Autonomous Inspection of a Tumbling Satellite in Low Earth Orbit

Afrah Ghedira and Hanspeter Schaub
University of Colorado Boulder

ABSTRACT

Space object inspection is a critical capability for rendezvous and proximity operations, enabling applications such as damage assessment, docking port localization, and space asset characterization. However, inspection is inherently constrained by factors including illumination conditions, viewing geometry, relative distance, and collision avoidance requirements. These challenges are further compounded when the target is tumbling, as the accessibility of inspection regions depends not only on orbital geometry but also on the target's attitude dynamics. This paper presents a deep reinforcement learning approach for the autonomous inspection of spinning resident space objects, considering both general tumbling motion and stable rotation about the orbit-normal axis with constant spin rate. The problem is formulated as a semi-Markov decision process for sequential decision making. By leveraging natural orbital dynamics, the agent executes impulsive maneuvers followed by variable-duration coasting phases to efficiently complete the constrained inspection task under time and fuel budgets. Unlike conventional planning methods that optimize a fixed observation sequence offline, the proposed framework continuously adapts its decisions online based on the observed environment. The learned policies achieve safe and efficient inspection across a wide range of low Earth orbit scenarios with varying target spin rates and lighting conditions. To assess performance, reinforcement learning policies spanning different time-versus-fuel optimization objectives are compared with baseline heuristic strategies, demonstrating the benefits of adaptive decision making for autonomous on-orbit inspection.

1. INTRODUCTION

Orbital rendezvous and proximity operations have become increasingly active areas of research due to growing interest in applications such as active debris removal, on-orbit servicing, refueling, upgrading, and in-space assembly. Among these applications, inspection plays a critical and often prerequisite role. Prior to docking, servicing, or debris capture, it is essential to assess structural integrity, identify damage, locate docking interfaces, and characterize the target's geometry and rotational state. Inspection is also important for the characterization of unknown or non-cooperative space assets. Consequently, spacecraft inspection has been studied from several perspectives, ranging from vision-based relative navigation [1–3], mapping and target-motion estimation [4, 5], and autonomous guidance and trajectory optimization [6–8], to reinforcement-learning-based decision-making [9, 10]. Despite these advances, autonomous inspection of a resident space object (RSO) remains a challenging sequential decision-making problem. The inspector spacecraft must satisfy strict geometric and operational constraints, including limits on relative distance for imaging while also maintaining a safety distance, camera incidence angle constraints relative to RSO surface normals, and illumination requirements dictated by the Sun direction. At the same time, the mission must minimize resource use, particularly fuel and time. These constraints can become more complex when the target is spinning or tumbling, as the visibility and accessibility of inspection points depend on both orbital motion and the target's rotational dynamics.

These challenges motivate the need for autonomous inspection strategies capable of adapting to uncertain and evolving mission environments without ground intervention. Reinforcement learning (RL) has recently emerged as a promising framework for spacecraft autonomy and sequential decision-making under uncertainty. Previous work has demonstrated its effectiveness for a variety of space applications, including Earth observation scheduling [11–13], rendezvous and proximity operations [14–17], planetary landing [18, 19], and small-body exploration [20–23]. These studies demonstrate the ability of RL to autonomously generate guidance and observation strategies, while adapting to uncertain environments and changing mission conditions.

For autonomous spacecraft inspection, early work focused on the perception, estimation, and control capabilities required to operate around unknown or tumbling targets. References 2 and 3 demonstrate autonomous vision-based navigation and control for circumnavigating non-cooperative spacecraft using prescribed inspection trajectories, while References 4 and 5 extend these efforts to simultaneous localization, mapping, and inertial-property estimation. Although these methods enable target characterization, they do not directly optimize inspection trajectories according to mission objectives or inspection progress. Autonomous inspection guidance has also been addressed through online planning and optimization. Reference 7 proposes a receding-horizon guidance strategy for autonomous inspection of unknown RSOs, while Reference 8 formulates inspection of a stochastically tumbling target as a constrained stochastic optimization problem accounting for attitude uncertainty, sensing constraints, and fuel consumption. Although these methods adapt trajectories online, they require repeatedly solving computational optimization problems during mission execution. More recently, RL has emerged as an alternative for autonomous inspection guidance. References 24 and 25 investigate coordinated and information-driven inspection strategies for multiple spacecraft, Reference 26 applies deep reinforcement learning to multi-agent satellite inspection, and Reference 27 proposes a hierarchical deep reinforcement learning framework for decentralized multi-agent inspection of tumbling targets. While these methods demonstrate the potential of RL for autonomous inspection, they either consider multiple inspector spacecraft or restrict the learned policy to transitions among predefined viewpoints.

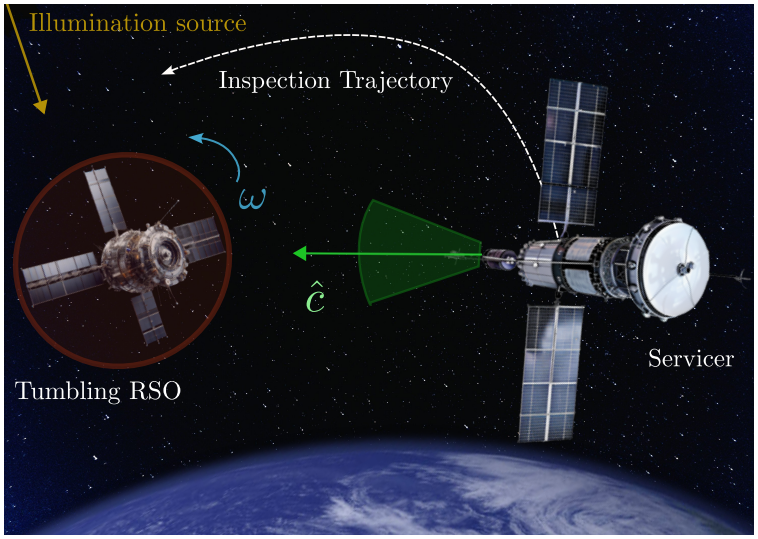


Fig. 1: Conceptual illustration of spinning target inspection

The present work builds upon a recently proposed reinforcement-learning framework for autonomous RSO inspection [9] that learns a continuous guidance policy for a single inspector that directly selects impulsive maneuvers and variable-duration coasting intervals. Reference 9 addresses the problem of safe and autonomous inspection for nadir-pointing satellites on circular Low Earth Orbits (LEO) using impulsive maneuvers by formulating the problem as a semi-Partially Observable Markov Decision Process (sPOMDP). Reference 28 subsequently extends to RSOs on eccentric orbits with multiple inspecting servicers while Ref. 29 benchmarks against probabilistic roadmap solutions. These studies demonstrate the feasibility of robust, fuel-efficient autonomous inspection with formal safety guarantees. However, these prior studies assume targets with Hill-frame-fixed attitudes. The present work extends the earlier training framework to RSOs undergoing either stable rotation about the orbit normal or general tumbling (Figure 1), introducing inspection constraints that depend jointly on orbital geometry and target attitude dynamics. The resulting policies are trained to remain effective across a range of spin rates and rotation axes, increasing robustness for realistic inspection scenarios. As in Reference 9, collision-avoidance safety can be enforced through an optimization-based shielding mechanism that filters actions predicted to violate a prescribed keep-out zone. Using Clohessy-Wiltshire (CWH) relative dynamics [30], the shield replaces unsafe actions with the nearest admissible alternative, providing collision-avoidance guarantees while minimally altering the learned policy.

