Loco-manipulation Tasks for Self-Relocatable Space Robots

Máximo A. Roa[‡], Ismael Rodríguez[‡], Anatoliy Huzynets[‡],
Hrishik Mishra[‡], Bernhard Brunner[‡], Gerhard Grunwald[‡]

I. Introduction

In-space assembly is a key technology for the future development of large infrastructures in space, from space stations and telescopes, to solar powered plants or planetary bases. Such structures are much larger than cargo volume in current launchers, therefore they must be sent as modules or separate pieces that are assembled in situ, typically using robotic manipulators. Different robotic manipulators are in service in space now, including for instance the Canadarm2[1], or the ERA[2]. These manipulators typically execute simple pick and place and manipulation tasks, and have limited relocation capabilities.

Recent projects within the European Union have explored new concepts for robotic loco-manipulation in space, i.e. robots that combine the capabilities of locomotion and object manipulation. Within the EU project MOSAR, a Walking Manipulator (WM) with 7 degrees of freedom (DoF) was developed to perform on-orbit assembly and reconfiguration of modular satellites using replaceable modules. The satellite is endowed with Standard Interconnects that provide power, mechanical and electrical connectivity to the modules and to the robot. The WM can self-relocate and can also perform manipulation tasks; a ground demonstrator was developed to show the capabilities under lab conditions (Fig. 1). However, a single arm has strong limitations in reachability and mobility. Within the project MIRROR, funded by ESA, a Multi-Arm Robot (MAR) is developed to perform assembly tasks for a large space telescope (Fig. 2). The robot consists of a torso and two 7-DoF arms attached to it. A ground demonstrator for this robot is currently under construction.

The efficient exploitation of the loco-manipulation abilities for such robots requires suitable planning tools to create plans for locomotion, grasping and manipulation of objects. The presence of contact and whole-body motions in the planning problem are two major common features of humanoids and space robots. Locomotion and manipulation are in fact conceptually similar problems, as they both deal with

This study was partially funded by the European Union H2020 Program under Grant No. 821966 (MOSAR: Modular Spacecraft Assembly and Reconfiguration) and by the European Space Agency (ESA) under contract No. 4000132220/20/NL/RA (MIRROR: Multi-arm Installation Robot for Readying ORUS and Reflectors).

[‡]All authors are with the Institute of Robotics and Mechatronics, German Aerospace Center (DLR), 82234 Wessling, Germany firstname.lastname@dlr.de

[1]https://www.asc-csa.gc.ca/eng/iss/canadarm2/default.asp

[2]https://www.esa.int/Science_Exploration/Human_and_Robotic_Exploration/International_Space_Station/European_Robotic_Arm

Fig. 1: Walking Manipulator (WM) for performing loco-manipulation tasks on a modular satellite.

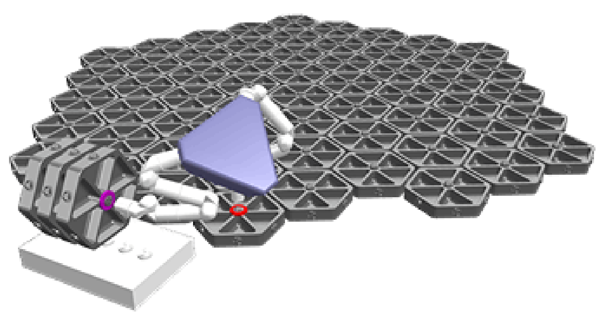

Fig. 2: Multi-arm robot (MAR) manipulating an hexagonal mirror tile during the assembly of a large space telescope.

underactuation and solve it through application of suitable contact forces [1]. In the humanoid robotics community, both problems have been considered together in whole-body, multicontact motion planning approaches [2], [3], [4].

