Tracking Control for the Grasping of a Non-cooperative Tumbling Satellite with a Free-Floating Robot

Roberto Lampariello, Hrishik Mishra, Nassir Oumer, Phillip Schmidt, Marco De Stefano, Alin Albu-Schäffer

***Abstract*— In this paper a novel method is presented for grasping a non-cooperative tumbling satellite with a free-floating robot. A reference trajectory, provided by an offline motion planner and adapted online to account for modelling uncertainties, is tracked with a visual servo for the approach phase and with a joint space controller for the rigidization phase which follows the grasp. An Extended Kalman filter ensures a Lipschitz continuity of the noisy pose estimates of a visual tracker, which makes the control method robust for executing the grasping task. The visual servo is in cascade with an impedance controller to achieve tracking while being compliant. The method is successfully validated on the OOS-SIM facility, which allows simulating gravity-free multibody dynamics with the hardware-in-the-loop simulation method.**

I. Introduction

Among the possible applications of robotic systems in an orbital scenario, this paper focusses on those which may operate in On-Orbit Servicing and in Active Debris Removal missions [1][2][3][4]. To tackle the related control challenges we present here an autonomy-based approach, in which the robot is commanded through a precomputed reference trajectory, corrected by a tracking controller, to account for intrinsic planning and execution errors. We focus on the task of grasping a defective, tumbling target satellite (from here on, the Target) by means of a free-floating robot, consisting of a non-actuated chaser satellite (from here on, the Chaser) carrying a kinematically redundant robot manipulator. These are shown in Fig. 1.

The task of grasping a target satellite with a space robot was already performed in [1] and [2], however only for the case of a cooperative Target. When the Target cannot be attitude controlled and does not present visual or structural features to aid its grasping, then it is generally termed non-cooperative. In this case, the accomplishment of the robotic grasping task in autonomous mode presents some substantial challenges. The robot motion is dictated by that of the Target, which is generally unknown. The robot motion also has to be correctly synchronized with that of the Target, in order to meet and grasp some preselected grasping point on it, while satisfying motion constraints, such as workspace limits, kinematic and dynamic singularities, collision avoidance, as well as sensor-driven constraints, such as camera field-of-view boundaries and pixel velocity limits. Furthermore, due to the given free-floating dynamics in play, the effect of an impact during the contact phase can be very detrimental for accomplishing the task, since it can quickly bring the Target out of the range of the robot.

All authors are from the Institute of Robotics and Mechatronics (DLR), 82234 Weßling, Germany, email: *firstname.lastname@dlr.de*

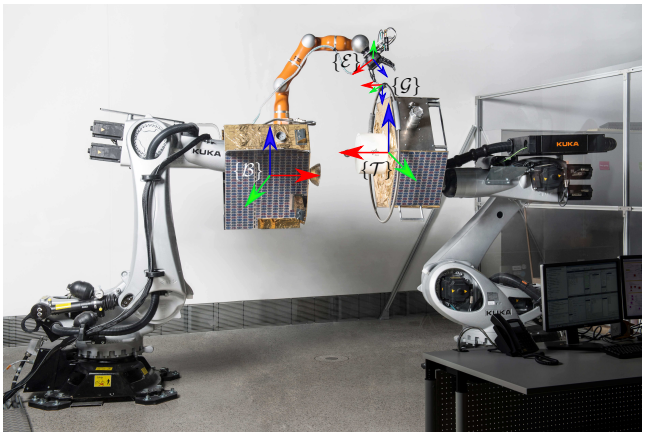

Fig. 1. OOS-SIM experimental facility used to reproduce the gravity-free dynamics of the Chaser (left), carrying a Light-Weight Robot, and the Target (right). A camera is mounted on the gripper of the kinematically redundant manipulator, to allow visual servoing. The reference frames of the Chaser $\mathcal{B}$, the robot end-effector ϵ, the Target $\mathcal{T}$ and the predefined grasping point $\mathcal{G}$ are also shown.

The grasping task of interest has been addressed extensively in the literature, first in the context of feedback control and then also in that of optimal control (see Section I-A). While in the former the aim is to solve a regulation control problem, often taking advantage of the free-floating dynamics and redundancy of the Chaser, in the latter an open-loop approach is preferred, based on the idea of computing a feasible and optimal trajectory in real-time. Due to the highly constrained task, explained above, the use of a feasible trajectory is recognized to be of great importance.

A gap in the methodology described above, which we want to close here, is that of having both the optimal control and the feedback control elements working together. A feedback controller alone does not provide any guarantee of convergence, given the presence of the constraints, which give rise to local minima. Furthermore, the grasping task has a limited time window for its execution and local control methods generally do not ensure finite time convergence. At the same time, an open-loop approach is not robust to

modelling errors (e.g., dynamic model) and to contingencies, such as an impact. We present here a tracking controller for executing the grasping task of interest, which takes a reference trajectory from a database, generated off-line with a motion planner in simulation and feasible with respect to all relevant constraints, and superimpose on it sensor-based corrections to account for the discrepancy between the simulated and the real world.

