Autonomous Task Scheduling for Earth-Observing Satellites Tracking Moving Targets with Low Maneuverability

Yumeka Nagano* and Hanspeter Schaub †
Department of Aerospace Engineering Sciences, University of Colorado Boulder, Boulder, Colorado

Agile Earth-observing satellites are essential for monitoring dynamic global events such as ships, hurricanes, wildfires, and floods. Scheduling imaging tasks for moving targets is challenging, particularly when future target locations are uncertain. This study investigates autonomous satellite task scheduling using deep reinforcement learning (DRL), explicitly accounting for probabilistic target position estimates that evolve over time. Unlike prior studies assuming perfect observability and fixed imaging locations, this approach incorporates growing location uncertainty and models the required search time for successful imaging based on the estimated target location and its variance. The scheduling problem is formulated as a partially observable semi-Markov decision process (POsMDP) to capture limited state observability and variable-duration actions. Policies are trained and evaluated within BSK-RL, a high-fidelity simulation framework integrating the Basilisk spacecraft simulator with the Gymnasium RL interface. Two key factors are explored: (1) the effect of training environments and observation spaces on policy performance, and (2) the influence of reward function design on balancing target priority and uncertainty reduction. Results show that policies leveraging uncertainty observations outperform those without, and that reward formulations balancing target priority and uncertainty allow effective trade-offs, improving overall mission performance.

I. Introduction

Earth-observing satellites collect data using imaging instruments, playing a vital role in disaster tracking, environmental monitoring, resource exploration, and weather forecasting [1]. Advances in satellite agility—specifically, the ability to maneuver along all three axes (roll, pitch, and yaw)—have expanded the flexibility of data collection [2]. However, this agility comes at the cost of increased scheduling complexity. As a result, the agile Earth-observing satellite scheduling problem has garnered significant attention, driven not only by the rising number of observation requests but also by the demand for more efficient and responsive imaging strategies [3].

Traditionally, planning and scheduling are conducted ground-based, where a plan is created using an offline optimizer. This plan is then sequenced and uplinked to the spacecraft for open-loop execution [4]. While effective for simple problems, this approach struggles to accommodate the rising number of target requests—particularly when rapid rescheduling is needed due to newly added tasks. These challenges are especially pronounced for dynamic, unpredictable, and moving targets, such as ships, aircraft, hurricanes, wildfires, floods, tornadoes, and cloud systems, which require real-time decision-making to ensure relevant data capture. NASA highlights the critical need for advanced Earth-monitoring technologies to improve disaster response and resource management [5].

The scheduling problem becomes even more complex when imaging a single target cannot be accomplished in a single pointing or when the target's location is uncertain. In such cases, searching strategies are required. Prior work has proposed combining low-fidelity sensors to detect potential targets with high-fidelity sensors for precise imaging [6]. Strip imaging has been explored for linear targets [7] and extended to cover area targets by combining multiple strips [2, 8]. Additionally, unified modeling approaches have been proposed to handle point, line, and area targets by discretizing areas into a grid system [9].

Various methods have been proposed to address the satellite scheduling problem. Optimization-based approaches, such as branch-and-bound (B&B) [10] and mixed-integer linear programming (MILP) [11], offer theoretical guarantees of optimality but are often computationally expensive and unsuitable for real-time or onboard applications [12]. Moreover, MILP-based formulations typically rely on simplified models that may not adequately reflect real-world

*Graduate Research Assistant, Ann and H.J. Smead Department of Aerospace Engineering Sciences, University of Colorado, Boulder, Boulder, CO, 80309.

†Distinguished Professor and Department Chair, Ann and H.J. Smead Department of Aerospace Engineering Sciences, University of Colorado, Boulder, Boulder, CO, 80309.

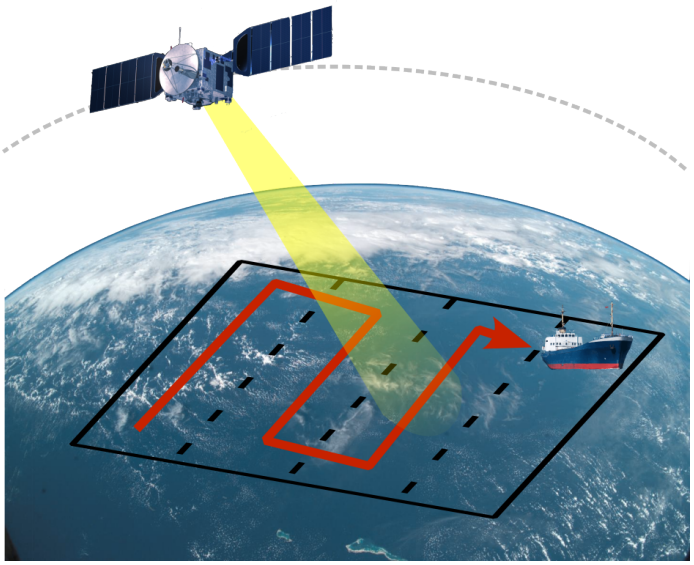


Fig. 1 Concept Illustration of satellite task scheduling with location uncertainty.

constraints. To reduce computation time for real-time applications, heuristic approaches—including greedy algorithms and local search techniques—have been developed [2]. While these methods can produce near-optimal solutions, they often struggle with the high dimensionality and dynamic nature of real-time scheduling, particularly when handling frequent task updates and a large number of requests. Stochastic optimization techniques such as genetic algorithms and simulated annealing have also been explored [13], but these face similar limitations in managing complex constraints under strict time requirements.

Further work has applied optimization techniques to the scheduling of moving targets in scenarios such as maritime surveillance, tropical storms, wildfires, and hurricanes [14–17]. However, these approaches still face significant challenges in handling the complexity and scale of real-time operations.

To overcome these limitations, recent studies have investigated autonomous satellite scheduling using machine learning (ML) techniques to enable onboard, real-time decision-making [18]. In this paradigm, the learning process is performed on the ground, and the trained model is deployed onboard, reducing computational burden while enabling closed-loop responses to dynamically changing tasks and unexpected events [19]. Despite their promise, many ML-based methods simplify spacecraft dynamics and often overlook critical safety and operational constraints, such as reaction wheel (RW) momentum dumping, power usage, and data storage limitations.