The remainder of the paper is organized as follows. Section 2 reviews the inspection problem formulation and summarizes the sPOMDP framework adopted from Reference 9, highlighting the modifications introduced for spinning and

tumbling targets. Section 3 analyzes policy performance for both stable-spin and general-tumble scenarios, including a benchmark study evaluating robustness across diverse orbital conditions and target dynamics, together with comparisons against baseline inspection heuristics. Finally, Section 4 concludes the paper and discusses future research directions.

2. PROBLEM FORMULATION REVIEW AND RL SOLUTION APPROACH

The reinforcement learning framework adopted in this work builds upon the formulation presented in Reference 9, with modifications to accommodate tumbling target dynamics and revised mission initialization conditions. This section reviews the original problem formulation and highlights the changes introduced for the present study.

2.1 Problem Statement

The objective of this work is to maximize the fraction of the RSO surface that is successfully inspected while satisfying illumination constraints. The servicer accomplishes this through impulsive maneuvers that modify its relative trajectory, enabling observations from multiple viewpoints while avoiding collision.

2.1.1 RSO Model

The RSO is modeled as a sphere with a radius of 2 m whose surface is discretized into 100 uniformly distributed inspection points. Each point is associated with an outward-pointing surface normal vector, $\hat{\boldsymbol{n}}_i$. The RSO is assumed to be in a circular LEO at an altitude between 500 and 1000 km, with orbital dynamics including J_2 perturbations and atmospheric drag. To expose the learning agent to diverse illumination conditions, the orbital inclination i and right ascension of the ascending node (RAAN) Ω are randomized during training. Unlike the nadir-pointing RSO considered in Reference 9, the target is modeled either as a stable spinner or as a generally tumbling body. In the stable-spin scenario, the RSO rotates about the orbit-normal axis at a constant angular rate $\omega_{B/H}$, where B and H denote the body and Hill [31] frames, respectively. To improve robustness, the spin rate is randomized during training and the learned policy is evaluated over a range of spin rates. For the general tumble case, both the initial spin axis and angular velocity are randomized. The randomized initial spin axis results in a general tumbling motion without a fixed rotation axis.

2.1.2 Servicer Spacecraft Model

The servicer spacecraft is equipped with a body-fixed camera whose boresight is represented by the unit vector $\hat{\boldsymbol{c}}$. The camera is assumed to remain continuously pointed toward the RSO using attitude control implemented through reaction wheels. A surface point is considered successfully imaged only if the servicer is within the inspection distance of $r_{\text{inspect}} = 250$ m and the camera incidence angle satisfies:

$$\theta_i = \angle(\hat{\boldsymbol{n}}_i, -\hat{\boldsymbol{c}}) < \theta_{\max} = 30^\circ \tag{1}$$

The servicer orbit is propagated using the same perturbation models as the RSO, including J_2 effects and atmospheric drag. Relative motion is described by the perturbed CWH equations [30]. The servicer can perform impulsive maneuvers in arbitrary directions with a maximum impulse magnitude of $\Delta v_{\max} = 1$ m/s, while the total mission Δv budget is limited to 10 m/s. Reference 9 initialized the servicer at a random location within a 1 km radius of the RSO, causing a significant fraction of the total Δv to be expended during the initial approach to the target. Because this cost depends primarily on the random initial separation rather than the inspection strategy, the present work instead initializes the servicer in a lead-follower formation 200 m behind the RSO, placing it immediately within the allowable imaging range. This enables a more consistent comparison with baseline methods by isolating the maneuvering cost associated with inspection.

2.1.3 Illumination Constraints

In addition to satisfying the camera viewing constraints, a surface point must be sufficiently illuminated by the Sun. A point is considered illuminated when the angle between its surface normal and the Sun direction, $\hat{\boldsymbol{s}}$, satisfies:

$$\phi_i = \angle(\hat{\boldsymbol{n}}_i, \hat{\boldsymbol{s}}) < \phi_{\max} = 60^\circ \tag{2}$$

2.1.4 Objective Function

The inspection objective is to maximize the fraction of the target surface that is successfully inspected while minimizing fuel consumption, subject to collision avoidance constraints,

$$\begin{aligned} \max \quad & \frac{|\mathscr{I}|}{N} - \alpha \Delta v \\ \text{s.t.} \quad & \|\mathbf{r}_{DC}\| > R_D + R_C, \quad \forall t. \end{aligned} \tag{3}$$

Here, $\mathscr{I}$ denotes the set of successfully inspected surface points and N is the total number of discretized surface points. The parameter α weights the fuel consumption penalty. The vector $\mathbf{r}_{DC}$ represents the relative position between the servicer and the RSO, while R_D and R_C denote their respective collision radii. The constraint enforces a minimum separation distance throughout the mission to guarantee collision avoidance.

2.2 Semi-Partially Observable Markov Decision Process

Following Reference 9, the inspection task is formulated as a Markov Decision Process (MDP) [32], a standard framework for sequential decision-making. The servicer executes an impulsive maneuver followed by a variable-duration coast phase before selecting the next action, naturally leading to a semi-Markov formulation in which each action has an associated duration Δt. Because the agent observes only a subset of the full system state, the problem is partially observable. The discounted reward accumulated over an action duration is given by:

$$r_t^{(\gamma)} = \int_{t_t}^{t_t + \Delta t_t} e^{-\gamma(t - t_t)} \rho_t(t)\, dt \tag{4}$$

where γ is the discount factor and $\rho_t(t)$ is the reward density. The elements of the POMDP are defined as follows:

2.2.1 State Space

The state space includes all the satellite dynamical states, flight software states, and relevant environmental variables availabe in the Basilisk simulation. An episode terminates when any of the following conditions are met:

- Completion: At least 90% of illuminated points are inspected
- Collision: $\|\mathbf{r}_{DC}(t)\| < R_D + R_C$
- Escape: $\|\mathbf{r}_{DC}(t)\| > 1$ km
- Fuel depletion: $\Delta v_{\text{available}} = 0$
- Time limit: $t > 10$ orbital periods

2.2.2 Observation Space

The observation space consists of a subset of the full system state, summarized in Table 1. Compared with Reference 9, the observation space has been extended to include the RSO attitude and angular velocity, enabling the policy to reason about the target rotational dynamics.