The first phase of the task, the approach phase, is executed by a visual servo in a cascade with an impedance controller, bringing the robot end-effector onto the grasping point on the tumbling Target. It is widely recognized that the robustness of a visual servo is significantly improved against modelling errors when performing tracking rather than regulation [5]. The pose of the Target is robustly estimated throughout the motion with an extended Kalman filter, which is fed with the noisy pose estimates of a model-based visual tracking algorithm. The Kalman filter is designed to reduce its computational footprint, in view of an implementation on a realtime system. The use of robot impedance control is intended to minimize the detrimental effect of an impact between the robot manipulator and the Target. In the following rigidization phase of the grasping task, in which the Target is already grasped but needs to be stabilized with respect to the Chaser, the reference trajectory provides a final configuration in which to let the system drift with decaying velocity, by virtue of the robot manipulator controller's dissipation.

In this paper we present the tracking controller just described and its validation on DLR's OOS-SIM robotic facility, shown in Fig. 1. This facility allows reproducing realistic orbital gravity-free dynamics in three-dimensions through hardware-in-the-loop simulation. The success of the controller requires a judicious interplay between the different elements involved: the motion planner, the visual servo (including the Kalman filter and the visual tracking) and the free-floating robot impedance controller. Furthermore, the intrinsic uncertainty in the control problem at hand is handled with partial on-line adaptation of the reference trajectory. Extensions of the functionality of the OOS-SIM facility to simulate the post-grasping tumbling motion of the satellite stack are also described.

The paper is then structures as follows. In the rest of this Section, we present a relevant bibliography and a more detailed problem statement. In Section II we provide details of the applied methods, while in Section IV we present the experimental results and their analysis. In Section V we provide a discussion as well as our conclusions and views on future work. The mathematical notation is such that scalar constants are written in plain text, scalar variables in italics, vectors, vector arrays and matrices are always bold, coordinates of vector quantities are expressed with the left upper superscript.

A. Related work

Many approaches for grasping a free-tumbling Target mostly focus on feedback control methods [6][7] and momentum control methods [8]. In [9], the motion planning problem is solved for the approach phase of the grasping task. The planning is supported by a prediction of the Target's motion, achieved through an extended Kalman filter and noisy pose measurements of a laser camera system. The motion prediction occurs before any motion of the robot and the control strategy does not include a continuous visual feedback. An optimization problem is solved partly analytically and partly numerically, to minimize a penaltly cost, function of four weighted quantities: travel time, distance, line-of-sight angle of the laser camera and robot joint acceleration. Experiments are conducted on an experimental facility which reproduces the dynamics of the tumbling Target and of a robot manipulator end-effector with an attitude-controlled base. In [10] a path planning method is presented for the rigidization phase which follows the grasping. It is assumed that the Chaser is attitude stabilized. As such, an optimal control problem is formulated for the detumbling of the Target to which an external moment is applied, solved analytically for the case of minimum time and zero final velocity.

In [11] an optimal control problem is solved numerically with nonlinear optimization in joint space, addressing the grasping task under the uncertainty of the initial and final positions of the robot end-effector resulting from tracking sensing data. The method is applied in simulation to a 2D problem, for a fully attitude controlled robotic system. In [12] a direct shooting method is used to treat the grasping problem in 3D, with inclusion of robot joint position and velocities contraints, as well as the Chaser free-floating dynamics. A look-up table approach is presented with which (close to) globally optimal soultions can be retreived in real-time for any possible tumbling state of the Target within a predefined range for the angular velocity.

In [13], an adaptive EKF variant was developed which used Laser-vision data for parameter and motion estimation. The authors used this variant for a Prediction-based controller in [9]. In [14], the authors implemented a filter cascade using range images for target state and parameter estimation. In [15], a prediction method was adopted keeping autonomous grasping in focus. In [16], a closed-loop system was implemented for autonomous target tracking with visual servo and EKF as a part of the incremental inverse-kinematics controller. The idea of visual servoing and position-based impedance control were brought together in [17] for tasks in which contact with the environment is expected. As such, the aim here was to fuse the visual and joint position to improve the target's pose-estimation and hence improve tracking accuracy. Model-based methods [18] which exploit robust edge features efficently, are used in position-based control.

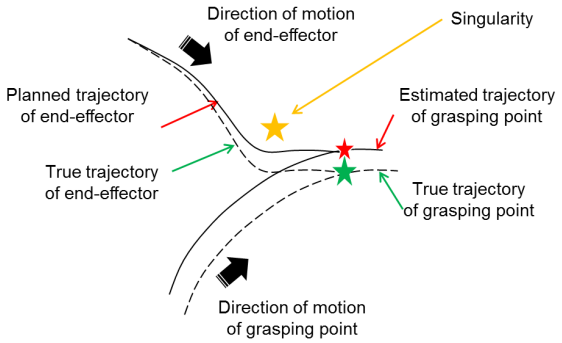

Fig. 2. The motion of the Target grasping point and of the robot end-effector are shwon for the model-based (solid line) and for the true (dotted line) cases.

B. Problem statement

In the grasping task of interest we assume that the Target has a known geometry, such that model-based computer vision pose estimation can be used. Its angular rate is limited here to 2 deg/s (due to hardware limitations, the limit for the rigidization phase is set to 1 deg/s). The grasping point on the Target is predefined. The Chaser is not actuated (free-floating) and is initially in the same orbit as that of the Target with zero translational and angular velocities.

The grasping task itself is commonly divided into three phases: an approach phase, in which the robot end-effector is brought in the vicinity of the moving grasping point of the Target; a tracking phase, in which the end-effector follows the grasping point with the same velocity, while homing in onto it and closing the grasp; a rigidization phase, in which the relative velocity between the Chaser and the Target is brought to rest by a suitable control of the robot manipulator.

