safety, the cost functions defined in Eqn (14) and Eqn (15) were then implemented. For the second cost to be computed, the robot geometric model was scale by a factor of 3 in the x direction. The result is such that the manipulability remained approximately the same, but the penetration depth decreased from an initial value of 1.68m to a final optimized value of 0.8m. The new initial position of the chaser satellite was (0.33, 0.0, -0.83)m. No numerical problems were encountered in the implementation of the collision avoidance within the optimization.

In the following example, the second scenario was considered. For this, the moments at the end-effector during the stabilization phase were found to be most critical. in fact, the principal components of inertia of the target were set to [5000 3000 7000] kgm^2. The solutions before and after implementing an inequality constraint of +/- 10 Nm in all directions is shown in

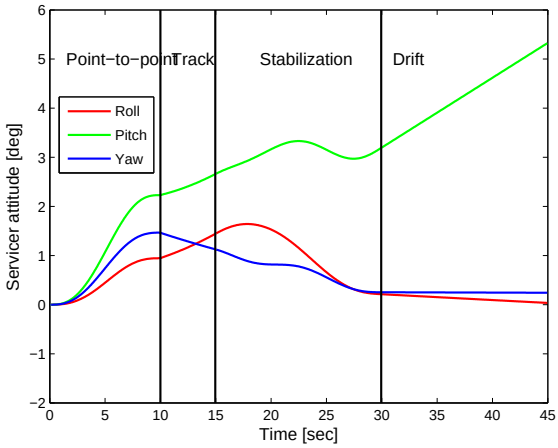

Figure 2. Flat spin for scenario I, ω=[0 4 0] deg./sec.: Chaser satellite attitude during the grasping task

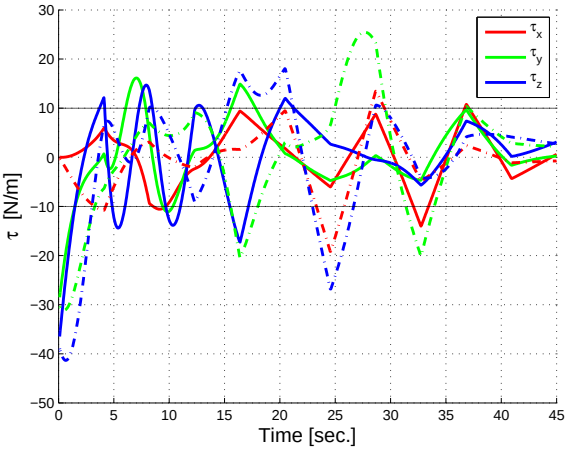

Figure 3. Flat spin for scenario II, ω_{target}=[0 0 4] deg./sec.: Torques at the end-effector during the stabilization phase

Fig. 2. The computational time was in the order of 30 min., although the number of B-spline parameters was set to N=70. The end-effector equality constraint on the orientation was relaxed in one direction, since the solar panel boom was chosen as grasping point, thus allowing a whole range of grasping angles around it.

In order to be able to make use of the optimized solutions in the allocated operational time span of 100 seconds, a look-up table may be built off-line for a given target satellite. The range of target orientations and rotational velocities can be divided into regular intervals, thus generating a set of grid points. Optimal solutions can then be computed for each of these points off-line and stored in the look-up table. Further work will address the details of this approach.

With the assumption of the flat-spin axis being in the orbital plane, a suitable time during an orbit will exist, for which the distance between the Observation point and the optimal grasping pose, can be covered in the allocated operational time of 100 seconds, while still allowing to perform the proposed grasping sequence (observation, planning and execution) without collisions. The rotational rate is in fact in the worst case 90 minutes, for a low-earth orbit, providing two time slots for the grasp. This limits the subsequent approach maneuvers to V-bar or R-bar maneuvers. Cases for which the spin axis is out of the orbital plane are not considered here, for which out-of-plane maneuvers might have to be considered.

6 Aknoledgments

The author would like to thank Klaus Landzettel and Ulrich Hillenbrand for the useful discussions.

7 Conclusion

A novel method for the grasping of a tumbling target, based on nonlinear optimization and collision avoidance, is presented. Typical motion constraints are addressed, including robot joint position and velocity limits and end-effector forces during the target motion stabilization. Impacts are carefully avoided, by ensuring that the end-effector velocity equals that of the grasping point on the target during grasping. Spacecraft approach phases and post-grasping detumbling phases are not considered. The effectiveness of the method was shown with some simulation examples, in two operational scenarios. Further work will concentrate on allowing optimized grasping motions to be available within useful operational times.

With this proposed operation approach, a solution of the motion planner can be observed in a preview screen. This allows to determine if a maneuver is safe with respect to collisions, end-effector force limits, robot singularities and workspace limits, and eventually satellite control force limits and attitude deviations of the chaser satellite. These are all issues which make the grasping task non-intuitive and as such might be difficult for an operator to handle in the telepresence control mode.

