(Preprint) AAS 22-841

SMALL BODY RECONNAISSANCE BY MULTIPLE SPACECRAFT VIA DEEP REINFORCEMENT LEARNING

Kento Tomita[*], Yuri Shimane[†], and Koki Ho [‡]

Small-body investigations by spacecraft are one of the most scientifically important space exploration missions. Due to the strong uncertainty of the dynamics around the body, geological surface features, and scientific values of candidate target sites, these missions require dedicated planning and execution from the ground. As a study of automated operations for asteroid investigation, this paper investigates how small-body reconnaissance operations could be performed by multiple spacecraft. By comparing baseline policies with different model parameters and a policy trained via deep reinforcement learning, we discuss the optimal balance of exploration and exploitation for our science model.

INTRODUCTION

Studying small bodies, such as asteroids and comets, is crucial to unraveling the formation and evolution of the solar system, which have motivated many missions for detailed observation. After the first sample-return of dust particles from comet Wild 2 by Stardust[1] in 1999, missions involving touchdown on the surface of small bodies were conducted by Hayabusa[2] (2003), Rosetta[3] (2004), Hayabusa2[4] (2014), and OSIRIS-REx[5] (2016). These missions demonstrated the scientific impact brought by proximity surveys, and also their operational challenges. Detailed survey and reconnaissance of surface cost dedicated planning and execution from the ground, and there is a need for more advanced autonomy for operations. In this regard, this paper studies how small-body reconnaissance operations could be performed by multiple spacecraft with deep reinforcement learning.

The advantage of using a distributed, multiple spacecraft architecture for the exploration of small celestial bodies has been a topic gaining interest since the turn of this century.[6] Such architectures are typically collaborative, potentially hierarchical, and have relatively high autonomy; this makes them suitable for rapid investigation in-situ, adapting to the a priori unknown environment with little or no human involvement in the loop. NASA's Autonomous Nanotechnology Satellites (ANTS) concept mission[6–8] is one such example, where a distributed system of 1000 picosatellites was to explore the asteroid main belt. In this architecture, some spacecraft would assume the role of "workers", conducting scientific observation in-situ, while a smaller number of "coordinator" spacecraft would determine the overall action of the swarm. This type of organization, otherwise referred to as "mother-daughter" architectures, has since also been adapted for a more ambitious sample return

[*] PhD Student, Daniel Guggenheim School of Aerospace Engineering, Georgia Institute of Technology, Atlanta, GA 30332, AIAA Student Member.

[†] PhD Student, Daniel Guggenheim School of Aerospace Engineering, Georgia Institute of Technology, Atlanta, GA 30332, AIAA Student Member.

[‡] Assistant Professor, Daniel Guggenheim School of Aerospace Engineering, Georgia Institute of Technology, Atlanta, GA 30332, AIAA Senior Member.

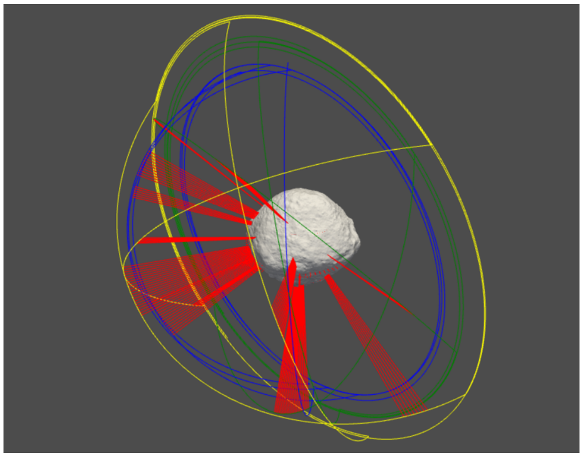


Figure 1: Multi spacecraft reconnaissance operations using the trained RL agents, shown in inertial frame. Blue, green, yellow trajectories represent each of three different spacecraft in the swarm. The red lines represent the line-of-sight connecting spacecraft and surface targets. The detailed results are described in the section on Numerical Analysis.

mission from multiple main-belt asteroids.[9] In contrast, studies such as ESA's Asteroid Population Investigation and Exploration Swarm (APIES) concept[10] consider a swarm constituted by a smaller number of identical small satellites, deployed once arriving at the asteroid belt.

Recently, a significant number of authors have studied the operational aspect of having multiple spacecraft in the vicinity of small bodies. For the purpose of motion planning, Bandyopadhyay et al[11] combines a distributed assignment algorithm for the space allocation problem and sequential convex programming for the trajectory design problem of each spacecraft to demonstrate motion-planning in a cluttered, time-varying environment. Wang et al[12] posed an optimal control problem of multiple spacecraft that minimizes the fuel consumption while also ensuring no relative drift in terms of mean orbital elements, as small bodies typically lead to osculating orbital elements due to strong perturbing forces present. In a similar fashion, Lippe and D'Amico[13] leveraged an extended Kalman filter (EKF) to map the osculating elements to mean relative orbital elements (ROE), which is then used as part of the control problem to maintain the formation in the mean ROE space. Nallapu and Thangavelautham[14] developed the Automated Swarm Designer for small body observation by multiple spacecraft by optimizing their attitudes during a flyby.

