

REINFORCEMENT LEARNING FOR SPACECRAFT NAVIGATION & ENVIRONMENT CHARACTERIZATION IN THE PLANAR-RESTRICTED TWO-BODY PROBLEM

Kenneth M. Getzandanner^{*†} and John R. Martin[‡]

OVERVIEW

During mission planning and execution, spacecraft operators must balance data collection and downlink, systems constraints, human factors, and navigation. As missions become increasingly complex and ambitious, these factors become more intricately entwined and conflicted. For example, a spacecraft's position must be known accurately in order to point to and image a target. Large position errors may cause missed observations or require additional scanning that increases operations complexity and data volume. Some observations require imaging from specific relative geometries which adds orbit control and timing considerations. Adjusting the orbit may allow for optimal observability of environmental parameters and/or enable more efficient sensor coverage, but maneuver execution error adds uncertainty to the current state which impacts both characterization and coverage objectives.

For Earth-orbiting spacecraft, onboard, autonomous navigation systems that utilize the Global Positioning System (GPS) and other measurements simplify operations planning. However, resource constraints may limit the ability to perform navigation and other operations tasks simultaneously. Spacecraft that rely on ground-based radiometrics must consider tracking and ephemeris update schedules. Furthermore, interplanetary and small-body navigation relies on Earth-based radiometric tracking and, in some cases, *in situ* observations. Generally, this type of tracking conflicts with science observations and data storage, e.g., the spacecraft must slew an antenna towards Earth, capture Optical Navigation (OpNav) images, or collect altimetry measurements. Environment characterization is moreover tightly coupled with navigation and operations. Uncertainty in geophysical parameters, such as gravity field, spin state, atmosphere, etc., of uncharacterized bodies affect navigation performance, and are often valuable scientifically independent of navigation. The small-body environment presents a particularly complex and a stressing case, as experienced by the Origins, Spectral Interpretation, Resource Identification, Security – Regolith Explorer (OSIRIS-REx) spacecraft at the near-Earth asteroid 101955 Bennu^{1,2} and other missions.

Deep Reinforcement Learning (DRL) is a State-of-the-Art (SOA) technique for autonomous spacecraft operations planning and autonomy. Examples in the literature include applications of DRL for the Agile Earth Observing Satellite Scheduling Problem (AEOSSP),^{3,4} Mars station-keeping and Low-Earth Orbit (LEO) observations,⁵ multiple aspects of spacecraft control,⁶ Au-

^{*}Flight Dynamics Lead, NASA/GSFC Code 595, 8800 Greenbelt Rd, Greenbelt, MD 20771.

[†]PhD Student, Department of Aerospace Engineering, University of Maryland, College Park, MD 20742.

[‡]Assistant Professor, Department of Aerospace Engineering, University of Maryland, College Park, MD 20742.

onomous Rendezvous & Docking (AR&D),⁷ and operations about a small celestial body,^{8–11} among others. To date, realistic navigation considerations and environment characterization for the spacecraft scheduling and autonomy problem have been largely simplified or omitted. The goal of this project is to address this gap in the literature by developing a simple environmental characterization training environment in the Planar-Restricted 2-Body Problem (PR2BP), establishing benchmarks and heuristic baselines, and designing a previously unstudied Markov Decision Process (MDP) formulation that will enable the spacecraft DRL agents to appropriately balance navigation and actuation capabilities. Starting with a lower-fidelity model provides valuable insight into the structure of the problem and potential solutions that can be applied to future iterations with increasing levels of fidelity.