A feasible trajectory to accomplish this task is provided by a motion planner, first described in [12] and extended here. The motion planner relies on a prediction of the Target motion, which can be accomplished as described in [15] [9]. However, in order to complete the task, the tracking controller will initially need to deviate from the reference trajectory, given that the latter is model-based and will differ from the real-world conditions. This fact is shown pictorially in Fig. 2. Note that the gross motion of the motion planner is still maintained, in order to avoid motion constraints such as, for example, a singularity.

Due to the deviation from the reference trajectory in the approach and tracking phases, the reference trajectory for the rigidization phase is adapted online, such that the deviation in joint space is recovered. This favours the fullfillment of joint position related constraints (collisions avoidance, singularity avoidance) throughout it.

II. Methods

In this Section we present the methods used to solve the grasping task, to include elements of the motion planning, the visual-servo with manipulator dynamisc, used in the approach and the tracking phases and the joint-position-based tracking control used in the rigidization phase. The control system architecture is shown in Fig. 3.

A. Motion planning and motion synthesizer

1) Approach and tracking phases: The motion planning method described in Section I-A and in detail in [12] is extended here to account for the requirements stemming specifically from the visual tracking. It is in fact required that throughout the approach phase sufficient features of the Target are in view of the camera (e.g., the Target's solar panel on its top suface) and that the pixel velocity in the image remains low. These result in motion constraints, the first of which is treated with an inequality constraint on the pitch angle of the end-effector frame $\mathcal{E}$ relative to the inertial frame $\mathcal{I}$, referred to as ϕ_e (see Fig. 1), defined as

$$\phi_e{}^2 \leq \phi_{e\,mid}{}^2 + \phi_{e\,delta}{}^2, \tag{1}$$

where $\phi_{e\,mid}$ is a parameter to define a middle value for ϕ_e and $\phi_{e\,delta}$ is a paremeter which relates to the field-of-view of the camera.

Defining the pixel velocities in the x and y components of the end-effector frame $\mathcal{E}$ to be ${}^e v_{px,py} = {}^e v_{y,x}{}^{et} / {}^e r_z{}^{et}$, an inequality constraint is also introduced as

$$\Delta^e v_{px,py} \leq {}^e v_{px,py} \leq \Delta^e v_{px,py}, \tag{2}$$

where $\Delta^e v_{px,py}$ are parameters which relate to the pixel velocity limits in the camera image.

In order to minimize the relative velocity between the robot end-effector and the Target, and therefore the pixel velocity in the camera image, the robot joints velocity at the end of the approach maneuver was defined to be equal to that given by the inverse kinematics of the tracking phase which follows. In this way, the transition between these to phases, unlike in [12], is more smooth and less prone to large pixel velocity values.

The output of the motion planner is judiciously defined to account for two important aspects. Firstly, a reference trajectory of the end-effector is defined relative to the grasping point frame on the Target, $[\mathbf{H}_{eg,ref}, \mathbf{V}_{eg,ref}]$. The tracking controller is designed to track this relative motion, where its pose is given by $\mathbf{H}_{eg} = \mathbf{H}_{eI}\mathbf{H}_{Ig}$. This ensures that any error in $\mathbf{H}_{eI}$ (tracking control errors, robot forward kinematic errors) and in $\mathbf{H}_{Ig}$ (Target motion prediction errors) are intrinsically eliminated in following the desired relative motion $[\mathbf{H}_{ge,ref}]$, which is independent of these uncertainties. Note however, that the motion of the robot relative to absolute space will deviate from the motion planning solution. This relates to the second aspect to be considered, namely that

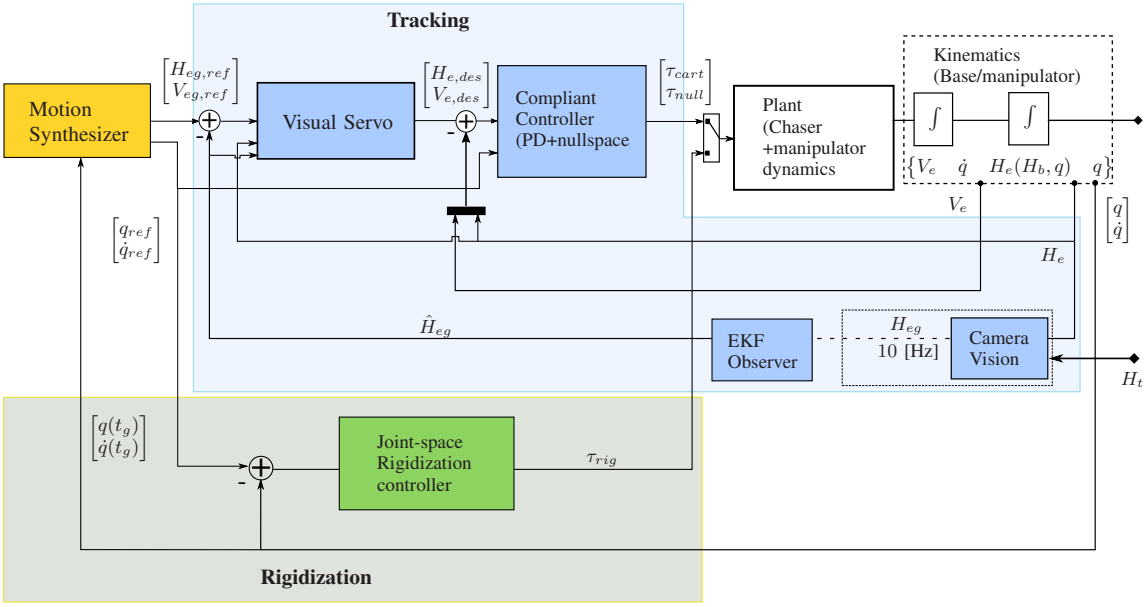

Fig. 3. Control system architecture for tracking or rigidization controllers.

