Deep Reinforcement Learning for Reactive IOS Space Manipulator Operations

Matteo D'Ambrosio, **Lorenzo Capra**, Michèle Lavagna

17th Symposium on Advanced Space Technologies in Robotics and Automation

18-20 October 2023

POLITECNICO MILANO 1863

DIPARTIMENTO DI SCIENZE E TECNOLOGIE AEROSPAZIALI

Advanced Space Technologies for Robotics and Astrodynamics

Outline

- Research motivation
- Research objectives
- Motion synchronization
- Reinforcement Learning guidance
- Simulation results
- Conclusions

Research motivation

In orbit servicing (**IOS**) activities are a way forward sustainability for future space assets and leading space agencies are focusing more and more on their exploitation and reusability.

Motivation:
- **robotic** manipulation and capture as key enabler for IOS capabilities
- **enhanced autonomy** and **adaptivity** as crucial mission improvement.

- Robotics
  - Space manipulator *kinematics*
  - Space manipulator *dynamics*
  - Space manipulator *guidance*
  - Space manipulator *control*
- Autonomy
  - *Adaptive* guidance & control
  - *Reinforcement Learning (RL)*

Research objectives

- Development of the **motion synchronization** scenario
- **Guidance** and **control RL** implementation
- **Training** and **testing** in nominal conditions
- **Robustness** analysis

Motion synchronization

Close-range approach → **Motion synchronization** → Grappling & Capture → Stabilization & Detumbling

<u>Objectives:</u>

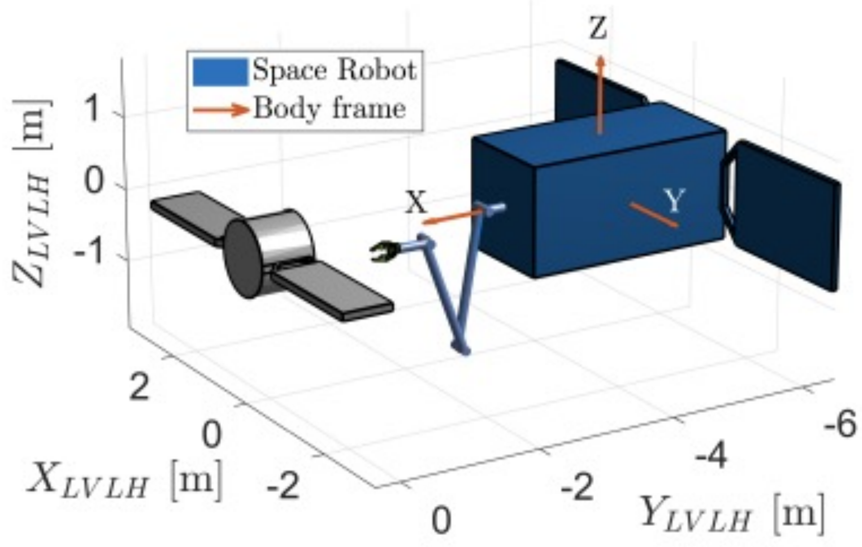
- **Adjust** the chaser end effector **pose** so that the grasping point is **stationary: pose tracking**

$$[\, x_{ee} \,, y_{ee} \,, z_{ee} \,] = [\, x_{grasp} \,, y_{grasp} \,, z_{grasp} \,] - [\, 0.4 \,, 0 \,, 0 \,]$$