For the purpose of autonomous navigation, the information collected by members of a swarm may be combined to augment both the spacecraft state and knowledge of the dynamical environment; Verisano and Vasile[15] demonstrated an improvement in navigation performance through the use of such data links. Similarly, Stacey et al[16] proposed an algorithm that leverages optical observation of multiple spacecraft via inter-satellite RF communication links to estimate both the states and the asteroid's gravitational field, rotational motion, and landmark positions.

Applications of reinforcement learning (RL) in the context of operations around asteroids, albeit for a single spacecraft, have also been gaining traction. A particular advantage of using RL is due to its ability to adapt the control law to a dynamics that is unknown ahead of time; this is particularly suitable for asteroid exploration, where the irregularities of the body may not be known to a great extent before taking in-situ measurements. Willis et al[17] achieve hovering motion over a tumbling small body with unknown gravitational parameters through the use of a direct policy search, relying only on optic flow observables. Such hovering motions, enabled by the relatively high maneuverability in the vicinity of small bodies, are advantageous both for concentrated observation of a target of interest via a body-fixed hovering, or mapping the entire surface of the body via an inertial-frame hovering. For the purpose of close-proximity operations, Gaudet et al[18,19] propose a policy mapping LIDAR measurements to controls for a 6-degree-of-freedom spacecraft. Federici et al[20] apply a similar policy mapping for an impactor, demonstrating the algorithm on the DART mission scenario. The use of a physics-informed neural-network (PINN)-based gravity model together with RL has been proposed by,[21] where a highly representative gravity model is used for training the RL agent without compromising on the computational cost for propagating the dynamics. Another notable use of an RL architecture, proposed by Piccinin et al,[22] aims to improve the surface mapping efficiency by scheduling the image acquisition process.

This paper focuses on autonomous asteroid reconnaissance operations and studies the performance of multiple spacecraft architecture with a policy trained by deep reinforcement learning. We consider a cooperative sequential decision-making problem where every step, each spacecraft determines which site to investigate in what condition. In a simulated environment, the trained policy exhibits the sixth best performance with the narrowest performance variation against nine different baseline policies. By comparing the best and worst policies, we discuss the balance of exploration and exploitation for our problem.

PROBLEM STATEMENT

The objective of our problem is to maximize the science about the target small body with multiple spacecraft, assuming the sites of interest are already identified. We assume that a global mapping phase reveals the location of these sites but their detailed values are still unknown. Through reconnaissance operations, each spacecraft accumulates knowledge about these sites, under restricted inter-spacecraft communication. Within a certain mission window, the swarm of spacecraft needs to cooperate in collecting as much science as possible.

Science Model

The objective function of our problem is the cumulative sum of the science gain. This subsection details the assumptions of our science model that defines science gains by each reconnaissance operation, which results in Eq. 4.

Ground Truth Site Values For a given list of sites on a small body, we define a segmented *value* distribution (Fig. 2). One reconnaissance operation about a site reveals a part of the segmented value up to the quality of observation. Specifically, let $i \in I = \{1, 2, 3, ...\}$ denote the site of interest with I being the set of sites. We define $V_i \in R^n$ that represents the value distribution of the site i. For example, $V_i = [0, 10, 0.5]$ means that the scientific value of site i has three segments whose value ranges from 0 to 10. Depending on the observation condition, the spacecraft may cover all or part of the segments and collect their value up to a certain quality.

Property of Reconnaissance We give each reconnaissance operation the property of *coverage*, *quality*, and *emission angle*. Coverage, $c \in R$, represents how many segments the operation covered. Quality, $q \in R$, defines the rate of value collection for each segment. Emission angle, $\theta \in R$, is the angle between the surface normal vector and the direction from the surface to the spacecraft. Coverage, quality, and the emission angle are determined by the mean of the feasible observation

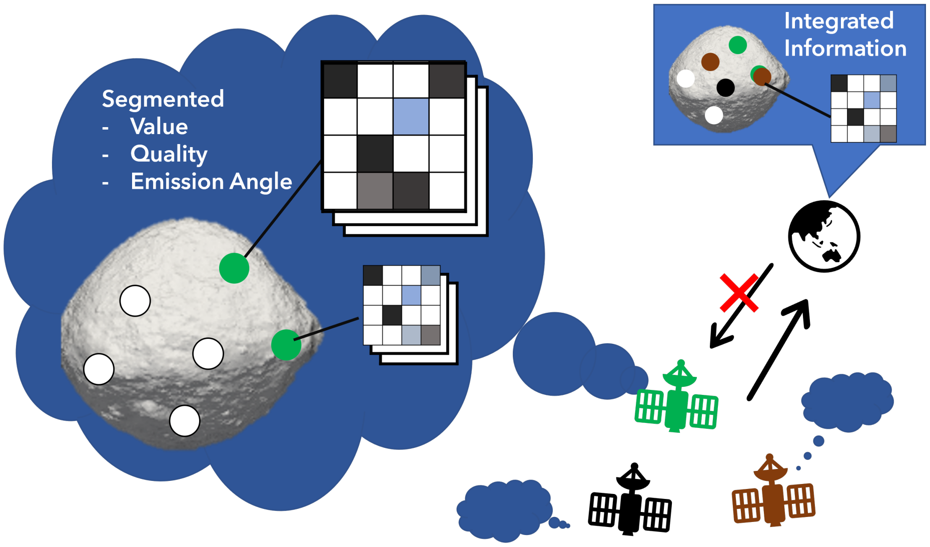


Figure 2: Concept of small body reconnaissance by multiple spacecraft.

period over the fly-over trajectory. We model the feasibility of observation, coverage, and quality as the function of the Sun direction and the position of spacecraft and target site by setting minimum and maximum bounds for slant range, slant angle, and solar incidence angles.

