AAS 26-052

AUTONOMOUS SPACE-BASED IMAGING: REINFORCEMENT LEARNING SCHEDULING WITH IMAGING–DOWNLINK TRADEOFFS ACROSS ORBITAL REGIMES

Daniel Huterer Prats* **and Hanspeter Schaub†**

The rapid growth of resident space objects (RSOs) is increasing demands on space situational awareness (SSA) and space domain awareness (SDA). Space-based space surveillance (SBSS) can provide timely, illuminated observations, but scheduling is challenging due to rapidly changing line-of-sight and illumination, tight pointing constraints, emergence of short-notice targets, and limited onboard resources such as energy, momentum, and storage. This paper researches a resource-aware reinforcement learning (RL) scheduler for a single low Earth orbit (LEO) platform that images targets across LEO, medium Earth orbit (MEO), and geosynchronous Earth orbit (GEO), a "LEO-to-any" setting. The problem is posed as a partially observable Markov decision process in which the agent selects among target imaging, downlink, charging, and momentum desaturation actions. A tunable objective combines imaging reward and downlink reward through a mixing parameter α that controls the trade between imaging and delivery. Monte Carlo results show that a policy trained in a LEO-only catalog transfers to a "LEO-to-any" mixed-regime catalog without retraining, maintaining stable acquisition performance while improving total return and illuminated-image throughput. Analysis of actions during eclipse periods further indicates eclipse-effective behavior, with most eclipse-time imaging decisions selecting illuminated targets or substituting higher-altitude targets, and LEO selections exhibiting a measurable sunward pointing bias in Hill coordinates. These results support scalable multi-regime autonomous SBSS scheduling that balances collection against timely ground delivery under realistic resource and geometry constraints.

INTRODUCTION

The rapid growth in resident space objects (RSOs), fueled by the deployment of numerous low Earth orbit (LEO) constellations and reduced barriers to space access, has led to a surge in cataloged space assets. Estimates indicate over 9700 active satellites currently orbiting in LEO, with projections of tens of thousands more in the next decade.[1,2] This proliferation places increasing stress on space situational awareness (SSA) and space domain awareness (SDA) architectures tasked with maintaining safe and sustainable operations. For decades, satellite tracking and imaging have been predominantly conducted from ground-based telescopes and radar networks, which provide mature, high-quality data but are subject to weather, atmospheric scintillation, seeing, and nighttime-only optical access. These environmental and geometric constraints can limit optical systems to operability fractions as low as 25% in some locations.[3]

Space-based space surveillance (SBSS) has long been recognized as a compelling complement to ground-based sensing.[4,5] By operating above the atmosphere, space-based optical sensors are not limited by clouds, can observe closer to bright bodies such as the Sun and Moon, and over the

*Ph.D. Student, Department of Aerospace Engineering Sciences, University of Colorado Boulder, Boulder, Colorado 80303

†Distinguished Professor and Department Chair, Department of Aerospace Engineering Sciences, University of Colorado Boulder, Boulder, Colorado 80303, Fellow of AAS and AIAA.

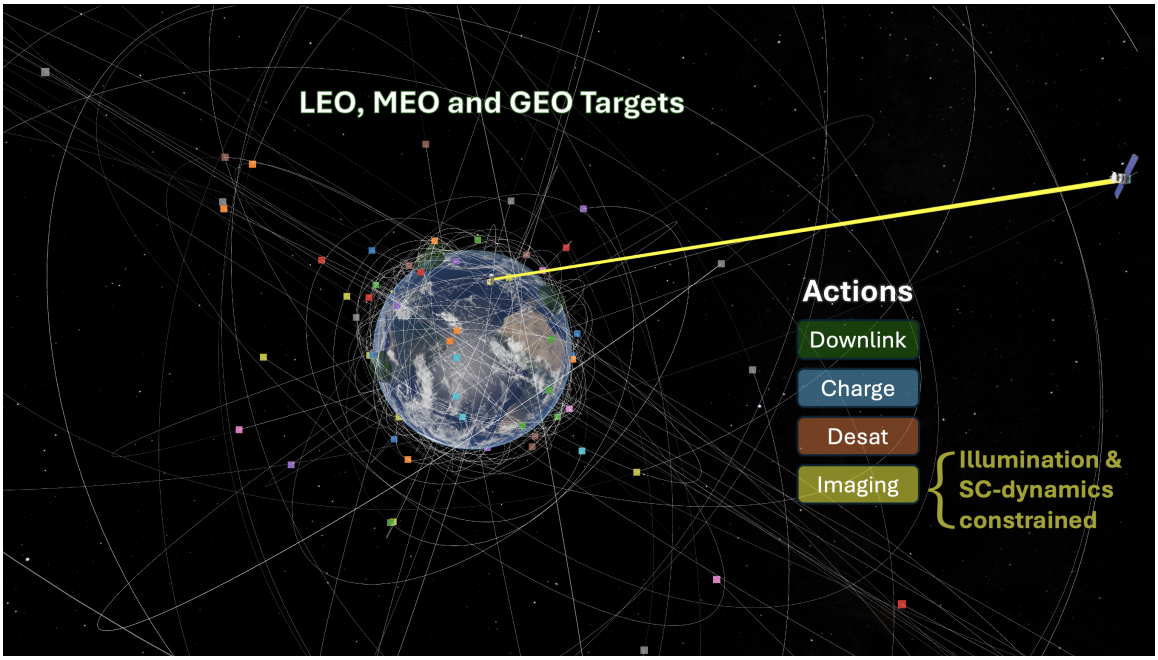

Figure 1 Space-based RSO imaging under illumination and spacecraft dynamics/sensor constrains. LEO-platform imaging targets in LEO, MEO and GEO orbit.

course of an orbit can achieve an effective field of regard that covers the entire sky.[6] Recent SBSS architecture studies have examined how single or distributed space-based platforms can detect and track debris and satellites in GEO, MEO, and LEO, quantifying detection limits, cataloging performance, and revisit statistics for different orbital configurations and optical payloads.[7,8] These works highlight both the geometric advantages of SBSS and the operational challenges that arise once the sensor itself becomes a rapidly moving observer with its own lighting, pointing, and communication constraints.

A parallel line of research focuses on orbit determination (OD) from short-arc optical measurements collected by narrow field-of-view (FoV) space-based or ground-based sensors. The short duration of these passes, combined with high relative angular rates, yields significant initial state uncertainty.[9,10] To address this, a variety of estimation techniques have been explored, including admissible-region–based initial orbit determination, batch and sequential estimators, genetic algorithms, and multiple shooting methods.[11–14] This paper assumes that target orbits are known with sufficient fidelity for tasking and focuses instead on the onboard scheduling problem: given a catalog of RSOs with known orbits, how should a space-based inspector allocate limited pointing, energy, and downlink resources over time to maximize mission value?

Tasking and scheduling of sensors for SSA/SBSS have been shown to be NP-hard, with strong analogy to the vehicle routing problem with time windows.[15] Consequently, exact mixed-integer formulations and constraint satisfaction approaches[16] can only guarantee optimality for small problem instances within a reasonable timeframe. To improve scalability, numerous heuristic and metaheuristic schedulers have been proposed, including greedy policies,[17] multi-objective genetic algorithms (e.g., NSGA-II) for multi-sensor, multi-regime scenarios,[18] and auction or information-

gain–driven schedulers.[19] While these methods can produce high-quality schedules, they typically must be resolved whenever new targets or updated orbital data become available, limiting their responsiveness in highly dynamic environments and making it difficult to consistently account for detailed spacecraft subsystem constraints.

Reinforcement learning (RL) offers an alternative by learning reusable policies that generalize across scenarios and enable fast, reactive decision-making at run time.[20] In the SSA context, RL has been used to schedule ground-based sensor networks, where agents learn to select which ground telescopes or radars should track which RSOs over time.[21–25] Beyond SSA, RL and deep RL have shown potential to solve the Earth observation (EO) satellite scheduling problem, where targets are stationary or slow-moving and the agent manages trade-offs between imaging opportunities, energy, momentum, and onboard storage.[26–33] In these EO settings, the orientation of the imaging satellite is predominantly one-way (looking down toward Earth and slewing along and across track), which simplifies access modeling and leads to more predictable imaging windows in terms of illumination constraints.

In contrast, space-based sensor tasking for space surveillance is inherently more challenging. Targets span multiple orbital planes and regimes, and their relative motion with respect to a LEO inspector is rapid and varies substantially across LEO, MEO, and GEO. Consequently, usable imaging windows are brief, highly geometry dependent, and tightly constrained by illumination. Early RL-based SBSS work used a LEO-based sensor to image GEO targets, optimizing metrics such as mean trace covariance over the catalog while benefiting from the quasi-stationary nature of GEO orbits.[34] That formulation focused on target selection and did not model realistic spacecraft subsystem or safety constraints such as battery or data management, reaction-wheel momentum buildup and desaturation, or explicit downlink operations. More recently, a LEO-to-LEO baseline was demonstrated using a high-fidelity simulator with flight-like attitude, power, and communication models and a shared safety shield.[35] That work trained a Proximal Policy Optimization (PPO)[36] policy to schedule imaging and eclipse-time downlink under realistic line-of-sight (LOS) and illumination constraints, but focused on a single orbital regime and used a single variable objective function.

However, many SSA and EO missions must balance multiple competing objectives, such as maximizing image collection, optimizing timely delivery of data to the ground, and preserving energy and momentum margins for future operations. Multi-objective reinforcement learning (MORL) provides a formal framework for such problems, using scalarization or Pareto-based methods to trade different reward components.[37,38] In satellite scheduling, MORL and deep RL have been explored for agile EO satellites to jointly optimize observation yield and other mission metrics under resource constraints.[39] Yet, to the best of the authors' knowledge, MORL has not been systematically applied to space-based SSA in a way that explicitly trades off image acquisition against downlink timeliness, nor has it been combined with a unified, multi-regime (LEO–MEO–GEO) SBSS scheduling environment with flight-like platform constraints.