Element	Dim	Description
$\boldsymbol{r}_{DC}^{H}$	3	Hill-frame relative position
$\boldsymbol{v}_{DC}^{H}$	3	Hill-frame relative velocity
$\boldsymbol{\omega}_{B/H}^{N}$	3	RSO angular velocity (inertial frame)
$\boldsymbol{\sigma}_{B/H}^{N}$	3	RSO MRP attitude coordinates relative to Hill frame
$\Delta v_{\text{available}}$	1	Remaining Δv budget
i	1	Orbital inclination
Ω	1	RAAN
$\hat{\boldsymbol{s}}$	3	Sun direction vector
$[\tau_{\text{ecl}}^{o}, \tau_{\text{ecl}}^{c}]$	2	Eclipse entry and exit times
t	1	Elapsed time since episode start
Inspection status	M	Regional inspection coverage

Table 1: Observation space elements

The inspection status represents the fraction of inspected points within $M = 15$ surface regions of the RSO. These regions are chosen to approximately match the servicer camera field of view.

2.2.3 Action Space

The action consists of an impulsive maneuver followed by a variable-duration coast interval:

$$a \in \Delta\boldsymbol{v}_{\max} \times [0, \Delta t_{\max}]$$

2.2.4 Reward Function

The reward is defined as:

$$R = R_{\text{inspect}} + R_{\text{goal}} - R_{\text{fuel}} - R_{\text{fail}} \tag{5}$$

where the four terms respectively reward newly inspected surface points and successful mission completion while penalizing fuel consumption and mission failure.

2.2.5 Transition Model

The transition model is a deterministic generative process. After each impulsive maneuver, the environment propagates the spacecraft dynamics over the selected coast duration Δt before the next decision step.

2.3 Solution Approach

The inspection problem is solved using deep-RL, with simulations conducted in BSK-RL [33], a modular satellite RL framework that wraps the high-fidelity astrodynamics simulator Basilisk [34] within the Gymnasium [35] interface. This combination provides realistic spacecraft dynamics, including J_2 perturbations, atmospheric drag, and attitude control. The sPOMDP formulation described above is implemented within this framework by specifying the observation and action spaces, reward function, termination conditions, RSO spin dynamics, and illumination constraints. An example of the environment setup is available on the BSK-RL GitHub repository.[1] Proximal policy optimization (PPO) is used to train a policy for the inspection task [36]. The learning rate follows a piecewise-constant decay schedule, starting at 10^{-4} and remaining constant until 20 million training steps. It is then reduced to 10^{-5} and further decreased to 10^{-6} after 30 million steps to improve stability and promote convergence during the later stages of training. The discount factor is set to $\gamma = 0.9999$, while the success and failure reward weights are fixed at $\beta_{\text{success}} = \beta_{\text{fail}} = 1$. A batch size of 1550 is used, with all remaining hyperparameters following the default PPO settings in RLlib 2.35.0. Each

[1] `https://avslab.github.io/bsk_rl/examples/rso_inspection.html`

policy is trained for 48 hours using 32 CPU cores. At the start of each episode, the initial conditions are randomized to ensure diverse orbital and dynamical scenarios. The RSO is placed on a different circular orbit for each episode, all at varying altitudes ranging from 500 to 1000 km but with varying orbital orientations. This variation changes both the illumination conditions and overall mission geometry. Although all evaluation results use a lead-follower initial configuration for the servicer, the policies are trained with randomized initial servicer states within a 1 km radius of the chief. This training strategy improves policy robustness and results in better overall performance.

2.4 Shielded Reinforcement Learning

Although reward penalties can discourage unsafe behavior during training, they cannot guarantee collision avoidance when a learned policy is deployed under varying operating conditions. Shielded reinforcement learning provides an additional mechanism for explicitly enforcing safety constraints, as introduced for spacecraft inspection in Reference 9. In general, shielding methods can be categorized as pre-shields, which restrict the available action space before action selection, and post-shields, which modify or override an action selected by the policy when it is deemed unsafe. The optimization-based shield employed by Reference 9 follows a post-shielding approach. An action is considered unsafe if the resulting trajectory is predicted to violate a prescribed keep-out radius within a finite prediction horizon. Collision prediction is performed using the CWH relative equations of motion [30]. Because the shield operates on the translational relative dynamics and assumes a spherical keep-out region, its formulation is independent of the target attitude and rotational dynamics. The same shield developed for the nadir-pointing target in Reference 9 can therefore be directly applied to the spinning-target scenarios considered here. When the policy selects an unsafe maneuver, the shield replaces it with the closest admissible action that maintains the required separation while respecting the maneuver constraints. Both the keep-out radius and prediction horizon are configurable, allowing the degree of conservatism to be adjusted according to mission requirements. A key advantage of this formulation is that the safety constraints are decoupled from the policy training process. The keep-out radius and prediction horizon can therefore be adjusted at deployment without retraining the policy, allowing more conservative safety requirements to be imposed when operating under increased uncertainty.

3. RESULTS

The results are presented in two parts. The first considers the inspection of RSOs undergoing stable rotation about the orbit-normal axis, while the second extends the analysis to targets exhibiting general tumbling motion and compares the learned policies against baseline heuristics.

3.1 Stable spinning targets

The first set of results considers targets undergoing stable rotation about the orbit-normal axis at a constant spin rate. The objective is to learn a robust inspection policy that generalizes across varying orbital configurations, illumination conditions, and target spin rates. To investigate the trade-off between inspection performance and fuel consumption, policies are trained using fuel penalty weights $\alpha \in \{0.0, 0.1, 0.2, 0.3, 0.4, 0.5\}$. Figure 2 shows the evolution of the reward, fuel consumption, inspection time, and inspection coverage relative to the illuminated surface points during training.

The fuel penalty α has a strong influence on the behavior learned by the agent. As expected, smaller values of α encourage more aggressive inspection strategies, resulting in higher fuel consumption but faster task completion. In contrast, larger values promote fuel-efficient behavior at the expense of longer inspection times and, in some cases, incomplete inspections. For large α, the policy tends to increase the coast duration between maneuvers and rely more heavily on the natural relative dynamics, whereas low-α policies prioritize rapid completion through more frequent and larger maneuvers. The longer simulated episode durations associated with high-α policies also reduce the number of training steps that can be completed within the fixed 48-hour training period.