Map State To compute the collected science, we define a map state $M(a) = \{M_i(a)\}_{i \in I}$, where $a \in A$ denotes a spacecraft or the ground station and A represents the union of the ground station and the set of spacecraft. The map state of spacecraft a for a site i is represented as $M_i(a) = \{\tilde{V}_i(a), Q_i(a), T_i(a), H_i(a)\}$. $\tilde{V}_i(a) \in R^n$ represents the collected site value for each segment of the site i. $Q_i(a) \in R^n$ records the highest reconnaissance quality for each segment. $T_i(a)$ stores a unique set of the emission angles of observation for each segments and $H_i(a) \in R^n$ represents the variation of the emission angles for each segments for site i. The variation of the emission angles for each segment is calculated as an *entropy*. Suppose indices j represent each segment of site i and indices $k(j)$ denotes each of the unique emission angles of observation for the segment j of site i. We can describe $T_i(a)$ and $H_i(a)$ as $T_i(a) = \{T_{i,j}(a)\}_i = \{\theta_{i,j,k}(a)\}$ and $H_i(a) = \{h_{i,j}(a)\}_i$ where $h_{i,j} \in R$. Then the variation of the emission angles are,

$$h_{i,j} = -\sum_k \frac{1}{n(j)} \log_2\left(\frac{1}{n(j)}\right) = \log_2\left(n(j)\right) \tag{1}$$

where $n(j)$ is the number of unique emission angles for segment j of site i. Note that emission angles are grouped by 10 degrees interval, so the number of unique emission angles per segment can be as large as eight because the emission angle are in the interval of $[0, 90)$ in degree. Therefore, the maximum entropy per segment is three.

Updating Map State Each reconnaissance operation for site i updates the map state of the observer spacecraft about the observed site i, $M_i(a)$, and that of ground station, $M_i(g)$, where g represents the ground station. Suppose the reconnaissance operation has the property of coverage, c, quality, q, and emission angle, θ. Then, the segments to be updated are randomly chosen whose number is proportional to c. Any chosen segment of $Q_i(a)$ and $Q_i(g)$ whose value is less than q is updated to be q, and the rest remains the same. Similarly, if any chosen segment of $T_i(a)$ and $T_i(g)$ adds the emission angle θ in the set unless it has been registered. The entropy $H_i(a)$ and $H_i(g)$ are then updated. The collected value, $\tilde{V}_i$ are updated by multiplying the true site value V_i and the observation quality Q_i for each segment:

$$\tilde{V}_i(a) \leftarrow V_i \odot Q_i(a) + \epsilon \tag{2}$$

where $\odot$ is an element wise multiplication and $\epsilon \in R^n$ is the observation noise. Note that for the ground station, this noise is set zero:

$$\tilde{V}_i(g) \leftarrow V_i \odot Q_i(g). \tag{3}$$

As an example, let's consider a site value of $V_i = [0, 10, 0.5]$. By the reconnaissance with the property of $c = 0.7$, $q = 0.5$, and $\theta = 20$ deg, the map state at $t = t_1$ is updated as follows:

$$\begin{aligned} M_i(t = t_1) = \{\tilde{V}_i &= [0, 1.0, 0], \quad Q_i = [0.5, 0.1, 0], \\ T_i &= [[10,], [5,], \emptyset], \quad H_i = [0, 0, 0]\} \\ M_i(t = t_2) = \{\tilde{V}_i &= [0, 5.0, 0], \qquad Q_i = [0.5, 0.5, 0], \\ T_i &= [[10, 20], [5,], \emptyset], \quad H_i = [1, 0, 0]\} \end{aligned}$$

Science Gain The science gain brought by each reconnaissance operation is computed based on the collected site value $\tilde{V}_i(g)$ and the entropy of the emission angles $H_i(g)$ of the ground station. Let r_{sci} denote the science gain by the reconnaissance for site i in the step from time $t = t_1$ to $t = t_2$, then

$$\begin{aligned} r_{sci} = &\sum_j \tilde{v}_{i,j}(g, t_2)\,(1 + h_{i,j}(g, t_2)) \\ &- \sum_j \tilde{v}_{i,j}(g, t_1)\,(1 + h_{i,j}(g, t_1)) \end{aligned} \tag{4}$$

where $v_{i,j}$ and $h_{i,j}$ represents the value, $\tilde{V}_i(g)$, and the emission angle entropy, $H_i(g)$, for j segment of site i, respectively. To maximise the science gain Eq. 4, each spacecraft need to visit a site with high scientific value that has not been collected much while maximising the variation of emission angles of observation expressed by the entropy. It is important to take the balance of exploration and exploitation among the swarm of spacecraft. Here exploitation is to increase the variation of emission angles, H_i, for the site with high estimated value, $\tilde{V}_i$, and exploration is to visit the site with less collected value irrespective of the estimated site value.