The present paper builds on the prior LEO-to-LEO RL baseline[35] and advances SBSS autonomy along two main axes:

1. **Multi-regime scheduling ("LEO-to-any"):** The environment is extended so that a single LEO platform can simultaneously schedule imaging of RSOs in LEO, MEO, and GEO, in addition to isolated LEO-to-MEO and LEO-to-GEO cases. This enables a unified assessment of how orbital regime affects lighting bias, day/night cadence, and attainable autonomy metrics relative to the LEO-only baseline.

2. **Reward-aware imaging vs. delivery:** A tunable, multi-objective reward formulation is introduced that explicitly trades illuminated image acquisition against timely downlink to the ground. A scalarization parameter α balances image and downlink rewards, enabling a reward trade study that exposes how the learned policy reallocates effort among imaging, downlink, charging, and desaturation as mission objectives shift from collection-heavy to latency-sensitive regimes.

All experiments are conducted in a Basilisk [*] and BSK-RL [†] simulation environment with coupled orbit, attitude, power, and communication dynamics, using PPO agents shielded by safety logic that enforces hard limits on battery state-of-charge, data storage usage, and reaction-wheel momentum.[40,41] The resulting performance shows that a LEO-only trained policy is well capable to adapting to an environment with RSOs in multiple orbit regimes. Moreover, the α tuning parameter can be used to significantly impact the delivery time of images from the spacecraft to the ground and change the behavior of the policy.

SPACE-BASED SPACE SURVEILLANCE ENVIRONMENT

Space-based SSA departs significantly from the well studied agile Earth observation satellite (AEOS) setting commonly used in RL scheduling studies.[32,33,42] In AEOS, targets are fixed or slow-moving ground locations causing visibility windows to be relatively long and predictable. In contrast, a space-to-space imaging platform must contend with rapidly evolving relative motion between the imaging spacecraft and the RSOs, including targets in different planes and orbital regimes (LEO, MEO, GEO). Targets can enter and exit the field of view (FOV) from any direction, with imaging opportunities varying quickly due to lighting, LOS constraints, and the spacecraft's own attitude and resource state.

Compared to the prior work,[35] which considered a single LEO inspector imaging LEO RSOs, this work extends the environment in two key ways:

1. The RSO catalog now spans multiple orbital regimes, enabling a single LEO platform to schedule imaging of targets in LEO, MEO, and GEO ("LEO-to-any"), as well as LEO-to-MEO and LEO-to-GEO subsets. Adaptability is demonstrated by training policies in a LEO-only environment and deploying them, without retraining, in a mixed-regime LEO-to-any setting for evaluation.

2. The reward function is decomposed into separate terms for illuminated image acquisition and timely downlink, which are combined via a tunable parameter α to study reward-aware tradeoffs between collection and delivery. As detailed in section , a range of alpha values is tested to study policies across the entire spectrum of the tradeoff balance between imaging and downlinking.

The underlying spacecraft model, controller, and message-passing architecture in Basilisk are kept consistent with the baseline so that observed changes in behavior can be attributed to multi-regime geometry and reward balancing rather than to platform modeling differences.

[*] https://avslab.github.io/basilisk/
[†] https://avslab.github.io/bsk_rl/

Simulation Environment Overview

The core environment is implemented in Basilisk which is a fast, physics-based and flight-proven spacecraft simulation environment. Basilisk uses a modular, message-passing architecture to assemble closed-loop simulations of multi-physics spacecraft systems. In this SBSS configuration, a single LEO imaging spacecraft is simulated together with an RSO catalog and a ground-station network. Each episode spans several orbits and is discretized into decision steps at which the RL agent selects a high-level action. Between decisions points, Basilisk propagates the coupled orbit, attitude, power, and data-handling dynamics.

The LEO imaging spacecraft is placed in a near-circular 500 km orbit, consistent with the earlier LEO-to-LEO study. The RSO catalog contains multiple shells: a dense LEO shell, a sparser MEO shell, and GEO ring targets.

Table 1 Mixed-regime RSO catalog distribution and orbital-element sampling bounds.

Regime	Mix weight	Altitude range h [km]	e range	i range [deg]
LEO	50%	400–2000	0–0.02	0–180
MEO	30%	2000–35000	0–0.10	0–120
GEO	20%	35486–36086	0	0–15

Notes: h is sampled uniformly above mean Earth radius ($R_E = 6371$ km). LEO/MEO draws reject samples until $a(1-e) - R_E \geq 400$ km.

Orbital elements are randomly drawn within regime-specific bounds to create diverse viewing geometries and lighting cadences. A set of fixed ground stations provides downlink opportunities and the visibility to these stations is computed using standard slant-range and horizon geometry. The exact orbital parameters and ground-station locations used in this work are summarized in the Appendix.

Illumination is not modeled as a simple binary shadow/light flag. Instead, a continuous eclipse factor allows partially eclipsed geometries to still receive reward. Figure 2 from[35] illustrates how LOS and eclipse geometry combine to determine whether an imaging attempt receives full or zero reward.

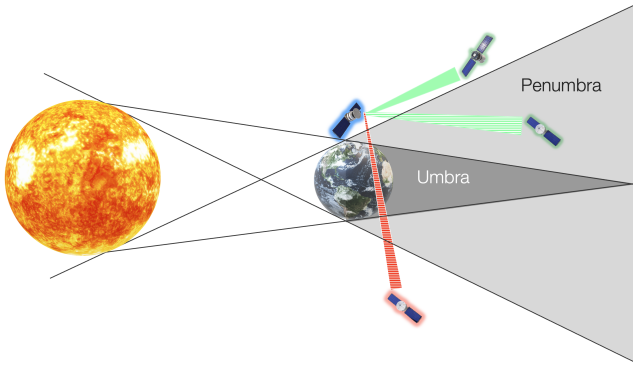

Figure 2 Space-based RSO imaging under eclipse and LOS restrictions. Green: fully illuminated; shaded: partially illuminated; red: LOS violation.

At each environment step, RSOs whose elevation satisfy visibility criteria are potentially considered candidates. The environment then constructs an observation vector for the RL agent (described

in Section) and applies the chosen high-level action (image, downlink, charge, desaturate) to propagate the environment accordingly, until that step is concluded.

Episodes are finite-horizon, typically spanning the equivalent of 150 imaging actions worth of several orbits of the LEO inspector. The horizon is chosen so that, in the absence of resource constraints, the spacecraft would have at least one opportunity to view most or all RSOs in the catalog. For a catalog of $\mathcal{T}$ targets the episode duration is set to approximately 150% of the time needed to image $\mathcal{T}$ targets back-to-back, following the design of the prior LEO-only study.[35]

Spacecraft Dynamics, Power, and Control

The inspector spacecraft configuration follows the AEOS-style platform used in prior Basilisk and BSK-RL studies.[32,43] The rigid-body dynamics, reaction-wheel assembly, and power system are modeled explicitly. Key properties include:

- A 330 kg spacecraft with a diagonal inertia matrix representative of a small agile observatory.
- Three orthogonal reaction wheels with realistic torque and speed limits, enforcing actuator saturation and the need for periodic momentum desaturation.
- A finite-capacity battery and body-fixed solar arrays, with power generation dependent on attitude and eclipse state and power consumption driven by base loads, imaging payload operation, transmitter use, and desaturation events.

Attitude control is provided by Basilisk's `mrpFeedback` controller,[‡] an MRP-based feedback law implemented as in Schaub and Junkins.[44] A guidance module computes the desired attitude to track a selected target or ground station and then the controller drives the body toward this reference while respecting reaction-wheel limits. The closed-loop architecture mirrors the previous LEO-to-LEO work and is illustrated in Fig. 8 (Appendix).

To ensure that the RL agent must actively manage energy and storage, the battery state of charge is initialized in a mid-range band (25–50% of capacity), and the data buffer is sized to hold only a fraction of the images that would be required to image the entire catalog. This prevents trivial solutions (e.g., imaging everything without needing to charge or downlink) and forces the policy to trade imaging against charge and downlink actions. A detailed list of spacecraft and controller parameters used is given in.[35]

Targeting, Imaging, and Downlink Constraints

At each decision point, the environment identifies unimaged (or unsuccessfully imaged) RSOs within the current field of regard and adds those to the current potential target list. This list is subsequently organized in increasing order of elevation and then size adjusted for an appropriate dimension of the observation space described in .

For a successful `Image` action, additional constraints must hold at the time of capture:

[‡] `https://avslab.github.io/basilisk/Documentation/fswAlgorithms/attControl/mrpFeedback/mrpFeedback.html`

- **Attitude error and rate:** the MRP attitude error and attitude rate error must be below a specified threshold.
- **Illumination:** the target must be sufficiently illuminated, and quantified using a continuous eclipse illumination factor $s_i \in [0, 1]$ that also considers penumbra. Only images taken when $s_i > e_{\text{thresh}}$ are treated as sufficiently illuminated in the reward function.
- **Timing:** the spacecraft must satisfy the above conditions for at least one full iteration of the flight software evaluation during the imaging action.

Each successful capture consumes a fixed amount of onboard storage, regardless of illumination quality, reflecting the assumption that images are not processed or filtered onboard. Downlink actions point the spacecraft toward a visible ground station and transmit data at a fixed baud rate subject to access constraints. Storage is freed as data is downlinked, but downlink also consumes power and occupies attitude and pointing time that could otherwise be used for imaging or charging. The key targeting, imaging, and downlink parameters are summarized in Table 8.