a reference trajectory is also provided in the joint space of the robot manipulator. As will be shown in Section II-C, the tracking control law contains a main term to minimize the Cartesian tracking error and a second term, projected in the null-space of the robot manipulator, to minimize the tracking error in joint space. This is to minimize the deviation of the robot motion in joint space from the feasible reference trajectory. The null-space term also cares for the kinematic redundancy of the robot manipulator.

2) Rigidization phase: In this phase, the offline motion planner provides a reference trajectory in joint space. The tracking controller here is a joint space controller. The task is to rigidize the compound after the grasp, bringing the relative motion between the two satellites to zero. Given that only internal forces come into play, the angular momentum of the Target is shared with the Chaser and at the end of the maneuver the two satellites tumble with a new angular velocity, function of their total inertia. The cost function of the optimization problem is chosen to be the mechanical energy of the robot manipulator, in order to minimize its loads, i.e.

$$\mathbf{\Gamma} = \int_{t_{ri}}^{t_{rf}} (\boldsymbol{\tau}^T(t)\, \dot{\mathbf{q}}(t))^2 dt \tag{3}$$

where t_{ri} t_{rf} are the initial and final times of the motion.

In the online setting however a modification of the reference trajectory is necessary, to account for the deviation from the reference trajectory in the previous approach phase. Given that the trajectory is parameterized with a B-spline, the optimization parameters of the solution generated offline are still used to synthesize online a new rigidization trajectory, by modifying them with the measured initial robot manipulator joint states. Let the measured initial position and velocity for a given joint be q_{0m} and $\dot{q}_{0m}$ respectively, and the B-spline parameters $q_i, 1 \leq i \leq N_B$. The modified parameters are then taken to be:

$$q_{i\,mod} = q_i + (q_0 - q_{0m} + \dot{q}_{0m} * (t_{rf} - t_{ri})/N_B) * (N_B - i)/N_B, \tag{4}$$

where the second term accounts for the initial position deviation $q_0 - q_{0m}$ and the joint velocity at the initial time times the duration of the first B-spline segment, $(t_{rf} - t_{ri})/N_B$, in which the spline cannot be modified. This effectively smoothly brings the joint configuration back to the one planned offline. As such, the fullfilment of the joint position related motion constraints is favoured. The chosen modification of the trajectory could however give rise to an increase in robot joint torques, since it implies pushing the Target onto the predicted trajectory. Note however, that the choice of the cost function in Eqn. (3) minimizes these, allowing for a larger margin from their limits.

B. Extended Kalman Filter (Model-based control)

The Extended Kalman Filter (EKF) as a nonlinear observer is ubiquitous in all disciplines germane to this paper. It is a nonlinear observer in the sense that the model propagation is nonlinear whereas, the error injection is linear and hence the covariance time-propagation is performed through a linear approximation of the state dynamics. In the context of this paper, the EKF provides robustness to the tracking controller against incoming measurement outliers from the computer vision algorithm by ensuring Lipschitz continuity of the measurement. Due to its predictor-update structure, it also provides a state prediction which allows us

to compensate for image-processing time-delays and ensure continuous control errors during occlusion or measurement failures caused by degenerate observability conditions. From a tracking controller perspective, the EKF also estimates the target velocities which can be used as a feed-forward term for the Visual Servo, without which, the controller design is constrained.

The state of the target is defined by its inertial pose, $\mathbf{H}_T = \begin{bmatrix} R(q) & r \\ 0 & 1 \end{bmatrix}$ where q and r are the quaternion representation of the Target's orientation and the analogous position relative to the inertial frame. The pose is driven by a mixed velocity representation $\mathbf{V}_t^m = \begin{bmatrix} \dot{r} \\ \omega^b \end{bmatrix}$, where the superfix b and m denote body and mixed velocity forms. The dynamic system is modeled with the state, $x_a \in \mathbb{R}^{13} = \begin{bmatrix} q^T r^T \omega^T \dot{r}^T \end{bmatrix}^T$ as,

$$\frac{d}{dt}\begin{bmatrix} \dot{q} \\ r \\ \omega \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \frac{1}{2}\omega \otimes q \\ \dot{r} \\ -\omega_\times I \omega \\ \Phi(n) \end{bmatrix} \tag{5}$$

where $\otimes$ denotes the quaternion multiplication tensor, $(.)_\times$ denotes the skew-symmetric form of $\mathbb{R}^3$. $\Phi(n)$ is the linear approximation of orbital dynamics given by the Clohessy-Wiltshire equations. Since, for the tracking phase, the total time, $T < 20$ [sec], we assume this can neglect this term.