Action Space and Trajectory Design

Over the mission window, each spacecraft need to optimize its choice of 1) departure time, 2) target site, 3) periapsis altitude, and 4) solar incidence angle for their reconnaissance, under the trajectory restrictions as follows. We consider two types of trajectories; Sun-terminator orbits for the parking orbit and flyby trajectories for the reconnaissance operations, referencing the OSIRIS-REx reconnaissance trajectory design.[23] Sun-terminator orbits are near circular orbit on the Sun-terminator plane. We consider the spacecraft stay in the parking orbit unless it performs reconnaissance operations. For the reconnaissance, the spacecraft leaves to a trajectory that flies over the

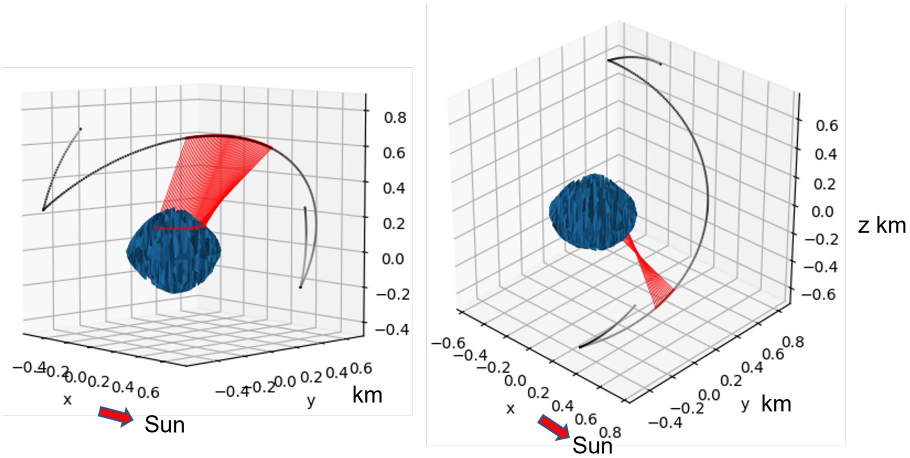


Figure 3: Sample reconnaissance trajectory.

target site latitude at a designated solar incidence angle. Depending on the phase of the small body rotation and parking orbit, the angle and range conditions for the observation change. Then the fly-over trajectory returns to the parking orbit. Thus, the fly over trajectory is identified by choice of departure time, periapsis altitude, and target site. Figure 3 shows the examples of trajectories considered. To reduce the collision risk, each spacecraft is assigned to an initial parking orbit with a different radius. We also restrict only one spacecraft that operates reconnaissance at a time.

COOPERATIVE RECONNAISSANCE AGENT

This paper investigates and compare the performance of two types of agents for the cooperative asteroid investigation. The first agent adopts a model-free deep reinforcement learning algorithm called Proximal Policy Optimization[24] (PPO). The second agent is a handcrafted baseline agent with a rule-based policy. Note that the baseline agent, unlike the RL agent, simulates reconnaissance operation for every candidate to explicitly use feasible observation time and delta-v required in their policy.

Proximal Policy Optimization

Proximal policy optimization is an online policy gradient method for reinforcement learning. It optimizes a parameterized policy function by gradient descent to maximize an objective tied to cumulative episode reward. PPO restricts the resulting difference of the surrogate objective by one step of the policy gradient, and in tern enables multiple-step minibatch updates more safely. This objective is called clipped surrogate objective, which is defined as

$$L^{CLIP} = \mathrm{E}\left[\min(r_t(\theta)\hat{A}_t, \mathrm{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t)\right], \tag{5}$$

where θ is the parameter of policy function and $r_t(\theta)\hat{A}_t$ represents the advantage of the parameter being θ than the old value. To compute $\hat{A}_t$, we need to approximate value function too and often the value function and policy function share some parameters. The objective then includes the terms for the value function, which result in

$$L_T = \hat{\mathrm{E}}_t\left[L^{CLIP}(\theta) - c_1 L_t^{VF}(\theta) + c_2 S[\pi]\right], \tag{6}$$

where c_1, c_2 are coefficients, and S denotes an entropy bonus to ensure sufficient exploration, and L_t^{VF} is a squared-error loss for the value function.

We train a single policy network by PPO used for all of the spacecraft. The definition of output action and input observation are described in the followings.

Action Space As described in subsection , the action space for each spacecraft consists of 1) departure time, 2) target site, 3) periapsis altitude, and 4) solar incidence angle for their reconnaissance. Since the choice of the target site is discrete, this study makes all the other actions also discrete for a simpler implementation. For the first item of action, the departure time, every time step agent makes the decision if it goes reconnaissance or not. For the other actions, agent picks from the given list depending on the environment configuration.