Problem Objective

From the environment perspective, the high-level objective is to:

- maximize the number of RSOs that are both imaged under valid illumination and successfully downlinked to the ground, and
- maintain safe operation with respect to battery state-of-charge, reaction-wheel momentum, and storage capacity.

In the earlier LEO-only work, this objective was expressed as maximizing the fraction of illuminated targets imaged out of all targets, $\mathcal{I}_{\text{ill}}/\mathcal{T}$, under hard constraints. That environment-level objective is retained here but is now coupled to a multi-objective RL reward: separate reward components R_{image} and R_{downlink} are combined via the scalarization

$$J(\alpha, R_{\text{image}}, R_{\text{downlink}}) = (1 - \alpha)\, R_{\text{image}} + \alpha\, R_{\text{downlink}}, \qquad \alpha \in [0, 1].$$

Episodes terminate either when the horizon is reached, all targets are imaged or when critical safety limits (e.g., depleted battery, unrecoverable wheel saturation) are violated, encouraging policies that remain resource-aware and operationally safe while exploiting the rich, multi-regime SBSS geometry.

The key challenges of this SBSS task include:

- adapting to orbital dynamics (including elliptical target orbits) and rapidly changing eclipse conditions;
- operating under limited energy, momentum, and data storage;
- respecting geometric constraints (LOS, FOV, attitude, and rate limits);
- balancing image collection against timely downlink under a tunable, multi-objective reward.

REINFORCEMENT LEARNING SETUP

Autonomous space-based tasking in a dynamic orbital environment is a natural setting for reinforcement learning (RL), where an agent must make sequential decisions to maximize long-term returns under uncertainty.[20] In this work, the autonomous imaging and downlink scheduling problem for SBSS is formulated as a partially observable Markov decision process (POMDP). The POMDP framework allows the policy to reason over incomplete state information, adapt to changing target visibility and lighting, and manage limited onboard resources while satisfying operational constraints.

A POMDP is defined by a state space $\mathcal{S}$, action space $\mathcal{A}$, transition function $\mathcal{T}$, reward function $\mathcal{R}$, observation space $\mathcal{O}$, and observation model $\mathcal{Z}$. Because the true simulator state is not directly observable, the agent receives observations $\boldsymbol{o}_k \in \mathcal{O}$ at discrete decision times t_k and selects actions $a_k \in \mathcal{A}$ to maximize the expected discounted sum of future rewards

$$\mathbb{E}\left[\sum_{k=0}^{T} \gamma^k r_k\right], \tag{1}$$

where $\gamma \in [0, 1)$ is the discount factor, r_k is the scalar reward at decision k, and T is the episode horizon. Due to the varying step durations across the four different actions the problem is structured as a semi-POMDP, where the discount factor γ is not a per-step discount factor but rather per-second. This is done to account for the different time-opportunity cost of each action. The formulation of the semi-POMDP is done as in equation (5) of Reference 42.

POMDP Formulation

The elements of the POMDP tuple for the SBSS task are:

State space: The underlying simulator state in Basilisk provides the generative model for the Markov process and includes:

- inspector orbit and attitude states
- RSO orbital states (LEO/MEO/GEO) and priorities
- internal subsystem states (battery state-of-charge, data storage, wheel momentum)
- environmental states (illumination factors, eclipse intervals, ground-station visibility).

This full state is not exposed directly to the agent, but is used to compute observations, rewards, and transitions.

Observation space: The agent receives a compact, normalized observation vector containing quantities relevant to scheduling and resource management. The observation design is intentionally regime-agnostic: the policy is not given explicit labels indicating whether a target is in LEO, MEO, or GEO. Hence, regime-specific behavior arises solely through geometric and lighting differences.

Table 2 Observation space elements provided to the agent at each decision step.

Category	Element	Description	Dim.
Spacecraft	s_{data}	Fraction of onboard data storage used	1
	s_{batt}	Normalized battery state-of-charge	1
	s_{mom}	Normalized wheel momentum magnitude	1
Targets	$\boldsymbol{\epsilon}_i$	Elevation angles of candidate targets i	N
	$\boldsymbol{r}_{BR,i}^{\mathcal{H}}$	Relative position to target i in Hill frame	$3N$
	$\boldsymbol{\phi}_i$	Boresight–target angles for target i	N
	$\boldsymbol{d}_i$	Distance to target i	N
	$\boldsymbol{s}_i$	Illumination factors for target i	N
Environment	$\boldsymbol{e}_{\text{start}}, \boldsymbol{e}_{\text{end}}$	Normalized eclipse start/end times	2
	$\boldsymbol{g}_{\text{open}}, \boldsymbol{g}_{\text{close}}$	Normalized GS window open/close times	$2 \times N_{\text{GS}}$

At each decision step, the environment constructs the observation from global spacecraft features and a subset of candidate targets. Table 2 summarizes the observation elements.

A key design choice is that only the top $N = 10$ candidate RSOs are provided at each step. These are selected from all LOS-valid targets and sorted by *ascending* local elevation angle. Thus, when more than 10 RSOs are visible, the observation favors targets closer to the horizon. In multi-regime catalogs, this mechanism tends to expose more LEO targets in the earlier episode phase (which sweep rapidly through low and mid elevations) and can, in dense scenes, temporarily exclude some high-elevation MEO/GEO targets from the candidate set. Nonetheless, further-away MEO and GEO targets would still appear with lower elevation angles and be considered. As a result, a mild preference for LEO targets observed in the early phase may be interpreted primarily as a consequence of the candidate-selection heuristic, not as an inherent bias arising from the fact that the policy was trained in a LEO-only environment. Over the course of the episode, higher elevation targets have been observed to enter the list of $N = 10$ candidates.

Action space: At each decision step, the agent chooses one action from a discrete set:

- `Image`(i): slew and settle to image candidate target $i \in \{1, \ldots, N\}$ for a fixed duration (Δt =300 seconds)
- `Charge`: orient for solar charging for a fixed interval (Δt =300 seconds)
- `Downlink`: point toward a visible ground station and transmit stored data (Δt =180 seconds)
- `Desat`: perform a momentum desaturation maneuver (Δt =150 seconds).

Transition model: The transition model is implemented as a deterministic generative simulator. Given the current state and chosen action, Basilisk propagates the environment for a period of $\Delta t =$ (based on the chosen action):

- inspector orbit and attitude,
- RSO motion in their respective orbits,

- battery state-of-charge and power flows,
- data storage usage and downlink throughput,
- eclipse status and LOS to RSOs and ground stations.

The next observation and reward are computed from the resulting state.

Reward Decomposition and α Trade Study

To capture the competing operational goals of maximizing illuminated image collection and maximizing timely delivery of useful imagery to the ground, a scalarized multi-objective reward of the form

$$J(\alpha) = (1-\alpha)\, R_{\text{image}} + \alpha\, R_{\text{downlink}}, \qquad \alpha \in [0,1], \tag{2}$$

is adopted, where R_{image} measures illuminated image acquisition and R_{downlink} measures delivery of those images to the ground. This is a standard linear scalarization in multi-objective RL,[37,38] and sweeping α yields a family of policies ranging from imaging-heavy to delivery-heavy mission preferences.

At each decision step k, the agent selects an action and the environment updates the onboard image buffer. The episode return is scalarized using (2), where the imaging term rewards the acquisition of an illuminated image and the downlink term rewards the delivery of illuminated images to the ground. For the purposes of this paper, targets are equally weighted with $w_i = 1$ for all $i \in \mathcal{T}$.

Imaging component: Each `Image` action targets a single candidate RSO, so at most one new image is generated per decision. Let i_k denote the selected target at step k. Executing `Image` produces one stored image associated with target i_k provided that line-of-sight and pointing constraints are met. However, only sufficiently illuminated captures contribute to reward. An illuminated-acquisition reward is defined as

$$r_k^{\text{image}} = \begin{cases} w_{i_k}, & \text{if } \mathcal{C}_{\text{img}}(i_k) \text{ holds}, \\ 0, & \text{otherwise}, \end{cases} \tag{3}$$

so that an imaging step yields $(1-\alpha)\, r_k^{\text{image}}$ in (2).

The $\mathcal{C}_{\text{img}}(i)$ denotes the conjunction of the imaging conditions

$$\begin{aligned} \mathcal{C}_{\text{img}}(i): \quad & s_i > e_{\text{thresh}}, \\ & \text{LOS}_i = 1, \\ & \left\|\boldsymbol{\sigma}_{B/R}(i)\right\| \le \sigma_{\text{img}}, \\ & \left\|\boldsymbol{\omega}_{B/R}(i)\right\| \le \omega_{\text{img}}, \end{aligned} \tag{4}$$

with $\sigma_{\text{img}} = 0.0025$ and $\omega_{\text{img}} = 0.01$ rad/s.

Here, w_i is the priority weight of target i, s_i is its illumination factor with threshold e_{thresh}, LOS_i indicates unobstructed line-of-sight, $\boldsymbol{\sigma}_{B/R}$ is the MRP attitude error between the spacecraft body frame and the target-pointing reference frame, and $\boldsymbol{\omega}_{B/R}$ is the corresponding body-rate error. The attitude-error bound $\|\boldsymbol{\sigma}_{B/R}\| \le 0.0025$ corresponds to a principal rotation error $\phi \le 0.573°$, and

the rate-error bound is $\|\boldsymbol{\omega}_{B/R}\| \leq 0.01$ rad/s. Compared to,[35] these pointing constraints have been significantly tightened, which is also reflected in longer acquisition times needed during each imaging action in order to hone in on the target precisely enough.

Importantly, the act of imaging can still generate data even when $s_{i_k} \leq e_{\text{thresh}}$, simulating an image with insufficient target illumination. These special images occupy onboard storage but are treated as "non-useful" and do not increase r_k^{image}.

