Deep Reinforcement Learning for Rapid Spacecraft Science Operations Scheduling to Maximize Science Return

Alex M. Zhang[1], **Lara Waldrop**[1]

[1]Department of Electrical and Computer Engineering, University of Illinois Urbana-Champaign, IL, USA
{alexmz2, lwaldrop}@illinois.edu

Abstract

Science operations scheduling is crucial for any spacecraft to deliver high-value scientific results after launch. Unfortunately, finding the optimal operations schedule that maximizes science return is very difficult (NP-hard) due to complexities arising from a wide range of operational constraints including power, thermal, telemetry, target visibility, in addition to constraints arising from the science objectives themselves. The scale of the scheduling problem is far too large to optimize manually or via traditional methods such as mixed-integer linear programming or even classical reinforcement learning methods. We introduce a deep reinforcement learning framework based on the Maskable Proximal Policy Optimization (MPPO) algorithm to perform science operations scheduling and demonstrate its application to NASA's upcoming Carruthers Geocorona Observatory mission, where science returns are maximized by enhancing the absolute sensitivity characterization achieved via optimal scheduling of stellar calibration observations. Our approach is fast (training and scheduling all in under 6 hours), reliable, and represents, to our knowledge, the first demonstration of a deep reinforcement learning framework for science operations scheduling on a large-scale NASA heliophysics mission.

Introduction

Spacecraft science operations scheduling to maximize the science return of the overall mission is an NP-hard problem (Garey and Johnson 2002) due to many overlapping constraints and the large number of tasks that need to be scheduled. Common constraints include power budgets that restrict spacecraft pointing to a certain angular window, target visibility windows due to the spacecraft's orbit geometry, inter-observation setup/teardown times, accommodations for mission-critical operations, and science-specific constraints. Traditionally, much of this planning has been done manually or with ad hoc tools, which are highly labor-intensive. Exact optimization methods, such as mixed-integer programming, constraint programming, graph search, or even classical reinforcement learning approaches have been explored, but these approaches either demand significant domain expertise, lack the speed necessary for time-critical replanning, or do not scale to the complexity and size of the Carruthers mission scheduling problem. Heuristic methods offer the scalability but at the cost of optimality (Jacquet et al. 2024).

Deep reinforcement learning (DRL) has recently shown great promise for tackling large-scale scheduling tasks by combining neural network function approximation with reinforcement learning to construct schedules by sequentially choosing the best operation to schedule at particular points in time (Herrmann and Schaub 2023). Although prior studies have demonstrated encouraging results on satellite observation scheduling benchmarks (Liu et al. 2025), in realistic mission scenarios these approaches often demand days of training to fully enforce complex constraints. This is an impractical requirement when constraint sets and science priorities can quickly evolve due to new telemetry, science targets of opportunity, or anomaly responses.

In this work, we introduce a DRL framework built on the Maskable Proximal Policy Optimization (MPPO) algorithm that guarantees per-step compliance with hard constraints via dynamic action masking. We demonstrate the framework on NASA's upcoming Carruthers Geocorona Observatory mission set to launch in September 2025. The Carruthers mission is a dual-camera ultraviolet observatory with six configurable filters per camera, where precisely scheduling stellar observations is essential for accurately characterizing absolute sensitivity across every camera–filter combination, all while respecting power, telemetry, and visibility budgets.

Crucially, our MPPO-based scheduler meets stringent operational demands: in pre-launch stress tests simulating a mission anomaly that required a constraint change, our scheduler retrained and replanned a feasible three-month operations sequence composed of 1216 science operations in under six hours. This rapid, reliable replanning capability is an advance over existing DRL and classical scheduling methods and represents, to our knowledge, the first demonstration of a deep reinforcement learning framework for science operations scheduling on a large-scale NASA heliophysics mission.

Problem Statement

Suppose we have J activity blocks a_j, $1 \leq j \leq J$, where each activity block represents a science operation that may contain multiple science targets of interest, described by the tuple $(i, p, q, \ell, d_{\min}, d_{\max}, b_1, b_n, T_1, T_2, T_3)$, where

- i denotes the sequence of images to be taken.
- p is the minimum number of times activity block a_j needs to be scheduled.
- q is the maximum number of times activity block a_j needs to be scheduled.
- ℓ is the length of time, in seconds, of the activity block.
- d_{min} is the required minimum time, in seconds, between different scheduled instances of this activity block.
- d_{max} is the required maximum time, in seconds, between different scheduled instances of this activity block.
- b_1 and b_n describe the first and last target coordinates for the spacecraft to point in for this activity block.
- T_x for $x = 1, 2, 3$ is a list of visibility intervals in which power regime x applies.

The power-use regimes are as follows:

1. Regime 1: No restriction.
2. Regime 2: Only 8 hours of continuous imaging per 24 hours allowed.
3. Regime 3: Only 8 hours of continuous imaging at a time allowed, and only 12 hours of imaging every four weeks is allowed.

Note that the same activity block can be part of multiple different power-use regimes, which complicates the overall problem even further. For example, the first science target of an activity block may be part of power regime 1, but the last science target of an activity block can be in power regime 3. Additionally, the schedule must satisfy the following requirement motivated by power: 'For all instances in time, there must exist a continuous block of at least 8 hours spent within power regime 1 in the last 24 hours.'