Observation Space To make optimal decisions, ideally, the spacecraft needs to know for each target 1) the relative position, 2) your map state of science, and 3) other spacecraft's map state of the science. Although we give complete information for the first item, for the other two information, we restrict their knowledge; to reduce the input dimension for your own map state, and to limit the communication between spacecraft for other spacecraft's map state. Table 1 shows the observation space for each spacecraft.

Table 1: Observation space for the RL agent. n_i represents the number of target sites.

Variable	**Size**
Position of spacecraft	3
Target position	$3n_i$
Estimated value*	n_i
Cumulative coverage*	n_i
Mean quality*	n_i
Mean entropy of emission angle*	n_i
Number of visits by other spacecraft	n_i

* These values are based on the map state M_i.

Reward For each reconnaissance operation, the agent takes the reward defined by the weighted sum of the scientific gain for the entire system (Eq. 4) and delta-v cost for the reconnaissance. Let $w_{sci} > 0, w_{dv} < 0$ represents the coefficient for the science gain and delta-v cost, respectively. Then the reward is described as

$$r = w_{sci} r_{sci} + w_{dv} dv. \tag{7}$$

Baseline Agent

As a baseline, we consider an agent with a rule-based policy. Note that this baseline agent uses additional information than shown in the RL agent. Specifically, every time step, the baseline agent simulates reconnaissance operation for each site to compute the feasible surface observation time and delta-v required. The rule-based policy takes the input shown in Table 2. We consider three parameters to define the rule-based policy; minimum observation time, t_{min}, maximum delta-v, dv_{max}, and maximum number of visits by others, n_{max}. The algorithm of the baseline agent is shown in Algorithm 1.

The baseline agent first checks if the minimum number of visits by others among all sites, denoted by n_{min}, is less than the given n_{max}. If this condition is violated, it updates n_{max} to $n_{max} \times$

$\lfloor n_{min}/n_{max} \rfloor$. Next, it screens the reconnaissance operation for each site by the thresholds of t_{min}, dv_{max}, and n_{max} to obtain feasible targets. If there is any site never visited among the feasible target, pick this target. If all the feasible targets have been visited at least once, it chooses to visit the site with the highest estimated value. If none of the sites meets the three thresholds, then stay in the parking orbit for that time step.

Table 2: Observation space for the baseline agent. Here n_i represents the number of target sites.

Variable	**Size**
Feasible observation time	n_i
Delta-V required	n_i
Estimated value*	n_i
Number of visits by other spacecraft	n_i

* These values are based on the map state M_i.

Algorithm 1 Baseline agent policy

```
Input: $t_{min}, dv_{max}, n_{max}$
  Set of target sites, $I$,
  Set of altitudes, $R_a$,
  Set of solar incidence angles, $S$
Output: Action
  Action $\leftarrow$ No Go                                        ▷ Initialize action
  $v_{best} \leftarrow 0$                                           ▷ Initialize best site value
  $n_{min} \leftarrow$ The minimum number of visits for all sites by others
  $n_{max} \leftarrow n_{max}(1 + \lfloor n_{min}/n_{max} \rfloor)$   ▷ Adjust if all sites are visited more than $n_{max}$ times by others
  For: $i, r_a, s \in I \times R_a \times S$ do                        ▷ Iterate for all actions
     $t_{obs}$, $dv \leftarrow f(i, r_a, s)$                        ▷ Simulate observation time and delta-v
     Fetch $n_i$, the number of visits by others for site $i$
     if $t_{obs} > t_{min}$ and $dv < dv_{max}$ then
        if Site $i$ is never visited by yourself then
            Action $\leftarrow$ {Go, $i, r_a, s$}
           break
        else if $n_i < n_{max}$ then                         ▷ If $i$ is visited by others less than $n_{max}$ times
           if $v(i) > v_{best}$ then                          ▷ If $i$ has the highest estimated value
              $v_{best} \leftarrow v(i)$                                 ▷ Update best value
              Action $\leftarrow$ {Go, $i, r_a, s$}
           end if
        end if
     end if
  end For:
```

NUMERICAL ANALYSIS

Simulator Configuration

To train the RL agent and to demonstrate the performance of the agents, we built a randomized asteroid reconnaissance environment. This paper, as an example, uses the asteroid Bennu whose parameter is shown in Table 3. We demonstrate the results where we have three spacecraft and 12 target sites with three options for reconnaissance altitudes and two for solar incidence angles. The details of the environment parameters are shown in Table 4.

Table 3: Asteroid parameters.[25,26]

Parameter	Value
Name	101955 Bennu
GM, km^3/s^2	$4.8904e-9$
C20	-0.05812
C22	0.00320
Diameter, km	0.482
Rotation period, h	4.296061

Table 4: Environment parameters.

Parameter	Values
Time step, h	1
Maximum total number of reconnaissance	72
Number of spacecraft	3
Initial parking orbit radius, km	$0.9, 1.0, 1.1$
Initial fuel, km/s	0.1
Number of target sites	12
Mean site value	$1, 1, 1, 3, 3, 3, 5, 5, 5, 7, 7, 7$
Standard deviation of site value	$1, 1, 1, 1, 2, 3, 1, 3, 5, 1, 5, 7$
Observation noise (Eq. 2), 1-σ	0.03
Altitudes for reconnaissance, km	$0.2, 0.4, 0.6$
Solar incidence angles for reconnaissance, deg	$-45, 45$

Results and Discussion

The main objective of our problem is to maximize the collected science.