Downlink component: A `Downlink` action can transmit multiple buffered images within a single decision interval, limited by the instantaneous link availability and data rate. Let $\mathcal{B}_k$ denote the set of images in the onboard buffer at the start of step k, and let $\mathcal{D}_k \subseteq \mathcal{B}_k$ denote those that are successfully delivered during the action. Because only illuminated images are mission-relevant for the objective, downlink reward is only given for images that satisfy the illumination criterion at acquisition. Using $\mathcal{D}_k^+ \subseteq \mathcal{D}_k$ to denote the subset of delivered images that are illuminated, then the downlink reward is formulated as

$$r_k^{\text{downlink}} = \sum_{i \in \mathcal{D}_k^+} w_i. \tag{5}$$

so that a downlink step yields $\alpha\, r_k^{\text{downlink}}$ in (2).

Although delivering non-illuminated images yields no direct reward, it can still be operationally beneficial because it frees storage capacity and prevents the buffer from saturating, thereby preserving the ability to acquire future illuminated images. Thus, the downlink action simultaneously serves as a reward-bearing delivery mechanism for useful data and as a housekeeping mechanism for buffer management.

The number of images delivered in $\mathcal{D}_k$ depends on the effective contact time within the fixed action duration (180 s). Transmission occurs only during propagation steps in which the spacecraft satisfies the necessary downlink conditions, including (i) a ground station in view above the elevation/range constraints and (ii) the required downlink attitude (nadir pointing in this implementation). As a result, the effective contact time may be substantially shorter than the nominal 180 s (for example if the access window opens late, closes early, or the spacecraft is still slewing into nadir), and hence the buffer is often not fully emptied in a single action.

Episodes in which the policy selects consecutive `Downlink` actions have also been observed. Because this behavior maintains the downlink attitude between decisions, it can reduce time lost due to repeated attitude maneuvers and increase the fraction of the action interval spent in a downlink-enabled state, effectively "pre-positioning" the spacecraft to exploit short or late-opening ground-station contacts.

Effect of α Parameter: The scalarization parameter α is introduced to modulate the relative emphasis between illuminated image acquisition and timely downlink within the reward function. Small values of α bias the reward toward image collection, such that illuminated acquisitions contribute nearly their full weight w_i, while downlink provides comparatively little immediate incentive unless driven by storage or safety constraints. Conversely, large values of α bias the reward toward delivery, downweighting image acquisition and increasing the incentive to transmit stored imagery during available ground-station contacts. Intermediate values of α are intended to balance these

competing objectives, allowing the policy to trade image collection against delivery in a manner consistent with mixed mission priorities.

Training Procedure

Given the high dimensionality of spacecraft dynamics and the complexity of the observation space, a deep RL using the PPO algorithm is used.[36] PPO is an actor–critic method in which the actor (policy network) maps observations to action distributions, and the critic (value network) estimates the expected return to guide policy updates. PPO constrains successive policy updates via a clipped surrogate objective, which helps maintain training stability by preventing overly large parameter changes between iterations. In this work, PPO is implemented using Ray RLlib framework, and modified to account for timestep-aware discounting as the problem is posed as a semi-POMDP.[45]

In this study, separate PPO policies are trained for each value of α on a LEO-only catalog, and then evaluated zero-shot on mixed LEO/MEO/GEO catalogs to assess both the imaging–delivery trade and cross-regime generalization under the candidate-target selection mechanism described in Section .

Compared to the AMOS study,[35] the training setup here is adapted to better support learning of successful downlinks, proactive charging behavior and to accommodate the multi-objective reward:

- **Training horizon and batch size:** Training runs are extended to approximately 48 hours wall-clock time per α value, with larger batch sizes of up to 5000. This increased sample budget allows the agent to more reliably discover and refine its own downlink and charging strategy, rather than relying on safety-shield interventions.

- **Safety shield usage:** The same safety shield logic is retained (e.g., forced `Charge` when battery falls below a critical threshold, forced `Downlink` when storage is nearly full), but the larger training batches and multi-objective reward lead to policies that invoke the shield less frequently than both heuristic schedulers and the earlier AMOS policies. Most charge and downlink actions become proactive decisions by the RL policy rather than emergency overrides.

- **Hyperparameter search:** A hyperparameter sweep is conducted over learning rate, discount factor, batch size, PPO clipping parameter, and network size. Unless otherwise noted, RLlib defaults are used outside this sweep. A time-discounted generalized advantage estimator is used, with the discount applied per second rather than per decision step to ensure fair credit assignment across actions with different durations.

Representative training hyperparameters are summarized in Table 9 in the Appendix. Training remains challenging due to the long horizons, multi-regime geometry, and multi-objective reward, however, the resulting policies demonstrate stable behavior across values of α and generalize well from LEO-only training to mixed LEO/MEO/GEO evaluation, with regime-dependent preferences shaped predominantly by geometry and the candidate-target selection mechanism rather than by explicit regime labels.

RESULTS

A reward trade study is conducted by sweeping the mixing parameter α from image-focused ($\alpha = 0$) to downlink-focused ($\alpha = 1$). The Monte Carlo statistics report the mean $\pm$ standard deviation across random seeds.

Robustness of a LEO-trained policy in mixed-regime deployment

To test whether a policy trained in a LEO-only catalog remains effective when deployed in a multi-regime setting, the same $\alpha = 0.5$ policy is used to evaluate on (i) a LEO-only environment and (ii) a mixed-regime environment. The mixed catalog is generated with regime sampling weights $(0.5, 0.3, 0.2)$ for LEO/MEO/GEO (Section).

Table 3 Policy robustness: the same 50d/50i policy ($\alpha = 0.5$) trained in LEO and evaluated in LEO vs. mixed catalogs. Values are mean $\pm$ std over $N = 100$ Monte Carlo seeds.

Env	Total reward	Illum. images	Illum. images downlinked	Downlink actions	Imaging actions
LEO	87.11 ± 3.08	87.52 ± 3.21	86.69 ± 3.11	28.94 ± 4.13	131.50 ± 2.90
MIXED	93.18 ± 2.56	93.41 ± 2.59	92.95 ± 2.61	35.26 ± 6.30	128.00 ± 3.99

Env	Frac. imaged LEO/MEO/GEO	Mean dt_{acq} [s]	Acq. success
LEO	$1.0/0.00/0.00$	148.88 ± 19.51	0.645 ± 0.038
MIXED	$0.59/0.22/0.19$	145.36 ± 24.58	0.666 ± 0.039

Table 3 shows that the policy remains stable under the distribution shift: the acquisition success rate and mean inter-acquisition time are similar across environments, and performance remains comparable when the policy is exposed to higher-altitude targets. In the mixed catalog the policy selects more downlink actions and achieves an increase in total return and delivered illuminated images over the fixed-horizon episodes.

Importantly, in the mixed setting the policy allocates imaging across regimes in roughly the same proportions as the catalog composition, indicating that the LEO-trained policy does not stick to exclusively imaging LEO targets once higher-altitude targets are present. That said, the policy under-selects MEO targets relative to their 30% presence in the scenario (Table 3: 0.22 imaged fraction versus 0.30 in the catalog). A plausible explanation is that the MEO subset in this study contains the most eccentric orbits, which can induce larger variability in range and apparent motion over short windows.

Overall, the RL-policy trained in a pure LEO environment is performing well in scenarios with significantly different RSO catalogs. Moreover, the performance of total illuminated images taken actually increases when some LEO targets are replaced with MEO and GEO RSOs. This is likely due to multiple factors including: (i) the much larger difference in semi-major axis ensures that over a given timespan the imaging spacecraft will be in view of the target even if starting on opposite sites of the planet (ii) secondly, the much larger altitude of those newly introduced targets ensure that their eclipse period when their illumination factor is below the e_{thresh} is much shorter, allowing for more opportunities to image them successfully (iii) the slower relative velocity compared to the LEO-LEO environment also tends to prolong visibility windows, facilitating a successful image capture before LOS is lost.

α sweep: how the policy reallocates actions

Across $\alpha \in [0, 1]$, the total episode reward remains nearly flat with slight downward trend towards the $\alpha = 1.0$ policy. Notably, the policy shifts how it spends decisions: increasing α induces more frequent downlink actions and fewer imaging actions, as expected. Figure 3 visualizes this trade in terms of total reward, illuminated images, and delivered (downlinked) illuminated images. Throughout this section, colors correspond to the downlink reward weight α using the same mapping shown by the colorbar in Fig. 3.

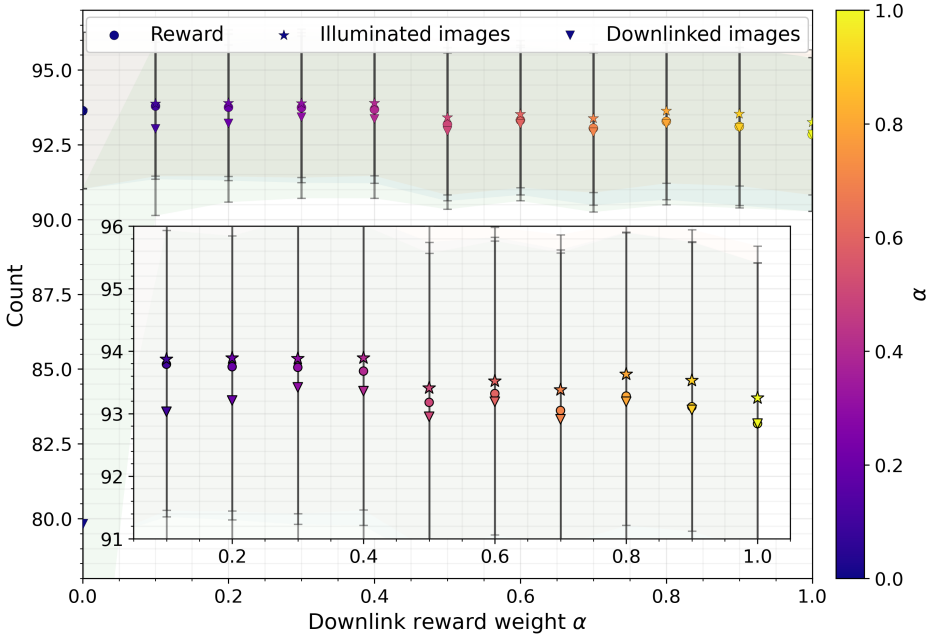

Figure 3 Mixed-regime α sweep summary using a consistent color encoding for α (colorbar at right). Each marker shows the mean count (with error bars) across Monte Carlo seeds for total reward, illuminated images, and delivered (downlinked) illuminated images. Colors in subsequent α-sweep plots follow this same α mapping.