PROBLEM DEFINITION

Consider a spacecraft in a planar, 2D orbit about an arbitrary, planar central body. The central body rotates at a fixed rate about an axis perpendicular to the spacecraft’s orbital plane. The only force acting on the spacecraft is the two-body gravitational attraction of the central body, thus dynamics are governed by the PR2BP. The spacecraft’s initial position and velocity vectors, $\vec{r}(t_0), \vec{v}(t_0)$, the central body’s gravitational parameter, μ , initial rotation offset, θ_0 , and constant rotation rate, ω , and cartesian positions of a configurable number of optical surface landmarks, $\vec{r}_l^T$, are random variables with uncorrelated, Gaussian priors. The system state vector, $\vec{X}$, and covariance matrix, P , are defined as:

$$\vec{X} = [\vec{r}^T, \vec{v}^T, \mu, \theta, \omega, \vec{r}_l^T]^T \quad (1)$$

$$P = \mathbb{E} [(\hat{X} - \vec{X})(\hat{X} - \vec{X})^T] \quad (2)$$

where $\hat{X} - \vec{X}$ is the difference between the estimated and true state vectors. As discussed in Reference 12, the covariance matrix can be factorized into the following form for improved numerical stability:

$$P = UDU^T \quad (3)$$

where U is an upper triangular matrix with 1’s along the diagonal and 0’s below it, and D is a diagonal matrix.

The state is propagated at fixed time steps using numerical integration subject to PR2BP dynamics for the spacecraft and constant-rate central body rotation. The State Transition Matrix (STM) is also numerically-integrated along with the state and used for linear covariance propagation. Fixed process noise, S , is also applied at each time update to capture the effects of un-modeled accelerations as well as aid in numerical stability.* The full time update is shown in Eqs. (4) & (5), where F is the non-linear time update function using numerical integration.

$$\vec{X}(t_{k+1}), \Phi(t_k, t_{k+1}) = F(t_k, \vec{X}(t_k)) \quad (4)$$

*Unfortunately, the process noise matrix symbol, S , is similar to the MDP state space symbol, $\mathcal{S}$, used later in the document. Note the use of script font for the latter to help differentiate.

$$P(t_{k+1}) = \Phi(t_k, t_{k+1})P(t_k)\Phi(t_k, t_{k+1})^T + S(t_k) \quad (5)$$

For the UD-factorized form, the covariance time update, Eq. (5), can be restructured to solve for $U(t_{k+1})$ and $D(t_{k+1})$ directly using the Modified Gram-Schmidt algorithm (see Reference 12). The analytical approximation for the process noise covariance, $S(t_k)$, based on a 1st-order Taylor Series truncation of $\Phi(t_k, t_{k+1})$, presented in Reference 12, is used here. Process noise is applied to the translational and rotational portions of the state with separate Power Spectral Densities (PSDs), q_t and q_r . $S(t_k)$ is decomposed into the factors Q_k and G_k according to Equations (6), (7), and (8) as inputs into the Modified Gram-Schmidt algorithm.

$$S(t_k) = G_k Q_k G_k^T = \begin{bmatrix} \frac{\Delta t^3}{3} q & \frac{\Delta t^2}{2} q \\ \frac{\Delta t^2}{2} q & \Delta t q \end{bmatrix} \quad (6)$$

$$Q_k = \begin{bmatrix} \frac{\Delta t^3}{12} q & 0 \\ 0 & \Delta t q \end{bmatrix} \quad (7)$$

$$G_k = \begin{bmatrix} 1 & \frac{\Delta t}{2} \\ 0 & 1 \end{bmatrix} \quad (8)$$

A configurable number of ground stations reside on the central body at specified body-fixed coordinates. At any time step, the spacecraft may receive a single instantaneous range measurement from one ground station subject to visibility constraints, i.e., the spacecraft must be above the station's local horizon. There are also a configurable number of landmarks on the surface which may be used to generate optical bearing measurements, also subject to visibility constraints. A station or landmark is considered visible if the spacecraft is above the local horizon, i.e.,

$$\frac{\vec{r} - \vec{r}_n}{\|\vec{r} - \vec{r}_n\|} \cdot \frac{\vec{r}_n}{\|\vec{r}_n\|} \geq 0 \quad (9)$$

where $\vec{r}$ and $\vec{r}_n$ are the inertial positions of the spacecraft and station or landmark, respectively.