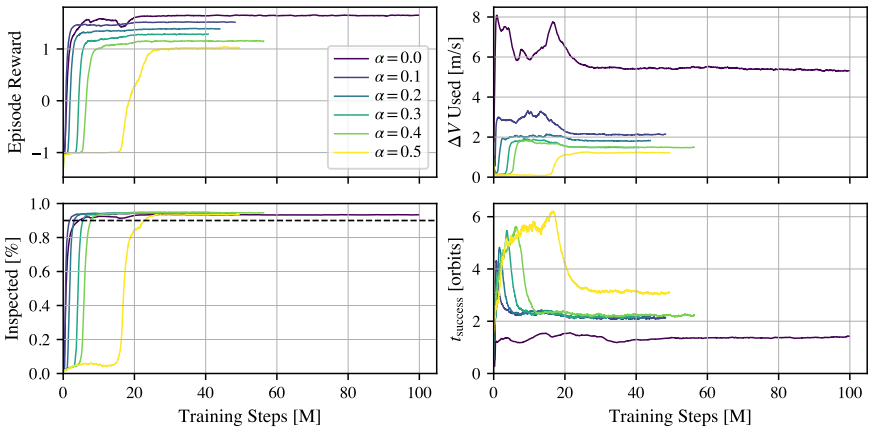
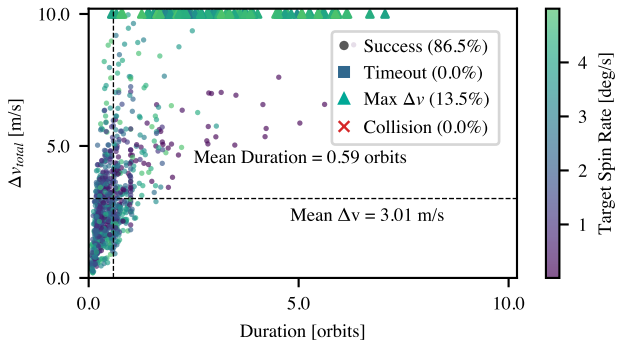


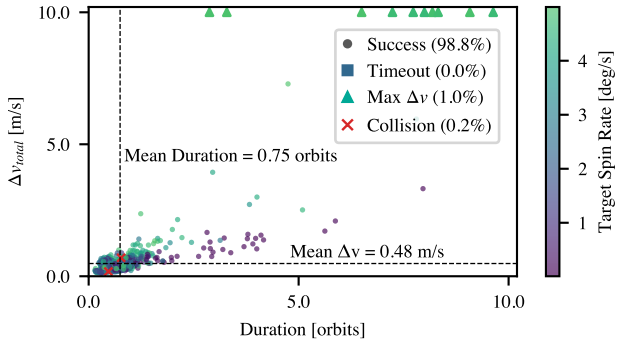
Fig. 2: Training curves for different values of the fuel penalty α

To further characterize these differences, Figure 3 compares three representative policies, with $\alpha = 0$, 0.2, and 0.5, over 1000 randomized environment instances. The unshielded evaluation varies the target spin rate as well as the orbital orientation, resulting in different illumination conditions across the test cases.

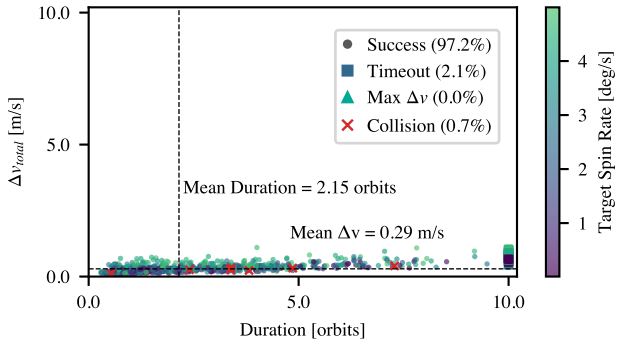
The distributions of successful episodes in Figure 3 clearly illustrate the trade-off induced by the fuel penalty. With no fuel penalty ($\alpha = 0$), the policy achieves the shortest mean successful inspection duration of 0.59 orbits, but at the cost of the highest mean fuel consumption, with a total Δv of 3.01 m/s. At the other extreme, the policy trained with $\alpha = 0.5$ reduces the mean Δv to only 0.29 m/s, while increasing the mean inspection duration to 2.15 orbits. The intermediate policy with $\alpha = 0.2$ provides a more balanced solution, completing successful inspections in an average of 0.75 orbits while consuming a mean Δv of 0.48 m/s. These results demonstrate that the reward formulation can be tuned to produce inspection policies corresponding to different operational priorities, ranging from rapid inspection to fuel conservation. The effect of α, however, extends beyond the trade-off between mission duration and fuel consumption. Allowing unrestricted fuel expenditure within the available budget does not necessarily improve overall mission performance. For $\alpha = 0$, 13.5% of the evaluation cases terminate due to fuel depletion before the inspection is completed. Thus, although the policy is encouraged to complete the inspection rapidly, excessive maneuvering can exhaust the available Δv and ultimately prevent mission completion. Conversely, the $\alpha = 0.5$ policy achieves a higher success rate of 97.2%, with most failures resulting from reaching the 10-orbit mission time limit. This behavior is consistent with the stronger fuel penalty, which encourages the policy to rely more heavily on passive relative motion and longer coast intervals. Among the evaluated policies, $\alpha = 0.2$ provides the most favorable balance between these competing objectives. In addition to achieving moderate inspection times and low fuel consumption, it yields the highest success rate of 98.8%. Of the remaining cases, 1.0% terminate due to fuel depletion and 0.2% due to collision. Although collisions occur only rarely, even a small collision probability is undesirable for proximity operations and motivates the introduction of an explicit safety mechanism. The optimization-based safety shield proposed in Reference 9 is therefore applied in the subsequent analysis to assess whether collision risk can be mitigated without substantially degrading inspection performance. Based on its overall balance of inspection time, fuel consumption, and success rate, the policy trained with $\alpha = 0.2$ is selected for the remainder of the evaluation.



(a) $\alpha = 0$



(b) $\alpha = 0.2$



(c) $\alpha = 0.5$

Fig. 3: Benchmark evaluation of policies with different fuel penalties α over 1000 environment instances

Shield

As discussed previously and demonstrated by the collision cases observed in the unshielded evaluation, reward penalties can discourage unsafe behavior but cannot guarantee collision avoidance under varying operating conditions. This section therefore evaluates the selected policy using the safety shield described in the previous section. The $\alpha = 0.2$ policy is combined with a shield enforcing a 50 m keep-out radius and a prediction horizon of one orbit. As shown in Figure 4, no collisions are observed across the 1000 shielded evaluation instances, increasing the overall success rate from 98.8% to 99.0%. The remaining failures are associated with mission constraints other than collision.

