

Enhancing Space Manipulator Fault Tolerance for In-Orbit Servicing through Meta-Reinforcement Learning

Matteo D'AMBROSIO^{1,a,*} and Michèle LAVAGNA^{1,b}

¹ASTRA Lab, Department of Aerospace Science and Technology, Politecnico di Milano, Via la Masa 34, 20156 Milan, Italy

^amatteo.dambrosio@polimi.it, ^bmichelle.lavagna@polimi.it

Keywords: Orbital Robotics, Failure Management, Deep Reinforcement Learning

Abstract. This study develops a data-driven motion planning framework for a 7-DoF robotic manipulator to enhance fault tolerance against joint failures during the pre-capture phase of In-Orbit Servicing missions. The system is trained using meta-Reinforcement Learning (meta-RL), exposing the agent to a wide range of randomized scenarios, including stochastic joint-locking. This process achieves two main goals: it equips the agent with strong generalization capabilities for conditions beyond its training domain, and it bypasses traditional inverse kinematics by learning a direct mapping from sensor data to joint-space commands. The meta-learning framework is shown to significantly improve the system's ability to autonomously manage unforeseen failure events. In non-critical scenarios, the agent successfully learns to repurpose redundant joints to complete its task. However, the system cannot recover from failures that cause an unrecoverable reduction of the manipulator's workspace. The results demonstrate that this approach leads to more predictable autonomous behavior during failures and, most importantly, shows how some unexpected anomalies may be overcome by simply reframing them as an operational constraint, before critical operations begin.

Introduction

In-Orbit Servicing (IOS) activities represent a transformative shift in space operations, demanding highly autonomous robotic systems for critical close-proximity tasks like satellite capture, maintenance, and disposal under strict safety constraints [1, 2]. Increasing the autonomy of Guidance, Navigation, and Control (GNC) pipelines for these missions remains challenging, as traditional methods struggle to adapt in real-time to environmental variability or unexpected system anomalies. Deep Reinforcement Learning (DRL) has emerged as a promising approach to enhance these GNC architectures [3], offering the ability to learn complex dynamics through simulated interactions with the environment [4, 5, 6].

Overview. This study, carried out within the ASTRA Lab at Politecnico di Milano, presents a DRL-based guidance framework for a 7-DoF space manipulator mounted on a free-flying satellite. Its specific goal is to synchronize the end-effector with a fixed point on a tumbling, uncooperative target object (Fig. 1) during the pre-capture operations. To achieve this, the proposed framework employs a meta-learning approach [7, 8, 9]. The agent is trained across a wide range of randomized nominal and off-nominal scenarios, including varied initial poses, target tumbling rates, and potential joint failures. This work specifically explores the agent's ability to leverage the system's kinematic redundancy to maintain performance under degraded conditions. The proposed guidance agent receives information from ideal sensors in the task space and directly produces joint-space trajectory references, and demonstrates that a learning-driven manipulator guidance architecture can inherently adapt to failures on-the-go under specific conditions.



Space Robot Dynamics and GNC implementation

The robotic system is modeled through the open-source SPART toolkit [10], assuming a free-flying space robot model. The coupled equations of motion between base and manipulator are expressed in the compact form $\mathbf{H}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} = \boldsymbol{\tau}$ where $\mathbf{q}$ is the state variable (comprising the full state of the base and manipulator joints), and $\mathbf{H}$, $\mathbf{C}$ are the generalized inertia and convective inertia matrices. The implemented GNC system (Fig. 2) operates under the assumption of ideal navigation sensor outputs. These sensors feed information to the DRL-based guidance system, which in turn provides the desired manipulator trajectory.

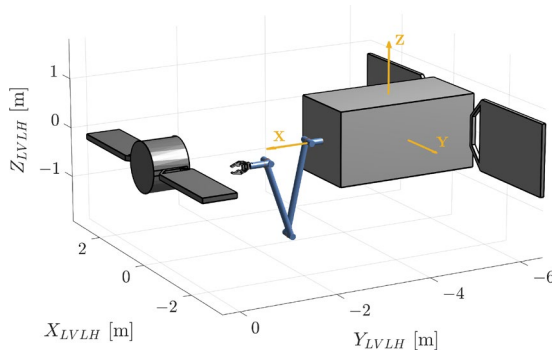


Figure 1: Satellite capture scenario.