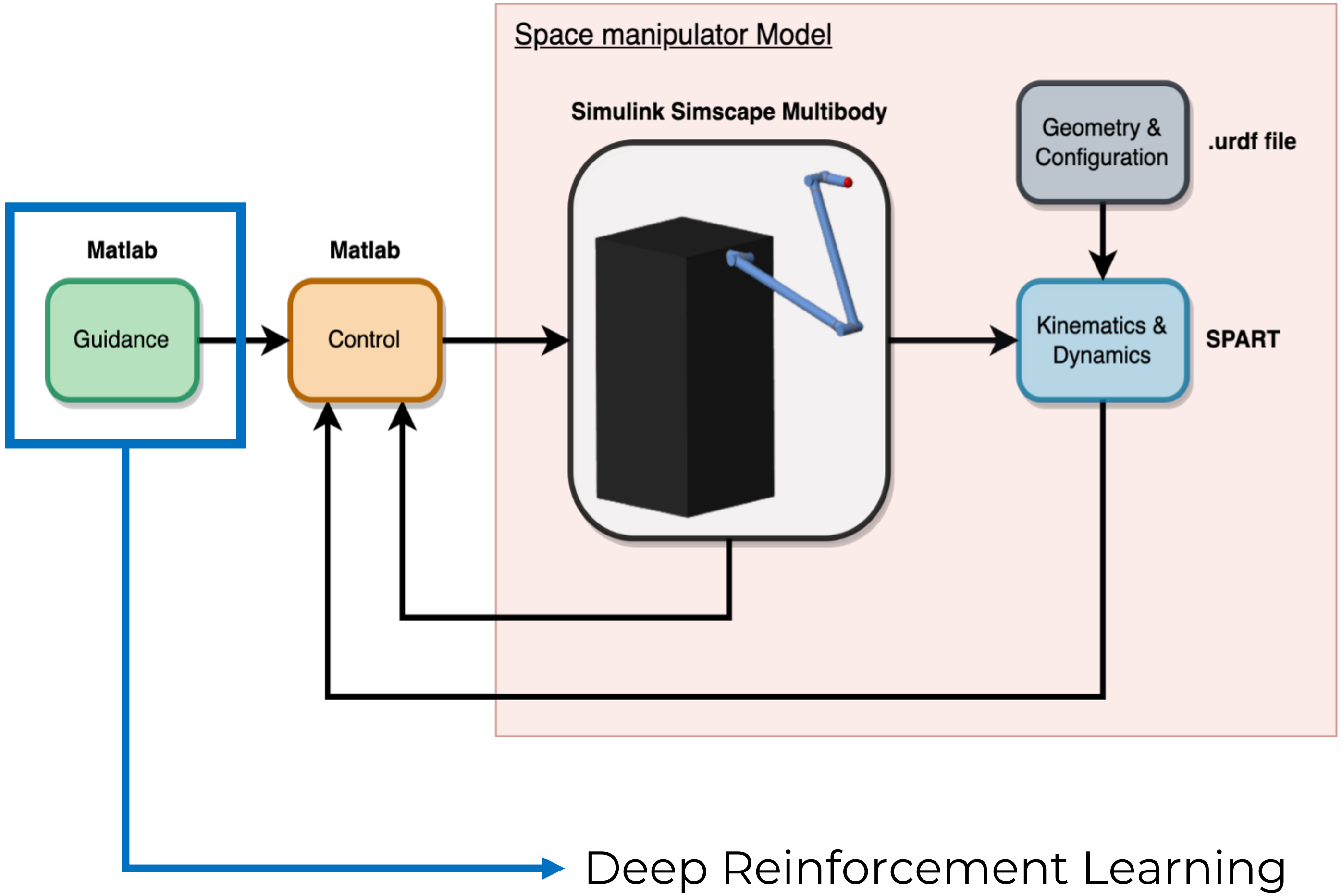
- **Thresholds** set at:

$$e_{p_{ax}} < 5cm \qquad e_{p_{tx}} < 5cm \qquad e_{\alpha} < 5°$$

- **Robustness** against parametric and navigation uncertainty
- Successful if **consecutive timesteps** inside desired region



Simulator architecture

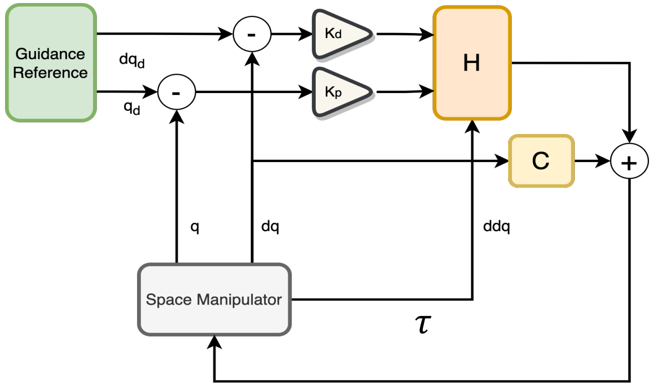


Deep Reinforcement Learning

- 6-dof spacecraft equipped with 7-dof robotic arm
- A model-based feedback linearization control