At each time step, the spacecraft may choose to receive a range measurement from one of the specified number of ground stations, generate bearing measurements for all visible landmarks, apply an impulsive Δv to adjust the orbit configuration, or do nothing. If either of the measurement actions are taken, the covariance is updated using the standard Kalman update:

$$K = P(t_{k+1})H(HP(t_{k+1})H^T + R)^{-1} \quad (10)$$

$$P^+(t_{k+1}) = (I - KH)P(t_{k+1}) \quad (11)$$

where H is the measurement partials matrix, and R is the user-defined measurement noise. Reference 12 provides a measurement update for the UD-factorized form using a Carlson rank-one update. If the maneuver action is selected, an impulsive $\Delta \vec{v}$ is applied to the spacecraft's velocity component. Maneuver execution errors, defined via the Gates model,¹³ are also applied to the

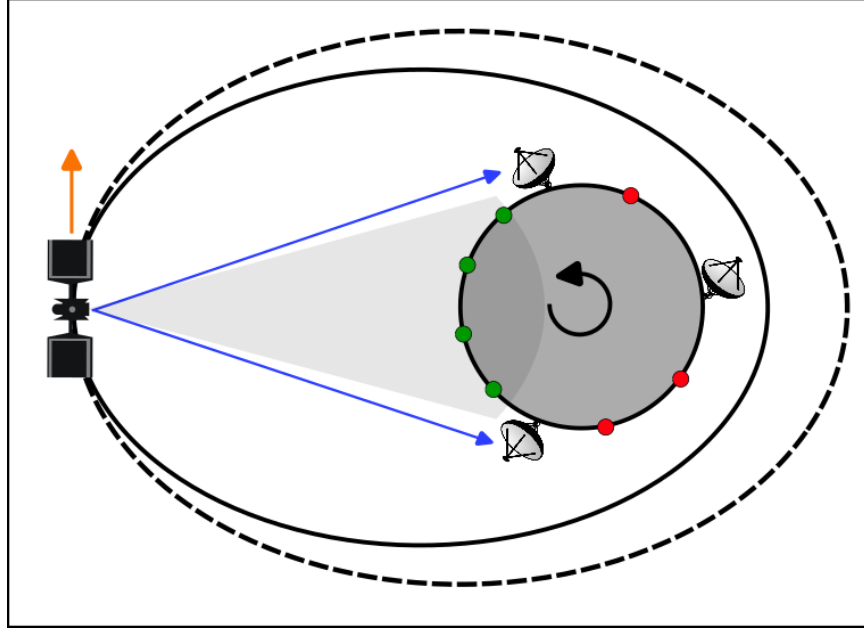


Figure 1. Graphical representation of the proposed planar, 2D spacecraft navigation and characterization example problem.

covariance factors using a Carlson rank-one update. A graphical representation of the example problem formulated above is shown in Figure 1.

This problem can be formulated as a MDP:

$$\mathcal{P} : \{\mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \gamma\} \quad (12)$$

At minimum, the MDP state space, $\mathcal{S}$, consists of the state vector, $\vec{X}$:

$$\mathcal{S} = \{\vec{X}\} \quad (13)$$

An alternative form of $\mathcal{S}$ includes the vectorized covariance factors $\vec{U}$ and $\vec{D}$, as well:

$$\mathcal{S} = \{\vec{X}, \vec{U}, \vec{D}\} \quad (14)$$

where the lower diagonal and identity terms in U are omitted from $\vec{U}$ and $\vec{D} = \text{diag}(D)$. For network training and stability purposes, it may be beneficial to instead use a triangular, square-root form of the factorized covariance matrix:

$$\mathcal{S} = \{\vec{X}, \vec{L}^T\} \quad (15)$$

where $\vec{L}^T$ is a vectorized form of the triangular, square-root covariance matrix*:

*In literature, the upper-triangular square-root factor of the covariance matrix is typically represented by the symbol R , as in $P = RR^T$. However, R is already used to express measurement uncertainty. Therefore, we instead use the relation $P = RR^T = L^T L$, where R is the typical upper-triangular matrix expressed in literature, and L^T is the transpose of the corresponding lower-triangular matrix.

$$L^T = U\sqrt{D} \quad (16)$$

Yet another approach is to omit correlation information and instead provide just the formal sigmas, $\vec{\sigma}$, extracted from the covariance factors:

$$\mathcal{S} = \{\vec{X}, \vec{\sigma}\} \quad (17)$$

The intent for including $\vec{U}$ and $\vec{D}$ (or $\vec{L}^T$ or $\vec{\sigma}$) in $\mathcal{S}$ is to allow knowledge of the formal state uncertainty to influence action selection in addition to the current filter state, $\vec{X}$. Assessing the utility of including formal covariance information in the state space is an expected outcome of the study.

The action space, $\mathcal{A}$, is defined as:

$$\mathcal{A} = \{\text{Null}, \rho_1, \dots, \rho_n, \beta, \Delta v\} \quad (18)$$

where “Null” means take no action, ρ_i means process a single range measurement from station i , β means process bearing measurements from all visible landmarks, and Δv means apply a control action via a specified impulsive $\Delta \vec{v}$. All actions are mutually-exclusive, i.e., only one action can be performed per time step. For this specific scenario, there is no incentive to select the “Null” action and it is largely included for evaluation purposes, e.g., the policy should not select a Null action if it is working properly. However, there are practical circumstances where a Null action could be valid for alternate scenarios. For example, if the total amount of tracking was limited due to resource and/or spacecraft constraints.

The inclusion of a control action, Δv , introduces both discrete and continuous actions to the action space. If the control action is selected, a continuous vector, specified as a magnitude and direction, must also be supplied. Having continuous action parameters that depend on the selection of a discrete action choice forms a parametric action space.¹⁴ How to efficiently incorporate a parametric action space as part of this MDP is a focus of this study.

The transition function, $\mathcal{T}$, is deterministic and equal to the generative function, G , where G performs the time update described in Eqs. (4) & (5) and, depending on the action, a , the measurement update described in Eqs. (10) & (11).

$$\mathcal{T}(s'|s, a) = G(s, a) \quad (19)$$

Since the ultimate goal of this work is to incorporate realistic navigation and environment characterization in a DRL spacecraft planning framework, the reward, $\mathcal{R}$, is constructed as a function of the change in the formal gravitational parameter uncertainty, σ_μ :

$$\mathcal{R}(s(t_k), a(t_k)) = \frac{1}{1 + \sigma_\mu(t_k)} - \frac{1}{1 + \sigma_\mu(t_{k-1})} \quad (20)$$

The last component of the MDP, the discount factor, $\gamma \in \{0, 1\}$, is left as a user-defined value.

PROPOSED METHODOLOGY

The goal of this project is to generate an optimal policy, $\pi(s)$, for the MDP described in the previous section. In other words, we seek a policy that selects the spacecraft navigation or control action that maximizes the future discounted reward, thus minimizing the formal uncertainty in the gravitational parameter, σ_μ , as a proxy for overall environment characterization. The expected value, $\mathcal{V}$, at the current state, s given policy, π is defined as:

$$\mathcal{V}^\pi(s) = \mathcal{R}(s, \pi(s)) + \gamma \sum_{s'} \mathcal{T}(s'|s, a) \mathcal{V}^\pi(s') \quad (21)$$

The policy that maximizes the value for any state is the optimal policy, $\pi^*(s)$:

$$\pi^*(s) = \arg \max_{\pi} \mathcal{V}^\pi(s) \quad \forall s \in \mathcal{S} \quad (22)$$