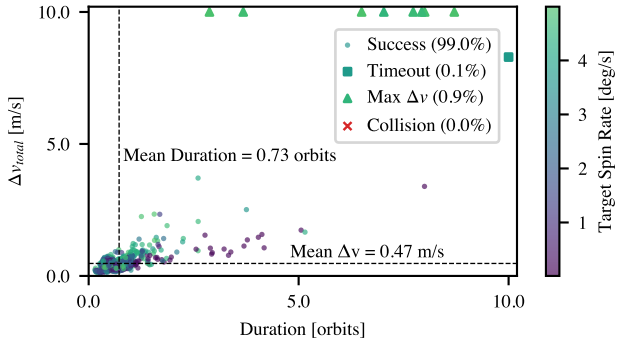


Fig. 4: Shielded benchmark evaluation of the policy with $\alpha = 0.2$, a 50 m keep-out radius, and a one-orbit prediction horizon over 1000 environment instances

For this configuration, the impact of the shield on nominal inspection performance is negligible, which is expected given the low collision rate of the unshielded policy and the relatively moderate safety constraints. Increasing the keep-out radius or prediction horizon results in more frequent shield interventions and therefore provides a more conservative safety configuration. However, these additional interventions can increase computational cost and may also increase inspection duration and Δv expenditure. To illustrate this trade-off, a second evaluation is performed using a more conservative shield with a 100 m keep-out radius and a two-orbit prediction horizon, as shown in Figure 5.

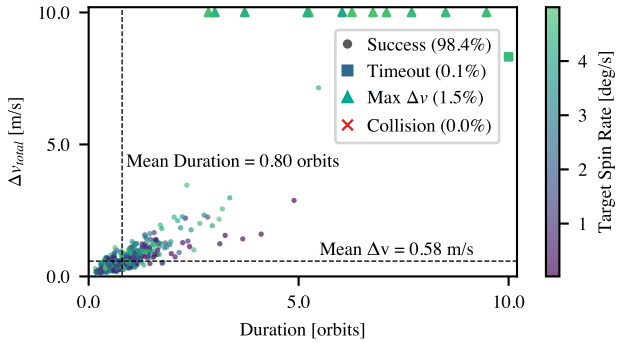


Fig. 5: Shielded benchmark evaluation of the policy with $\alpha = 0.2$, a 100 m keep-out radius, and a two-orbits prediction horizon over 1000 environment instances

With the more conservative shield configuration, the mean Δv increases to 0.58 m/s and the mean mission duration increases to 0.8 orbits. This illustrates the trade-off introduced by the shield parameters: more conservative safety constraints can provide greater robustness to uncertainty but may require additional maneuvering and longer inspection times. The shield therefore provides a tunable mechanism for balancing collision avoidance and inspection performance without modifying or retraining the underlying RL policy.

3.2 Targets on a general tumble

Following the strong performance observed for stable spinning targets, the analysis is extended to RSOs undergoing general tumbling motion. In these scenarios, both the initial spin axis and spin rate are randomized, requiring the policy to accommodate a broader range of target rotational dynamics. The policy is trained using the same procedure and computational budget as in the stable-spin case, with the target rotational dynamics replaced by randomized tumble conditions. Based on the trade-offs identified in the previous section, the policy trained with a fuel penalty of $\alpha = 0.2$ is selected for the following evaluations. Figure 6 presents the benchmark performance over 1000 randomized environment instances, varying the RSO orbit, initial spin axis, and spin rate.

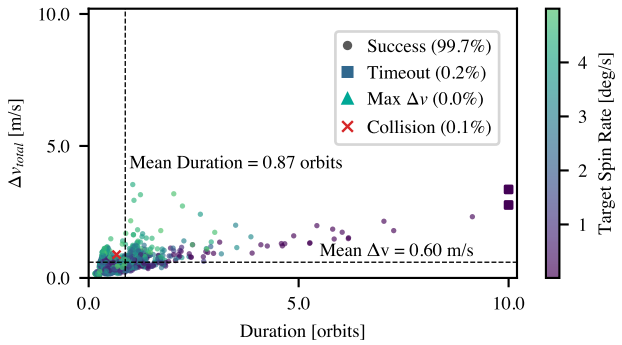


Fig. 6: Benchmark evaluation of the policy with $\alpha = 0.2$ over 1000 environment instances

The policy maintains strong performance when extended to generally tumbling targets, achieving a 99.7% success rate with a mean successful inspection duration of 0.87 orbits and a mean total Δv of 0.60 m/s. Collisions account for only 0.1% of the evaluation cases and can be mitigated using the safety shield demonstrated in the previous section. No failures due to fuel depletion are observed, while 0.2% of the cases terminate upon reaching the mission time limit. Interestingly, these timeouts occur primarily at low target spin rates, suggesting that faster target rotation may facilitate inspection rather than increase its difficulty. Because the benchmark simultaneously varies the orbital and rotational conditions, the effect of target rotation cannot be isolated from Figure 6 alone. A second benchmark is therefore performed using a fixed, high-illumination orbit and fixed servicer initial conditions, while randomizing only the target spin rate and initial spin axis. This isolates the influence of the target rotational dynamics while maintaining a large number of inspectable surface points. The results over 1000 environment instances are shown in Figure 7.

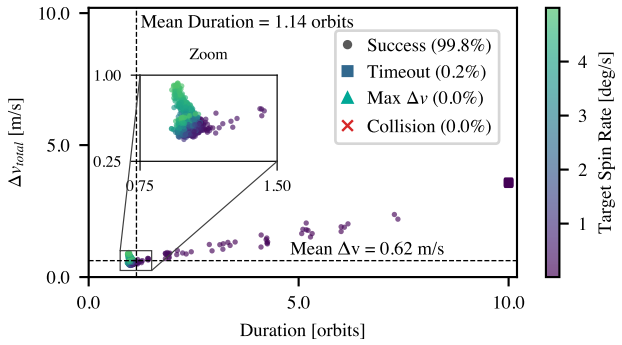
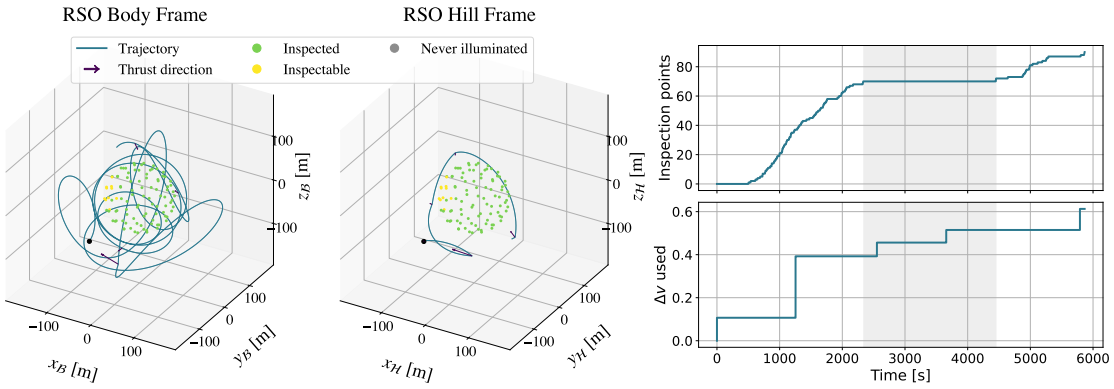


Fig. 7: Benchmark evaluation of the policy with $\alpha = 0.2$ over 1000 fixed-orbit environment instances

A clear dependence on target spin rate emerges from this evaluation. Successful inspections at higher spin rates are concentrated in the lower-left region of Figure 7, corresponding to shorter inspection durations and lower Δv expenditure. In contrast, targets with lower spin rates, particularly below approximately 1.5°/s, generally require more time and maneuvering effort to complete the inspection. This behavior suggests that the learned policy exploits the natural rotational dynamics of the target: as the RSO tumbles, previously inaccessible surface regions rotate into favorable viewing geometries, allowing the servicer to inspect them without repositioning. At lower spin rates, these changes in viewing geometry occur more slowly, requiring the servicer to perform additional maneuvers to observe different regions within the mission time limit.

