

The OOS-SIM: An On-ground Simulation Facility For On-Orbit Servicing Robotic Operations

Jordi Artigas, Marco De Stefano, Wolfgang Rackl, Roberto Lampariello, Bernhard Brunner, Wieland Bertleff, Robert Burger, Oliver Porges, Alessandro Giordano, Christoph Borst, Alin Albu-Schaeffer*

Abstract—On-orbit servicing involves a new class of space missions in which a servicer spacecraft is launched into the orbit of a target spacecraft, the client. The servicer navigates to the client with the intention of manipulating it, using a robotic arm. Within this framework, this work presents a new robotic experimental facility which was recently built at the DLR to support the development and experimental validation of such orbital servicing robots. The facility allows reproducing a close-proximity scenario under realistic three-dimensional orbital dynamics conditions. Its salient features are described here, to include a fully actuated macro-micro system with multiple sensing capabilities, and analyses on its performance including the amount of space environment volume that can be simulated.

I. INTRODUCTION

Ideas to assemble space structures by a flying robot in orbit or to conduct servicing missions to existing satellites have been discussed since the 1980s [1]. The current International Space Station (ISS) has not been constructed by such free-flying robots but through many hours of human Extra Vehicular Activities (EVA) with the assistance of the Shuttle Remote Manipulator System (SRMS, Canadarm) and Space Station Remote Manipulator System (SSRMS, Canadarm2). Cost studies of system failures (up to billions of dollars) are summarized in [2]. Examples of earth-orbiting spacecraft failures are:

- OAO-A2: loss of star field sensor due to debris collision
- OAO-C, Olympus 1 and Exosat: loss of attitude control
- NOAA-6: hydrazine accidentally vented, causing uncontrolled tumble
- Hipparcos: launched into the wrong orbit due to apogee engine failure
- ATS-6: Thruster failure.

Another example that shows the extensive costs associated to on-orbit equipment is the Hubble Space Telescope (HST). Its original costs of \$500 million in 1990 increased up to \$10 billion until 2010, mainly due to four maintenance manned missions [3].

Space robots can facilitate manipulation, assembling, or servicing functions in orbit, either assisting or surrogating astronauts. On the other hand, space debris, the collection of defunct materials in the space orbits, have continually increased in number over recent years [1]. Commonly adopted debris mitigation measures might be insufficient to keep the space environment safe from these accidental collisions

All authors are with the Institute of Robotics and Mechatronics in the German Aerospace Center (DLR), 82234 Wessling, Germany, email: jordi.artigas@dlr.de.



Fig. 1: Overview of the OOS-SIM. Left: Client satellite; Right: Servicer satellite and manipulator

[2]. Rather than disposing mal-functional satellites, On-Orbit Servicing (OOS) can undertake maintenance tasks and put systems back into operation. An OOS system must be capable of grasping, stabilizing and berthing uncooperative free-floating satellites. Space robotics provides a reliable solution to accomplish these tasks.

The system presented in this article considers a *servicer* satellite equipped with a robot manipulator. The servicer is capable of approaching a target satellite, named *client* and represents any orbiter that needs to be maintained, repaired or de-orbited. Once the servicer is in the near range of the target, the servicer can deploy its manipulator to grasp the client and execute the servicing tasks, that is, manipulation tasks. While manipulation is not a new technology in robotics, micro-gravity in the orbital environment requires special attention. Micro-gravity affects the motion dynamics of the robot arm and that of the manipulated object; produces reaction dynamics at the robot base body (i.e. the servicer satellite); introduces impact dynamics issues between the robot arm's tool and the manipulated object; and can provoke vibration dynamics due to structural flexibility.

Servicing spacecraft technologies must be thoroughly tested before launch. Ground testing of orbital servicing tasks is a critical step along the path of ensuring successful OOS missions. One of the main difficulties in developing space robots is the difficulty of reproducing true micro-gravity environments on Earth. In general, computer simulations fail in reproducing true physical interaction, e.g. when the servicer manipulator grasps the target satellite. Air-bearing simulators

are limited to small angular motion or planar motion. The free-fall approach [4] can provide three dimensional micro-gravity environment but only for a few seconds (20-30) and limited cargo. The neutral buoyancy method introduces notable undesired fluid dynamics. Suspended systems can effectively simulate three dimensional micro-gravity but become rather complex for a macro-micro robot configuration, as the one presented in this work. See [4] for a discussion on methods for testing free-flying robots.