II. Walking space robots

The Walking Manipulator is a serial robot with 7 degrees of freedom (DoF) and 8 links [5]. It has a total length of 1.6 m. Its kinematic chain is symmetric, with alternated roll and pitch revolute joints. The WM has Standard Interconnects (SIs) at both extremities, allowing the robot to interact with the environment: SIs are used both to relocate by stepping onto other objects that have standard interfaces, and to manipulate them. To latch two interfaces, they must be properly aligned along the axial direction of the mating device. The considered SIs are HOTDOCK [6], which present a 90-degrees axial symmetry with geometrical guidance to facilitate the latching, and provide an active and a passive

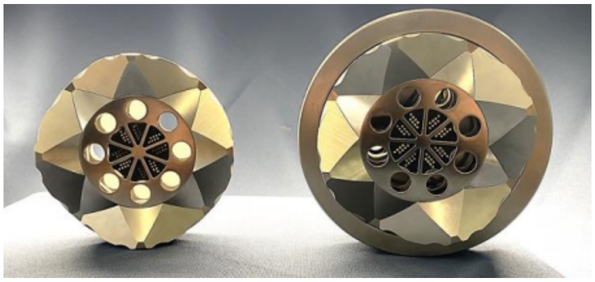

Fig. 3: HOTDOCK Standard Interconnect, with 90 deg axial symmetry. Left: passive SI. Right: active SI.

variant (Fig. 3). Consequently, four rotations around the axial direction are allowed, i.e., the mating is possible with four different relative orientations of 0, 90, 180, and 270 deg. This introduces different latching postures for creating a contact with an object or with the environment.

The Multi-Arm Robot consists of a central torso with 2 arms attached to it [7]. The torso is endowed with 3 SIs, one for each arm, and one additional SI that acts as additional contact point for the torso and provides a rotational DoF. An illumination and vision system is installed on the belly, and a solar array is installed on the back. Each arm of the MAR is similar to the WM, with 7 DoF and a length of 1.8 m. Its kinematic structure is symmetric, with alternate roll and pitch revolute joints. At both extremities, HOTDOCK devices are present to allow the robot to latch with any other SI. Both robots, the WM and the MAR, are torque controlled.

III. Loco-manipulation planning and control

The planner developed for these robots provides suitable trajectories for locomotion and manipulation tasks, either isolated or combined in the so-called loco-manipulation (whole-body) tasks. The input to the planner is specified as actions to be performed, such as “take an object and move it to a desired pose,” or “relocate to a new position in the environment,” provided in a suitable format that specifies relevant information including the initial and final robot pose, the desired object and its final location. The output contains the actuation commands, i.e. trajectories and torques at the joint level, as well as latching/unlatching commands for the end effectors. A valid output motion must fulfill the following constraints: 1) geometric: collision avoidance, 2) kinematic: joint range, velocity and acceleration limits, and 3) dynamic: joint torque limits and contact force/torque limits at coupling interfaces.

Our approach is based on a hybrid planner that combines a high-level layer that provides discrete transitions from one contact state to another (with contact states defined by a set of connections to SIs), and a low level layer that provides the joint trajectories to move from a contact state to the next one [8]. They are complemented with a validation layer that verifies the fulfillment of all the considered constraints. The low-level layer is based on an RRT motion planner (RRTConnect, [9]), which respects joint position and torque limits, to provide collision-free trajectories.

The high-level layer utilizes a graph representation of the locomotion problem to provide a trajectory that e.g. minimizes the overall displacement of the robot, while guaranteeing the execution of the intended task, either locomotion, manipulation or loco-manipulation. The nodes of the graph are the robot states and the edges are the transitions between states. The graph is constructed with a breadth-first search with priority queue reordering. Each evaluation considers the reachability of potential support locations from the current robot pose. The overall path in the graph is found via a cost function that combines the distance from the current node to the goal, and the cost for performing the transition, measured as the maximum torque saturation for the robot joints while performing the transition motion, where the (normalized) torque saturation for joint i is defined as $\hat{\tau}_i = \tau_i/\tau_{lim_i}$. The cost for the start node is null. The cost of node v in the graph is $cost_v = cost_{parent} + w_v c_v + w_e c_e$, i.e. a weighted sum of node and edge costs, cumulative along the graph traversal path. For each latching action, there are four possible latching locations provided by the HOTDOCK connector; the planner objective considers finding the optimal HOTDOCK rotation that minimizes the cost $cost_v$.

The planner provides valid solutions for locomotion (relocate), manipulation (pick-place) and loco-manipulation (relocate-pick-place) problems. The two-layer strategy allows a quick elimination of non-feasible plans, e.g., the high-level planner already discards plans that violate the constraints considered at that stage. When a plan results invalid, it is possible to (i) reduce velocities and accelerations in the trajectory planning, (ii) modify or re-plan the path layer, (iii) modify or re-plan the contact layer. In this regard, the planner provides not a single optimal plan but a set of candidate plans based on the graph representation.