To further illustrate this learned behavior, Figure 8 presents an example inspection trajectory in both the RSO Hill frame and body-fixed frame for a randomly tumbling target in the same high-illumination orbit used in the preceding benchmark. For visualization purposes, the RSO is displayed at 35 times its actual scale to improve the visibility of the surface inspection points. In this scenario, all surface points become inspectable during the mission. The policy successfully inspects 90% of the surface within 5865 s while using a total Δv of 0.6128 m/s. Figure 8(b) shows the corresponding evolution of inspection progress and Δv expenditure, with the shaded region indicating the

period spent in eclipse. As expected, no additional surface points are inspected during eclipse because the illumination constraint cannot be satisfied. The difference between the trajectories expressed in the Hill and body-fixed frames provides further insight into the learned strategy. The relatively limited motion observed in the Hill frame indicates that the servicer performs minimal translational repositioning, whereas the body-frame trajectory exhibits substantial motion around the target as the RSO tumbles. This demonstrates that a significant portion of the changing observation geometry is generated by the target rotation itself rather than by active servicer maneuvering.



(a) Trajectory plots

(b) Inspection progress and fuel usage over time

Fig. 8: Example inspection trajectory for a tumbling target in a high-illumination orbit

The observation that target tumbling can facilitate inspection raises an important question: if the target's natural rotation provides much of the required variation in viewing geometry, is an adaptive maneuvering policy necessary, or can simple heuristics achieve comparable performance? To investigate this question, the RL policies trained with $\alpha = 0$, 0.2, and 0.5 are compared against three baseline heuristics, as shown in Figure 9.

The first baseline, referred to as the *do-nothing* heuristic, performs no maneuvers. Because the servicer is initialized in a lead-follower configuration within the allowable inspection distance, this strategy relies entirely on the target's natural rotation to expose surface regions to the camera. The second baseline, referred to as the *Sun heuristic*, similarly maintains the initial configuration until the servicer transitions to the shaded side of the RSO as the orbital geometry evolves. It then performs a maneuver to reverse the lead -follower configuration, repositioning the servicer on the Sun-facing side of the target. This procedure is repeated as necessary until the inspection is completed. Finally, the *Sun-side vertical* heuristic places the servicer on a bounded relative orbit near the illuminated side of the target, with an along-track offset biased toward the Sun-facing side and a prescribed out-of-plane oscillation. This produces repeats variations in viewing geometry while maintaining proximity to the illuminated region, providing a simple rule-based inspection strategy without explicitly adapting to the target spin or inspection history. Additional heuristic configurations and initial orbital positions relative to the Sun were also investigated but did not produce significant improvements. Figure 9 summarizes the performance of the RL policies and the three representative baseline heuristics over the randomized benchmark cases.

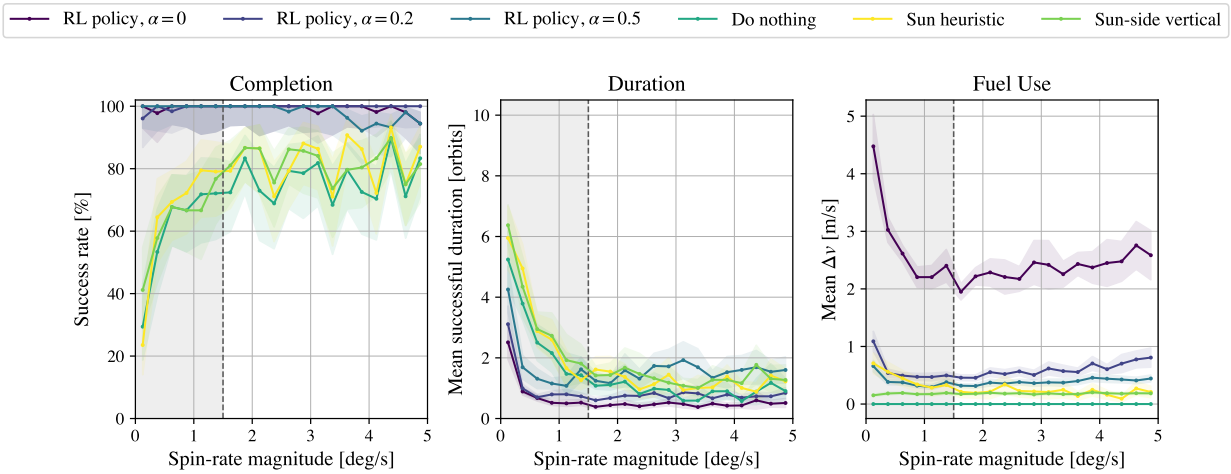


Fig. 9: Comparison of RL policies and baseline heuristics over 1000 environment instances

As expected, the heuristic strategies generally require less Δv than the RL policies, with the do-nothing heuristic requiring no active maneuvering. However, this reduction in fuel expenditure comes at the expense of lower success rates and longer inspection times. The difference is particularly pronounced at low target spin rates, where the performance of the heuristics deteriorates substantially. In this regime, relying primarily on target rotation is insufficient to expose the required surface regions within the mission time limit. The RL policies compensate by increasing both maneuvering effort and inspection duration, allowing them to maintain comparatively high success rates across a broader range of spin conditions. The heuristic strategies can nevertheless perform well when tailored to a particular orbital configuration. To illustrate this, Figure 10 repeats the comparison for the fixed high-illumination orbit considered previously. Under these more specific conditions, the Sun-side vertical heuristic achieves success rates comparable to the RL policies while requiring less Δv, although at the expense of longer inspection durations. Its behavior is therefore most similar to that of the RL policy trained with the largest fuel penalty. Nevertheless, its performance again degrades at low target spin rates. This comparison highlights the primary advantage of the RL approach: rather than being designed for a particular orbital or rotational configuration, the learned policy maintains strong performance across variations in orbit, illumination, spin rate, and spin-axis orientation.