Robot based facilities, i.e. hardware-in-the-loop simulators, can effectively implement active gravity compensation, can accommodate complex systems, e.g. a free-flying robot, and allow unconstrained motions within the robot workspaces. Moreover, they provide unlimited time to perform the simulations. [5] discuss the use of industrial robots to simulate autonomous rendezvous and docking maneuvers between two satellites. The Lockheed Martin Space Operations Simulation Center (SOSC) allows ground testing of rendezvous, proximity operations and docking maneuvers [6]. EPOS is a robotic facility designed to simulate on-orbit rendezvous and docking between two satellites [7]. One fundamental difference between existing rendezvous simulation facilities and the OOS-SIM is the servicing robot manipulator: The goal of the facility presented in this paper is to simulate the dynamics involved in the physical interaction between a robot manipulator, - which is mounted on a servicer satellite - and a target satellite.

II. SYSTEM DESCRIPTION

Fig. 1 shows a general view of the OOS-SIM facility. The system consists of four main *actors*: The servicer satellite, the servicer manipulator, the client satellite and an on-ground station.

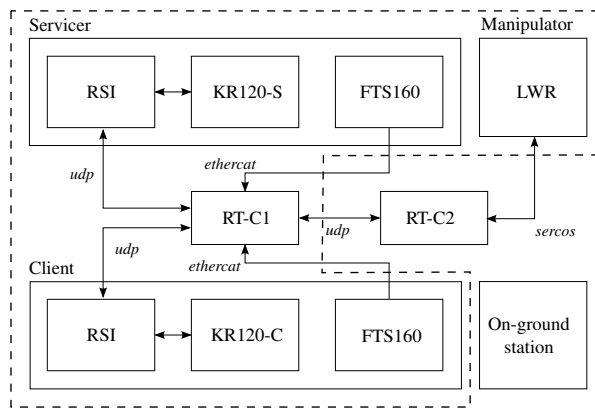


Fig. 2: System overview. The dash line divides simulated and mission-like elements

A system overview with the main hardware elements and their respective interfaces is shown in Fig. 2. The figure highlights the elements that are simulated, that is, those corresponding to the space environment and those that are not simulated but taken from the real mission setup. Thus, on-ground station and main control CPU (RT-C2) are regarded as mission-like hardware.

The RT-C1 is a computer running on VxWorks at a sampling rate of 4ms. This computer computes the simulated servicer and target free-floating dynamics whose outputs (Position and orientation signals in the Cartesian space) are commanded to both KR120's through the RSI interface. For instance, real time computation of the later defined dynamics (1) and (2) takes place in this computer. Furthermore, the RSI interface is synchronized to RT-C1 in order to preserve real time determinism. The RT-C2 is a computer running on VxWorks at 1ms. This computer is the actual robot control CPU for the servicer manipulator, denoted in Fig. 2 as LWR (Light-Weight-Robot) in Table I, performing joint level and Cartesian level real time control. The sensor data from both FTS160 are fed to the RT-C2 using the real time clock of the RT-C2. Furthermore, the RT-C2 is equipped with a Sercos interface to communicate with the LWR in real time.

Table I and Table II provide details of the robots involved in the facility and the sensors used respectively. A second LWR as Haptic Manipulator is used by a human operator to control the servicer manipulator in telepresence mode.

	Robot	DoF	Interface, T_s	I/O
Servicer	KR120	6	Ethernet, 4ms	X/X
Manipulator	LWR	7	Sercos, 1ms	X, F/X, F
Client	KR120	6	Ethernet, 4ms	X/X
Gripper	Robotiq 3f	4+8	Ethercat, 1ms	X, I/X, I
Haptic M.	LWR	7	Sercos, 1ms	X/X

TABLE I: Main hardware elements of the OOS-SIM. X=position; F=force; I=current; T_s =sampling time

	Sensor	Interface, T_s	Function
Client FTS	FTS160	EtherCat, 1ms	Physical interaction
Servicer FTS	FTS160	EtherCat, 1ms	Physical interaction
Haptic FTS	FTS78	RS485, 1ms	Dynamics compens.
Cameras	GigE	Ethernet, 1ms	Visual servoing
IMU	Xsense MTI	RS232	Star field simulation

TABLE II: Sensors used in the facility. FTS=Force-Torque Sensor; IMU=Inertial Measurement Unit

A. Servicer satellite (KR120-S)

The KR120 (denoted as KR120-S in Fig. 2) is a well suited industrial robot to perform the simulation of a free floating satellite. It can carry payloads up to 120kg with a repeatability of 0.06 mm and joint speeds up to 240°. Furthermore, the KRC-4 (Kuka Robot Controller) offers an external interface with control rates up to 4ms through the RSI (Robot Sensor Interface). In order to perform the servicing simulations, a satellite mockup has been mounted at the end-effector of the KR120-S. Furthermore it integrates a docking interface and some OOS elements such as a re-fueling interface.

B. Servicer manipulator (LWR)

The KR4+ is a light-weight-robot with seven degrees-of-freedom (DoF) equipped with joint torque sensors. With its position, torque and impedance interfaces on joint level, it is an adequate robot to perform manipulation tasks. Furthermore, its weight-payload ratio close to 1 makes it suitable for the envisaged simulations. It is mounted on the end-effector of the servicer KR120 with a force-torque sensor FTS160 in between.