A key qualitative change occurs between $\alpha = 0$ and $\alpha > 0$: when downlink is not rewarded ($\alpha = 0$) at all, the policy tends to postpone communications and instead relies on infrequent, large transfers (effectively waiting until the buffer is nearly full before downlinking).

Once $\alpha > 0$, delivered illuminated images increase rapidly and then plateaus below the number of illuminated images acquired (Fig. 4). Given the number of ground stations used in this scenario, it suggests that using less would cause this gap to increase due to the more limited downlink opportunities.

Table 4 summarizes the mixed-regime evaluation and includes the mean downlink fraction: the fraction of the onboard image buffer delivered per successful downlink event (mean $\pm$ std). This metric is useful because policies with similar end-of-episode delivered-image totals can differ substantially in delivery latency. At $\alpha = 0$, the mean downlink fraction is high (0.85 ± 0.13), consistent with bulk transfers that clear most of the buffer when a downlink opportunity is taken. As α increases, the mean downlink fraction drops sharply (e.g., 0.32 ± 0.07 at $\alpha = 0.1$ and ≈ 0.15–0.20 for $\alpha \geq 0.3$), indicating more frequent, smaller sized downlinks that reduce time-to-ground for

Table 4 Mixed-regime α sweep summary (mean $\pm$ std). "Useful downlinks" counts only illuminated images delivered. Mean downlink fraction is the fraction of onboard image storage delivered per successful downlink event (mean $\pm$ std). $N = 50$ for $\alpha = 0.6$, otherwise $N = 100$.

α	Total reward	Illum. images	Good images downlinked	Acq. success
0.0	93.64 ± 2.62	93.64 ± 2.62	79.82 ± 11.22	0.656 ± 0.039
0.1	93.79 ± 2.44	93.87 ± 2.42	93.03 ± 2.90	0.672 ± 0.034
0.2	93.75 ± 2.45	93.89 ± 2.45	93.21 ± 2.63	0.674 ± 0.034
0.3	93.74 ± 2.51	93.88 ± 2.48	93.42 ± 2.71	0.676 ± 0.037
0.4	93.68 ± 2.47	93.89 ± 2.43	93.36 ± 2.65	0.681 ± 0.038
0.5	93.18 ± 2.56	93.41 ± 2.59	92.95 ± 2.61	0.666 ± 0.039
0.6	93.63 ± 2.30	93.82 ± 2.30	93.50 ± 2.32	0.674 ± 0.037
0.7	93.05 ± 2.57	93.38 ± 2.48	92.91 ± 2.66	0.662 ± 0.031
1.0	92.84 ± 2.57	93.25 ± 2.43	92.84 ± 2.57	0.671 ± 0.033

α	Downlink actions	Imaging actions	Mean downlink frac.
0.0	3.27 ± 2.23	151.73 ± 3.85	0.85 ± 0.13
0.1	16.09 ± 4.93	136.74 ± 4.37	0.32 ± 0.07
0.2	20.73 ± 5.45	137.48 ± 3.64	0.24 ± 0.05
0.3	24.43 ± 5.52	133.65 ± 3.85	0.20 ± 0.04
0.4	23.69 ± 5.63	130.99 ± 4.51	0.19 ± 0.03
0.5	35.26 ± 6.30	128.00 ± 3.99	0.15 ± 0.03
0.6	29.58 ± 5.56	131.92 ± 3.58	0.17 ± 0.03
0.7	32.01 ± 5.19	131.00 ± 3.10	0.16 ± 0.03
1.0	29.13 ± 4.91	132.80 ± 3.01	0.16 ± 0.02

newly acquired imagery.

Notably, the best-performing policies in this study occur in the intermediate range $\alpha \approx 0.1$–0.4, which achieves the highest total reward while also maintaining near-saturated delivered-image counts. Combining these trends with the action-frequency summary (Fig. 5), it can be seen that intermediate α values can achieve comparable (or better) delivered-image performance without requiring the largest number of downlink actions, whereas very large α induces more frequent downlinking with smaller transfers per event or often times unsuccessful downlink actions due to an already empty storage buffer. From an operator perspective, α tunes a bulk-versus-latency preference where a low α prioritizes accumulating data and unloading it in larger bursts, whereas high α favors frequent partial transfers that reduce time-to-ground, improving delivery latency at the cost of more downlink actions and little additional total delivered imagery.

Single-scenario case study: how α changes behavior

To complement the Monte Carlo statistics, a representative single rollout in the mixed-regime environment is examined across several values of α. Table 5 summarizes the key outcomes, note here that the total reward uses the same α from training in the evaluation. Although the scalarized objective changes with α, the rollout exhibits the same qualitative pattern seen in the Monte Carlo sweep: larger α shifts decision allocation from imaging toward downlink, while maintaining comparable illuminated-image throughput.

Action reallocation with α: Comparing the extremes illustrates the behavioral shift. For $\alpha = 0$ (image-heavy), the agent spends the episode almost entirely imaging (149 imaging actions) and rarely downlinks (2 downlink actions), resulting in a large number of illuminated images onboard but fewer of them delivered (82 useful downlinks). For $\alpha = 1$ (downlink-heavy), the agent increases

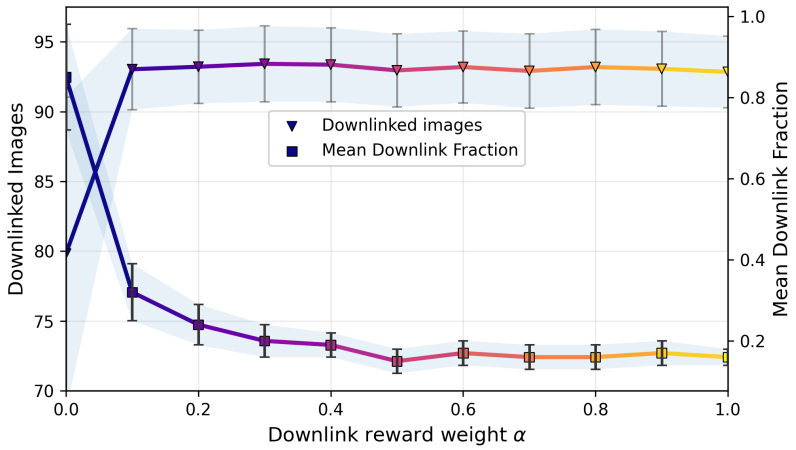

Figure 4 Delivered illuminated images versus α (left axis) and mean downlink fraction (right axis). The sharp drop in mean downlink fraction from $\alpha = 0$ to $\alpha > 0$ indicates a transition from infrequent bulk transfers to more frequent, smaller "maintenance" downlinks, which can reduce delivery latency even when final delivered-image totals are similar.

Table 5 Mixed-regime rollouts across α for a representative scenario. "Useful downlinks" counts only illuminated images delivered.

α	Total reward	Illum. images	Useful downlinks	Imaging acts	Downlink acts	Acq. success	Mean dt_{acq} [s]
0.0	92.0	92	82	149	2	0.678	154.4
0.1	94.7	95	92	142	9	0.676	138.2
0.2	93.8	94	93	143	13	0.692	125.3
0.3	93.7	94	93	138	16	0.650	158.7
0.4	93.0	93	93	140	18	0.657	147.8
0.5	94.0	94	94	134	25	0.694	162.6
0.7	92.3	93	92	140	18	0.621	114.1
1.0	92.0	93	92	134	27	0.697	123.5

downlink actions substantially (27) while reducing imaging actions (134), yielding nearly the same number of illuminated images (93) but a much higher delivered fraction (92 useful downlinks). This is consistent with downlink actions serving both as a delivery mechanism for useful imagery and a buffer-management mechanism when non-illuminated images occupy storage.

Intermediate values show that only a modest increase in downlink weighting is needed to achieve near-saturation in useful delivery: for $\alpha \in [0.2, 0.5]$, essentially all illuminated images are delivered (93–94 useful downlinks) while maintaining 134–143 imaging actions. The highest total reward in this rollout occurs at $\alpha = 0.1$, suggesting that a lightly downlink-aware policy can capture most of the delivery benefit without substantially reducing imaging opportunity.

Time-series interpretation (battery/storage, contacts, and cumulative counts): Figure 6 overlays battery and storage fraction, eclipse intervals, ground-station access windows, and cumulative illuminated images and downlinked targets. In image-heavy settings, storage tends to build between contacts, whereas in downlink-heavy settings, contacts are exploited more frequently and the downlinked-target count tracks the illuminated-image count more closely. The same plots also show how downlink actions align with station windows and eclipse segments.