To address these challenges, deep reinforcement learning (DRL) has been introduced as a way to optimize decision-making in complex, uncertain environments. DRL leverages reinforcement learning (RL) algorithms to train policies parameterized by deep neural networks, typically formulated as a Markov decision process (MDP) [20, 21]. Recent work has extended this formulation to account for more realistic operational constraints, including power charging schedules, downlink windows, and RW momentum management [4]. Alternative formulations, such as semi-Markov decision processes (sMDPs) and partially observable MDPs (POMDPs), have also been proposed to handle variable time steps and limited state observability, respectively.

Some studies have also incorporated environmental and target uncertainties. For instance, cloud coverage forecasts are used to improve optical imaging tasking [22, 23], while wildfire detection studies account for uncertainty in wildfire locations by modeling probabilistic growth patterns [24]. Additionally, satellite state uncertainty, including actuator faults, has been explored in prior work [25].

While RL-based approaches have demonstrated promise for scheduling tasks involving static targets, moving targets introduce additional challenges due to uncertainty in their future positions. Many prior studies assume perfect observability of targets—a reasonable simplification for static or well-characterized objects, but an unrealistic assumption for moving targets such as ships or aircraft. For these dynamic targets, location uncertainty grows over time if the target is not periodically observed, meaning that a satellite must allocate additional time to search for each target before imaging. Figure 1 illustrates this concept, showing how the satellite must search over an uncertainty region to successfully image a moving target.

This work explicitly models the expansion of target uncertainty for slowly moving objects (ships) and incorporates it into a RL-based scheduling framework. Unlike prior studies, which often assume instantaneous imaging or perfect knowledge of target locations, this framework requires the satellite to dwell on targets for a true imaging duration that depends on the target's current uncertainty. This introduces a temporal trade-off: spending more time on a single target reduces its uncertainty but may limit the coverage of other targets. Key challenges addressed in this study include: (1) uncertainty-aware decision-making, where the agent must reason about which targets to image based on both priority and the current uncertainty; (2) dynamic environment modeling, where target uncertainty evolves over time and the satellite must account for search durations; and (3) reward function design, which balances competing objectives of imaging high-priority targets and minimizing overall uncertainty. By systematically investigating the effects of observation space, training environment, and reward formulation, this work develops policies capable of effective task scheduling under realistic, time-varying target uncertainty. Compared to prior work, this approach explicitly integrates the search-time constraints and evolving target uncertainty into both the environment and the reward structure, enabling more practical deployment for dynamic Earth-observation scenarios.

The remainder of this paper is organized as follows. Section II introduces the formulation of the satellite imaging task scheduling problem and highlights the location uncertainties considered. Section III describes the methodology, including the simulation environment and training strategies. Section IV provides a comprehensive analysis of policy performance across different observation space and reward function. Finally, Section V concludes with a discussion of key findings and future research directions.

II. Problem Formulation

This work investigates methods for tracking moving targets with location uncertainty using DRL algorithms for autonomous satellite task scheduling.

A. Target Uncertainty Modeling

Each imaging target is modeled with inherent location uncertainty to reflect realistic observation conditions. To successfully capture an image, the satellite must maintain pointing requirements toward the selected target for a sufficient duration while searching the uncertain area. Each target i is characterized by the following parameters and sampling models:

- **Search Area** (A_i): A square region on the surface within which the estimated location is the center of the area. The initial side length s_i is uniformly sampled between 10 and 100 km to represent uncertainty:

$$s_i \sim \mathcal{U}(10,\ 100) \quad \text{(km)} \tag{1}$$

$$A_i = s_i^2 \quad \text{(km}^2\text{)} \tag{2}$$

- **Uncertainty** u_i: The uncertainty associated with target i is defined as the area of the search region, normalized by the maximum initial search area (100 km × 100 km):

$$u_i = \frac{A_i}{100 \times 100} \quad (-) \tag{3}$$

  This value may exceed 1.0 if the search area expands over time.
- **Acquisition speed** v_i. The apparent ground-track speed is sampled uniformly to reflect orbital geometry and latitude effects:

$$v_i \sim \mathcal{U}(3.0,\ 6.0) \quad \text{(km/s)} \tag{4}$$

- **Scan width** w. The sensor swath is fixed:

$$w = 20 \quad \text{(km)} \tag{5}$$

- **Search duration** T_i: The total imaging duration, proportional to the number of sensor swaths $\lceil s_i/w \rceil$ required to cover the area:

$$T_i = \frac{s_i \lceil s_i/w \rceil}{v_i} \quad \text{(s)} \tag{6}$$

- **True imaging time** t_i^*: The true imaging time within the duration window, drawn from a Gaussian distribution centered at the mid-point of T_i (μ_i) with standard deviation $0.3T_i$ (σ_i), truncated to nonnegative values. Approximately 4.8% of samples occur after the window ends:

$$t_i^* \sim \max\bigl(0,\ \mathcal{N}(\mu_i,\ \sigma_i^2)\bigr) \quad \text{(s)} \tag{7}$$

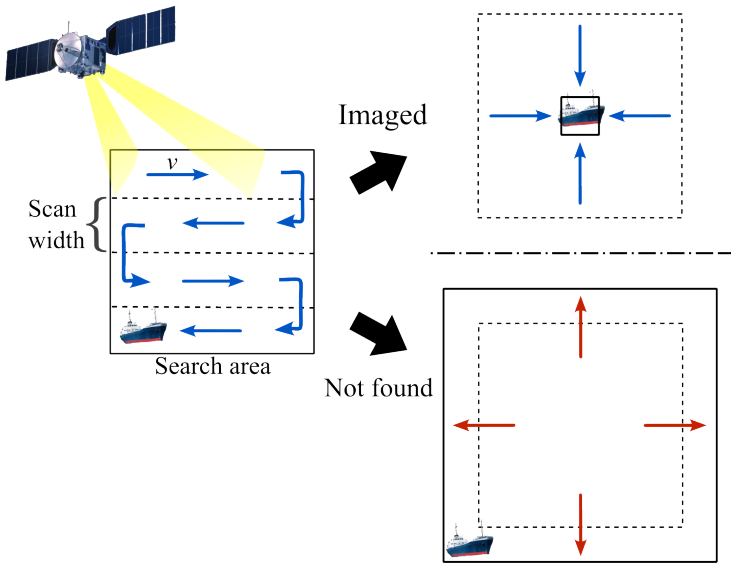


Fig. 2 Illustration of location uncertainty growth for a moving target and its effect on satellite imaging.