Table 5 and Fig. 4 shows the mean and distribution of final collected science of over 10 random episodes, respectively. We have nine baseline agents with different model parameters and one RL agent. The baseline agents have wide variations of performance depending on the parameter of n_{max} and t_{min}. The RL agent's performance ranks sixth against all the nine baseline agents. It also shows that the variance of the collected science is the smallest for the RL agent.

Figure 5 shows the collected science over steps and the relationship between step and time for the RL agent and the best and worst baseline agents, in the sense of mean final collected science. The step-to-time relation is not constant because any reconnaissance operation is considered one

Table 5: Average final collected science over 10 random episodes. The arguments of baseline corresponds to (n_{max}, t_{min}), respectively. The unit for t_{min} is seconds. dv_{max} are set 0.001 km/s for all the baseline agent.

Rank	Agent	Science
1	Baseline(1, 0)	28.8
2	Baseline(1, 1800)	27.8
3	Baseline(5, 3600)	25.4
4	Baseline(1, 3600)	23.7
5	Baseline(10, 3600)	22.5
6	RL	21.1
7	Baseline(10, 0)	19.6
8	Baseline(10, 1800)	19.4
9	Baseline(5, 0)	18.8
10	Baseline(5, 1800)	18.3

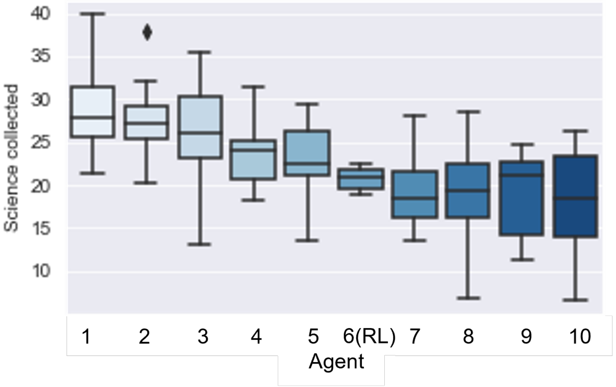
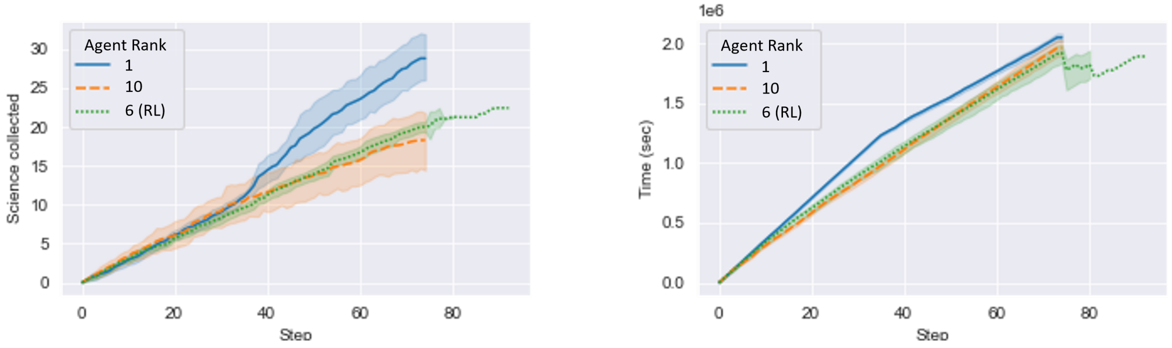


Figure 4: Distributions of final collected science over 10 random episodes. Ten agents are sorted in terms of the average final collected science. The arguments of baseline corresponds to (n_{max}, t_{min}), respectively. The unit for t_{min} is seconds. dv_{max} are set 0.001 km/s for all the baseline agent.

step. If no spacecraft is going reconnaissance, the default time step of 3600 seconds is taken. You can see the RL agent takes more steps before reaching the maximum reconnaissance operations of 72 steps than the other baseline agents. This means that the RL agent often chooses not going reconnaissance, unlike the other baseline agents. This difference in total number of steps also explains that the RL agent outperforms the worst baseline agent in collecting science although they have similar step-to-science ratio.

Figure 5 also shows that the best baseline agent increases the gap from the other two in the steps between 30 to 40. To study the cause of the gap increase, we further look into the distribution of collected value and entropy, which are the two main factors of the science gain of Eq. 4.

Figure 6 shows the distribution of collected value and entropy over four different steps. Note that the entropy represents the variation of the emission angles. Each row from top to bottom represents the best baseline, the worst baseline agent, and RL, respectively. The top row of Fig. 6



(A)

(B)

Figure 5: (A) Collected science over steps and (B) relationship between step and time for the RL agent and the best and worst baseline agents. The step-to-time relation is not constant because any reconnaissance operation is considered as one step. If no spacecraft is going reconnaissance, the default time step of 3600 seconds are taken.