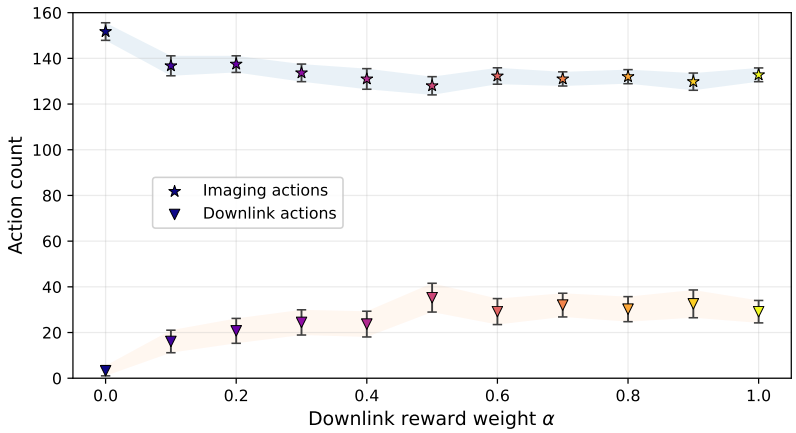

Figure 5 Imaging and downlink action counts versus α (mean $\pm$ std). Downlink actions increase with α while imaging actions decrease from the $\alpha = 0$ baseline and then stabilize, consistent with the intended reward trade.

Eclipse-phase pointing and target selection

Eclipse behavior is evaluated by conditioning on decision steps where the imaging spacecraft itself is in eclipse. In the representative mixed-regime rollout of the $\alpha = 0.1$ based RL-policy, there are 41 imaging decisions that occur during eclipse which are analyzed in more detail, since these steps are non-trivial and many nearby LEO objects are likely also eclipsed.

To quantify whether the policy behaves in an eclipse-aware manner, the following criteria are used to classify an eclipse-time imaging decision as "eclipse-effective". A decision is considered eclipse-effective if at least one of the following holds:

1. The selected target is illuminated at the time of selection, meaning its eclipse factor is above the illumination threshold.

2. The selected target is in a higher-altitude regime (MEO or GEO), which increases the likelihood of remaining illuminated during eclipse and often yields longer usable viewing intervals.

3. The selected target is in LEO and the pointing direction is biased toward the sunward side of the eclipse, computed from Hill-frame alignment between the target line-of-sight direction and the sun direction.

Under these rules, the RL-policy makes 37 eclipse-effective selections out of 41 eclipse-time imaging decisions. This indicates that the policy consistently exploits the limited opportunities available during eclipse. Among the four remaining cases that do not satisfy the eclipse-effective criteria, three coincide with situations where the candidate set is effectively exhausted, meaning no alternative unimaged target is available within the line-of-sight constraint. These events occur when the chosen target has low elevation beyond the approximate visibility bound near -21°, which strongly limits feasible imaging regardless of target illumination.

The eclipse-time event logs clarify how this consistency is achieved. Out of 41 eclipse-time imaging decisions, 36 select a target that is illuminated at decision time. In 18 cases the selected

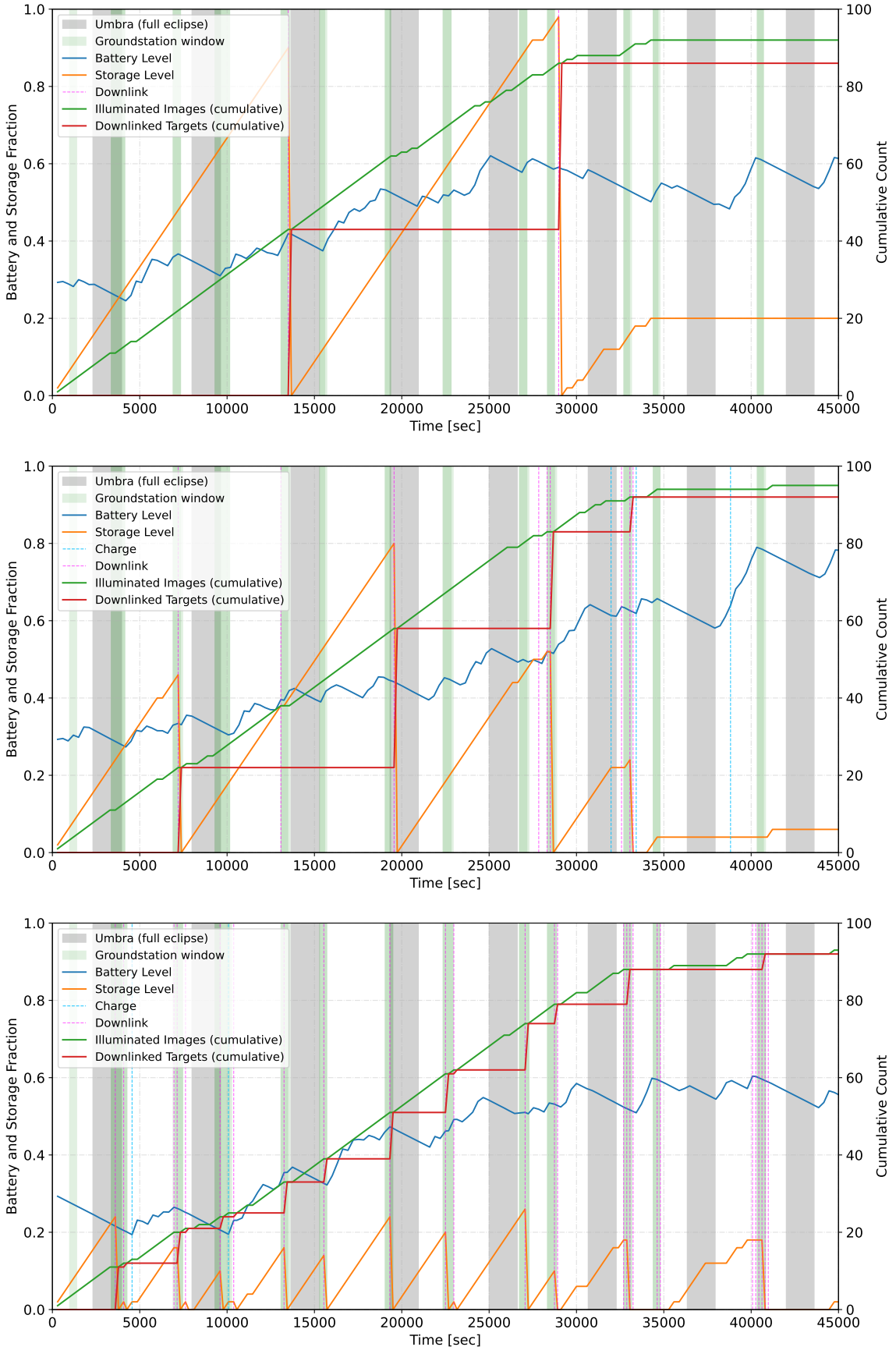

Figure 6 Mixed-regime time series for $\alpha = 0.0$ (top), $\alpha = 0.1$ (middle), and $\alpha = 1.0$ (bottom). Plots show battery and storage fractions, eclipse intervals, ground-station windows, and cumulative illuminated images/downlinked targets.

target is in a higher-altitude regime (MEO or GEO). In addition, 17 eclipse-time LEO selections satisfy the sunward-pointing test. Sunward pointing is measured by the Hill-frame alignment score

$$a \triangleq \hat{\boldsymbol{\ell}}^{H} \cdot \hat{\mathbf{s}}^{H},$$

where $\hat{\boldsymbol{\ell}}^{H}$ is the unit line-of-sight vector to the selected target and $\hat{\mathbf{s}}^{H}$ is the unit sun-direction vector, both expressed in the Hill frame. A LEO selection is classified as sunward if $a \geq 0$. Averaged over all eclipse-time imaging decisions, the mean alignment is $\bar{a} \approx 0.15$, noting that this value includes both LEO and higher-altitude targets and would likely be higher when restricted to LEO-only eclipse selections. These counts are not mutually exclusive, since a single selection can satisfy multiple criteria.

Compared to the rest of the episode, eclipse-time targeting also exhibits a relative shift away from LEO. This is consistent with the expected geometry, since a large fraction of nearby LEO objects enter eclipse concurrently with the imaging spacecraft. When LEO targets are selected during eclipse, the pointing direction frequently favors the sunward side of the eclipse, which is physically meaningful in scenarios where the imaging orbit intersects the shadow boundary at an angle rather than passing straight through the center of the Earth shadow. In such cases, one side of the local sky lies closer to the illuminated boundary, and targets in that direction are more likely to remain illuminated or to exit eclipse sooner. When this sunward-side preference cannot be satisfied because the remaining LEO targets are dark or geometrically infeasible, the policy more often selects higher-altitude targets whose illumination persistence improves eclipse-time acquisition probability.

Table 6 Episode segmentation statistics for target illumination and pointing elevation (Hill frame). The horizon is approximated by an elevation threshold near $-21°$.

Segment	Illuminated target fraction	Mean elevation $\pm$ std [°]
Segment 1/3	0.958	-13.59 ± 7.03
Segment 2/3	1.000	-6.58 ± 13.50
Segment 3/3	0.766	-28.73 ± 9.15

Table 6 further supports the interpretation that late-episode geometry, rather than policy preference, drives the few eclipse-inconsistent selections. The first two thirds of the episode maintain near-saturated illuminated targeting, while the final third shows both a lower illuminated fraction and a mean elevation below the line-of-sight threshold. This shift matches a regime where the remaining unimaged opportunities are increasingly low elevation, forcing occasional riskier selections that are constrained by visibility rather than by eclipse-aware strategy.

CONCLUSIONS

This work extends high-fidelity RL-based SBSS scheduling from a LEO-to-LEO setting to a multi-regime "LEO-to-any" environment with LEO, MEO, and GEO targets, while explicitly trading off image acquisition and downlink delivery using a scalar mixing parameter α. Monte Carlo results show that a policy trained in a LEO-only catalog transfers to a mixed-regime catalog without retraining, maintaining stable acquisition performance and achieving higher total return and illuminated-image throughput.