The optimal path is then forwarded to a suitable control structure to perform the desired actions on the robot. The control framework for the execution of the plans on both torque-controlled robots is presented in Fig. 5, exemplified for the MAR; however, the same structure is also valid for the WM. The green box on the left includes all the plan execution layer. This includes the motion planner that provides suitable joint and approach trajectories for performing loco-manipulation tasks. The central blue box includes a Cartesian and a joint impedance controller. The control communicates with the real hardware via an EtherCAT interface.

IV. Implementation

For prototyping, a Co-simulation was developed, in which the complete system dynamics was simulated by an external engine, `CoppeliaSim`[3], and the control software was developed in `MATLAB/Simulink`. For the planning layer, the inverse kinematics and collision avoidance use Coppelia Kinematic Routines[4], and the planning algorithms reuse the Open Motion Planning Library[5].

[3] https://www.coppeliarobotics.com/
[4] https://github.com/CoppeliaRobotics/coppeliaKinematicsRoutines
[5] https://ompl.kavrakilab.org/

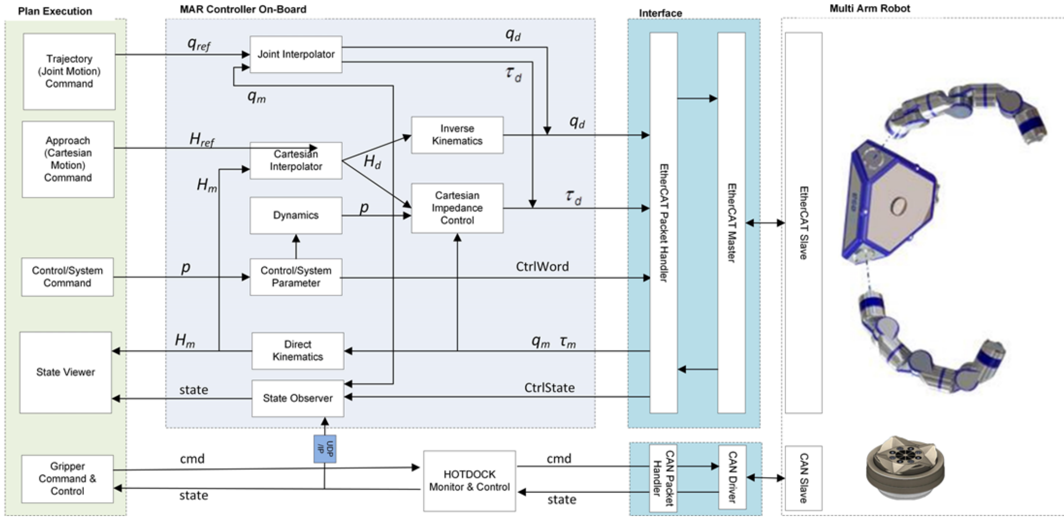

Fig. 4: Control framework for the MAR. Green box: plan execution layer. Light blue box: onboard robot controller. Darker blue box: EtherCAT interface to communicate with the hardware.

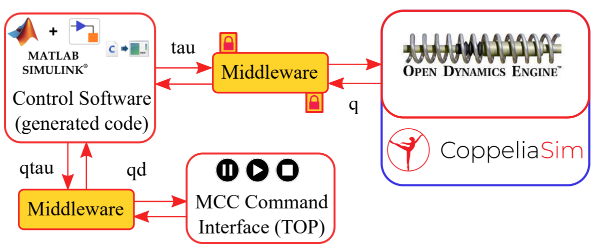

Fig. 5: Co-simulation framework for prototyping of the WM and MAR systems.

The holonomic constraints of the SIs were implemented using the suctionPad functionality in `CoppeliaSim`, which creates an overlap constraint, i.e., the two pertinent objects in close proximity will overlap their respective position/orientation to create dynamics loop closure constraints. Specifically, the suctionPad objects were added to the tiles in a way that reflected the kinematics of the SIs. Thus, in the event that the end-effector of the robot was in close proximity (below a specified threshold) of the suctionPad object, the former will be constrained to the supporting object. An overview of the developed framework is shown in Fig. XX. Note that the dynamics engine (`CoppeliaSim`) provides feedback of the plant state, and the control software possesses recursive kinematics and dynamics computation to track the Cartesian trajectory provided by the planner. Note that the Cartesian trajectory is provided for the free end-effector during each contact state, while the compatible joint trajectory is exploited to stabilize the redundancy of the WM. Furthermore, a Monitoring and Control Command (MCC) interface is used to aid a human supervisor to issue the offline-computed trajectory and high-level commands to the control software on the space robot. The developed Co-simulation framework will be used for validation of the proposed planning approach and the developed controller for loco-manipulation tasks.