C. Client satellite (KR120-C)

The client satellite is simulated using another KR120 placed 3 meters away from the servicer robot with a base rotation of 180°. Similar to the Servicer Satellite a mockup is mounted at the end-effector of the KR120-C with a force-torque sensor FTS160 in between. This mockup allows grasping maneuvers on its circular ring (resembling an Orbital Servicing Adapter (OSA)).

D. Ground Control Station

The ground station is not a simulated element as it can be the same one used in a real mission. It holds one or more operators and provides the necessary interfaces to control the mission environment or the simulated one. In general, the station will allow two main operational modes from ground: semi-autonomy and telepresence, both described in Sec. IV. For the latter, a human scaled force-feedback device is considered [8].

III. FUNCTIONAL DESCRIPTION

In this section the satellite dynamics implemented on the client robot and on the servicer with the manipulator (free-floating) are presented. As it will be seen and in order to simulate on-orbit servicing operations, both, servicer and client robot, operate in a common workspace. This allows the testing of tracking, grasping and stabilization of the free-tumbling client satellite through the servicer manipulator. Client docking maneuvers on to the servicer can be tested with some limitation (see Sec. IV). A complete workspace has been also developed to analyze the system capability maps, shown below.

1) *Client Robot (KR120-C)*: The client robot is a six DoF industrial robot able to simulate a free-floating dynamics of a satellite in six DoF. The robot can be controlled only in admittance mode, that is, the modified variable can only be position. Thus, the free-floating dynamics of a rigid body (6 DoF), based on the Newton-Euler equations [9], is used to compute the desired position command to the industrial robot as:

$$\mathbf{v} = \int_0^t \frac{\mathbf{F}_{CM}}{M} dt, \quad (1)$$

$$\boldsymbol{\omega} = \int_0^t \mathbf{I}^{-1}(\boldsymbol{\tau}_{CM} - \boldsymbol{\omega} \times \mathbf{I}\boldsymbol{\omega}) dt. \quad (2)$$

where $\mathbf{v} \in \mathbb{R}^{3 \times 1}$ is the linear velocity of the satellite and $\boldsymbol{\omega} \in \mathbb{R}^{3 \times 1}$ is the satellite angular velocity. $\mathbf{F}_{CM} \in \mathbb{R}^{3 \times 1}$ are the forces and $\boldsymbol{\tau}_M \in \mathbb{R}^{3 \times 1}$ are the torques applied on the center of

mass (CM) where the gravity components are compensated in the model. $M \in \mathbb{R}$ and $\mathbf{I} \in \mathbb{R}^{3 \times 3}$ are the mass and the inertia matrix of the satellite to be simulated respectively. In order to measure the external forces and torques, needed to compute the dynamics, a FTS is placed at the end effector of the client robot. Thus, forces and torques measured by the sensor need to be transformed in the satellite center of mass, as:

$$\mathbf{F}_{CM} = R_{EE,CM} \mathbf{F}_{FS} \quad (3)$$

$$\boldsymbol{\tau}_{CM} = \mathbf{p}_{EE,CM} \times (R_{EE,CM} \mathbf{F}_{FS}) + R_{EE,CM} \boldsymbol{\tau}_{TS} \quad (4)$$

where $R_{EE,CM}$ is the rotation matrix between the end-effector and the center of mass (CM); and $\mathbf{p}_{EE,CM}$ is the vector from the end-effector to the center of mass. By integrating (1) and (2) and using the Euler-Rodrigues formulation [10], an homogeneous transformation matrix ΔX_d (that represents a translational and rotational increment) is computed and commanded through the KR120's inverse kinematics. The data flow is shown in Fig. 3. The input to the simulated

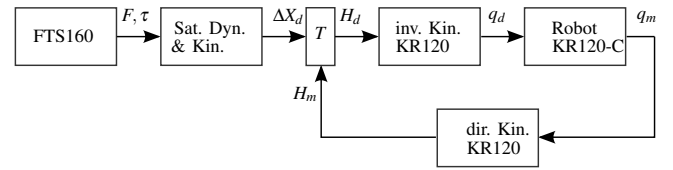


Fig. 3: Data flow for the client dynamics simulation. (Sat.Dyn.= Satellite Dynamics; Dir/Inv Kin= direct / inverse Kinematics).

dynamics are the measured forces and torques by the sensor. Thus, the satellite dynamics can be computed according to the external physical interaction, which results in *desired* translational and rotational increment, i.e. ΔX_d . The command is then projected onto the current KR120 Cartesian frame, $H_d = H_m \Delta X_d$, which is the command to the KR120 through the inverse kinematics. This is how the simulated satellite motion is implemented on the robot Cartesian space.