Sweeping α yields an operator-interpretable shift in behavior. As α increases, downlink actions become more frequent and imaging actions decrease modestly, while total reward remains nearly

flat. Delivered illuminated imagery rises sharply once $\alpha > 0$ and then saturates near the number of illuminated images acquired, indicating that delivery is limited mainly by access geometry and pointing constraints. Eclipse-conditioned analysis further shows eclipse-effective target selection, with 37 of 41 eclipse-time imaging decisions satisfying criteria based on selecting illuminated targets, substituting higher-altitude targets, or choosing sunward-biased LEO pointings in Hill coordinates. The few exceptions occur primarily when remaining unimaged opportunities fall below the line-of-sight elevation bound and no better imaging option is available. This is particularly relevant as it can potentially improve onboard decision-making in particularly challenging situations, which a human operator may have a hard time analyzing.

ACKNOWLEDGMENTS

This work utilized the Alpine high-performance computing resource at the University of Colorado Boulder. Alpine is jointly funded by the University of Colorado Boulder, the University of Colorado Anschutz, Colorado State University, and the National Science Foundation (award 2201538).

REFERENCES

[1] A. Williams, O. Hainaut, A. Otarola, G. H. Tan, A. Biggs, N. Phillips, and G. Rotola, "A Report to ESO Council on the Impact of Satellite Constellations," tech. rep., European Southern Observatory (ESO), 2021.

[2] J. A. Hernandez and P. Reviriego, "A brief introduction to satellite communications for NTN," *arXiv*, 2023.

[3] M. R. Ackermann, R. R. Kiziah, P. C. Zimmer, J. T. McGraw, and D. D. Cox, "A Systematic Examination of Ground-Based and Space-Based Approaches to Optical Detection and Tracking of Satellites," Tech. Rep. SAND2015-3276C, Sandia National Laboratories, 2015.

[4] G. Stokes, C. Vo, R. Sridharan, and J. Sharma, "The Space-Based Visible Program," *Space 2000 Conference and Exposition*, American Institute of Aeronautics and Astronautics, 2000-09.

[5] E. M. Gaposchkin, C. v. Braun, and J. Sharma, "Space-Based Space Surveillance with the Space-Based Visible," Vol. 23, No. 1, 2000, pp. 148–152.

[6] B. Jia, K. D. Pham, E. Blasch, D. Shen, Z. Wang, and G. Chen, "Cooperative Space Object Tracking Using Space-Based Optical Sensors via Consensus-Based Filters," Vol. 52, No. 4, 2016, pp. 1908–1936.

[7] T. Flohrer, H. Krag, H. Klinkrad, and T. Schildknecht, "Feasibility of Performing Space Surveillance Tasks with a Proposed Space-Based Optical Architecture," *Advances in Space Research*, Vol. 47, No. 6, 2011, pp. 1029–1042.

[8] J. Silha, T. Schildknecht, A. Hinze, J. Utzmann, A. Wagner, P. Willemsen, F. Teston, and T. Flohrer, "Capability of a Space-Based Space Surveillance System to Detect and Track Objects in GEO, MEO and LEO Orbits," *Proceedings of the 65th International Astronautical Congress*, Toronto, Canada, 2014. IAC-14.A6.1.1.

[9] L. Ansalone and F. Curti, "A genetic algorithm for Initial Orbit Determination from a too short arc optical observation," *Advances in Space Research*, Vol. 52, No. 3, 2013, pp. 477–489.

[10] D. A. Vallado and S. S. Carter, "Accurate orbit determination from short-arc dense observational data," *Journal of the Astronautical Sciences*, Vol. 46, No. 2, 1998, pp. 195–213.

[11] J.-S. Ardaens and G. Gaias, "A numerical approach to the problem of angles-only initial relative orbit determination in low Earth orbit," *Advances in Space Research*, Vol. 63, No. 12, 2019, pp. 3884–3899.

[12] S. M. Lenz, H. G. Bock, J. P. Schlöder, E. A. Kostina, G. Gienger, and G. Ziegler, "Multiple shooting method for initial satellite orbit determination," *Journal of Guidance, Control, and Dynamics*, 2012.

[13] G. Sciré, F. Santoni, and F. Piergentili, "Analysis of orbit determination for space based optical space surveillance system," *Advances in Space Research*, Vol. 56, No. 3, 2015, pp. 421–428.

[14] G. M. Goff, J. T. Black, and J. A. Beck, "Tracking maneuvering spacecraft with filter-through approaches using interacting multiple models," *Acta Astronautica*, Vol. 114, 2015, pp. 152–163.

[15] E. Hedenström, "Tracking Sensor Network Schedule Optimization for Space Surveillance," master's thesis, KTH Royal Institute of Technology, 2025.

[16] N. K. Dhingra, C. DeJac, A. Herz, and R. Green, "Space Domain Awareness Sensor Scheduling with Optimality Certificates," *Proceedings of the Advanced Maui Optical and Space Surveillance Technologies (AMOS) Conference*, Maui, HI, USA, 2023. Accessed 2025-08-17.

[17] N. Herz and R. Wimmer-Schweingruber, "Scheduling algorithms for ground-based optical space surveillance," *Acta Astronautica*, Vol. 87, 2013, pp. 1–11.

[18] T. Hinze, J. Strohmer, and B. Bastida Virgili, "A genetic algorithm for optimizing GEO object observation schedules," *Proceedings of the AMOS Technical Conference*, 2012.

[19] B. Shteinman and M. K. Jah, "Information-gain driven auction-based scheduling for space surveillance sensors," *Proceedings of the AMOS Technical Conference*, 2018.

[20] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. MIT Press, 2nd ed., 2018.

[21] D. Jang, P. M. Siew, D. Gondelach, and R. Linares, "Space Situational Awareness Tasking for Narrow Field of View Sensors: A Deep Reinforcement Learning Approach," *Proceedings of the 71st International Astronautical Congress (IAC)*, International Astronautical Federation, 71st International Astronautical Congress. International Astronautical Federation, the International Academy of Astronautics, and the Interna- tional Institute of Space Law, 2020-10.

[22] R. Linares and R. Furfaro, "An Autonomous Sensor Tasking Approach for Large Scale Space Object Cataloging," 2017.

[23] P. M. Siew, T. Smith, R. Ponmalai, and R. Linares, "Scalable Multi-Agent Sensor Tasking Using Deep Reinforcement Learning," 2023.

[24] R. Linares and R. Furfaro, "Dynamic Sensor Tasking for Space Situational Awareness via Reinforcement Learning," 2016.

[25] B. Oakes, J. F. Ralph, and J. Barr, "Deep Reinforcement Learning Applications to Space Situational Awareness Scenarios," 2024.

[26] M. Nazari, A. Oroojlooy, L. Snyder, and M. Takac, "Reinforcement Learning for Solving the Vehicle Routing Problem," *Advances in Neural Information Processing Systems (NeurIPS)*, Vol. 31, 2018.

[27] A. Harris, T. Teil, and H. Schaub, "Spacecraft Decision-Making Autonomy Using Deep Reinforcement Learning," *Advances in the Astronautical Sciences, Volume 164: 29th AAS/AIAA Space Flight Mechanics Meeting*, Univelt, 2019, pp. 1757–1775.

[28] A. Hadj-Salah, R. Verdier, C. Caron, M. Picard, and M. Capelle, "Schedule Earth Observation Satellites with Deep Reinforcement Learning," 2019.

[29] D. Eddy and M. Kochenderfer, "Markov Decision Processes for Multi-Objective Satellite Task Planning," *2020 IEEE Aerospace Conference*, IEEE, 2020, pp. 1–12.

[30] A. Harris, T. Valade, T. Teil, and H. Schaub, "Generation of Spacecraft Operations Procedures Using Deep Reinforcement Learning," Vol. 59, No. 2, 2022, pp. 611–626.

[31] A. Herrmann and H. Schaub, "Reinforcement Learning for the Agile Earth-Observing Satellite Scheduling Problem," Vol. 59, No. 5, 2023-10, pp. 5235–5247.

[32] L. Q. Mantovani, Y. Nagano, and H. Schaub, "Reinforcement Learning for Satellite Autonomy Under Different Cloud Coverage Probability Observations," *AAS/AIAA Astrodynamics Conference*, 2023.

[33] M. Stephenson and H. Schaub, "Reinforcement Learning for Earth-Observing Satellite Autonomy with Event-Based Task Intervals," *AAS/AIAA Space Flight Mechanics Conference*, 2024.

[34] P. M. Siew, D. Jang, T. G. Roberts, and R. Linares, "Space-Based Sensor Tasking Using Deep Reinforcement Learning," Vol. 69, No. 6, 2022-12-01, pp. 1855–1892.

[35] D. H. Prats, H. Schaub, and C. Wheeler, "Reinforcement Learning for Space-to-Space Surveillance: Autonomous Scheduling for Resident Space Object Imaging," *Advanced Maui Optical and Space Surveillance Technologies Conference*, Maui, Hawaii, September 16–99 2025.

[36] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal Policy Optimization Algorithms," Aug. 2017.

[37] D. M. Roijers, P. Vamplew, S. Whiteson, and R. Dazeley, "A Survey of Multi-Objective Sequential Decision-Making," *Journal of Artificial Intelligence Research*, Vol. 48, 2013, pp. 67–113.

[38] C. F. Hayes, R. Rădulescu, E. Bargiacchi, J. Källström, M. Macfarlane, M. Reymond, T. Verstraeten, L. M. Zintgraf, R. Dazeley, F. Heintz, E. Howley, A. A. Irissappane, P. Mannion, A. Nowé, G. Ramos, M. Restelli, P. Vamplew, and D. M. Roijers, "A Practical Guide to Multi-Objective Reinforcement Learning and Planning," *Autonomous Agents and Multi-Agent Systems*, Vol. 36, No. 1, 2022, p. 26.