Figure 2 illustrates the concept of target location uncertainty growth. If a target is successfully imaged, its uncertainty resets to zero. If the target is not imaged, the search area expands over time at a rate proportional to the target's speed, reflecting increasing positional uncertainty.

B. Partially Observable Semi-Markov Decision Process Formulation

To account for variable time steps and partial observability, the satellite scheduling problem is formulated as a partially observable semi-Markov decision process (POsMDP), defined by the tuple $(\mathcal{S}, \mathcal{A}, T, R, O, Z)$ [12], where:

- State Space $\mathcal{S}$: Comprises the satellite and environmental states required to propagate the simulation.
- Action Space $\mathcal{A}$: Includes imaging actions corresponding to the next 32 target requests, consistent with the variable target set sizes analyzed in [12]. When image action is executed, it evaluates imaging constraints such as the relative pointing angle and angular rate. A target is considered successfully imaged if these constraints are satisfied for the true imaging duration (t_i^*). When the agent has access to the target's search area observation, the imaging duration is limited to the search time, even if the true imaging time extends longer. Without access to this uncertainty information, the maximum imaging duration is constrained to one orbital period. Once the imaging window closes, the satellite proceeds to its next decision step.
- Transition Function $T(s'|s, a)$: The environment uses a deterministic generative model. Given a state s and action a, the simulator deterministically generates the next state s'.
- Observation Space O: Represents the subset of the state space observable by the agent for decision-making. Table 1 summarizes the observation space used in this study, adapted from [12] with some observations removed (e.g., power consumption, which is not considered here). When target uncertainty is observable, the search area of each target is also included. Observations are provided for all $N = 32$ upcoming targets.
- Observation Function $Z(o|s, a)$: Defines the mapping from the environment state s to the agent's observation o. The satellite observes its own state and the estimated locations of upcoming targets, represented by their mean positions and associated uncertainties (standard deviations). While the mapping from the simulator state to these estimates is deterministic, the true target locations are not directly observable and evolve stochastically, rendering the environment partially observable from the agent's perspective.
- Reward Function $R(s, a, s')$: Defines the immediate gain obtained when transitioning from state s to s' after executing action a. Two reward formulations are considered:
  - **Weighted Uncertainty (*Multiply*):**

$$R(s, a, s') = \begin{cases} p_n \cdot u_n, & \text{if } a = a_{\text{image},n} \text{ and } t_{\text{search}} = t_n^* \\ 0, & \text{otherwise} \end{cases} \tag{8}$$

This formulation rewards imaging high-priority targets with large uncertainty, encouraging both value-driven selection and uncertainty reduction.

– **Additive Uncertainty (*Additive*):**

$$R(s,a,s') = \begin{cases} p_n + \lambda u_n, & \text{if } a = a_{\text{image},n} \text{ and } t_{\text{search}} = t_n^* \\ 0, & \text{otherwise} \end{cases} \tag{9}$$

Here, λ adjusts the relative importance of uncertainty versus target priority. In both cases, t_{search} is the time spent searching the target, and $a_{\text{image},n}$ denotes the imaging action for the selected target n.

Two reward assignment strategies are also considered:

– **Unique Target Imaging:** Each target yields a reward only once and is subsequently removed from the candidate list.

– **Multiple Target Imaging:** Repeated imaging of the same target yields repeated rewards, promoting additional images and uncertainty refinement.

Each episode simulates three orbits, and the episode terminates after completing these orbits.

Table 1 Observation Space of the Satellite

Parameter	Normalization	Description
${}^{\mathcal{E}}\boldsymbol{r}_{\mathcal{B}/\mathcal{E}}$	Earth's radius	Earth-fixed satellite position
${}^{\mathcal{E}}\boldsymbol{v}_{\mathcal{B}/\mathcal{E}}$	Orbital velocity	Earth-fixed satellite velocity
${}^{\mathcal{H}}\hat{\boldsymbol{c}}$	-	Instrument pointing direction in the Hill frame
${}^{\mathcal{E}}\boldsymbol{\omega}_{\mathcal{B}/\mathcal{E}}$	0.03 rad/s	Body angular rate
${}^{\mathcal{H}}\boldsymbol{r}_{n\in N}$	Earth's radius	Hill-frame positions of upcoming N targets
$\delta\theta_{n\in N}$	$\pi/2$ rad	Pointing error to upcoming N targets
$\delta\omega_{n\in N}$	0.03 rad/s	Pointing rate error to upcoming N targets
$t^o_{n\in N}$	300 s	Time until upcoming N target windows open
$t^c_{n\in N}$	300 s	Time until upcoming N target windows close
D	5	Upcoming request density
$p_{n\in N}$	-	Priority of upcoming N target
$u_{n\in N}$	Maximum initialized search area	Uncertainty of upcoming N target (only when observable)

C. Reward Function Design

The primary objective of the satellite scheduling problem is to maximize the cumulative reward from successfully imaging targets. When target uncertainty is present, using only the sum of target priorities can lead the agent to favor low-uncertainty targets that are easier to image. This behavior is suboptimal, because high-uncertainty targets, if ignored, can continue to grow more uncertain and may eventually become impossible to locate. Therefore, the reward function must consider both target priority and uncertainty to encourage the agent to image valuable targets while managing overall uncertainty growth.

Previous studies have explored multi-objective reward functions in satellite scheduling, balancing competing objectives such as target type, image quality, and revisit frequency [26, 27]. These works emphasize that carefully designed reward structures are essential for guiding reinforcement learning agents in complex decision-making environments.

The reward formulations considered in this study—weighted uncertainty (*Multiply*) and additive uncertainty (*Additive*)—capture complementary design objectives. The weighted formulation encourages imaging targets that are both high-priority and high-uncertainty, promoting robust information gain. The additive formulation enables a tunable trade-off between priority and uncertainty, allowing the agent to balance coverage and risk.

Two reward assignment strategies are also explored: unique target imaging, where each target contributes to the reward only once, and multiple target imaging, which allows repeated rewards for the same target to promote additional observations and uncertainty reduction. These design choices provide flexibility in shaping agent behavior to meet different operational objectives.