2) *Servicer robot (KR120-S + LWR)*: The Servicer system comprises the KR120-C, which emulates the servicer satellite dynamics, and the 7 DoF LWR mounted on the end-effector of the KR120-S. This configuration results in a macro-micro system of 6+7 DoF. The combined robot can be considered as a hybrid system since the motion of the satellite (KR120-S) is simulated as an admittance, i.e., the command to the KR120 can only be position, while the LWR can be torque or position controlled. The general equation of motion for free-flying robots can be expressed as [11]

$$\begin{bmatrix} \mathbf{H}_b & \mathbf{H}_{bm} \\ \mathbf{H}_{bm}^T & \mathbf{H}_m \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{x}}_b \\ \ddot{\boldsymbol{\theta}}_m \end{bmatrix} + \begin{bmatrix} \mathbf{c}_b \\ \mathbf{c}_m \end{bmatrix} = \begin{bmatrix} \mathbf{F}_b \\ \boldsymbol{\tau} \end{bmatrix} + \begin{bmatrix} \mathbf{J}_b^T \\ \mathbf{J}_m^T \end{bmatrix} \mathbf{F}_e, \quad (5)$$

where $\mathbf{H}_b \in \mathbb{R}^{6 \times 6}$, $\mathbf{H}_m \in \mathbb{R}^{7 \times 7}$, $\mathbf{H}_{bm} \in \mathbb{R}^{6 \times 7}$ are the inertia matrices of the base, manipulator and coupling inertia matrix between the base and the manipulator, respectively.

The vectors $\mathbf{c}_b \in \mathbb{R}^{6 \times 1}$ and $\mathbf{c}_m \in \mathbb{R}^{7 \times 1}$ are the non-linear velocity dependent term on the base and on the manipulator, respectively. $\mathbf{F}_b \in \mathbb{R}^{6 \times 1}$ and $\mathbf{F}_e \in \mathbb{R}^{6 \times 1}$ are the force torque wrenches acting on the center of mass of the base body or the end effector, respectively. The integration of the upper set of Eq. (5) leads to the total momentum of the system

$$\mathcal{L} = \int_0^t \mathbf{F}_b dt + \int_0^t \mathbf{J}_b^T \mathbf{F}_e dt = \mathbf{H}_b \dot{\mathbf{x}}_b + \mathbf{H}_{bm} \dot{\boldsymbol{\theta}}. \quad (6)$$

The motion of the servicer base to be rendered at the OOS-SIM can be either obtained from the integration of the complete dynamics equation in Eq. (5) or by integration of Eq. (6), which leads to

$$\begin{bmatrix} \mathbf{v}_b \\ \boldsymbol{\omega}_b \end{bmatrix} = \mathbf{H}_b^{-1} (\mathcal{L}_0 - \mathbf{H}_{bm} \boldsymbol{\theta}) - \int_0^t \mathbf{F}_b dt - \int_0^t \mathbf{J}_b^T \mathbf{F}_e dt, \quad (7)$$

where \mathcal{L}_0 is the total momentum at the initial time.

The data flow of the dynamics calculation is shown in Fig. 4. The main difference with respect to the previous

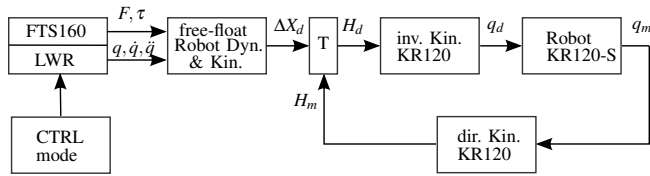


Fig. 4: Data flow for the servicer dynamics simulation. (Sat.Dyn.= Satellite Dynamics; Dir/Inv Kin= direct / inverse Kinematics).

control scheme in Fig. 3 is given by the joint states of the servicer manipulator (LWR). In Sec. IV two operational modes for space robots are discussed.

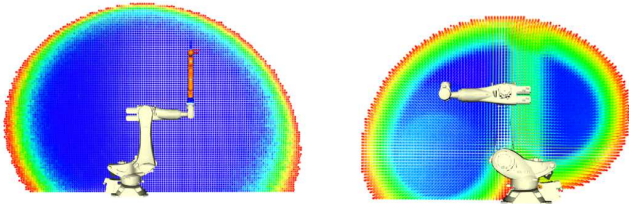


Fig. 5: Cross-sections of the *Servicer robot*(left) and *Client robot*(right) capability maps

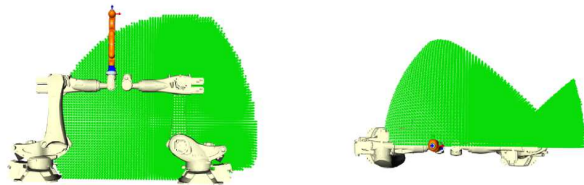


Fig. 6: Common workspace volume (cross-section) of the OOS-SIM, side-view (left) and top-view (right)