[39] L. Wei, Y. Chen, M. Chen, and Y. Chen, "Deep Reinforcement Learning and Parameter Transfer Based Approach for the Multi-Objective Agile Earth Observation Satellite Scheduling Problem," *Applied Soft Computing*, Vol. 110, 2021, p. 107607.

[40] P. W. Kenneally, S. Piggott, and H. Schaub, "Basilisk: A Flexible, Scalable and Modular Astrodynamics Simulation Framework," *Journal of Aerospace Information Systems*, Vol. 17, Sept. 2020, pp. 496–507.

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

[42] M. Stephenson, D. H. Prats, and H. Schaub, "Autonomous Satellite Inspection in Low Earth Orbit with Optimization-Based Safety Guarantees," *International Workshop on Planning & Scheduling for Space*, Toulouse, France, April 28–30 2025.

[43] M. A. Stephenson and H. Schaub, "Optimal Agile Satellite Target Scheduling with Learned Dynamics," *Journal of Spacecraft and Rockets*, Vol. 62, No. 3, 2025-05, pp. 793–804.
[44] Hanspeter Schaub and J. Junkins, "Nonlinear Spacecraft Stability and Control," *Analytical Mechanics of Space Systems, Fourth Edition*, AIAA Education Series, pp. 387–518, American Institute of Aeronautics and Astronautics, Inc., 4 ed., 2018-01-19.
[45] E. Liang, R. Liaw, P. Moritz, R. Nishihara, R. Fox, K. Goldberg, J. E. Gonzalez, M. I. Jordan, and I. Stoica, "RLlib: Abstractions for Distributed Reinforcement Learning," *Proceedings of the 35th International Conference on Machine Learning*, Vol. 80, June 2018, pp. 3053–3062.
[46] S. Cakaj, B. Kamo, V. Koliçi, and O. Shurdi, "The Range and Horizon Plane Simulation for Ground Stations of Low Earth Orbiting (LEO) Satellites," Vol. 04, No. 09, 2011, pp. 585–589.

APPENDIX: ENVIRONMENT AND SIMULATION DETAILS

This appendix collects the detailed environment parameters and figures referenced in Section . Unless otherwise noted, values follow the BSK-RL defaults and the configuration used in the prior LEO-to-LEO study.[35] The LEO-only setup is recovered by restricting the catalog to the LEO column in Table 7.

Orbital and Ground-Station Parameters

Table 7 summarizes representative orbital element ranges for the inspector and RSOs across regimes. These distributions are sampled independently for Monte Carlo experiments.

Table 7 Representative orbital parameters for scanning satellite and passive RSOs

Element	Inspector	LEO RSOs	MEO RSOs	GEO RSOs
Semi-major axis a (km)	6871[a]	6871–8371[b]	∼16371–31371	∼42164
Eccentricity e	0 (circular)	[0.0, 0.02]	[0.0, 0.05]	[0.0, 0.02]
Inclination i (deg)	0–180	0–180	0–180	0–20
Right Ascension Ω (deg)	0–360	0–360	0–360	0–360
Argument of Periapsis ω (deg)	0–360	0–360	0–360	0–360
True Anomaly f (deg)	0–360	0–360	0–360	0–360

a Fixed at 500 km altitude above Earth's mean radius (6371 km).

b LEO shell used in the original LEO-to-LEO baseline[35] .

Figure 7 illustrates the ground-station visibility footprint for the 500 km LEO inspector, with slant ranges computed as in Ref.[46]

Spacecraft and Control Parameters

The Basilisk module interconnections are shown in Fig. 8. The guidance module (`LocPointTask`) uses the selected target's navigation message to compute a desired attitude; `mrpFeedback` generates wheel torques, which are applied by the reaction-wheel assembly and returned as bus torques to the rigid-body dynamics. Power and data-handling modules monitor energy and storage, providing the state variables used in the RL observation.

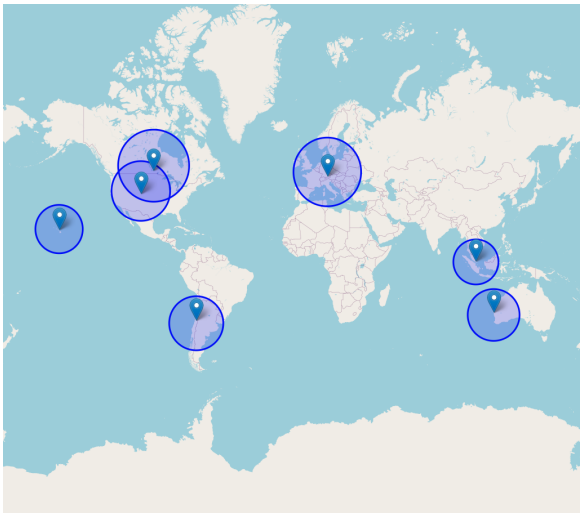

Figure 7 Ground-station visibility for LEO imaging satellite at 500 km altitude

Targeting, Imaging, and Downlink Constraints

Table 8 lists the main targeting, imaging, and downlink constraints used in the environment. These are consistent with the prior LEO-only study[35] and are applied identically across regimes.

Table 8 Targeting, imaging, and downlink constraints

Constraint	Value	Unit / Description
Attitude error requirement	≤ 0.0025	MRP norm
Attitude rate requirement	≤ 0.01	rad/s
Eclipse threshold e_{thresh}	0.5	illumination factor
Single-image data size	0.5	Mb
Storage capacity	25	Mb (50 images)
Initial storage fill level	0	Mb
FSW evaluation interval	0.5	s
Transmitter baud rate	1	Mbps (above elevation mask)

Training

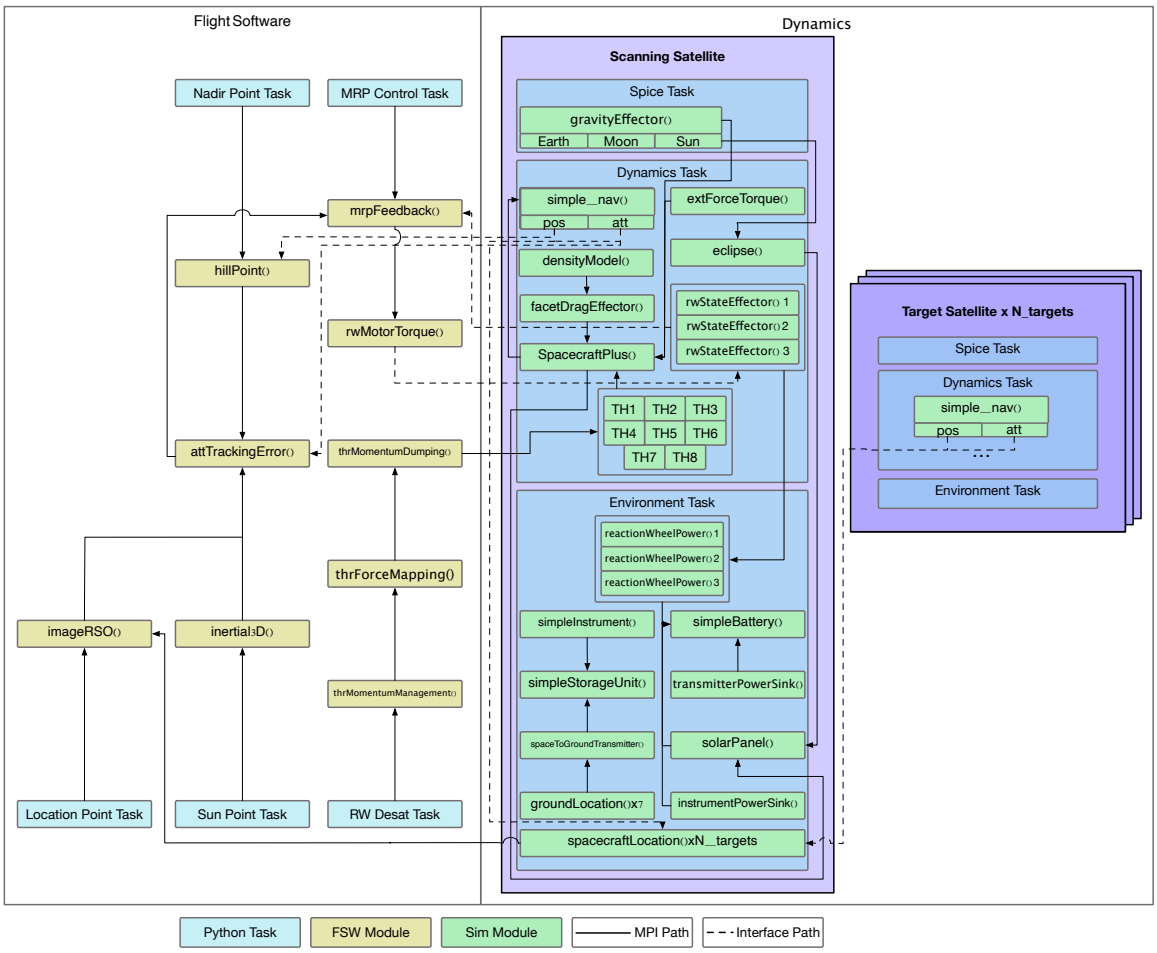

Figure 8 Basilisk simulation architecture used for SBSS scheduling

Table 9 RL Training Parameters

Name	Value
Learning rate	1×10^{-6}
Discount factor (γ)	0.9997
Gradient clip	1.0
PPO clip parameter (ϵ)	0.15
Training batch size	5000