In the multiplicative variant of the EKF, the quaternion state is maintained as a 3-component vector $\delta q_v \in T_Q$, which is the tangential space to the quaternion manifold. A detailed treatise on such vector expressions was given in [19]. In this text, the Modified Rodriguez Parameters (MRP), a_p was chosen so that singularity is at 360. The tangential vector is related to the quaternion as follows,

$$\dot{a}_p(q) = \frac{4}{1+q_0} q_v \tag{6}$$

$$\frac{1}{16}\begin{bmatrix} 8a_p \\ 16 - a_p^T a_p \end{bmatrix} \tag{7}$$

The differential form for a_p is given as,

$$\dot{a}_p = \Big(-\frac{1}{2}\omega_\times + \frac{1}{8}\Big) a_p + \Big(1 - \frac{1}{16} a_p^T a_p\Big)\omega \tag{8}$$

Among the alternatives for scaled inertia parameterization for zero-torque motion propagation, [20, §4.6] the 3-parameter form proposed in [13] is used. So, the angular dynamics can be rewritten as,

$$\dot{\omega} = \psi(\omega) = \begin{bmatrix} p_x \omega_y \omega_z \\ p_y \omega_x \omega_z \\ p_z \omega_x \omega_y \end{bmatrix} \tag{9}$$

which is assumed to have a linear approximation about the estimate, $\hat{\omega}$ given by,

$$\delta\dot{\omega} = \mathbf{M}\delta\omega \tag{10}$$

where $M = \left(\frac{\partial \psi(\omega)}{\partial \omega}\right)_{\omega=\hat{\omega}}$ is a hollow matrix [13].

Finally, the linear approximation of the system (5) is given in [20, eq. 4.79] as $\mathbb{R}^{12}$ form,

$$\dot{x} = \mathbf{A}x + \mathbf{B}w \tag{11}$$

Based on Fig.1, the measurement equation is given as,

$$y = \begin{bmatrix} r_c \\ \mu \end{bmatrix} = \begin{bmatrix} \rho_c + R(\eta_c)(r + R(\eta_t)\rho_t) \\ \eta_t^* \otimes q \otimes \eta_c^* \end{bmatrix} \tag{12}$$

The linear approximation for the measurement is derived many times [13], [20], [14], and we avoid the derivation and state the formula as,

$$y = \mathbf{H}x = \begin{bmatrix} -R(\overline{q})\rho_{t_\times} & 0_{3,3} & 1_{3,3} & 0_{3,3} \\ 1_{3,3} & & & 0_{3,9} \end{bmatrix} \tag{13}$$

Since the Kalman Filter equations are standard, we do not explicitly write the prediction-update form. The interested reader is referred to [21] for seminal paper on equations. By using the aforementioned system of equations, the EKF was implemented with the Outlier-rejection (OR) scheme suggested in [22] to provide robustness against local outliers from the vision system.

C. Visual Servo for Approach and Tracking Phases

In the context of On-Orbit Servicing, although a target may be non-cooperative in terms of navigation aids, its model variance is more likely to be low. As a result, this volume of work focuses on being able to visually track a well identified object using the image-space measurements to reconstruct its pose information. Based on the model-based visual tracking and EKF's results, this control law is used to provide a velocity and pose for the inner dynamics controller. This is a position-based Visual Servo. For the case of tracking, the time-evolution of the error in configuration space is more meaningful than in the image space.

The error for the PBVS system is related to the output of the EKF through the logarithm map of $\hat{\mathbf{H}}_{eg}$ described as $\log_G : SE(3) \to \mathfrak{se}(3)$. The kinematic tracking velocity $V_m = \begin{bmatrix} \dot{\rho}_m^T & \omega^T \end{bmatrix}^T$ is given using the visual servo law. Firstly, the reference trajectory is used to compute a desired end-effector pose $H_{e,des} = (H_e \hat{H}_{eg}) H_{ge,ref}$, which is then used to construct the visual servo law:

$$V_m = K_1 \log_G (H_{e,v}^{-1} H_{e,des}) \tag{14}$$

where, $\dot{H}_{e,v} = Ad_{R_{e,v}} V_m \; \forall H_{e,v}(n) = H_e$ and $n \in [1..k]$ are the EKF sample times. This virtual frame acts like a trajectory for the manipulator and is solved as an initial value problem between two camera sampling times.

Finally, a compliant controller is placed in cascade after the visual servo which acts upon the Cartesian error. The

impedance error is a Cartesian error observed in the inertial frame and is denoted as,

$$e = \log_G(H_e^{-1} H_{e,v}) \tag{15}$$

This is the Cartesian error used for the compliant PD control, $\mathcal{F}_e = -K_P e - K_D \dot{e}$. From [7] the stability of the PD controller is established.

The corresponding joint torques are given as,

$$\tau = J_g^T \mathcal{F}_e + (1_{n,n} - {J_g}^T {J_g^*}^T)\Gamma \tag{16}$$

where Γ is given by

$$\Gamma = -K_{PN}(q - q_{ref}) - K_{DN}(\dot{q} - \dot{q}_{ref}) \tag{17}$$

1) Computer Vision for Approach and Tracking Phases: The visual pose estimation is based on work of [18] which relies on a known object geometry. The algorithm consisits of sampling model contours and edge detection, followed by a local optimization of the 3D rigd-body transform. The re-projection error is then minmized been visible model contour and image edges using Levenberg-Marquardt algorithm. The complex motion trajectory (tumbling and an abrupt change of angular velocity) of the target poses difficulty in visual tracking. In order to address this problem, a motion prediction scheme based on the extended kalman filter is integrated into the local optimization. The prediction relies on a simple kinematic motion model assuming constant frame to frame motion. Under this assumption, the visual tracker is able to cope with relatively higher image motion between two camera frames. Moreover, the short-term prediction scheme helps the visual tracker reduce the search space in order not to trap in potential local optima during minimization of the cost function. The qualitative tracking results are illustrated in Fig. 4, with an examplar tracked images during approach and tracking phases. Those in red indicate the projection of model contours on to the image edges at the estimated pose. The axes of the coordinate frame of grasping point are indicated in red, green and blue. The overlap of the model contours and image edges shows the successful tracking and pose estimation.