Table 2 Satellite Parameters

Parameter	Value
Altitude	800 km
Inclination	45°
Mass	330 kg
Inertia	[82.1, 98.4, 121.0] kg m^2
Relative angle limit for imaging	0.01 MRP norm (2.29°)
Relative angular rate limit for imaging	0.01 rad/s

III. Numerical Simulation Setup for Performance Evaluation

To investigate effective training strategies for satellite scheduling under target location uncertainty, two key aspects are examined:

1) **Effect of Training Environment and Uncertainty Observation:** This analysis investigates how different observation spaces influence policy performance. Specifically, policies trained in environments with and without target uncertainty are compared. Within the uncertain environment, two conditions are tested: one where the agent can observe the target's search area and another where it cannot. These are further compared with a policy trained in an idealized, uncertainty-free environment, where imaging is instantaneous once the satellite points to the true target location. This comparison elucidates how incorporating uncertainty in both the environment and observation space affects learning and decision-making.
2) **Effect of Reward Function Design:** This analysis evaluates how different reward formulations guide the agent's learning behavior. By comparing two proposed formulations—the weighted uncertainty and additive uncertainty—the study identifies which reward structure better incentivizes the agent to balance high-priority imaging and uncertainty reduction.

A. Numerical Simulation Environment

All simulations are conducted using BSK-RL,* an open-source Python package that provides modular, high-fidelity reinforcement learning environments for spacecraft tasking and planning [28]. BSK-RL integrates the Gymnasium framework, widely used in reinforcement learning research, with Basilisk,† a high-fidelity astrodynamics simulation software developed for spacecraft flight dynamics and operations [29].

Basilisk, written in C and C++ with a Python interface, serves as the generative model $G(s, a)$ in this framework, enabling realistic simulation of spacecraft dynamics, subsystems, maneuvering times, and power constraints. Its fast runtime and flight-proven capabilities make it particularly well-suited for reinforcement learning applications that require both physical accuracy and computational efficiency.

The simulation models spacecraft dynamics, including reaction wheel (RW) control, enabling physically realistic imaging maneuvers. When an imaging action is selected, the simulator computes the control torque required to align the spacecraft with the target. For this study, explicit search path planning is not modeled—the satellite maintains pointing on a single location during the imaging duration.

Targets are uniformly distributed across Earth's surface, with the number of imaging requests (targets) ranging from 1,000 to 10,000, directly affecting target density. Target properties follow the definitions in Section II.A. The target distribution type can be extended to city targets [12].

An example script demonstrating how to model target-location uncertainty using the uncertainty-growth environment for spacecraft tasking is available on the BSK-RL website‡.

B. Satellite Configuration

The simulation employs satellite parameters, along with initialization and target parameters, as detailed in Table 2. The satellite follows a circular orbit with a period of about 100 minutes. The initial location and target parameters are randomly selected. Parameters not explicitly listed are set to their default values as provided by BSK-RL.

*https://avslab.github.io/bsk_rl

†https://avslab.github.io/basilisk

‡https://avslab.github.io/bsk_rl/examples/target_location_uncertainty.html

C. Training Algorithm and Computational Resources

Training is performed using the proximal policy optimization (PPO) algorithm [26] from RLlib, a library that provides scalable software primitives for RL, was used [30]. The hyperparameters used for training the policy are listed in Table 3, while all other parameters are set to their default values in RLlib. A detailed hyperparameter analysis can be found in Reference 31.

Experiments were conducted on the University of Colorado Boulder's high-performance computing cluster (CURC) using 32 CPU cores and up to 20 million environment steps per policy—equivalent to several years of on-orbit experience simulated over a few days.

Table 3 Training Parameters

Parameter	Value
Learning rate	$3 \cdot 10^{-5}$
Training batch size	3,000
Number of SGD iterations	10
Neural Network	2 layers with 2048 neurons each
Discount factor	0.997
Gradient clipping	0.5
PPO clip parameter	0.2
Generalized advantage estimation (GAE)	0.95
Failure penalty	0

D. Effect of Uncertainty Observation in Training

To assess the impact of environmental uncertainty and its observability, three policies were trained and evaluated under different configurations, summarized in Table 4.

Table 4 Training environments under different uncertainty observation conditions.

Policy	Training Environment	Uncertainty Observation
Baseline	Instant imaging (no search)	None
Obs None	True imaging time (Eq.7)	None
Obs Area	True imaging time (Eq.7)	Search area

The *Baseline* policy assumes ideal imaging—an image is captured immediately once pointing constraints are met—and is later tested in an environment requiring true imaging time. The *Obs None* policy is trained and tested with true imaging times but without access to uncertainty information. The *Obs Area* policy, by contrast, includes search area observations (Table 1), allowing it to reason about spatial uncertainty during decision-making. For all three cases, the reward corresponds to the sum of target priorities for uniquely imaged targets.

E. Reward Function Analysis

After evaluating the effect of uncertainty observation, the second analysis investigates how different reward formulations affect learning. Using the *Obs Area* configuration as the base environment, policies are trained with the two proposed reward structures described in Section II.C: (1) the weighted uncertainty formulation (*Multiply*) and (2) the additive uncertainty formulation (*Additive*) with varying coefficients λ.

Multiple weighting values ($\lambda = 0.5, 1, 2, 3, 5$) are trained and tested alongside a baseline ($\lambda = 0$) that ignores uncertainty. In this setup, target search areas expand over time when not imaged, simulating increasing positional uncertainty typical of moving objects such as ships. When a target is successfully imaged, its search area resets to zero. The rate of expansion is proportional to target velocity, ranging between 22–46 km/h, this is assigned for each target at the beginning of the episode and fixed during that episode. Under these conditions, the agent may choose to reimage the same target multiple times to reduce its uncertainty, and rewards are granted for repeated successful images.