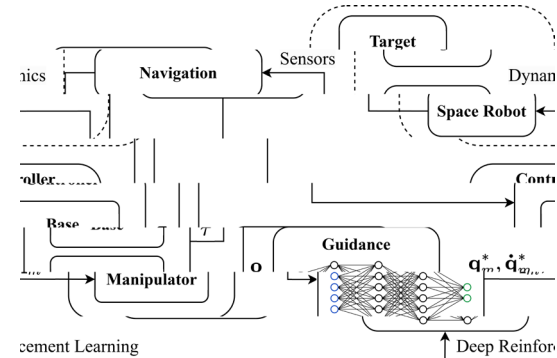


Figure 2: Space robot GNC architecture.

DRL agent guidance system. The core of the work regards the autonomous trajectory generation of the space manipulator, that is produced in real-time as each simulation progresses, based on the environmental perception of the agent at each instant. The inputs received by the agent correspond to the current state of the manipulator, and the errors between the current and target end-effector poses; the latter is ideally assumed as a known quantity for the scope of this work. The agent, implemented through a feedforward neural network, produces the desired joint angles and rates of the manipulator at each instant, bypassing the need for online optimization or inverse kinematics to produce a trajectory reference. A stochastic policy is applied during training to improve exploration of the action space, and consequently actions applied during training are sampled from a Gaussian distribution individually for each joint, encouraging the agent to avoid local minima during network optimization. The selected DRL algorithm is Proximal Policy Optimization [11], applying a reward function based on an Artificial Potential Field (Eq. 1), adapted from [12]. The value of U at each timestep k increases when the pose error of the end-effector decreases over time, such that the agent learns to guide the end-effector towards the desired pose. This goal-oriented reward is the core element of this formulation: the agent is motivated to move towards the desired goal, without directly prescribing how it should use the joint angles to achieve that. This is the crucial detail that makes the resulting trained policy adaptive, allowing it to autonomously decide how to exploit each of the joint angles, such that the final goal can be reached.

$$U_k = -\tilde{r} + \frac{10}{(1 + \tilde{r}_{ax})} + \frac{10}{(1 + \tilde{r}_{tx})} + \frac{10}{(1 + \tilde{\theta})} \quad (1)$$

Training and results in nominal environment

As defined through the reward function, the goal is for the agent to synchronize the end-effector with a target pose, that is taken fixed to a target spacecraft, that follows a free tumbling dynamics. In the nominal training phase, the tumbling only happens around a single axis, without any failures encountered. In these conditions, each simulation starts with a different distance between chaser and target satellites, initial manipulator configuration, tumbling rate of the target, as well as final

end-effector goal on the circular face of the target (Fig. 1). Fig. 3 shows that the network optimization process in the randomized environment, under nominal (non-failure) conditions, converges in approximately 8000 episodes. To compute an episode's outcome, the time that the end-effector can reliably stay below specific thresholds $\bar{r}$ and $\bar{\theta}$ is considered.

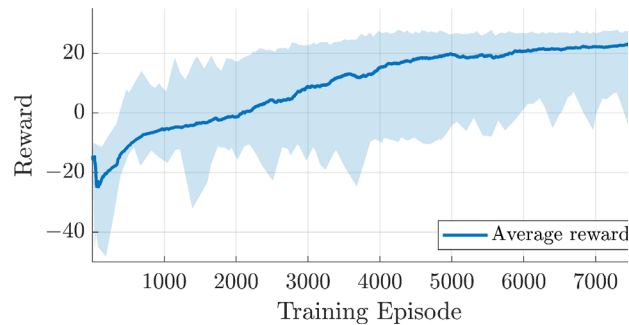


Figure 3: Training reward curve.