D. Joint Tracking Control for Stabilization Phase

The tracking controller for the rigidization phase is described in Fig. 3. The motion synthesizer provides an input expressed as $q_ref.\dot{q}_{ref}$. The control law for the tracking controller adopted here is then as follows

$$\boldsymbol{\tau}_{rig} = \mathbf{K}_p(q_{ref} - q) + \mathbf{K}_d(\dot{q}_{ref} - \dot{q}) \tag{18}$$

where $\mathbf{K}_p$ and $\mathbf{K}_d$ are (7x7) diagonal matrices including the stiffness and damping gains respectively. The first proportional term plays an inportant role, since itbrings the system into a feasible final configuration, as dictated by the motion planner.

III. Experimental facility

The OOS-SIM experimental facility is described in detail in [23]. In order to simulate the motion of the two satellites after the grasp, a master-slave approach was used, in which the Target followed the Light-Weight Robot through the internal actions which the latter imparted on it. The Chaser instead was commanded by the online integration of the equations of conservavation of linear and angular momentum of the multibody system consisting of the Chaser and the robot manipulator:

$$\mathbf{L}_C(\mathbf{V}_e) + \mathbf{L}_{LWR}(\mathbf{V}_e, \mathbf{q}, \dot{\mathbf{q}}) = \mathbf{L}_T(\mathbf{V}_t), \tag{19}$$

where $\mathbf{L}_C$, $\mathbf{L}_{LWR}$ and $\mathbf{L}_T$ are the momentum of the Chaser, of the Light-Weight Robot and of the Target respectively. Note in fact, that the left-hand side is zero before the grasp. After the grasp, the inertial parameters of the robot end-effector are updated to include those of the Target, in function of the measured relative pose with respect to it. As such, the total momentum remains constant and Eq. (19) can be solved for the twist of the Chaser $\mathbf{V}_e$ and integrated in time to obtain its pose $\mathbf{H}_e$.

IV. Results

Fig. 5 shows the motion planner solution for a Target spin velocity of 2 deg/s. The figure shows the three phases of the maneuver. The transition from the approach to the tracking phase is smooth, as the end-effector alligns itself with the grasping point on the Target. The values of the parameters $\phi_{e\,mid}$ and $\phi_{e\,delta}$ was set to 40 and 7.5 deg respectively. The tracking phase lasts 3.5 seconds to allow for the gripper to close onto it. The rigidization phase lasts five seconds and the number of parameters per joint was $N_B = 6$. The solution resulted from a global search, which found different local minima, as expected, due to nonlinearity of the optimization problem.

Fig. 6 shows the measured robot manipulator joint position error, defined as $\mathbf{q}_{ref} - \mathbf{q}$, for the cases of Target spin velocities of 1 deg/s and 2 deg/s respectively. In the experiments on the OOS-SIM facility it was in fact found that the rigidization task for the 2 deg/s case was intractable with the current setup. As such the experimental results presented here for the rigidization phase only relate to the 1 deg/s case. In Fig. 7 the measurement of the Target motion is shown, in which we observed a very slow damping of the angular velocity.

Returning to Fig. 6, the error in the joint position is shown to improve substantially for the second case, due to the fact that the EKF and the joint motion parameterization shown in Fig. 5 is used (the manuever was not possible without these new features). While the first provides a soomther estimate of $\mathbf{H}_{ge}$ (as shown below), the latter reduces the pixel velocity by a factor of two. At the point of transition between the tracking and the rigidization phase ($t = 28$), the joint position error becomes zero, due to the recomputation

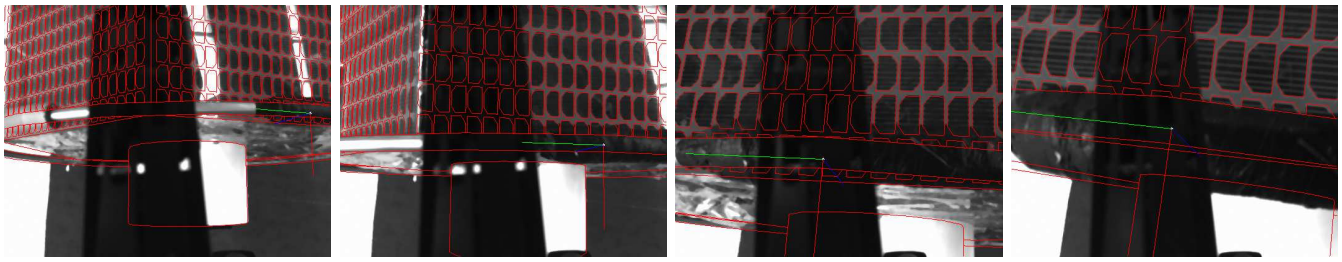

Fig. 4. Exemplar images during visual tracking for approach and tracking phase. The overlay of the model contours and image edges shows the successful tracking and pose estimation. The axes of the coordinate frame of grasping point is indicated in red, green and blue. The target features reduce significantly at very close range because of gripper finger occlusion (black in the image) and camera field of view.