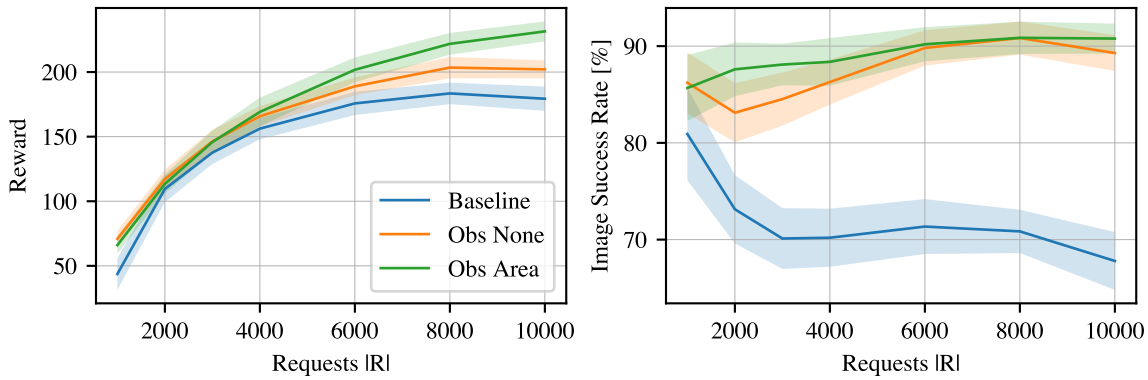
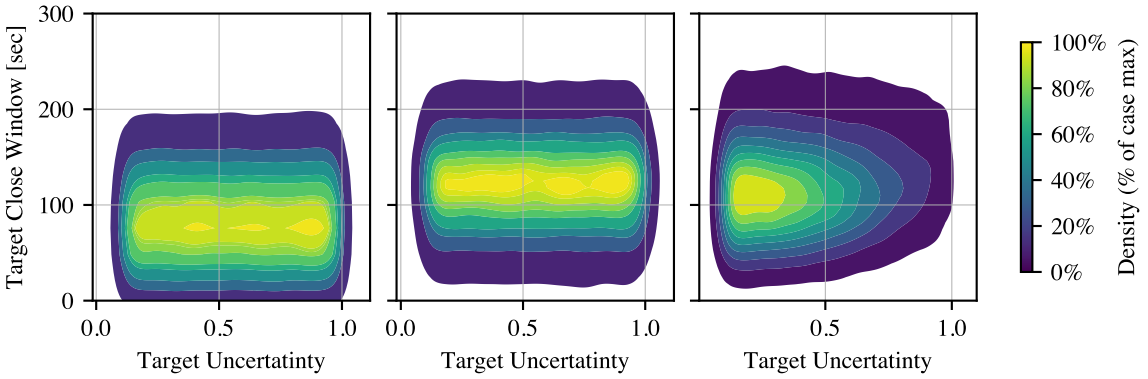


Fig. 3 Testing results for the three policies across varying numbers of targets.



(a) Baseline Policy **(b)Obs None Policy** **(c) Obs Area Policy**

Fig. 4 Heatmap of selected target characteristics for each policy, showing the distribution of target uncertainty and the remaining time until each target's imaging window closes.

IV. Results

Each policy is tested across different numbers of targets, ranging from 1000 to 10000 (1000, 2000, 3000, 4000, 6000, 8000, and 10000). For each case, simulations are run for three orbits with 50 random seeds, where the same seeds are used across all policies for fair comparison. In the plots, the solid line indicates the mean value, and the shaded region represents one standard deviation.

A. Results: Effect of Training Environment and Uncertainty Observation

Three policies described in Section III.D are trained and evaluated to assess how the training environment and the availability of uncertainty observations influence performance. For this analysis, the testing environment incorporates the uncertainty model, and the satellite must satisfy the true imaging duration specified in Equation 7. The reward function used is the sum of the priorities of uniquely imaged targets and does not incorporate uncertainty ($\lambda = 0$ for the *Additive* reward function).

Figure 3 shows the results across different target requests. The *Obs Area* policy—which has access to target uncertainty during both training and testing—consistently achieves the highest rewards and imaging success rates. This demonstrates that observing target uncertainty enables more informed and effective decision-making. The *Obs None* policy, trained in an environment that models uncertainty but without observing it directly, also outperforms the *Baseline* policy. This highlights the benefit of training under realistic stochastic dynamics, even when uncertainty is unobserved during execution. Overall, both uncertainty observation and realistic environment modeling significantly enhance policy performance under target-location uncertainty.

To further investigate behavioral differences among policies, Figure 4 illustrates the distribution of selected targets in terms of target uncertainty (u) and the remaining time until the target closes. The *Obs Area* policy preferentially selects

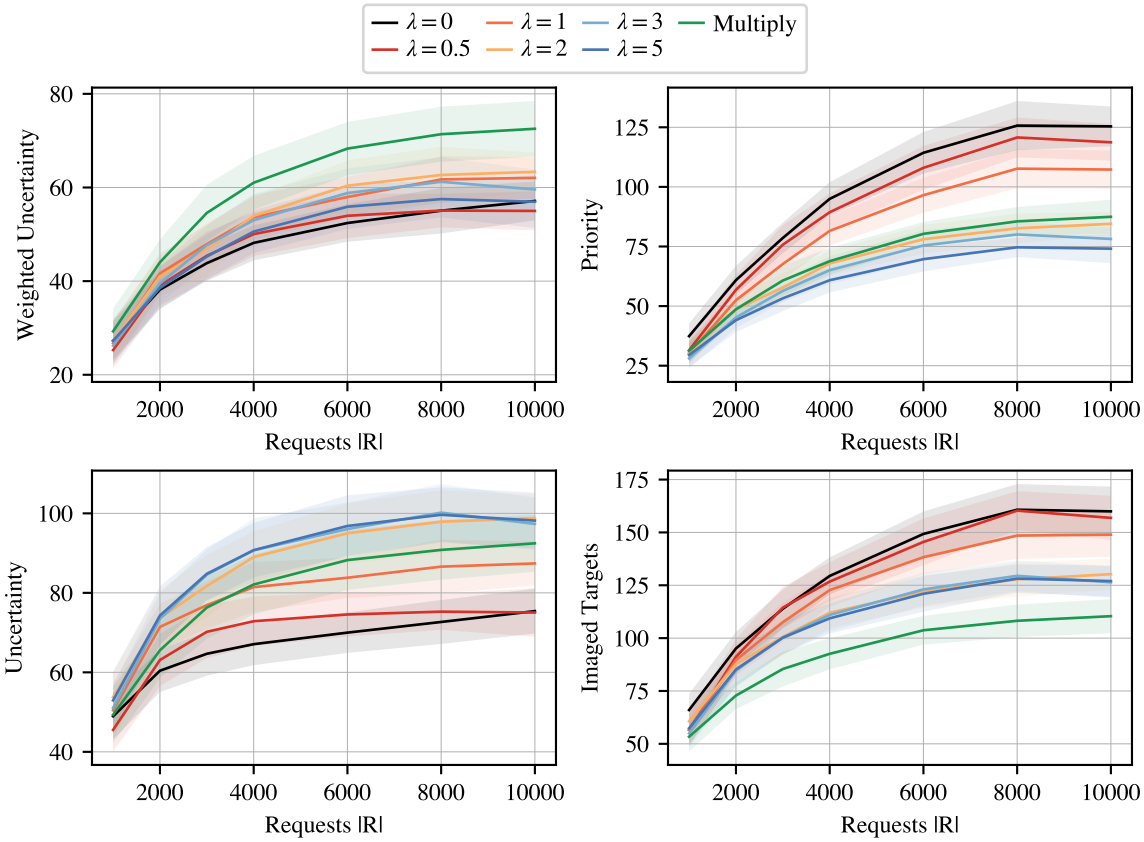


Fig. 5 Comparison of policies under different reward function formulations across varying numbers of targets.