3) *Workspace*: The OOS-SIM is designed to operate in the intersecting workspace of the three robots, referred to as common workspace. We use Reachability and Capability

maps to assess the OOS-SIM workspace quality, shape and volume. Reachability map is a 6 dimensional discrete representation of robot workspace introduced in [12]. Capability map encodes the Reachability index [13] in color intensity and makes it feasible to visualize. Methods for generating such maps were described in [14]. In order to obtain the common workspace we treat the set-up as two serial kinematic chains; a 13-DOF (KR120 with LWR) and 6-DOF (KR 120). Capability maps for the two cases are shown in Fig. [5]. The color indicates how many of the discretized directions are reachable in a voxel, ranging from red (close to 0%) to blue (close to 100%). The underlying voxel grids can be merged into a new map to obtain the common workspace volume including the direction information. The intersection of the two maps is visualized in Fig. [6].

It is concluded that a space environment volume of approximately 23.28 m^3 can be simulated ¹.

IV. SIMULATED OPERATIONS

The main objective of the OOS-SIM facility is to serve as a platform to develop and test space robotics control algorithms. The devised OOS tasks include the grasping of a non-cooperative tumbling target satellite by means of a free-floating robot as well as repair and maintenance tasks such as re-fueling or (tele)manipulation of an ORU (Orbital Replacement Unit). To that end, two main operational modes are foreseen: Semi-autonomy and teleoperation. This section gives an overview of these operational modes and highlights the benefits of using the described simulation platform.

A. *Semi-autonomy*

The principal aspect of this operational mode is that the robot motion is based on a reference trajectory provided by a motion planner [15]. The goal is to provide extra operational safety, with respect to the motion constraints, such as collision avoidance, camera field of view limits, etceteras. The reference trajectory, which is itself based on a motion prediction of the tumbling target [16], is computed on ground and then uploaded to the robot in space, where a controller accounts for disturbances, as well as for modeling and prediction errors. The grasping point on the target is assumed to be predefined by an operator.

The main elements of this mode are shown schematically in Fig. 7. The tracking module includes a visual servoing algorithm that makes use of a pair of cameras located on the servicer manipulator end-effector. Fig. 8 provides visual detail on the cameras. Another aspect to be tested and evaluated is the performance of impedance control for free-floating robots. Indeed, compliance (see the tracking module in Fig. 7) can be very beneficial in these types of applications. Model inaccuracies or time delays are expected in space missions and can clearly affect the controller performance. This can result in potential damages on the client satellite

¹Note that the volume presented here is not precise with respect to a particular mission. The collisions of KR120 robots with satellite mockups and a particular grasping point on the client satellite would alter this volume one way or the other.

or the servicer manipulator. In contrast to classical position control, Cartesian impedance control offers *friendlier* interaction with the manipulated body and can therefore better cope with system inaccuracies.

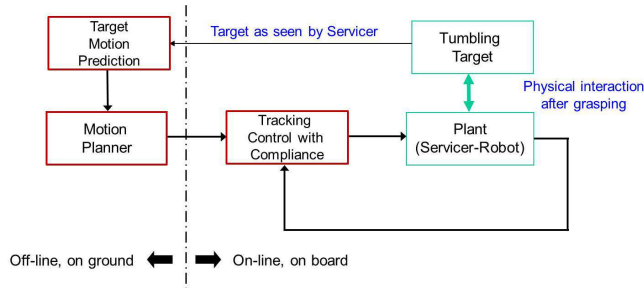


Fig. 7: System overview for semi-autonomous grasping mode

The OOS-SIM facility allows not only to develop the needed algorithms but also to validate the interaction between them. For instance, visual servoing can be tested along impedance control in a true free-floating setup. The control strategies which are implied in this operational mode, are thus analyzed under realistic motion dynamics and robot control behavior, as well as with true sensor signals (e.g. stereo camera at the robot end-effector).

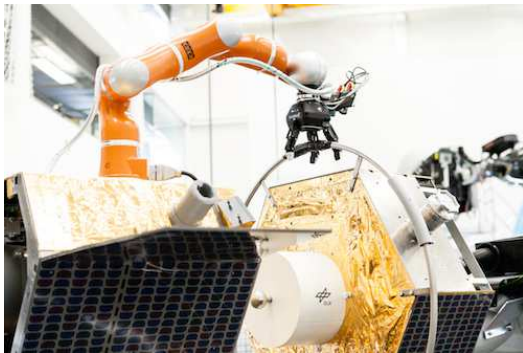


Fig. 8: The servicer manipulator is about to grasp the client satellite

B. Teleoperation

In teleoperation, the operator sees the images captured by the stereo camera and *feels* the interaction forces between the servicer manipulator and the client satellite from the ground station. Bilateral control is a distributed control mechanism that the exchange of mechanical between the user and the manipulated environment. To that end, a haptic device is used in order to capture the user desired motions for the space manipulator and to render feedback forces from the interaction between the manipulator and the spacecraft. In a sense, teleoperation combines the strength of robotics of interacting in remote environments with human skills. One of the main benefits of teleoperation is that is less model dependent than autonomy-based approaches and can therefore better cope with unstructured or poorly modelled