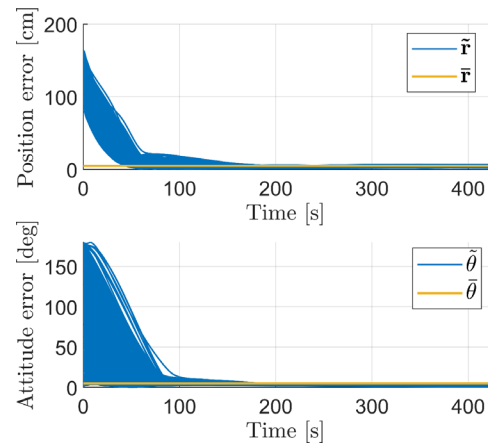


Figure 4: Montecarlo results.

A Monte Carlo analysis (Fig. 4) confirms the agent's ability to reliably synchronize its end-effector with a desired pose fixed to the tumbling target. In all simulations, the agent maintained a pose error below $\bar{r} = 1.5$ cm and $\bar{\theta} = 1^\circ$ for at least 30 consecutive seconds, regardless of the randomized end-effector goal. This performance constitutes a 100% success rate, establishing the baseline used for comparison against non-nominal simulation scenarios.

Agent behavior under joint failures

This section explores how the agent manages the kinematic redundancy of the system, to maintain performance in degraded conditions where one of the joints is assumed failed, and is locked from the beginning of the simulation. The specific joint that is locked is chosen randomly at the beginning of each simulation: the joints are assumed ordered from 1 to 7, from base of the manipulator to end-effector. To evaluate the potential of the meta-reinforcement learning approach, three testing scenarios are considered: i) the baseline agent from the previous section is directly tested under degraded conditions, ii) additional training is conducted starting from the baseline agent, exposing it to a failure of any of the 7 joints, and iii) additional training is conducted with the baseline agent, exposing it to failures of only the joints deemed non-critical in terms of workspace reduction, i.e. joints 1, 5, 6, 7, based on the results from ii). The results in terms of synchronization success percentage are reported in Tab. 1. Despite the large performance decrease with respect to the nominal behavior of 100% synchronization success rate, it is clear that the failure of some specific joints is more critical than others. Through a deeper examination of the results, the unsatisfactory performance of the agent found consequently to the locking of joints 2-4, is due to the drastic reduction of the workspace and dexterity of the manipulator, that is unrecoverable.

Table 1: Testing results with failed joints for cases i), ii), iii).

Joint	1	2	3	4	5	6	7
Case i)	64%	5%	11%	0%	46%	54%	52%
Case ii)	57%	1%	9%	0%	35%	68%	65%
Case iii)	69%	N/A	N/A	N/A	40%	77%	51%

Regarding the non-critical joints, focusing specifically on iii), the performance with respect to nominal conditions is heavily impacted, with a percent reduction in the order of 23%-60%. Despite this, further analysis of the failure locations for each test case demonstrate that the proposed method leads to some important insight on the agent's behavior under failed conditions. An example is provided for joint 1 (Fig. 5), overlaying the randomized end-effector targets, together with the success/failure of each episode, on top of the target's circular face. While only the results from joint 1 are shown, the same behavior is found for the others deemed non-critical (5-7). While the percentage of failures itself does not improve through additional training that subjects the agent to joint failures, it can be seen in the results from case iii) (Fig. 5) that the region where failures are encountered becomes more predictable; in cases i) and ii), this region almost overlaps the whole surface of the target, demonstrating that the agent fails unpredictably. The greater predictability of failures in test case iii) is the primary benefit of the meta-training approach, demonstrating that for each joint there exist relative configurations between the servicer spacecraft and target, such that the agent can reliably provide a successful synchronization of the end-effector to the desired pose after encountering a manipulator failure.