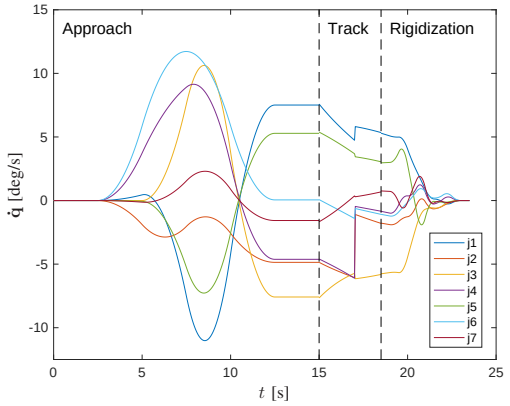

Fig. 5. Simulated robot manipulator joint velocities for a Target spin velocity of 2 deg/s.

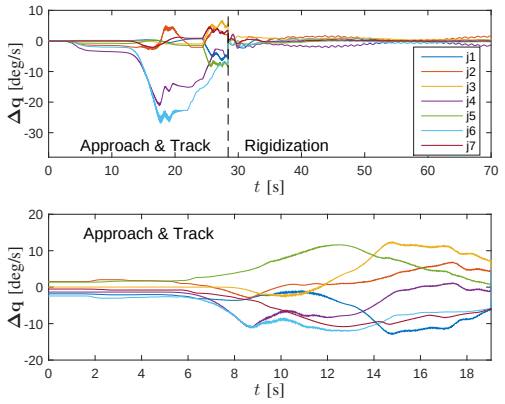

Fig. 6. Measured robot manipulator joint position error for Target spin velocity of 1 deg/s (top) and 2 deg/s (bottom). The the latter case only the approach phase is shown, in which however the use of the EKF and of the extended parameterization of the joint motion give a net increase in joint-space tracking performance.

of the reference trajectory (see SectionII-A.2). The tracking error in the rigidization phase is then shown to remain small.

In Figures 8, the output error norm of the position r_c of EKF and Vision is compared with respect to the ground

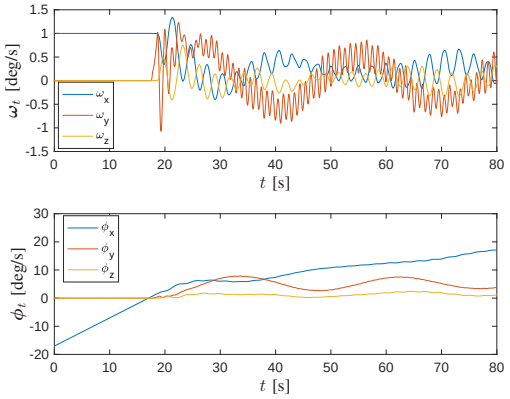

Fig. 7. Measured Target angular velocity (top) and orientation (bottom) for the 1 deg/s case. The post-grasping tumbling of the sattellite stack is visible in the (reduced) rotation about the x-axis.

truth. The EKF error norm is represented as a function of χ^2 of the incoming Vision data. It is observed that the χ^2 measure degrades for $t > 0$ and $||r_c|| \to 0$. In the fig.8, the degradation of vision data culminates in measurement failures towards the end. The EKF estimates though stay bounded. In 9 shows the velocity convergence of the filter. It also demonstrates the orientation degradation as $||r_c|| \to 0$. This observation betrays a general problem with the eye-in-hand configuration of visual servoing, since, in close proximity to target, the observability degeneracy degrades camera-based measurement. The EKF is computationally efficient owing to the three parameter, quaternion-propagation instead of 4.

V. Discussion and Conclusion

The main achievement of this paper is to demonstrate the feasibility of a grasping strategy based on Target motion prediction, robot motion planning and trajectory tracking, specifically for the latter step. In fact, due to modelling uncertainties, the reference trajectory has to be partially adapted online. A method is presented and validated here, which does that while favouring joint position related motion constraints. The tracking controller is then shown to

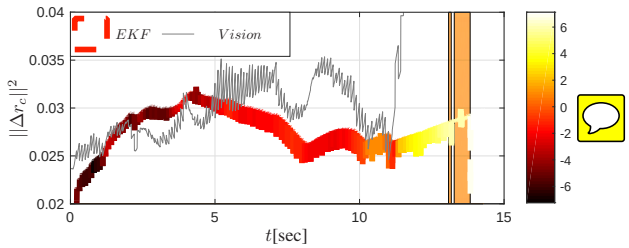

Fig. 8. Comparison of direct-Vision measurements with EKF, based on the norm $||\Delta r_c^2||$. As $||r_c|| \to 0$, the statistical measure of fitness, $\chi^2 \to \infty$ eventually leading to pose estimation failures due to degenerate observability.

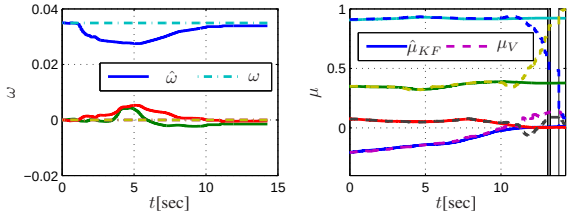

Fig. 9. a) Target state estimation (ω) in EKF for tracking control. b) Divergence of visual tracking's orientation estimate towards end of maneuver.

succesfully perform the task. In this way, the fesible (and optimal) motion planning solutions computed offline, with much computational effort, are shown to be useful for control purposes.