environments. This is specially useful in those tasks that require dexterous physical manipulation. Good examples are peg-in-hole tasks, (un)screwing tasks or of bionet / umbilical connectors manipulation. On the other hand, teleoperation for OOS missions presents two technical challenges:

1) System Stability in Time-Delayed Communications:

The closed-loop control system created between the haptic device and the servicer manipulator requires special attention. Some available space communication infrastructures provide real-time communication capabilities. These links are affected usually affected by considerable time delay, jitter and data losses. The CCSDS recommendations for tele-robotics set a limit of 1 second to allow force-feedback teleoperation. As shown in [17] and [18], dedicated GEO-based communication relays permit round-trip delays between on-ground station and orbiter of less than 650ms and contact windows close to one hour. Robust control methods that guarantee closed-loop stability in the presence of time delays, jitter and package loss are discussed in [19] and [20].

2) Teleoperation in Microgravity:

While this is a rather unexplored topic, the presented facility is a suitable platform for developing and testing bilateral control strategies that take into account both, time-delayed teleoperation with free-floating robots. These two factors combined can result in non-intuitive systems if not properly addressed. A key element for allowing intuitive control of the space manipulator is to *mask* the free-floating dynamics to the user such that he/she feels as if controlling a fixed base robot in the Cartesian space.

V. PERFORMANCE RESULTS

The main factors affecting the performance of the OOS-SIM facility are in the following discussed. Sensor noise, discretization, time delays and model inaccuracies can affect the simulated dynamics in the described set-up. Clearly, the performance of the simulation is highly dependent on the signal quality of the FTS. Therefore, its performance is here discussed. Further, experimental results are shown and validated with a off-line simulated dynamics in order to show the error given by the computed motion in the hardware in the loop and the off-line dynamics which represent the on-orbit scenario.

A. Force-Torque sensor performance

Sensor noise and drift play an important role. The DLR-FTS160 force-torque sensor is able to operate in a range of 500N/265Nm and provides noise signals below 1% for the forces and below 0.5% for the torques (both centered on zero). Furthermore, the sensors are temperature compensated. Tests with an initial heating phase of 30 minutes have shown sensor drifts below 0.1 N after 1 hour on the three translational measurements. This drift is handled by implementing a dead band. By way of illustration, taking the mass value of the TERRASAR-X satellite, $m = 1200 \text{ kg}$, and a simulated

operation that lasts 20 min. results in an effective residual drift (after the operation) of 0.014 mm/s.

B. Satellite Dynamics

In order to evaluate the rendered dynamics of the satellites, a comparison between the motions resulting from a pure simulation (here called SIM) and from a hardware in the loop experiment (here called HIS) was done. For the servicer evaluation a trajectory was executed with the LWR and the resulting reaction motion of the Servicer base was then compared. As input for the simulation the same measured trajectory was used. The used scenario data are listed in Tab. III.

Parameter	Value	Unit
m_b	400	kg
CoM_b	[0.48750,00120.5515]	m
I_b	diag(900,900,900)	kgm ²
$m_m(total)$	18	kg
m_c	40.0	kg
CoM_c	[0.50,10.2]	m
I_c	diag(4.0,4.0,4.0)	kgm ²

TABLE III: Simulation model parameters. *b*: Servicer satellite base; *m*: manipulator; *C*: client

The difference of the corresponding satellite base motion is shown in Fig. 9. The servicer base moves according with the programmed trajectory and follows the simulated motion.

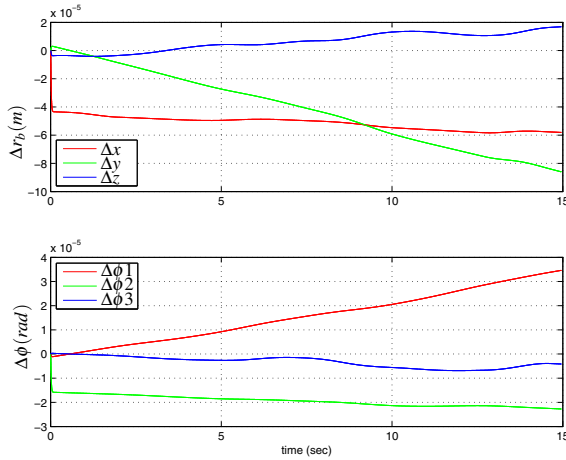


Fig. 9: Time history of the position (r_b) and orientation (ϕ) error for the Servicer between the SIM and the HIL measurements

For the position a small drift of less than $32 * 10^{-5} m/min$ is extrapolated from the measured motion. The orientation drift is extrapolated with less than $16 * 10^{-5} rad/min$.