In order to accommodate communications with NASA's Deep Space Network (DSN), the schedule cannot have science operations overlap with predetermined 6-hour windows that occur about twice a week. Moreover, within 8 hours of one of these predetermined windows, only activity blocks completely within power regime 1 are allowed to be scheduled in order to conserve power for the power-hungry DSN pass.

Finally, whenever adjacent blocks involve different pointing targets, the intervening slew duration must be included in the schedule. The slew period is subject to the most restrictive power-use regime (in order to be conservative).

Science constraints for the Carruthers mission are relatively straightforward - there are some blocks that should only be scheduled after 21 hours have been spent observing Earth, while others should be scheduled at least once daily. Numerous other science constraints (for example, an image must be taken with one of the filters on both cameras every three hours of imaging) are dealt with within each activity block and are thus not within the scope of the scheduling algorithm.

Any schedule that meets all of the above constraints is denoted as a *feasible schedule*. The *optimal schedule* is the feasible schedule that yields the best science. In the Carruthers mission's case, the best science is defined as the most accurate estimation of absolute sensitivity (least mean percent error) for each camera–filter pair after all operations are executed when evaluated by the mission's absolute sensitivity characterization algorithm (i.e., the real calibration routine used onboard and in ground processing) (Zhang et al. 2025).

Approach

In order to solve the scheduling problem defined in the previous section, we adopt a DRL approach using Masked Proximal Policy Optimization (MPPO), an extension of PPO (Schulman et al. 2017) that guarantees compliance with hard constraints by dynamically masking invalid actions from the policy's output distribution (Huang and Ontañón 2020). Due to the high number of constraints, standard PPO proved ineffective in our setting since it required excessive training time to obtain a feasible schedule, much less attempt to find the optimal schedule. In our approach, actions correspond to selecting the next activity block (or waiting), while states encode the current time, the number of times each activity block has already been scheduled, power regime usage, and upcoming DSN pass windows. We implemented a custom environment using the OpenAI Gymnasium interface (Towers et al. 2024), which handles state transitions, action masking for each state, and reward computation.

Reward-shaping is the most difficult part of any reinforcement learning application. In our case, we formulate a composite reward. First, the agent is given a large reward for activity blocks that need to be scheduled and a penalty once the required activity block can no longer be scheduled while still respecting constraints. If any of the daily activity blocks are missed, then a penalty is added at the end of each scheduled day. Finally, at the end of every three scheduled days, the absolute sensitivity characterization algorithm is used to obtain an estimate of the absolute sensitivity for each camera-filter pair based on all operations scheduled so far. Rewards are given for improvement on prior estimations, while a penalty is given on the first run if the error exceeds a certain threshold to prevent rewards-gaming. Regular monitoring of absolute sensitivity is critical for maintaining training stability, given that the algorithm is tasked with scheduling operations over extended periods spanning three months to a year. All other constraints are enforced via the action mask, thus eliminating the need for additional penalty terms.

For training, 24 independent instances of the environment are created and simulated in parallel. Training proceeds for 10^7 steps total, with updates to the policy and value neural networks every 256 steps. Once training is complete, we execute multiple Monte-Carlo rollouts of the stochastic policy and the best-performing schedule is selected as the final schedule. This approach hedges against the randomness in the learned policy and aims to further optimize the final schedule. On Ubuntu 22.04 with AMD Threadripper 5995WX (128GB RAM) and NVIDIA RTX 4070 (12GB VRAM) with CUDA 12.4 and PyTorch 2.2.2., the training takes nearly 5.5 hours while the evaluation takes under 25 minutes, bringing the total time needed to run the entire scheduling pipeline to under 6 hours. On-orbit, schedule replans can be expedited by skipping the training step if there is no major change to any constraints, thus dropping the required time to 25 minutes. The latter has not been fully optimized and can likely be made faster.

References

Garey, M. R.; and Johnson, D. S. 2002. *Computers and intractability*, volume 29. wh freeman New York.

Herrmann, A.; and Schaub, H. 2023. A comparative analysis of reinforcement learning algorithms for earth-observing satellite scheduling. *Frontiers in Space Technologies*, 4: 1263489.

Huang, S.; and Ontañón, S. 2020. A closer look at invalid action masking in policy gradient algorithms. *arXiv preprint arXiv:2006.14171*.

Jacquet, A.; Infantes, G.; Meuleau, N.; Benazera, E.; Roussel, S.; Baudoui, V.; and Guerra, J. 2024. Earth Observation Satellite Scheduling with Graph Neural Networks. *arXiv preprint arXiv:2408.15041*.

Liu, Z.; Xiong, W.; Jia, Z.; and Han, C. 2025. Two-stage deep reinforcement learning method for agile optical satellite scheduling problem. *Complex & Intelligent Systems*, 11(1): 35.

Schulman, J.; Wolski, F.; Dhariwal, P.; Radford, A.; and Klimov, O. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Towers, M.; Kwiatkowski, A.; Terry, J.; Balis, J. U.; De Cola, G.; Deleu, T.; Goulão, M.; Kallinteris, A.; Krimmel, M.; KG, A.; et al. 2024. Gymnasium: A standard interface for reinforcement learning environments. *arXiv preprint arXiv:2407.17032*.

Zhang, A. M.; Waldrop, L.; Filippini, H.; Clarke, J.; Joshi, P.; Cucho-Padin, G.; Karimi, P.; and Sirk, M. 2025. On-orbit Calibration of the Carruthers GCI: Radiometric Sensitivity. To be submitted to Space Science Reviews.