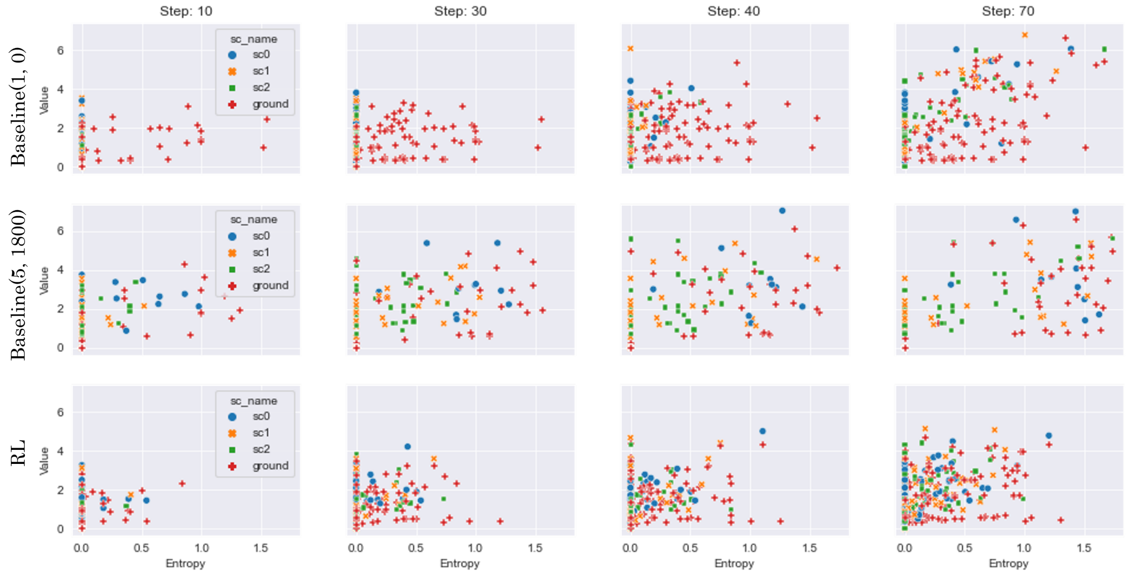


Figure 6: Comparison of the distribution of collected value and entropy over four different steps among three different policies.

shows that the trend of distribution switches from steps 30 and 40. Until step 30, the best baseline agent lets every spacecraft collect new values instead of increasing entropy. From step 40, the best baseline agent allows each spacecraft to increase the variety of sites where larger values are already collected. In step 70, the best agent successfully increases the ground science with high value and high entropy by increasing the points on the top right. On the other hand, the worst baseline agent, shown in the bottom row of Fig. 6, increases the variation of emission angles from step 10. The RL agent have denser distribution in the low-entroy segment. This means the RL policy weights more on exploration of value-increase than exploitation of entropy-increase, compared with the baseline agents.

CONCLUSION

This paper formulated a small body reconnaissance operation problem by multiple spacecraft and studied the performance of rule-based policies and a policy trained by deep reinforcement learning (RL). The rule-based policies result in a wide variation of performance depending on the choice of their parameters. The policy trained via RL performs outperforms baseline policies for four different parameter cases and performs less than five other parameter cases. We studied the exploration and exploitation trade-off for different policies and the RL policy exhibited more weight on exploration compared to our baseline policies. Future work will increase the fidelity of the dynamics and the uncertainty of the input for each agent with further investigation of the resulting policies.

REFERENCES

[1] D. Brownlee, "The Stardust mission: analyzing samples from the edge of the solar system," *Annual Review of Earth and Planetary Sciences*, Vol. 42, 2014, pp. 179–205.

[2] J. Kawaguchi, A. Fujiwara, and T. Uesugi, "Hayabusa—Its technology and science accomplishment summary and Hayabusa-2," *Acta Astronautica*, Vol. 62, No. 10-11, 2008, pp. 639–647.

[3] K.-H. Glassmeier, H. Boehnhardt, D. Koschny, E. Kührt, and I. Richter, "The Rosetta mission: flying towards the origin of the solar system," *Space Science Reviews*, Vol. 128, No. 1, 2007, pp. 1–21.

[4] S.-i. Watanabe, Y. Tsuda, M. Yoshikawa, S. Tanaka, T. Saiki, and S. Nakazawa, "Hayabusa2 mission overview," *Space Science Reviews*, Vol. 208, No. 1, 2017, pp. 3–16.

[5] D. Lauretta, S. Balram-Knutson, E. Beshore, W. Boynton, C. Drouet d'Aubigny, D. DellaGiustina, H. Enos, D. Golish, C. Hergenrother, E. Howell, *et al.*, "OSIRIS-REx: sample return from asteroid (101955) Bennu," *Space Science Reviews*, Vol. 212, No. 1, 2017, pp. 925–984.

[6] W. F. Truszkowski, M. G. Hinchey, J. L. Rash, and C. A. Rouff, "Autonomous and autonomic systems: A paradigm for future space exploration missions," *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, Vol. 36, No. 3, 2006, pp. 279–291, 10.1109/TSMCC.2006.871600.

[7] W. Truszkowski, J. Rash, C. Rouff, and M. Hinchey, "Asteroid exploration with autonomic systems," *11th IEEE International Conference and Workshop on the Engineering of Computer-Based Systems, ECBS 2004*, 2004, pp. 484–489, 10.1109/ECBS.2004.1316737.