References

[1] Papadopoulos, E., Moosavian, S.: "Dynamics and Control of Multi-arm Space Robots During Chase and Capture Operations", IROS 94, pp. 1554-1561, 1994.

[2] Yoshida, K., *et al*: "On the Capture of Tumbling Satellite by a Space Robot", IROS 06, Beijing, China, October 2006.

[3] Aghili, F.: "Optimal control for robotic capturing and passivation of a tumbling satellite with unknown dynamics", AIAA Guidance, Navigation and Control Conference and Exhibition, Honolulu, Hawaii, USA, August, 2008.

[4] Reintsema, D., Landzettel, K. and Hirzinger, G. (2007), "DLR's Advanced Telerobotic Concepts and Experiments for On-Orbit Servicing", Tracts in Advanced Robotics, Volume 31/2007, STAR - Springer, 22 S, ISSN 978-3-540-71363-0.

[5] Hillenbrand, U., Lampariello, R.: 'Motion and Parameter Estimation of a Free-Floating Space Object from Range Data for Motion Prediction', i-SAIRAS 2005, Germany, Sept. 2005.

[6] Dimitrov, D. Yoshida, K.: "Utilization of the Bias Momentum Approach for Capturing a Tumbling Satellite", IROS 06, Sendai, Japan, Oct. 2004.

[7] Dimitrov, D. Yoshida, K.: "Momentum Distribution in a Space Manipulator for Facilitating the Post-Impact Control", IROS 06, Sendai, Japan, Oct. 2004.

[8] Nagamatsu, H., Kubota, T., Nakatani, I.: *Capture Strategy for Retrieval of a Tumbling Satellite by a Space Robotic Manipulator*, ICRA 96, Minnesota, USA, April 1996.

[9] Wenfu Xu, *et al*: Autonomous target capturing of free-floating space robot: Theory and experiments. Robotica 27(3): 425-445 (2009).

[10] Panfeng Huang, *et al*, "Tracking Trajectory Planning of Space Manipulator for Capturing Operation", International Journal of Advanced Robotic Systems, Vol.3, No.3, pp.211-218, September, 2006.

[11] Schulz, V.H., "Reduced SQP methods for large scale optimal control problems in DAE with application to path planning problems for satellite mounted robots", PhD thesis, University of Heidelberg, 1995.

[12] Park, J., Bobrow, J.E.: *Minimum-Time Motions of Manipulators with Obstacles by Successive Searches for Minimum-Overload Trajectories*, ICRA 02, Washington, USA, May 2002.

[13] Stryk, O. von, M. Schlemmer: "Optimal control of the industrial robot Manutec r3". In: R. Bulirsch, D. Kraft (eds.): Computational Optimal Control. International Series of Numerical Mathematics 115 (Basel: Birkhuser, 1994) 367-382.

[14] *Space Robotics: Dynamics and Control*, ed. by Xu and Kanade, Kluwer Academic Publishers, 1993.

[15] F. Cusumano, R. Lampariello, G. Hirzinger, 'Development of Tele-operation Control for a Free-floating Robot during the Grasping of a Tumbling Target', IMG 2004, Genoa - Italy, 2004.

[16] Moosavian, S., Papadopoulos, E.: *Free-flying robots in space: an overview of dynamics modeling, planning and control*, *Robotica*, Vol.25, pp. 537-547, 2007.

[17] Ong, C.J., Gilbert, E.: "Robot Path Planning with Penetration Growth Distance", *Journal of Robotic Systems*, 15(2), 57-74, 1998.

[18] Stasse, O., Escande, A., Mansard, N., Miossec, S., Evrard, P., Kheddar, A.: *Real-Time (Self)-Collision Avoidance Task in a HPR-2 Humanoid Robot*, ICRA 08, Pasadena, California, USA, 2008.

[19] Joos, Hans-Dieter (2008) MOPS Multi-Objective Parameter Synthesis User's Guide V 5.3. DLR-Interner Bericht. DLR IB 515-08-37.

[20] Abiko, S., Lampariello, R., Hirzinger, G., "A Dynamic Library for Versatile Modeling of Free-Flying and Mobile Robotic Systems", ASTRA 08, ESA/ESTEC Noordwijk, The Netherlands, 2008.

[21] http://opende.sourceforge.net/wiki/index.php/Manual_(Coll

[22] Rogers, D.F., Adams, J.A.,*Mathematical Elements for Computer Graphics*, McGraw-Hill Publishing Company, Second Edition, 1990.

[23] Nenchev, D., Umetani, Y., Yoshida, K.: *Analysis of a Redundant Free-Flying Spacecraft Manipulator System*, IEEE Transactions on Robotics and Automation, Vol. 8, No. 1, Feb. 1992.