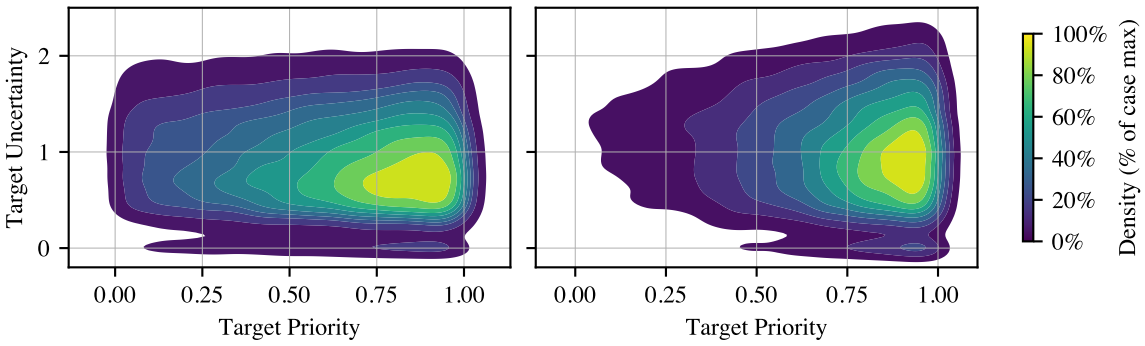
targets with lower uncertainty, improving the likelihood of successful imaging and resulting in higher overall rewards. The *Obs None* policy tends to select targets with longer closing windows compared to the *Baseline*, suggesting that it implicitly accounts for the time required to search each target, despite lacking direct access to uncertainty information. These patterns indicate that policies trained with exposure to uncertainty dynamics develop strategies that naturally prioritize search feasibility and imaging success.

B. Results: Effect of Reward Function Formulation

To evaluate the impact of different reward formulations, policies are trained and tested in the *Obs Area* configuration, where the agent observes target uncertainty during both training and evaluation. Target uncertainty grows over time if the target is not imaged, simulating the increasing positional uncertainty of moving objects. When a target is successfully imaged, its uncertainty resets. The agent may re-image targets multiple times, with rewards granted for each successful capture, allowing it to balance uncertainty reduction and coverage.

Four metrics are considered: (1) the cumulative weighted uncertainty of imaged targets, (2) the cumulative priority of imaged targets, (3) the cumulative uncertainty of imaged targets, and (4) the total number of targets imaged over the episode. These metrics provide insight into how effectively each policy balances the trade-off between imaging high-priority targets and reducing uncertainty. Because the optimal strategy is to image targets that are both high-priority and high-uncertainty, larger values in all four metrics indicate better performance.

Figure 5 summarizes the performance across reward formulations. The *Multiply* formulation consistently achieves the highest cumulative weighted uncertainty, as expected since this metric directly corresponds to the reward used during training. Within the *Additive* family, the best weighted uncertainty performance is observed for $\lambda = 1, 2, 3$. Increasing λ improves performance on uncertainty-related metrics while degrading priority-based performance, revealing the inherent trade-off between prioritizing high-value targets and reducing uncertainty. The *Multiply* formulation achieves a more balanced trade-off between these objectives.



(a) $\lambda = 2$ **(b)Multiply**

Fig. 6 Heatmap of selected target characteristics for *Additive* $\lambda = 2$ and *Multiply* reward function, showing the distribution of target priority and uncertainty.

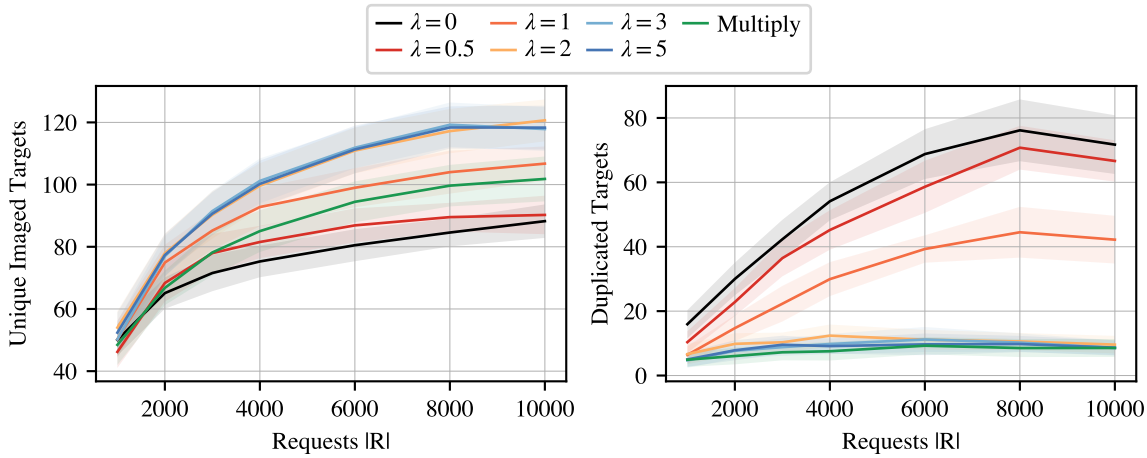


Fig. 7 The number of unique and duplicated imaged targets over episode across varying numbers of targets.

Although the *Multiply* formulation excels in weighted uncertainty, it selects the fewest targets overall, indicating a highly selective strategy focused on the most valuable targets rather than broad coverage. Figure 6 illustrates the distribution of selected targets in terms of priority and uncertainty, comparing *Multiply* to *Additive* $\lambda = 2$, which achieves the highest weighted uncertainty among the additive variants. The *Multiply* formulation strongly favors targets that are simultaneously high-priority and high-uncertainty, resulting in a concentrated selection strategy. In contrast, the *Additive* formulation allows selection of high-uncertainty but lower-priority targets, producing a broader distribution along the priority axis.