One approach is to employ Proximal Policy Optimization (PPO)¹⁵ as a solution method given that it is a SOA approach for solving MDPs. Specifically, we leverage the PPO implementation and associated utilities in PyTorch’s TorchRL¹⁶ library. PPO is an online, gradient method for policy optimization using a surrogate objective function which “clips” the ratio between the new and old policy, i.e., large changes to the policy are discouraged between iterations:

$$\mathbf{r}(\Theta) = \frac{\pi_\Theta(a|s)}{\pi_{\Theta_{old}}(a|s)} \quad (23)$$

$$L^{CLIP}(\Theta) = \mathbb{E} [\min (\mathbf{r}(\Theta) A_\Theta, \text{clip}(\mathbf{r}(\Theta), 1 - \epsilon, 1 + \epsilon) A_\Theta)] \quad (24)$$

where Θ is the policy parameterization and $A_\Theta(s, a) = Q(s, a) - V(s)$ is the advantage function. Initial experimentation with PPO has proved promising; however, other solution methods may be investigated, including off-policy methods that may be more sample-efficient and better enable policy exploration.

In order to address the challenge of balancing navigation and actuation actions in this problem, we propose employing a parametric action space formulation similar to Reference 14. This study will explore methods for structuring and modeling the parametric action space, including how to represent the underlying compound distribution consisting of both discrete and continuous components.

In all cases, policies are trained using episodes with spacecraft in a range of randomly-selected orbits. Scenario termination occurs after a fixed number of time steps. Both the maximum duration, t_{max} , and the step size, Δt , are configurable parameters. Dynamics and measurement modeling leverage PyTorch `Tensor` objects contained in TorchRL `TensorDicts`. State and covariance propagation is performed using the torchode numerical integration package,¹⁷ which accepts `Tensors` containing the initial states as inputs, as well as a pointer to a function that returns PR2BP derivatives.

EXPECTED OUTCOMES

Initial experiments using PPO on the MDP described above without the control action option resulted in policies that significantly exceeded an initial, random policy and had equivalent performance to pseudo-optimal heuristic policies. In this paper, we will expand on those results to incorporate the control action, which allows the spacecraft to reconfigure the orbit to improve measurement visibility and gravitational parameter sensitivity. The expected result is a methodology that produces optimal policies for the full MDP using PPO (or a similar DRL algorithm). **The expected innovations for this work include:**

- Including formal covariance propagation and measurement modeling that depends on action selection as part of the MDP state space and transition function and assessing the utility of uncertainty representation in the state space;
- Utilizing a reward function that incentivizes both navigation performance and environment characterization;
- Incorporating a parametric action space with mutually-exclusive discrete measurement and continuous control actions;
- Comparisons with alternative methods and pseudo-optimal heuristic policies;
- Gradual increase in environment fidelity to gain intuition and insight that is applicable to more realistic scenarios.

If successful, the resulting MDP formulation and solution methodology will address realistic navigation and environment characterization considerations in the spacecraft planning and autonomy problem, including actuation. The expected outcomes will enable more realistic studies with higher-fidelity dynamics and measurement models, additional environment parameters in the state space and reward (e.g., non-spherical gravity coefficients), other science observation objectives, and spacecraft systems and operations constraints, all of which must be considered for actual spaceflight missions.