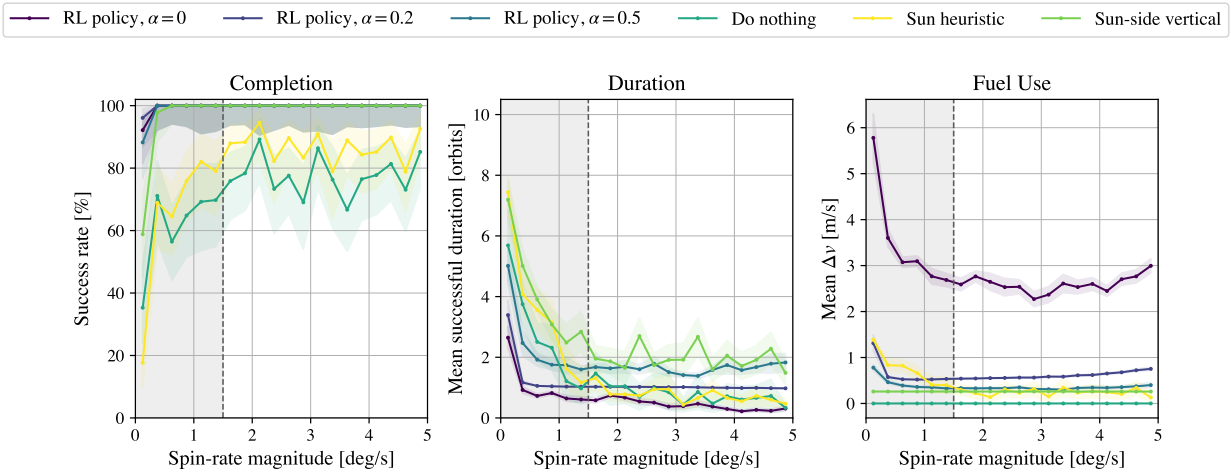


Fig. 10: Comparison of RL policies and baseline heuristics over 1000 fixed-orbit environment instances

Finally, the observed degradation in performance at low spin rates motivates a dedicated evaluation of this more challenging regime. The benchmark of Figure 6 is therefore repeated over 1000 randomized environment instances

while restricting the target spin rate to values below 1.5°/s. The results are shown in Figure 11.

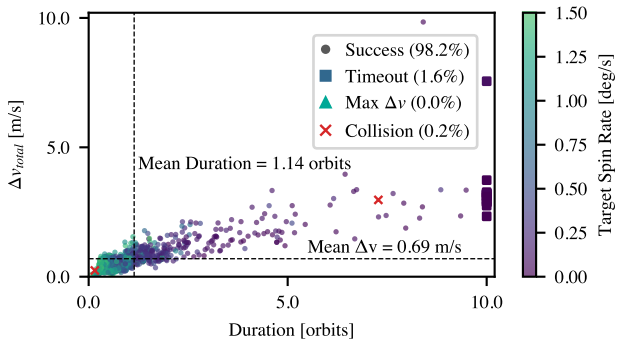


Fig. 11: Benchmark evaluation of the policy with $\alpha = 0.2$ over 1000 environment instances with target spin rates up to 1.5°/s

The results confirm that low-spin targets constitute a more challenging inspection regime. Restricting the spin rate to this range increases the mean successful inspection duration from 0.87 to 1.14 orbits and the mean total Δv from 0.60 to 0.69 m/s. The success rate decreases from 99.7% to 98.2%, with 0.2% of cases terminating due to collision and 1.6% reaching the mission time limit. The latter failures are concentrated among very slowly rotating and nadir-pointing targets, for which the target dynamics provide little or no passive variation in viewing geometry. Despite this degradation, the policy maintains a high success rate across the low-spin regime, demonstrating robustness to substantial variations in orbital and rotational conditions. However, the results also indicate a limitation in transferability: a policy trained predominantly on tumbling targets does not reliably generalize to the limiting nadir-pointing case. This suggests that the training distribution must adequately represent near-zero spin rates if performance in this regime is required. Conversely, the observed performance indicates that policies exposed to the more challenging low-spin conditions can generalize to faster rotations, where target motion provides additional opportunities for passive inspection.

4. CONCLUSION

This work extends reinforcement learning-based autonomous RSO inspection to targets exhibiting stable spin and general tumbling motion. The proposed policies leverage both the relative orbital motion of the servicer and the rotational dynamics of the target to efficiently inspect illuminated surface regions while balancing inspection time and fuel consumption. The results demonstrate robust performance across variations in orbital geometry, illumination conditions, spin rate, and spin-axis orientation. In particular, target rotation is shown to facilitate inspection by naturally exposing different surface regions, reducing the maneuvering required by the servicer. Comparisons with baseline heuristics highlight the advantage of the learned policies in maintaining high performance across varying mission conditions. An optimization-based safety shield was also applied to the learned policy, providing collision avoidance guarantees and enabling robust and safe autonomous inspection.

Future work will extend the framework to larger and geometrically complex targets, such as space stations. This will require significant modifications to the simulation and training environments, including shape-dependent surface discretization, explicit spacecraft pointing rather than center pointing, modeling of self-shadowing, and geometry-dependent keep-out regions. These extensions will enable the application of the proposed approach to autonomous inspection of more complex space assets.

5. REFERENCES

[1] Dehann Fourie, Brent Tweddle, Steve Ulrich, and Alvar Saenz Otero. Vision-based relative navigation and control for autonomous spacecraft inspection of an unknown object. In *AIAA guidance, navigation, and control (GNC) conference*, page 4759, 2013.

[2] Dehann Fourie, Brent E Tweddle, Steve Ulrich, and Alvar Saenz-Otero. Flight results of vision-based navigation

for autonomous spacecraft inspection of unknown objects. *Journal of spacecraft and rockets*, 51(6):2016–2026, 2014.

[3] Michael O'Connor, Brent Tweddle, Jacob Katz, Alvar Saenz-Otero, David Miller, and Konrad Makowka. Visual-inertial estimation and control for inspection of a tumbling spacecraft: experimental results from the international space station. In *AIAA Guidance, Navigation, and Control Conference*, page 4701, 2012.

[4] Charles Oestreich, Antonio Terán Espinoza, Jessica Todd, Keenan Albee, and Richard Linares. On-orbit inspection of an unknown, tumbling target using nasa's astrobee robotic free-flyers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2039–2047, 2021.

[5] Timothy Philip Setterfield. *On-orbit inspection of a rotating object using a moving observer*. PhD thesis, Massachusetts Institute of Technology, 2017.

[6] Michele Maestrini, Pierluigi Di Lizia, et al. Guidance for autonomous inspection of unknown uncooperative resident space object. In *Proceedings of the 2020 AAS/AIAA Astrodynamics Specialist Conference, South Lake Tahoe, CA, USA*, pages 9–12, 2020.

[7] Michele Maestrini and Pierluigi Di Lizia. Guidance strategy for autonomous inspection of unknown non-cooperative resident space objects. *Journal of Guidance, Control, and Dynamics*, 45(6):1126–1136, 2022.

[8] Karthik Sivaramakrishnan, Vignesh Sivaramakrishnan, Steven Cutlip, and Meeko Oishi. Trade-offs in on-orbit inspection of a stochastic, tumbling satellite. In *2025 American Control Conference (ACC)*, pages 1693–1700. IEEE, 2025.