Beyond aggregate metrics, it is also informative to examine the agent's imaging behavior, particularly repeated imaging of the same targets. While multiple images can help reduce uncertainty, excessive or closely timed repetitions may provide limited additional value. Figure 7 shows the number of unique and duplicated images per episode, and Table 5 summarizes the temporal spacing of duplicates.

Policies trained with smaller λ values frequently re-image the same targets within a single orbit, indicating prioritization of perceived high-value targets at the expense of overall coverage. As λ increases, both the frequency of duplicate imaging and the proximity of repeated captures decrease, reflecting a more balanced strategy: uncertainty is reduced while spatial coverage is preserved. The *Multiply* formulation behaves similarly to *Additive* $\lambda = 3$ and $\lambda = 5$, producing few duplicates that are well-spaced temporally, demonstrating effective and strategic distribution of imaging actions. Conversely, when $\lambda \leq 1$, repeated captures occur frequently within the same orbit, providing limited new information. Larger λ or multiplicative rewards encourage more informative, efficient, and spatially distributed imaging strategies.

Table 5 Time gap distribution of repeated images per policy. Each cell shows the number of images and percentage within each orbit-relative bin.

Time Gap	$\lambda = 0$	$\lambda = 0.5$	$\lambda = 1$	$\lambda = 2$	$\lambda = 3$	$\lambda = 5$	*Multiply*
Current orbit	**15367 (85.6%)**	**12867 (82.7%)**	**7027 (70.5%)**	398 (11.3%)	73 (2.4%)	81 (2.7%)	74 (2.8%)
Next orbit	2391 (13.3%)	2488 (16.0%)	2730 (27.4%)	**2866 (81.3%)**	**2729 (89.5%)**	**2633 (88.3%)**	**2279 (87.5%)**
Two orbits ahead	194 (1.1%)	188 (1.2%)	211 (2.1%)	262 (7.4%)	247 (8.1%)	269 (9.0%)	252 (9.7%)

V. Conclusion

This study investigated reinforcement learning (RL)-based approaches for satellite task scheduling under target location uncertainty, where successful imaging requires the satellite to dwell on a target for a specified duration to account for search time. Two key aspects were examined: (1) the influence of the training environment and uncertainty observation, and (2) the effectiveness of different reward function formulations.

The results show that policies trained with access to target uncertainty information consistently outperform those without it, highlighting the importance of incorporating uncertainty observations into both training and decision-making processes. Among the reward formulations evaluated, the *Multiply* approach—where the reward is defined as the product of a target's priority and its uncertainty—achieved the best balance between prioritizing high-value targets and reducing overall uncertainty. In contrast, the *Additive* formulation, where the reward is defined as the sum of priority and uncertainty with a tunable weighting coefficient λ, provides explicit control over this trade-off, which can be advantageous depending on specific mission objectives. These findings emphasize that both environmental realism and careful reward design are critical for developing effective RL-based satellite task scheduling policies under uncertainty.

In this study, only the search duration was considered, with simplifying assumptions about how targets are imaged. Future work will incorporate more realistic search strategies into the simulation, as well as more complex scenarios such as cooperative multi-satellite constellations collaboratively imaging targets with evolving uncertainty. Exploring adaptive or context-aware reward structures that dynamically adjust based on mission priorities and uncertainty levels could further enhance policy robustness. Finally, integrating real-world data on target motion and environmental effects will strengthen validation and demonstrate the applicability of these approaches to operational satellite missions.

Acknowledgments

This work utilized the Alpine high-performance computing resource, jointly funded by the University of Colorado Boulder, the University of Colorado Anschutz, Colorado State University, and the National Science Foundation (Award No. 2201538). Artificial intelligence tools were used to help improve the grammar and clarity of the manuscript. Final editing and approval were performed by the authors.

References

[1] Bianchessi, N., and Righini, G., "Planning and Scheduling Algorithms for the COSMO-SkyMed Constellation," Vol. 12, No. 7, 2008, pp. 535–544. https://doi.org/10.1016/j.ast.2008.01.001.

[2] Lemaître, M., Verfaillie, G., Jouhaud, F., Lachiver, J.-M., and Bataille, N., "Selecting and Scheduling Observations of Agile Satellites," Vol. 6, No. 5, 2002, pp. 367–381. https://doi.org/10.1016/S1270-9638(02)01173-2.

[3] Wang, X., Wu, G., Xing, L., and Pedrycz, W., "Agile Earth Observation Satellite Scheduling Over 20 Years: Formulations, Methods, and Future Directions," Vol. 15, No. 3, 2021, pp. 3881–3892. https://doi.org/10.1109/JSYST.2020.2997050.

[4] Herrmann, A., and Schaub, H., "Reinforcement Learning for the Agile Earth-Observing Satellite Scheduling Problem," 2023, pp. 1–13. https://doi.org/10.1109/TAES.2023.3251307.

[5] NASA Earth Science Division, "Earth Science to Action Strategy," , 2023. URL https://science.nasa.gov/earth-science/earth-science-to-action/.

[6] Candela, A., Swope, J., and Chien, S. A., "Dynamic Targeting to Improve Earth Science Missions," *Journal of Aerospace Information Systems*, Vol. 20, No. 11, 2023, pp. 679–689. https://doi.org/10.2514/1.I011233, URL https://doi.org/10.2514/1.I011233.

[7] Cheval, A., and Schaub, H., "Autonomous Strip Imaging Task Scheduling In Super-Agile Satellites Using Reinforcement Learning," *AAS/AIAA Astrodynamics Specialist Conference*, Boston, MA, 2025. Paper No. AAS 25-726.

[8] Du, B., Li, S., She, Y., Li, W., Liao, H., and Wang, H., "Area targets observation mission planning of agile satellite considering the drift angle constraint," *Journal of Astronomical Telescopes, Instruments, and Systems*, Vol. 4, No. 4, 2018, p. 047002. https://doi.org/10.1117/1.JATIS.4.4.047002, URL https://doi.org/10.1117/1.JATIS.4.4.047002.

[9] Qin, M., Xu, Z., Zhao, X., Sun, W., Xie, W., and Liu, Q., "A Unified Scheduling Model for Agile Earth Observation Satellites Based on DQG and PPO," *Aerospace*, Vol. 12, No. 9, 2025. https://doi.org/10.3390/aerospace12090844, URL https://www.mdpi.com/2226-4310/12/9/844.