REFERENCES

- [1] K. M. Getzandanner, K. E. Berry, P. G. Antreasian, J. M. Leonard, C. D. Adam, D. R. Wibben, M. C. Moreau, D. E. Highsmith, and D. S. Lauretta, “Small-Body Proximity Operations & TAG: Navigation Experiences & Lessons Learned from the OSIRIS-REx Mission,” *The 32nd AIAA/AAS Space Flight Mechanics Meeting, San Diego, CA*, Jan. 2022.
- [2] P. G. Antreasian, C. D. Adam, B. W. Ashman, K. Berry, J. L. Geeraert, K. M. Getzandanner, D. E. Highsmith, J. M. Leonard, E. Lessac-Chenen, A. Levine, J. V. McAdams, L. K. McCarthy, M. Moreau, D. Nelson, B. Page, J. Y. Pelgrift, S. Rieger, E. Sahr, D. Wibben, B. Williams, K. Williams, and D. Lauretta, “OSIRIS-REx Proximity Operations and Navigation Performance at (101955) Bennu,” *The 32nd AIAA/AAS Space Flight Mechanics Meeting, San Diego, CA*, Jan. 2022.
- [3] M. Stephenson and H. Schaub, “Reinforcement Learning for Earth-Observing Satellite Autonomy with Event-Based Task Intervals,” *46th AAS Guidance and Control Conference*, Breckenridge, CO, Feb. 2024.
- [4] A. Herrmann and H. Schaub, “Reinforcement Learning for the Agile Earth-Observing Satellite Scheduling Problem,” *IEEE Transactions on Aerospace and Electronic Systems*, Oct. 2023, 10.1109/TAES.2023.3251307.
- [5] A. Harris, T. Valade, T. Teil, and H. Schaub, “Generation of Spacecraft Operations Procedures Using Deep Reinforcement Learning,” *Journal of Spacecraft and Rockets*, Vol. 59, Feb. 2022, pp. 611–626, 10.2514/1.A35169.

- [6] M. Tipaldi, R. Iervolino, and P. R. Massenio, “Reinforcement learning in spacecraft control applications: Advances, prospects, and challenges,” *Annual Reviews in Control*, Vol. 54, Jan. 2022, pp. 1–23, 10.1016/j.arcontrol.2022.07.004.
- [7] A. Brandonsio, L. Capra, and M. Lavagna, “Deep Reinforcement Learning Spacecraft Guidance with State Uncertainty for Autonomous Shape Reconstruction of Uncooperative Target,” *Advances in Space Research*, 2023, 10.1016/j.asr.2023.07.007.
- [8] D. Chan and A. Agha-Mohammadi, “Autonomous Imaging and Mapping of Small Bodies Using Deep Reinforcement Learning,” *IEEE Aerospace Conference*, Big Sky, MT, Mar. 2019.
- [9] A. Herrmann and H. Schaub, “Reinforcement Learning for Small Body Science Operations,” *AAS Astrodynamics Specialist Conference*, Charlotte, NC, Aug. 2022.
- [10] S. Takahashi and D. J. Scheeres, “Autonomous Reconnaissance Trajectory Guidance at Small Near-Earth Asteroids via Reinforcement Learning,” *Journal of Guidance, Control, and Dynamics*, Vol. 46, July 2023, pp. 1280–1297, 10.2514/1.G007043.
- [11] Z. Chen, H. Cui, and Y. Tian, “Autonomous Maneuver Planning for Small-Body Reconnaissance via Reinforcement Learning,” *Journal of Guidance, Control, and Dynamics*, May 2024, pp. 1–13, 10.2514/1.G008011.
- [12] J. R. Carpenter and C. N. D’Souza, “Navigation Filter Best Practices,” Tech. Rep. NASA/TP–2018–219822, 2018.
- [13] C. R. Gates, “A Simplified Model of Midcourse Maneuver Execution Errors,” Tech. Rep. 32-504, JPL, Pasadena, CA, 1963.
- [14] W. Masson, P. Ranchod, and G. Konidaris, “Reinforcement Learning with Parameterized Actions,” Oct. 2015.
- [15] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, “Proximal Policy Optimization Algorithms,” 2017.
- [16] A. Bou, M. Bettini, S. Dittert, V. Kumar, S. Sodhani, X. Yang, G. D. Fabritiis, and V. Moens, “TorchRL: A data-driven decision-making library for PyTorch,” June 2023.
- [17] M. Lienen and S. Günnemann, “torchode: A Parallel ODE Solver for PyTorch,” *The Symbiosis of Deep Learning and Differential Equations II, NeurIPS*, 2022.