The planner and co-simulation approach effectively provides suitable plans for the desired problems. As illustrative examples, Fig. 6 shows a locomotion task using the WM, and Fig. 7 shows a loco-manipulation task using the MAR. The planning times are in the order of minutes, which makes this approach suitable for offline planning. In any case, this is typical in space robotics, where all plans must be approved by a ground operator before being uploaded and executed on the robotic hardware in space.

References

[1] B. Henze, M. A. Roa, and C. Ott, "Passivity-based whole-body balancing for torque-controlled humanoid robots in multi-contact scenarios," *Int. J. of Robotics Research*, vol. 35, no. 12, pp. 1522–1543, 2016.

[2] A. Escande, A. Kheddar, and S. Miossec, "Planning contact points for humanoid robots," *Robotics and Autonomous Systems*, vol. 61, no. 5, pp. 428–442, 2013.

[3] A. Werner, B. Henze, D. Rodriguez, J. Gabaret, O. Porges, and M. A. Roa, "Multi-contact planning and control for a torque-controlled humanoid robot," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2016, pp. 5708–5715.

[4] E. Yoshida, F. Kanehiro, and J. P. Laumond, "Whole-body motion planning," in *Humanoid Robotics: A Reference*. Springer, 2019, pp. 1575–1599.

[5] M. Deremetz, P. Letier, G. Grunwald, M. A. Roa, B. Brunner, A. Ferrán, M. Szydelko, J. Gancet, and M. Ilzkovitz, "MOSAR-WM: Integration and tests results of a relocatable robotic arm demonstrator for future on-orbit servicing missions," in *Int. Astronautical Congress (IAC)*, 2021.

[6] P. Letier, T. Siedel, M. Deremetz, E. Pavlovskis, B. Lietaer, K. Nottensteiner, M. A. Roa, J. Sánchez, J. Corella, and J. Gancet, "HOTDOCK: Design and validation of a new generation of standard robotic interface for on-orbit servicing," in *Int. Astronautical Congress (IAC)*, 2020.

[7] M. Deremetz., G. Grunwald, F. Cavenago, M. A. Roa, M. DeStefano, H. Mishra, M. Reiner, S. Govindaraj, A. But, I. Sanz, J. Gancet, P. Letier, M. Ilzkovitz, L. Gerdes, and M. Zwick, "Concept of operations and preliminary design of a modular multi-arm robot using standard interconnects for on-orbit large assembly," in *Int. Astronautical Congress (IAC)*, 2021.

[8] A. Huzynets, I. Rodriguez, M. Deremetz, P. Letier, G. Grunwald, and M. A. Roa, "Motion planning for relocatable robots performing on-orbit locomotion and manipulation tasks," in *Int. Astronautical Congress (IAC)*, 2022.

[9] J. Kuffner and S. LaValle, "RRT-Connect: An efficient approach to single-query path planning," in *IEEE Int. Conf. on Robotics and Automation*, 2000, pp. 995–1001.