C. Interaction

In this section the performance of the interaction between the Servicer and the Client is analyzed. For this purpose, we grasped the Client with the gripper mounted on the Servicer manipulator. In this connected configuration of the system the manipulator performed a motion to a different position.

The Client had to follow this motion. Ideally, the Client motion should be the same as the end-effector motion of the Servicer manipulator.

Fig. 10 shows the results of the performed experiment. In the first part of the motion ($0 < t < 1.3 sec$) the Servicer end-effector was holding the Client without any desired motion. In the second part of the motion ($t > 1.3 sec$) the Servicer end-effector started to move to the desired target point. At the beginning of the motion, a small jump of the error can be observed. A reason for that could be the fact, that the hold between the gripper and the grasped ring on the Client allows a certain tolerance to move. The grasp design of the OOS-SIM was meant to be non-cooperative.

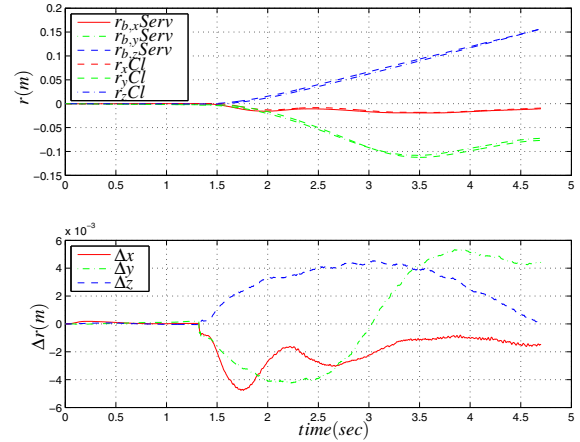


Fig. 10: Time history of the absolute motions of Servicer end-effector and Client (top) and the difference of the relative motion between the Servicer end-effector and the Client motion for the grasped interaction (bottom)

During the experiment a maximum error of less than $\Delta r < 0.005 m$ can be observed. Further reasons for the uncertainties in the interaction could be found in the flexibilities of the LWR and the FTS160 and the non-ideal grasp.

D. Dynamics due to discretization

The formality given by Newton-Euler equation (1)-(2), is partly broken due to the discrete computer-based implementation of the targeted simulated dynamics. Discretization, quantization and transmission delays are factors that can distort the simulation. This results, in additional undesired dynamics that can compromise the stability of the system, specially in physical contact situation. In particular, time delay is a well known factor that can disrupt system stability. Experiments have shown an intrinsic time delay of 16ms in the closed loop system depicted in Fig. 2. In general terms, it can be stated that the discretization induced dynamics do not pose a limitation in grasping and stabilization tasks, that is, in the physical contact between the servicer manipulator and the client satellite. On the other hand, the effects of time delay can become critical in servicer - client docking maneuvers. A formal analysis addressing this specific issue is presented in [21].

VI. CONCLUSIONS

The OOS-SIM represents a new class of hardware-in-the-loop facilities involving the concept of a space robot. Some results and limitations have been shown. One remarkable challenge is the stability of the system in the grasping configuration, that is, when the LWR grasps the ring of the KR120-C the three robots become mechanically connected, resulting in a kinematic chain of 19 DoF. This issue can be accommodated to some extent by virtue of the LWR, which is a torque controlled robot and allows impedance control. In general, impedance control is recommended for space robots at any rate since compliant behaviors are desirable. The compliance rendered by the impedance controller acts positively on the stability of the facility, specially in contact situations, as it is capable of absorbing some of the undesired energy generated due to signal discretization and internal time delays. While this effect facilitates the simulation, by no means the stability of the facility should rely on it. This remains as an issue and will be addressed in future work. First results in rendering the simulated dynamics in a passive manner are presented in [21]. On the other hand, one of the main objectives of the OOS-SIM is to define the specifications of the control algorithms that will fulfill the requirements imposed by future space missions. Future work will also deal with end-to-end tests, where a complete OOS mission infrastructure can be tested using the simulated microgravity space.