$$H(q)\ddot{q} + C(q,\dot{q})\dot{q} = \tau$$

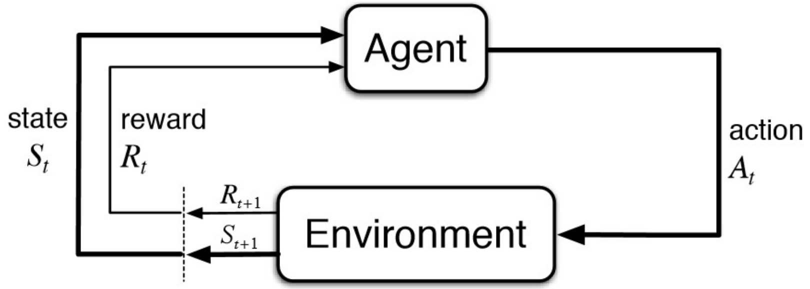
$$\tau = H(q)\left(\ddot{q} + K_d\dot{e}_q + K_p e_q\right) + C(q,\dot{q})\dot{q}$$

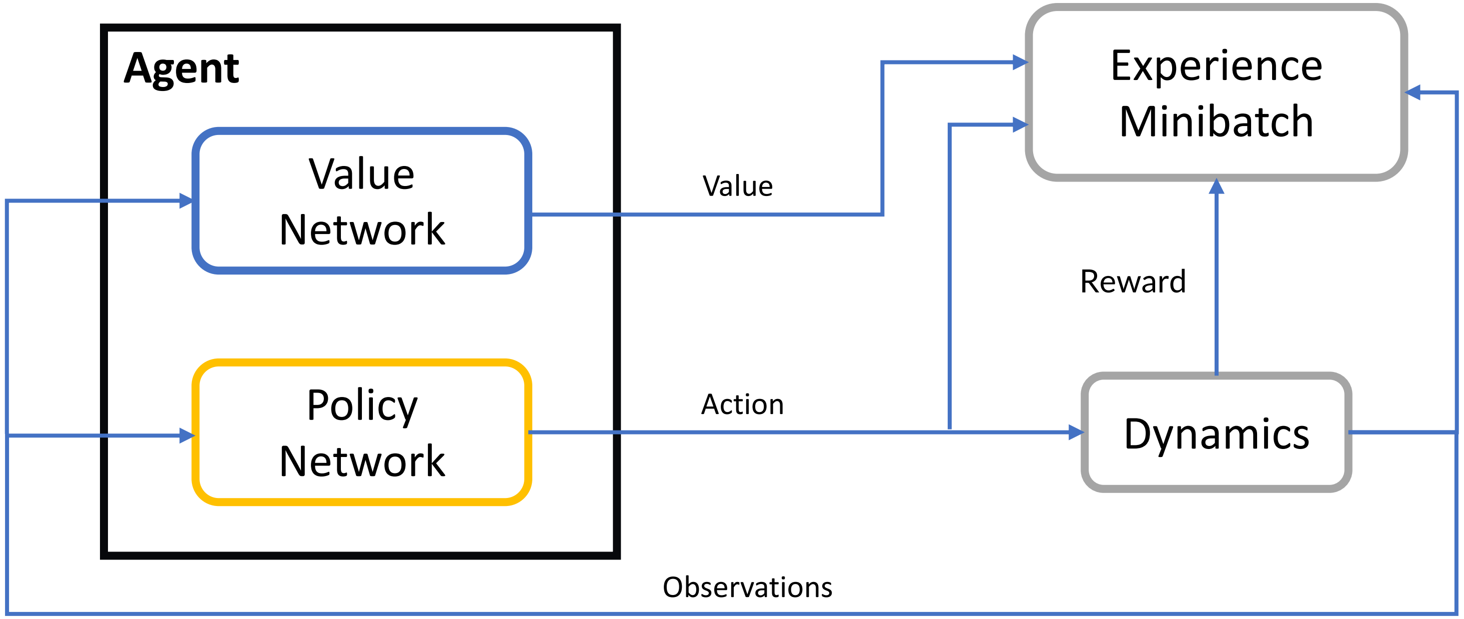


Reinforcement Learning guidance

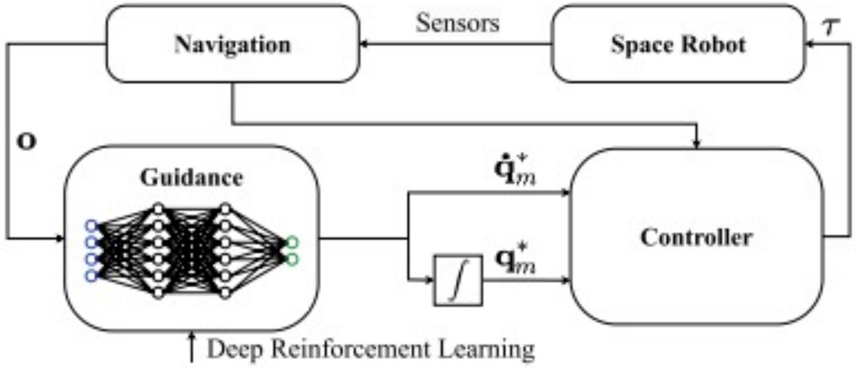
An agent improving its decision-making by interacting and receiving feedbacks from the surrounding environment.

Proximal Policy Optimization (PPO) selected as RL agent:





Reinforcement Learning guidance



Policy neural network:

- Joints angles and angular velocity
- End-effector position and orientation error
- End-effector velocity and angular rate error
- Desired manipulator joint rates

Reward function:

Artificial Potential Field (APF) ⟶ $U = -e_p + \frac{10}{1+e_{pax}} + \frac{10}{1+e_{ptx}} + \frac{10}{1+e_\alpha}$

Reward signal computed as:

$r = U_k - U_{k-1} + \delta$ with $\delta = \begin{cases} 0.01 & \text{if errors} < \text{threshold} \\ 0 & \text{if errors} > \text{threshold} \end{cases}$

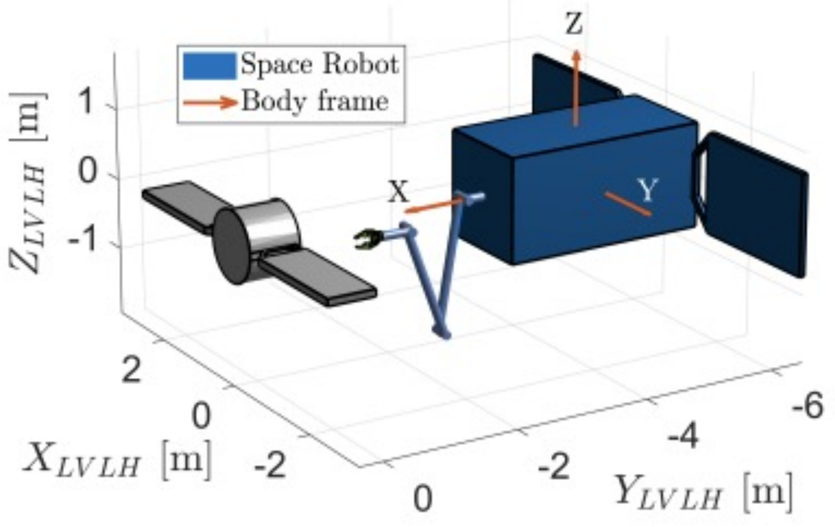
Simulation results

Agent training

Simulation set-up:

- Target tumbling around its major inertia axis with random angular velocity ⟶ $\omega_{trg} \in [-3, 3]$ deg/s
- **Spacecraft** kept **synchronized** with the target to minimize relative motion
- **Initial conditions** of the space manipulator are **randomized**

  Spacecraft position $^{+}/_{-}\ 25cm$ Manipulator joints $^{+}/_{-}\ 15°$
- **Grasping point randomized** on the target face
- Simulation stopped when $\mathbf{det(GJM)} \sim \mathbf{0}$, via SVD

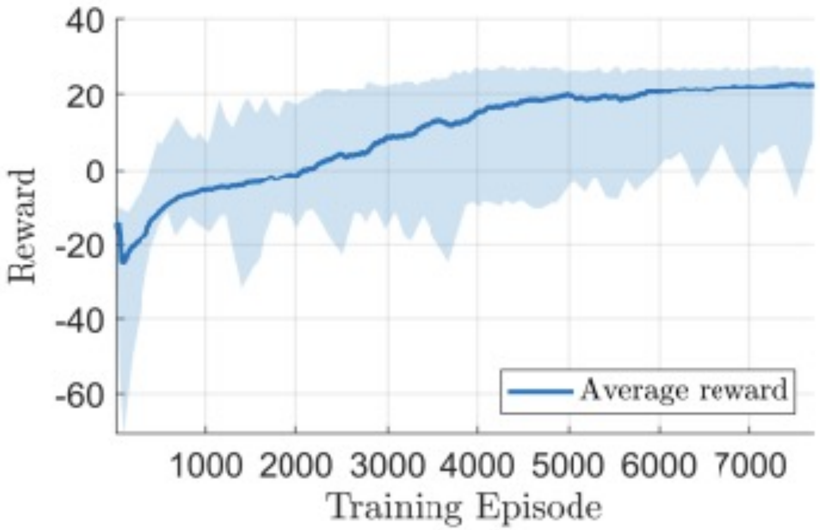


Simulation results

Agent training

Neural Network specifics:

- 3 hidden layers of 300 neurons
- *tanh* activation function
- Learning rate = $1e^{-5}$
- Output sampled from gaussian distribution $\mathcal{N}(\mu, \sigma)$



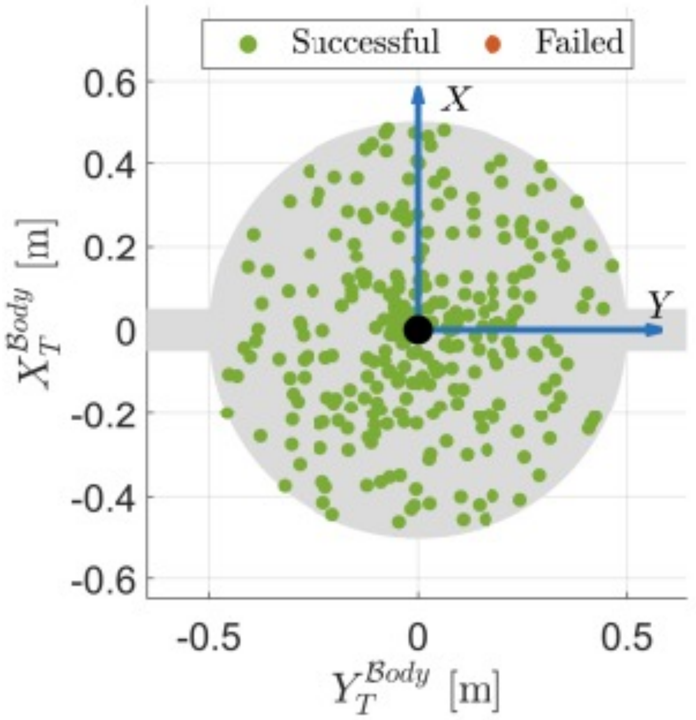
→ Training performed over 7500 episodes of 420s

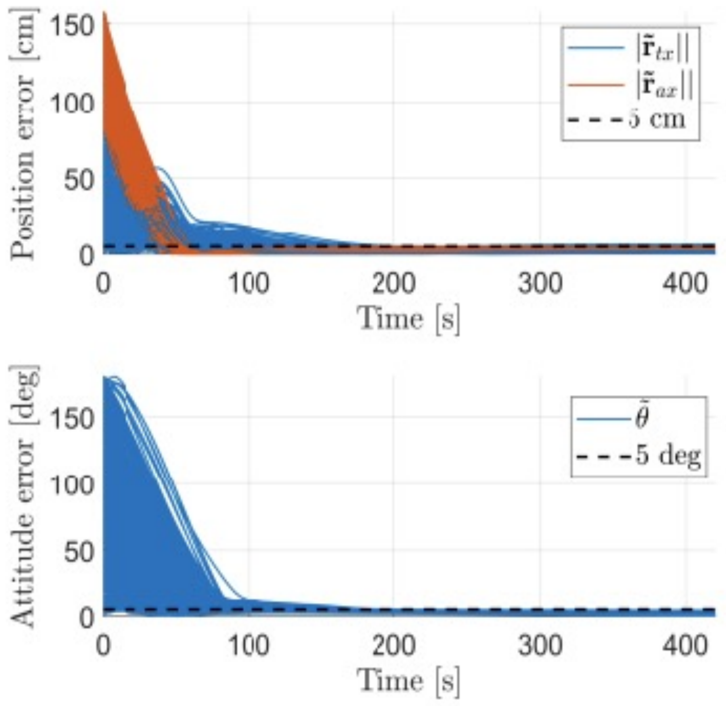
→ Monotonic increase of the average reward

Simulation results

Agent testing

Montecarlo analysis set-up to verify the performance over a new set of episodes with randomized conditions.