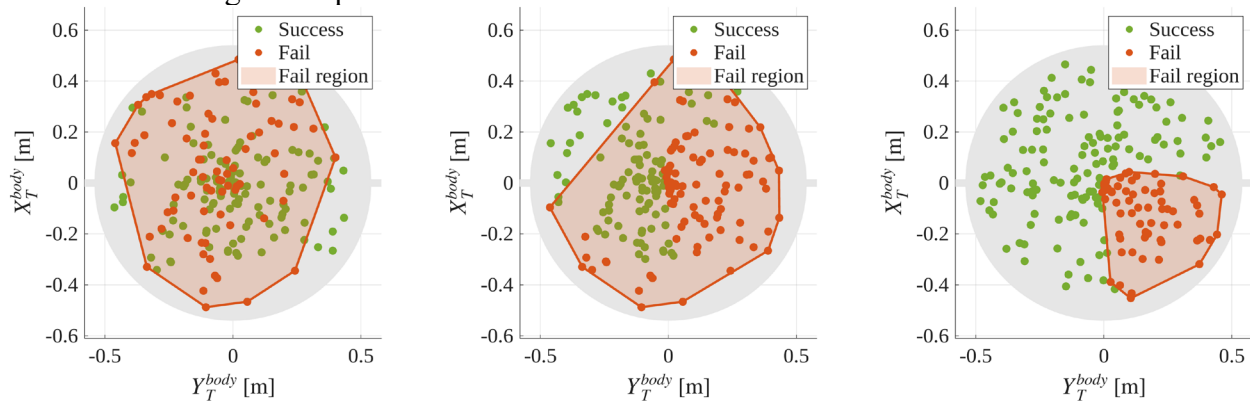


Figure 5: Joint 1 results for each testing case, i), ii), iii) (from left to right).

Overlaying the results from each of the non-critical joints (Figs. 6-7), it is clear that there are certain configurations between servicer spacecraft and target that allow the synchronization to be completed successfully, regardless of whether a non-critical joint is stuck in place at the beginning of the manipulator's motion. Indeed, by opportunely constraining the initial relative configuration between chaser and target, and maintaining it such that the grasping position falls outside of the failed-episode envelope, the 100% success rate of the agent can be fully retained even under failed conditions; this is achieved by autonomously repurposing the redundant joints. While a large reduction in success percent is obtained with respect to nominal conditions, the results show that a DRL-based manipulator guidance law trained through a meta-learning approach successfully improves the adaptability of the agent to unforeseen joint locks in the manipulator. This is a fundamental characteristic in safety-critical applications, and could be one of the main reasons to look into similar systems despite their novelty for space applications. By selecting an optimal relative configuration between chaser and target prior to the start of operations, together with an agent that can adapt to unforeseen failure events, the probability of a successful target capture can be increased even if a failure were to be encountered during operations.

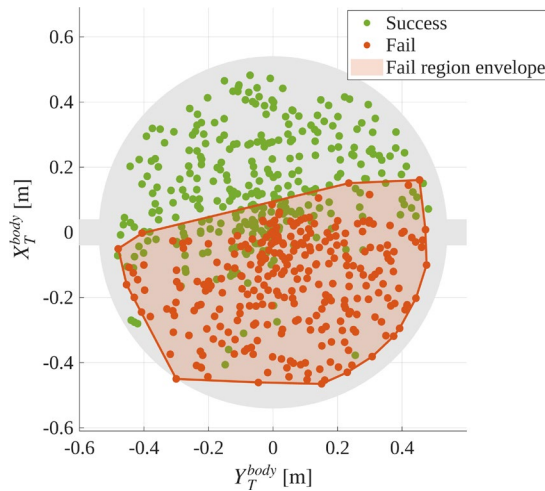


Figure 6: Fail region envelope.

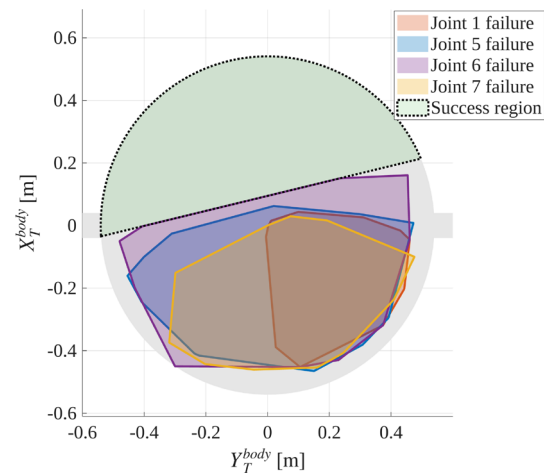


Figure 7: Fail regions overlayed.