A second major achievement of this paper is the implementation of an Extended Kalman Filter which ensures a Lipschitz continuity of measurements, which makes the control method robust for executing the grasping task. It is in fact shown that the visual tracking algorithm adoped may become degenerate in the vicinity of the Target. Furthermore, the new formulation to reduce the computational footprint has been shown to give a signifcant improvement.

Future work will also be dedicated to improving the performance of the facility in the coupled configuration. Furthermore, the sensing of the Chaser, to include typical sensor noise and sampling times, will be considered.

REFERENCES

[1] T. Imaida, Y. Yokokohji, T. Doi, M. Oda, and T. Yoshikwa, "Ground-space bilateral teleoperation experiment using ets-vii robot arm with direct kinesthetic coupling," in *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation*, vol. 1, pp. 1031–1038 vol.1, 2001.

[2] M. W. James Shoemaker, "Orbital express space operations architecture program," 2003.

[3] F. Sellmaier, T. Boge, J. Spurmann, S. Gully, T. Rupp, and F. Huber, "On-orbit servicing missions: Challenges and solutions for spacecraft operations," in *SpaceOps 2010 Conference*, April 2010.

[4] J. Telaar, I. Ahrns, S. Estable, W. Rackl, M. De Stefano, R. Lampariello, N. Santos, P. Serra, M. Canetri, F. Ankersen, *et al.*, "Gnc architecture for the e. deorbit mission," in *7th European Conference for Aeronautics and Space Sciences*, 2017.

[5] B. Siciliano and O. Khatib, *Springer handbook of robotics*. Springer, 2016.

[6] S. A. A. Moosavian and E. Papadopoulos, "Free-flying robots in space: an overview of dynamics modeling, planning and control," *Robotica*, vol. 25, no. 5, pp. 537–547, 2007.

[7] M. D. Stefano, J. Artigas, A. Giordano, R. Lampariello, and A. Albu-Schaeffer, "On-ground experimental verification of a torque controlled free-floating robot," in *13th Symposium on Advanced Space Technologies in Robotics and Automation (ASTRA 2015)*, 2015.

[8] K. Yoshida, D. Dimitrov, and H. Nakanishi, "On the capture of tumbling satellite by a space robot," in *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pp. 4127–4132, 2006.

[9] F. Aghili, "A prediction and motion-planning scheme for visually guided robotic capturing of free-floating tumbling objects with uncertain dynamics," *IEEE Transactions on Robotics*, vol. 28, no. 3, pp. 634–649, 2012.

[10] F. Aghili, "Optimal control of a space manipulator for detumbling of a target satellite," in *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pp. 3019–3024, IEEE, 2009.

[11] A. Flores-Abad, Z. Wei, O. Ma, and K. D. Pham, "Optimal control of a space robot to approach a tumbling object for capture with uncertainties in the boundary conditions," in *AIAA Guidance, Navigation, and Control (GNC) Conference*, p. 4522, 2013.

[12] R. Lampariello and G. Hirzinger, "Generating feasible trajectories for autonomous on-orbit grasping of spinning debris in a useful time," in *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pp. 5652–5659, IEEE, 2013.

[13] F. Aghili and K. Parsa, "Adaptive motion estimation of a tumbling satellite using laser-vision data with unknown noise characteristics," in *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 839–846, Oct 2007.

[14] M. D. Lichter and S. Dubowsky, "State, shape, and parameter estimation of space objects from range images," in *Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on*, vol. 3, pp. 2974–2979 Vol.3, April 2004.

[15] U. Hillenbrand and R. Lampariello, "Motion and parameter estimation of a free-floating space object from range data for motion prediction," in *Proceedings of i-SAIRAS*, 2005.

[16] Z. Z. Hong and D. Gangqi, *Active Space Debris Removal by Visual Servo Controlled Autonomous Robotics*. AIAA SPACE Forum, American Institute of Aeronautics and Astronautics, Sept. 2016.

[17] V. Lippiello, B. Siciliano, and L. Villani, "A position-based visual impedance control for robot manipulators," in *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pp. 2068–2073, April 2007.

[18] T. Drummond and R. Cipolla, *Real-time visual tracking of complex structures*. 2002.

[19] M. Landis, "Attitude Error Representations for Kalman Filtering," *Journal of Guidance, Control, and Dynamics*, vol. 26, no. 2, pp. 311–317, 2003. doi: 10.2514/2.5048.

[20] B. E. Tweddle, *Computer Vision-Based Localization and Mapping of an Unknown, Uncooperative and Spinning Target for Spacecraft Proximity Operations*. PhD thesis, Massachusetts Institute of Technology, 2013.

[21] R. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME–Journal of Basic Engineering*, vol. 82, no. Series D, pp. 35–45, 1960.

[22] H. Mishra and P. Schmidt, "Motion and parameter estimation for the robotic capture of a non-cooperative space target considering egomotion uncertainty," in *14th Symposium on Advanced Space Technologies in Robotics and Automation (ASTRA 2017)*, 2017.

[23] J. Artigas, M. D. Stefano, W. Rackl, R. Lampariello, B. Brunner, W. Bertleff, R. Burger, O. Porges, A. Giordano, C. Borst, and A. Albu-Schaeffer, "The oos-sim: An on-ground simulation facility for on-orbit servicing robotic operations," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2854–2860, May 2015.