REFERENCES

- [1] D. Akin, M. I. of Technology, G. C. M. S. F. Center, U. S. N. Aeronautics, S. A. Scientific, and T. I. Branch, *Space applications of automation, robotics and machine intelligence systems (ARAMIS) - phase II*, ser. NASA contractor report. National Aeronautics and Space Administration, Scientific and Technical Information Branch, 1983, no. v. 3.
- [2] A. Ellery, J. Kreisel, and B. Sommer, "The case for robotic on-orbit servicing of spacecraft: Spacecraft reliability is a myth," *Acta Astronautica*, vol. 63, no. 56, pp. 632 – 648, 2008.
- [3] J. William F. Ballhaus, *James Webb Space Telescope (JWST) Independent Comprehensive Review Panel (ICRP) Final Report*, ser. NASA contractor report. National Aeronautics and Space Administration, Scientific and Technical Information Branch, 2010, no. v. 3.
- [4] C. Menon, S. Busolo, S. Cocuzza, A. Aboudan, A. Bulgarelli, C. Betanini, M. Marchesi, and F. Angrilli, "Issues and solutions for testing free-flying robots," *Acta Astronautica*, vol. 60, no. 12, pp. 957 – 965, 2007.
- [5] O. Ma, A. Flores-Abad, and T. Boge, "Use of industrial robots for hardware-in-the-loop simulation of satellite rendezvous and docking," *Acta Astronautica*, vol. 81, no. 1, pp. 335 – 347, 2012.
- [6] Z. Milenkovic and C. D'Souza, *The Space Operations Simulation Center (SOSC) and Closed-loop Hardware Testing for Orion Rendezvous System Design*. American Institute of Aeronautics and Astronautics, 2015/02/27 2012. [Online]. Available: <http://dx.doi.org/10.2514/6.2012-5034>
- [7] T. Boge, T. Wimmer, O. Ma, and T. Tzschichholz, "Epos—using robotics for rvd simulation of on-orbit servicing missions," in *AIAA Modeling and Simulation Technologies Conference*, 2010, pp. 2–5.
- [8] T. Hulin, K. Hertkorn, P. Kremer, S. Schatzle, J. Artigas, M. Sagardia, F. Zacharias, and C. Preusche, "The dlr bimanual haptic device with optimized workspace," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, May 2011, pp. 3441–3442.
- [9] P. C. Hughes, Ed., *Spacecraft Attitude Dynamics*. Dover Publications, Inc., Mineola, New York, 2004.
- [10] B. Siciliano and O. Khatib, Eds., *Springer Handbook of Robotics*. Springer, 2008.
- [11] R. Featherstone, *Rigid Body Dynamics Algorithms*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2007.
- [12] R. Diankov, "Automated construction of robotic manipulation programs," Ph.D. dissertation, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, September 2010.
- [13] F. Zacharias, C. Borst, and G. Hirzinger, "Online generation of reachable grasps for dexterous manipulation using a representation of the reachable workspace," in *Advanced Robotics, 2009. ICAR 2009. International Conference on*, June 2009, pp. 1–8.
- [14] O. Porges, T. Stouraitis, C. Borst, and M. A. Roa, "Reachability and capability analysis for manipulation tasks," in *ROBOT (2)*, ser. Advances in Intelligent Systems and Computing, M. A. Armada, A. Sanfeliu, and M. Ferre, Eds., vol. 253. Springer, 2013, pp. 703–718.
- [15] 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*, Nov 2013, pp. 5652–5659.
- [16] U. Hillenbrand and R. Lampariello, "Motion and Parameter Estimation of a Free-Floating Space Object from Range Data for Motion Prediction," in *'i-SAIRAS 2005' - The 8th International Symposium on Artificial Intelligence, Robotics and Automation in Space*, ser. ESA Special Publication, vol. 603, Aug. 2005.
- [17] E. Stoll, U. Walter, J. Artigas, C. Preusche, P. Kremer, G. Hirzinger, J. Letschnik, and H. Pongrac, "Ground verification of the feasibility of telepresent on-orbit servicing," *J. Field Robot.*, vol. 26, no. 3, pp. 287–307, Mar. 2009. [Online]. Available: <http://dx.doi.org/10.1002/rob.v26:3>
- [18] R. Lamparello, N. W. Oumer, J. Artigas, W. Rackl, G. Panin, R. Purschke, J. Harder, W. Walter, J. Frickel, K. Ravandoor, J. Scharnagl, K. Schilling, K. Landzettel, and G. Hirzinger, "Forrost: Advances in on-orbit robotic technologies," in *2015 IEEE Aerospace Conference*. Montana, USA: IEEE, March 7-14 2015.
- [19] J. Artigas, J.-H. Ryu, C. Preusche, and G. Hirzinger, "Network representation and passivity of delayed teleoperation systems," in *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, Sept 2011, pp. 177–183.
- [20] J. Artigas, "Time domain passivity control for delayed teleoperation," Ph.D. dissertation, UPM - Universidad Politecnica Madrid, Madrid, Spain, 2014.
- [21] M. De Stefano, J. Artigas, W. Rackl, and A. Albu-Schaeffer, "Passivity of virtual free-floating dynamics rendered on robotic facilities," in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, 2015.