[9] Mark Stephenson, Daniel Huterer Prats, and Hanspeter Schaub. Autonomous satellite inspection in low earth orbit with optimization-based safety guarantees. In *International Workshop on Planning Scheduling for Space*, Toulouse, France, April 28–30 2025.

[10] Joshua Aurand, Steven Cutlip, Henry Lei, Kendra Lang, and Sean Phillips. Deep q-learning for decentralized multi-agent inspection of a tumbling target. *Journal of Spacecraft and Rockets*, 61(2):341–354, 2024.

[11] Lorenzzo Quevedo Mantovani and Hanspeter Schaub. Improving robustness of autonomous earth-observing spacecraft using training environment enhancements. *Journal of Aerospace Information Systems*, 23(3):293–304, 2026.

[12] Mark Stephenson, Lorenzzo Quevedo Mantovani, and Hanspeter Schaub. Learning policies for autonomous earth-observing satellite scheduling over semi-markov decision processes. *Journal of Aerospace Information Systems*, 22(9):741–803, Sept. 2025.

[13] Adam Herrmann and Hanspeter Schaub. Reinforcement learning for the agile earth-observing satellite scheduling problem. *IEEE Transactions on Aerospace and Electronic Systems*, 59(5):5235–5247, Oct. 2023.

[14] Lorenzo Federici, Boris Benedikter, and Alessandro Zavoli. Deep learning techniques for autonomous spacecraft guidance during proximity operations. *Journal of Spacecraft and Rockets*, 58(6):1774–1785, 2021.

[15] Giovanni Fereoli, Hanspeter Schaub, and Pierluigi Di Lizia. Meta-reinforcement learning for spacecraft proximity operations guidance and control in cislunar space. *Journal of Spacecraft and Rockets*, 62(3):706–718, 2025.

[16] Qingyu Qu, Kexin Liu, Wei Wang, and Jinhu Lü. Spacecraft proximity maneuvering and rendezvous with collision avoidance based on reinforcement learning. *IEEE Transactions on Aerospace and Electronic Systems*, 58(6):5823–5834, 2022.

[17] Kanta Prasad Sharma, Indradeep Kumar, Pavitar Parkash Singh, K Anbazhagan, Hussain Mobarak Albarakati, Mohammed Wasim Bhatt, Avlokulov Anvar Ziyadullayevich, Arti Rana, et al. Advancing spacecraft rendezvous and docking through safety reinforcement learning and ubiquitous learning principles. *Computers in Human Behavior*, 153:108110, 2024.

[18] Roberto Furfaro, Andrea Scorsoglio, Richard Linares, and Mauro Massari. Adaptive generalized zem-zev feedback guidance for planetary landing via a deep reinforcement learning approach. *Acta Astronautica*, 171:156–171, 2020.

[19] Brian Gaudet, Richard Linares, and Roberto Furfaro. Adaptive guidance and integrated navigation with reinforcement meta-learning. *Acta Astronautica*, 169:180–190, 2020.

[20] Adam Herrmann and Hanspeter Schaub. Autonomous small body science operations using reinforcement learning. *Journal Of Aerospace Information Systems*, 21(10):865–884, Oct 2024.

[21] Brian Gaudet, Richard Linares, and Roberto Furfaro. Terminal adaptive guidance via reinforcement meta-learning: Applications to autonomous asteroid close-proximity operations. *Acta Astronautica*, 171:1–13, 2020.

[22] Shota Takahashi and D Scheeres. Autonomous proximity operations at small neas. In *Proceedings of the 33rd International Symposium on Space Technology and Science*, volume 2, 2022.

[23] Zheng Chen, Hutao Cui, and Yang Tian. Autonomous maneuver planning for small-body reconnaissance via reinforcement learning. *Journal of Guidance, Control, and Dynamics*, 47(9):1872–1884, 2024.

[24] Benjamin Bernhard, Changrak Choi, Amir Rahmani, Soon-Jo Chung, and Fred Hadaegh. Coordinated motion planning for on-orbit satellite inspection using a swarm of small-spacecraft. In *2020 IEEE Aerospace Conference*, pages 1–13. IEEE, 2020.

[25] Yashwanth Kumar K Nakka, Wolfgang Hönig, Changrak Choi, Alexei Harvard, Amir Rahmani, and Soon-Jo Chung. Information-based guidance and control architecture for multi-spacecraft on-orbit inspection. In *AIAA Scitech 2021 Forum*, page 1103, 2021.

[26] Henry H Lei, Matt Shubert, Nathan Damron, Kendra Lang, and Sean Phillips. Deep reinforcement learning for multi-agent autonomous satellite inspection. In *Proceedings of the 44th Annual American Astronautical Society Guidance, Navigation, and Control Conference, 2022*, pages 1391–1412. Springer, 2022.

[27] Joshua Aurand, Steven Cutlip, Henry Lei, Kendra Lang, and Sean Phillips. Exposure-based multi-agent inspection of a tumbling target using deep reinforcement learning. *arXiv preprint arXiv:2302.14188*, 2023.

[28] Mark Stephenson and Hanspeter Schaub. Safe, autonomous multi-agent inspection of space objects leveraging relative orbit dynamics. In *Advanced Maui Optical and Space Surveillance Technologies Conference*, Maui, Hawaii, September 16–99 2025.

[29] Mark Stephenson and Hanspeter Schaub. Comparing probabilistic roadmaps and reinforcement learning for relative motion inspection of space objects. In *AAS Guidance, Navigation and Control Conference*, Breckenridge, CO, Jan. 31 – Feb. 4 2026.

[30] WH Clohessy and RS Wiltshire. Terminal guidance system for satellite rendezvous. *Journal of the aerospace sciences*, 27(9):653–658, 1960.

[31] George William Hill. Researches in the lunar theory. *American journal of Mathematics*, 1(1):5–26, 1878.

[32] Richard S Sutton, Andrew G Barto, et al. *Reinforcement learning: An introduction*, volume 1. MIT press Cambridge, 1998.

[33] Mark Stephenson and Hanspeter Schaub. Bsk-rl: Modular, high-fidelity reinforcement learning environments for spacecraft tasking. In *International Astronautical Congress*, Milano, Itlay, Oct. 14–18 2024.

[34] Patrick W. Kenneally, Scott Piggott, and Hanspeter Schaub. Basilisk: A flexible, scalable and modular astrodynamics simulation framework. *Journal of Aerospace Information Systems*, 17(9):496–507, Sept. 2020.

[35] Mark Towers, Ariel Kwiatkowski, Jordan Terry, John U. Balis, Gianluca De Cola, Tristan Deleu, Manuel Goulão, Andreas Kallinteris, Markus Krimmel, Arjun KG, Rodrigo Perez-Vicente, Andrea Pierré, Sander Schulhoff, Jun Jet Tai, Hannah Tan, and Omar G. Younis. Gymnasium: A standard interface for reinforcement learning environments, 2025.

[36] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. 2017.