[10] Gabrel, V., Moulet, A., Murat, C., and Paschos, V. T., "A New Single Model and Derived Algorithms for the Satellite Shot Planning Problem Using Graph Theory Concepts," Vol. 69, No. 0, 1997, pp. 115–134. https://doi.org/10.1023/A:1018920709696.

[11] Valicka, C. G., Garcia, D., Staid, A., Watson, J.-P., Hackebeil, G., Rathinam, S., and Ntaimo, L., "Mixed-Integer Programming Models for Optimal Constellation Scheduling given Cloud Cover Uncertainty," Vol. 275, No. 2, 2019, pp. 431–445. https://doi.org/10.1016/j.ejor.2018.11.043.

[12] Stephenson, M., Quevedo Mantovani, L., and Schaub, H., "Learning Policies for Autonomous Earth-Observing Satellite Scheduling over Semi-Markov Decision Processes," *Journal of Aerospace Information Systems*, Vol. 22, No. 9, 2025, pp. 741–803. https://doi.org/10.2514/1.I011649.

[13] Li, Y., Xu, M., and Wang, R., "Scheduling Observations of Agile Satellites with Combined Genetic Algorithm," *Third International Conference on Natural Computation (ICNC 2007)*, IEEE, 2007, pp. 29–33. https://doi.org/10.1109/ICNC.2007.652.

[14] Chao, T., Han, X., Li, X., and Yang, M., "Multi-Objective Optimization of Continuous Monitoring Scheduling for Moving Targets by Earth Observation Satellites," Vol. 144, 2025, p. 110056. https://doi.org/10.1016/j.engappai.2025.110056.

[15] Morgan, S. J., McGrath, C. N., and De Weck, O. L., "Optimization of Multispacecraft Maneuvers for Mobile Target Tracking from Low Earth Orbit," Vol. 60, No. 2, 2023, pp. 581–590. https://doi.org/10.2514/1.A35457.

[16] Wen, Z., Liu, Y., Zhang, S., and Hu, H., "Task Scheduling Method of Revisit Tasks for Satellite Constellation towards Wildfire Management," 2024. https://doi.org/10.22541/au.172487479.92448519/v1.

[17] Pearl, B. D., Miller, J. M., and Lee, H. W., "Developing the Reconfigurable Earth Observation Satellite Scheduling Problem," *AIAA SCITECH 2025 Forum*, 2025. https://doi.org/10.2514/6.2025-0589.

[18] Fukunaga, A., Rabideau, G., Chien, S., and Yan, D., "Towards an Application Framework for Automated Planning and Scheduling," *1997 IEEE Aerospace Conference*, IEEE, 1997, pp. 375–386 vol.1. https://doi.org/10.1109/AERO.1997.574426.

[19] Lu, J., Chen, Y., and He, R., "A Learning-Based Approach for Agile Satellite Onboard Scheduling," Vol. 8, 2020, pp. 16941–16952. https://doi.org/10.1109/ACCESS.2020.2968051.

[20] Harris, A., Valade, T., Teil, T., and Schaub, H., "Generation of Spacecraft Operations Procedures Using Deep Reinforcement Learning," Vol. 59, No. 2, 2022, pp. 611–626. https://doi.org/10.2514/1.A35169.

[21] Hadj-Salah, A., Verdier, R., Caron, C., Picard, M., and Capelle, M., "Schedule Earth Observation Satellites with Deep Reinforcement Learning," 2019. https://doi.org/10.48550/ARXIV.1911.05696.

[22] Mantovani, L. Q., Nagano, Y., and Schaub, H., "Reinforcement Learning for Satellite Autonomy Under Different Cloud Coverage Probability Observations," *AAS Astrodynamics Specialist Conference*, Broomfield, CO, 2024. Paper No. AAS 24-208.

[23] Hadj-Salah, A., Guerra, J., Picard, M., and Capelle, M., "Towards operational application of Deep Reinforcement Learning to Earth Observation satellite scheduling," , Aug. 2020. URL https://hal.science/hal-02925740, working paper or preprint.

[24] Stephenson, M., and Schaub, H., "Scalable Autonomous Decentralized Constellation Tasking on Asynchronous Semi-Markov Decision Processes," *International Workshop on Satellite and Constellations Formation Flying*, Kaohsiung, Taiwan, 2024.

[25] Nagano, Y., and Schaub, H., "Enhancing Fault Resilience In Rl-Based Satellite Autonomous Task Scheduling," *AAS/AIAA Astrodynamics Specialist Conference*, Boston, MA, 2025. Paper No. AAS 25-637.

[26] Tangpattanakul, P., Jozefowiez, N., and Lopez, P., "Multi-objective Optimization for Selecting and Scheduling Observations by Agile Earth Observing Satellites," *Parallel Problem Solving from Nature - PPSN XII*, edited by C. A. C. Coello, V. Cutello, K. Deb, S. Forrest, G. Nicosia, and M. Pavone, Springer Berlin Heidelberg, Berlin, Heidelberg, 2012, pp. 112–121.

[27] Li, L., Yao, F., Jing, N., and Emmerich, M., "Preference incorporation to solve multi-objective mission planning of agile earth observation satellites," *2017 IEEE Congress on Evolutionary Computation (CEC)*, 2017, pp. 1366–1373. https://doi.org/10.1109/CEC.2017.7969463.

[28] Stephenson, M. A., and Schaub, H., "BSK-RL: Modular, High-Fidelity Reinforcement Learning Environments for Spacecraft Tasking," *75th International Astronautical Congress*, IAF, Milan, Italy, 2024.

[29] Kenneally, P. W., Piggott, S., and Schaub, H., "Basilisk: A Flexible, Scalable and Modular Astrodynamics Simulation Framework," Vol. 17, No. 9, 2020, pp. 496–507. https://doi.org/10.2514/1.I010762.

[30] Liang, E., Liaw, R., Nishihara, R., Moritz, P., Fox, R., Goldberg, K., Gonzalez, J., Jordan, M., and Stoica, I., "RLlib: Abstractions for Distributed Reinforcement Learning," *Proceedings of the 35th International Conference on Machine Learning*, Proceedings of Machine Learning Research, Vol. 80, edited by J. Dy and A. Krause, PMLR, 2018, pp. 3053–3062. URL https://proceedings.mlr.press/v80/liang18b.html.

[31] Herrmann, A., and Schaub, H., "A Comparative Analysis of Reinforcement Learning Algorithms for Earth-Observing Satellite Scheduling," Vol. 4, 2023, p. 1263489. https://doi.org/10.3389/frspt.2023.1263489.