[8] C. Rouff, "Intelligence in future NASA swarm-based missions," *AAAI Fall Symposium - Technical Report*, Vol. FS-07-06, 2007, pp. 112–115.

[9] L. D. Vance, J. Thangavelautham, and E. Asphaug, "Evaluation of mother-daughter architectures for asteroid belt exploration," *AIAA Scitech Forum*, No. January, 2019, pp. 1–8, 10.2514/6.2019-0618.

[10] P. D'Arrigo and S. Santandrea, "APIES: A mission for the exploration of the main asteroid belt using a swarm of microsatellites," *Acta Astronautica*, Vol. 59, No. 8-11, 2006, pp. 689–699, 10.1016/j.actaastro.2005.07.011.

[11] S. Bandyopadhyay, F. Baldini, R. Foust, S. J. Chung, A. Rahmani, J. P. d. l. Croix, and F. Y. Hadaegh, "Distributed spatiotemporal motion planning for spacecraft swarms in cluttered environments," *AIAA SPACE and Astronautics Forum and Exposition, SPACE 2017*, No. 203999, 2017, pp. 1–10, 10.2514/6.2017-5323.

[12] W. Wang, G. Mengali, A. A. Quarta, and H. Baoyin, "Time-optimal formation establishment around a slowly rotating asteroid," *Journal of Guidance, Control, and Dynamics*, Vol. 44, No. 4, 2021, pp. 889–897, 10.2514/1.G005624.

[13] C. Lippe and S. D'Amico, "Spacecraft swarm dynamics and control about asteroids," *Advances in Space Research*, Vol. 67, No. 11, 2020, pp. 3426–3443, 10.1016/j.asr.2020.06.037.

[14] R. Teja Nallapu and J. Thangavelautham, "Attitude control of spacecraft swarms for visual mapping of planetary bodies," *2019 IEEE Aerospace Conference*, IEEE, 2019, pp. 1–16.

[15] M. Vetrisano and M. Vasile, "Autonomous navigation of a spacecraft formation in the proximity of an asteroid," *Advances in Space Research*, Vol. 57, No. 8, 2016, pp. 1783–1804, 10.1016/j.asr.2015.07.024.

[16] N. Stacey, K. Dennison, and S. D. . Amico, "Autonomous Asteroid Characterization through Nanosatellite Swarming," *IEEE Aerospace Conference*, 2022.

[17] S. Willis, D. Izzo, and D. Hennes, “Reinforcement learning for spacecraft maneuvering near small bodies,” *AAS/AIAA Space Flight Mechanics Meeting*, Vol. 158, 2016, pp. 1351–1368.

[18] B. Gaudet, R. Linares, and R. Furfaro, “Six degree-of-freedom hovering using lidar altimetry via reinforcement meta-learning,” *AIAA Scitech 2020 Forum*, Vol. 1 PartF, No. January, 2020, pp. 1–15, 10.2514/6.2020-0953.

[19] B. Gaudet, R. Linares, and R. Furfaro, “Terminal adaptive guidance via reinforcement meta-learning: Applications to autonomous asteroid close-proximity operations,” *Acta Astronautica*, Vol. 171, No. February, 2020, pp. 1–13, 10.1016/j.actaastro.2020.02.036.

[20] L. Federici, A. Scorsoglio, L. Ghilardi, A. D’ambrosio, B. Benedikter, A. Zavoli, and R. Furfaro, “Image-based Meta-Reinforcement Learning for Autonomous Terminal Guidance of an Impactor in a Binary Asteroid System,” *AIAA Science and Technology Forum and Exposition, AIAA SciTech Forum 2022*, 2022, pp. 1–22, 10.2514/6.2022-2270.

[21] J. R. Martin and H. Schaub, “Reinforcement Learning and Orbit-Discovery Enhanced By Small-Body Physics-Informed Neural Network Gravity Models,” *AIAA Science and Technology Forum and Exposition, AIAA SciTech Forum 2022*, 2022, pp. 1–19, 10.2514/6.2022-2272.

[22] M. Piccinin, P. Lunghi, and M. Lavagna, “Deep Reinforcement Learning-based policy for autonomous imaging planning of small celestial bodies mapping,” *Aerospace Science and Technology*, Vol. 120, 2022, p. 107224, 10.1016/j.ast.2021.107224.

[23] A. H. Levine, D. Wibben, and S. M. Rieger, “Trajectory Design and Maneuver Performance of the OSIRIS-REx Low-Altitude Reconnaissance of Bennu,” *AIAA SCITECH 2022 Forum*, 2022, p. 2388.

[24] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, “Proximal policy optimization algorithms,” *arXiv preprint arXiv:1707.06347*, 2017.

[25] “Small-Body Database Lookup,”

[26] D. Scheeres, J. McMahon, A. French, D. Brack, H. Ikeda, H. T. Y. Tsuda, and D. Lauretta, “Comparing the Estimated Dynamical Environments and Mass Distributions of Bennu and Ryugu,” *spectroscopy*, Vol. 364, 2019, pp. 272–275.