- 100% success rate
- Convergence reached in ¼ of the simulation time on average
- End-effector remains inside thresholds after convergence

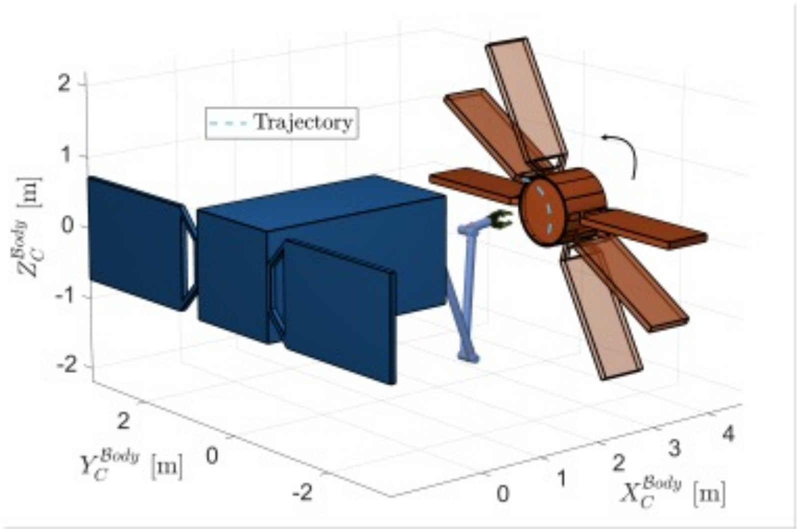
Simulation results

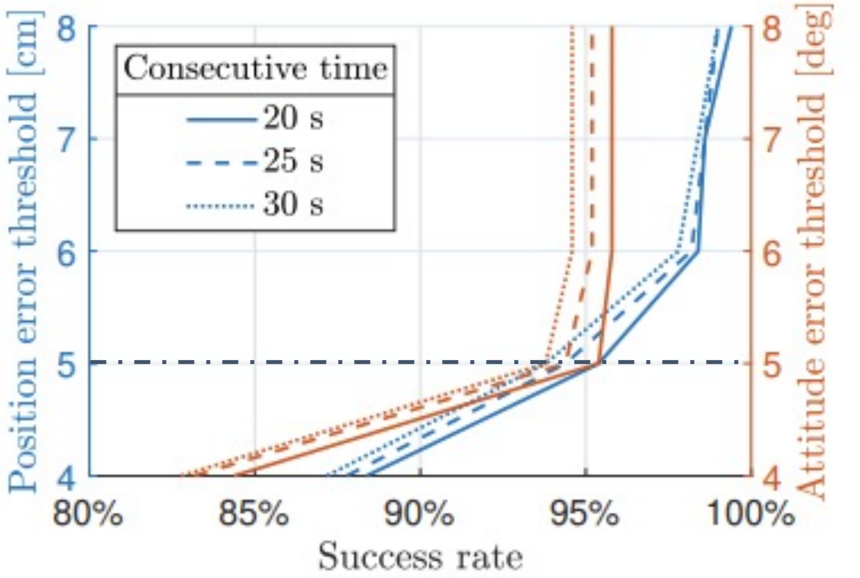
Robustness analysis

Errors in the spin rate synchronization around the target's rotation axis are added:

$$\omega_{err} \in [-0.5, 0.5] \text{ deg/s}$$

The grasping point is no more static with respect to the space manipulator, but the end-effector has to track a moving point.