Conclusion

This study demonstrates that meta-DRL approaches can significantly boost fault-tolerance capabilities of the space robot GNC system, that can be achieved by exposing the agent to targeted failures during the training phase. While overall performance still lags behind nominal conditions, the main outcome of the approach is the agent's ability to exploit the manipulator's kinematic redundancy to achieve its objective even under failed joint conditions, together with a much greater predictability of the agent's behavior during critical operations. Having similar knowledge of the agent's predictability consequently to failures is operationally invaluable, as it turns critical anomaly events into operational constraints.

Acknowledgments

The authors wish to express their gratitude to the ESA Academy Student Sponsorship Programme for supporting this work for presentation at the CEAS-AIDAA 2025 conference.

References

- [1] A. Flores-Abad, O. Ma, K. Pham, S. Ulrich, A review of space robotics technologies for on-orbit servicing, *Prog. Aerosp. Sci.* 68 (2014) 1-26. <https://doi.org/10.1016/j.paerosci.2014.03.002>
- [2] B.M. Moghaddam, R. Chhabra, On the guidance, navigation and control of in-orbit space robotic missions: A survey and prospective vision, (2021). <https://doi.org/10.1016/j.actaastro.2021.03.029>
- [3] A. Al Ali, Z.H. Zhu, Reinforcement learning for path planning of free-floating space robotic manipulator with collision avoidance and observation noise, *Front. Control Eng.* 5 (2024). <https://doi.org/10.3389/fcteg.2024.1394668>
- [4] S. Wang, X. Zheng, Y. Cao, T. Zhang, A multi-target trajectory planning of a 6-DoF free-floating space robot via reinforcement learning, in: *Proc. IEEE*, (2021) 3724-3730. <https://doi.org/10.1109/IROS51168.2021.9636681>
- [5] J. Blaise, M.C.F. Bazzocchi, Space manipulator collision avoidance using a deep reinforcement learning control, *Aerospace* 10 (2023). <https://doi.org/10.3390/aerospace10090778>
- [6] M. D'Ambrosio, L. Capra, A. Brandonisio, S. Silvestrini, M. Lavagna, Redundant space manipulator autonomous guidance for in-orbit servicing via deep reinforcement learning, *Aerospace* 11 (2024) 341. <https://doi.org/10.3390/aerospace11050341>

- [7] B. Gaudet, R. Linares, R. Furfaro, Adaptive guidance and integrated navigation with reinforcement meta-learning, *Acta Astronaut.* 169 (2020) 180-190.
<https://doi.org/10.1016/j.actaastro.2020.01.007>
- [8] B. Gaudet, R. Linares, R. Furfaro, Terminal adaptive guidance via reinforcement meta-learning: Applications to autonomous asteroid close-proximity operations, *Acta Astronaut.* 171 (2020) 1-13. <https://doi.org/10.1016/j.actaastro.2020.02.036>
- [9] M. D'Ambrosio, S. Silvestrini, M. Lavagna, Conditioned sequence models for warm-starting sequential convex trajectory optimization in space robots, *Aerospace* 13 (2026) 137.
<https://doi.org/10.3390/aerospace13020137>. <https://doi.org/10.3390/aerospace13020137>
- [10] J. Virgili-Llop, J.V. Drew II, M. Romano, SPART spacecraft robotics toolkit: an open-source simulator for spacecraft robotic arm dynamic modeling and control, (2016).
- [11] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, O. Klimov, Proximal policy optimization algorithms, *arXiv:1707.06347* (2017).
- [12] Y. Li, D. Li, W. Zhu, J. Sun, X. Zhang, S. Li, Constrained motion planning of 7-DoF space manipulator via deep reinforcement learning combined with artificial potential field, *Aerospace* 9 (2022). <https://doi.org/10.3390/aerospace9030163>