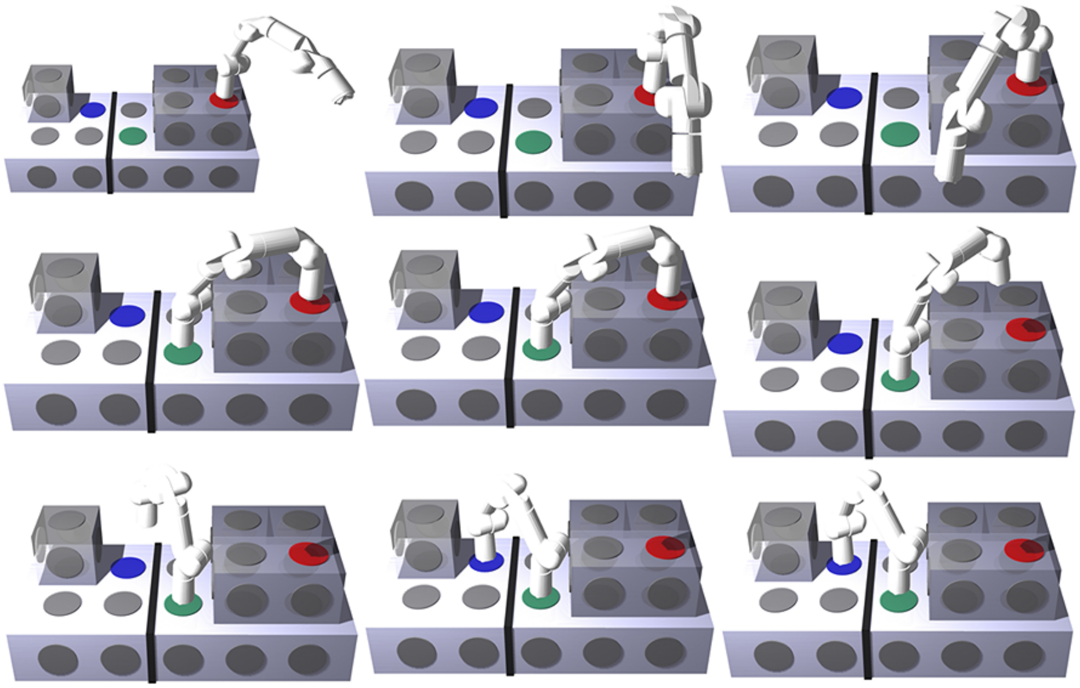

Fig. 6: WM executing a locomotion task, where the robot must move from an initial configuration, latched to the red connector, to a pose where both extremities must be latched, as defined by the blue and green connectors.

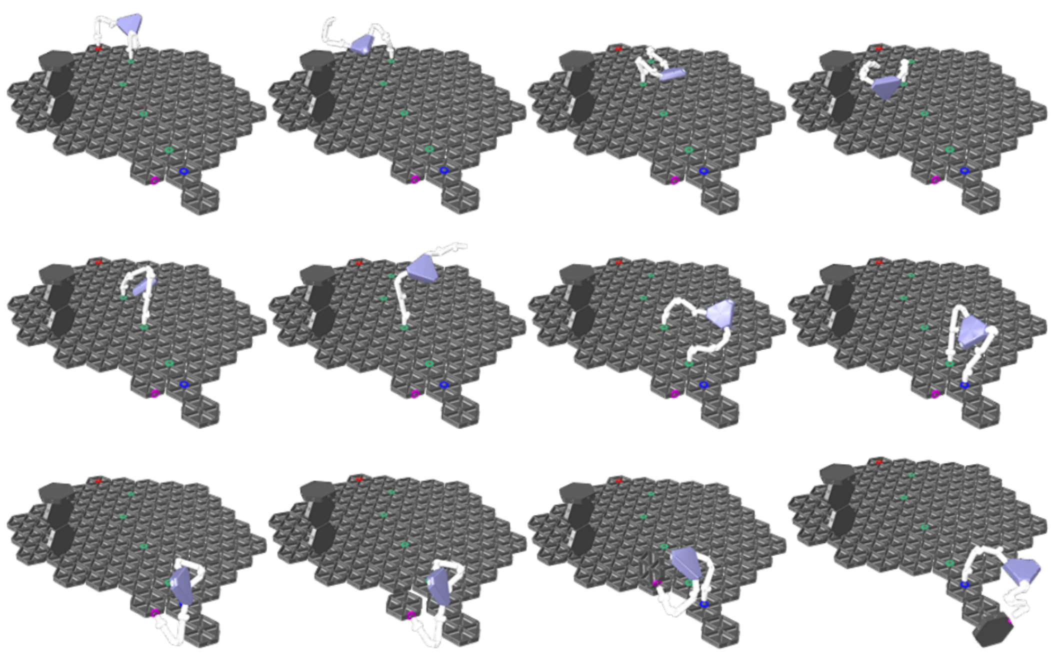

Fig. 7: MAR executing a loco-manipulation task, where the robot must change its initial pose (top left) in order to retrieve an hexagonal mirror tile and place it on a desired location (bottom right).