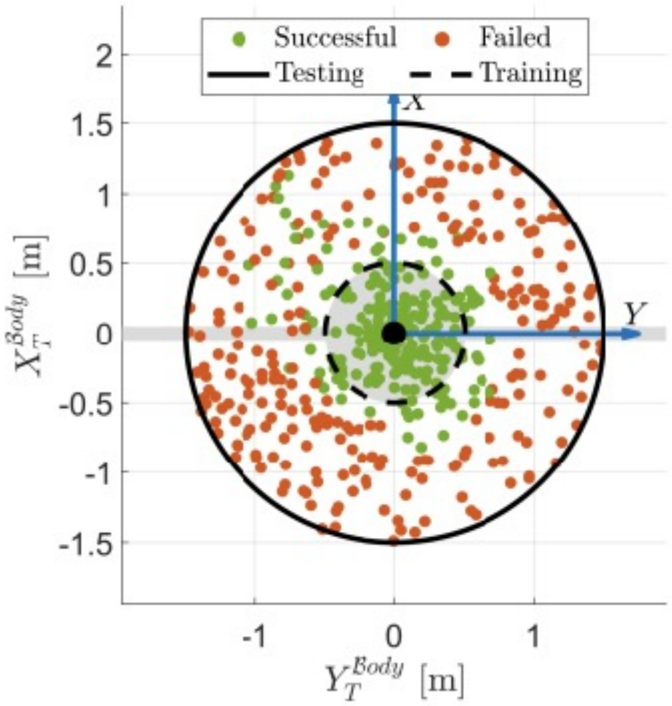


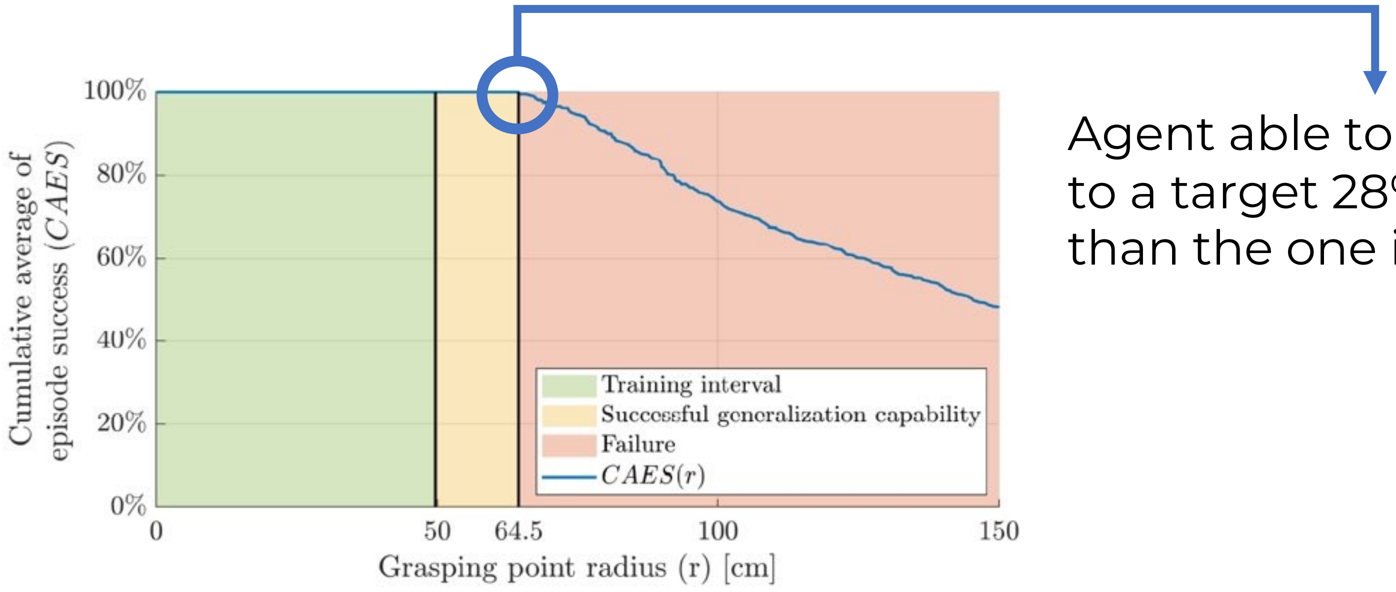


Simulation results

Robustness analysis

Parametric analysis on the dimension of the target is performed to verify generalizing capabilities of the RL agent. Radius of the target object now sampled from: $r_{trg} \in [50, 150]$ cm





Agent able to generalize to a target 28% larger than the one in training.

Conclusions

Deep Reinforcement Learning is successfully deployed to solve for the guidance of a space manipulator during the phase of motion synchronization of a potential IOS mission:

- PPO is employed as DRL agent
- Training is performed on the nominal scenario
- Montecarlo analysis is carried out to verify performances
- Robustness of the method is checked against synchronization errors and larger targets

Future developments:

- Collision avoidance can be added by tuning the reward function
- Full tumbling of the target must be investigated, adapting the synchronization scenario

Deep Reinforcement Learning for Reactive IOS Space Manipulator Operations

Matteo D’Ambrosio, **Lorenzo Capra**, Michèle Lavagna

17th Symposium on Advanced Space Technologies in Robotics and Automation

18-20 October 2023

POLITECNICO MILANO 1863

DIPARTIMENTO DI SCIENZE E TECNOLOGIE AEROSPAZIALI

Advanced Space Technologies for Robotics and Astrodynamics