Deep Reinforcement Learning-based Policy for Autonomous Imaging Planning of Small Celestial Bodies Mapping

Margherita Piccinin[a], Paolo Lunghi[b], Michèle Lavagna[c]

[a]*Ph.D. Student, Politecnico di Milano, Via La Masa 34, 20156, Milano, Italy.*
[b]*Assistant Professor, Politecnico di Milano, Via La Masa 34, 20156, Milano, Italy.*
[c]*Full Professor, Politecnico di Milano, Via La Masa 34, 20156, Milano, Italy.*

Abstract

This paper deals with the problem of mapping unknown small celestial bodies while autonomously navigating in their proximity with an optical camera. A Deep Reinforcement Learning (DRL) based planning policy is here proposed to increase the surface mapping efficiency with a smart autonomous selection of the images acquisition epochs. Two techniques are compared, Neural Fitted Q (NFQ) and Deep Q Network (DQN), and the trained policies are tested against benchmark policies over a wide range of different possible scenarios. Then, the compatibility with an on-board application is successfully verified, investigating the policy performance against navigation uncertainties.
Keywords: Small bodies shape reconstruction, Autonomous exploration, Deep learning for space applications

Email addresses: `margherita.piccinin@polimi.it` (Margherita Piccinin), `paolo.lunghi@polimi.it` (Paolo Lunghi), `michelle.lavagna@polimi.it` (Michèle Lavagna)

1. Introduction

Intelligent mapping is a crucial but challenging capability for small celestial bodies exploration. In fact, in proximity of small celestial bodies, the environment is extremely harsh and unknown: mission's operations related to the *mapping process* entail articulated phases, from far approach to close surveys, to gradually characterizing the body with several *mapping stages*, employing different instruments and techniques. In particular, body imaging is fundamental for the body shape reconstruction, that is performed entirely on-ground with stereophotoclinometry (SPC) [1] or stereophotogrammetry (SPG) [2] techniques. The mapping process for shape reconstruction of an unknown body requires several iterations: the shape model is refined during the subsequent observations of the body, until a high resolution model is obtained. Such process entails the collection of a large amount of images, to be sent to ground and elaborated together with navigation data in an iterative manner.

In order to achieve a good surface mapping, granting the maximum coverage and the adequate viewing and illumination conditions, trajectory design is the first necessary step. In proximity of small bodies, the gravitational field can be highly irregular and perturbations like Solar Radiation Pressure, gravitational perturbation due to the Sun and comet outgassing may play a dominant role. In such highly perturbed environments, the design of an orbit suitable for carrying out the mapping process entails many challenging aspects, related also to operational constraints and orbit maintenance, with several possible existing strategies. In the cases of binary asteroids systems, analyses dedicated to the orbit's stability are made in the three body problem accounting for the irregular shape of the body[3]. Families of stable orbits can be found in cases when the Solar Radiation Pressure is the dominant perturbation; these orbits do not require active control and therefore are inexpensive. In particular, three families of orbits are the most studied in literature: ecliptic, terminator and quasi-terminator orbits [4], [5], [6]. Terminator orbits lie in the plane perpendicular to the Sun and are highly stable. The main drawback of this solution is that the angle between spacecraft and Sun is always 90°, limiting the imaging opportunities. Quasi-terminator orbits are particularly good for global mapping campaigns because they are stable and also offer a good variation of Sun-relative geometries. Nevertheless, their applicability is in practice limited, depending on mission time scales, length scales and minimum allowable orbit radius. Another approach is to find surrounding frozen orbits that also satisfy the repeating ground track condition, which is a feature particularly useful for the surface mapping [7]. Other strategies are based on actively controlled trajectories, including but not limited to the heliostationary hovering [8], body-fixed hovering flight [9], as well as flybys, conic-like trajectories or ping-pong orbits [10], [11]. In fact, when the body mass is small such strategies can still be actuated with reasonable costs and offer the possibility to easily obtain the desired Sun-spacecraft-body relative geometry. Drawbacks are that fuel cost may become important if the strategy is extended for a long time and that maneuvers require ground supervision. A completely different approach is hopping exploration over the surface [12]. As a consequence of the rich and challenging dynamical environment, mapping trajectories are strictly mission-dependent and related to a complex and tailored design performed on-ground, including the planning of the orbital operations and the scheduling of data acquisition for mapping.

While orbit selection is the first step for collecting adequate data, mapping is tightly coupled with navigation and planning of the exploration [1]. Today, ground support is necessary for navigation and mapping tasks, and a large human effort is required even 24/7 for proximity operations supervision and planning [13].

The aim of this work is to make a step forward in the direction of autonomy. Autonomous explorations consists in intelligently acting in the unknown environment where the agent is moving. In terrestrial robotics, techniques for autonomous exploration have been developed for real applications: in particular, in Active SLAM (Simultaneous Localization And Mapping) the robot autonomously localizes itself, maps the environment and plans exploration, tasks that are tightly coupled [14],[15],[16]. Active SLAM has also been examined as being an instance of a Partially Observable Markov Decision Process (POMDP)[17]. In this context, a good - even if not optimal - policy can be found by means of Deep Reinforcement Learning (DRL) algorithms: the use of a DRL technique allows finding a good solution policy of an otherwise computationally intractable problem [18]. Such algorithms exhibit well proven generalizing capabilities, that are fundamental to design a flexible policy capable of exploring environments with different and unknown characteristics [19], and of handling problems with partial observability [20]. In particular, POMDPs can be tackled with DRL techniques such as Neural Fitted Q (NFQ) [21] and Deep Q Network (DQN) [22].

In the space field, autonomous exploration has never been accomplished.

For systematic asteroid exploration, light and robust algorithms are required, capable to promptly react to unexpected conditions, reducing risks for such a delicate phase [23]. In [24], supervised machine learning is applied to estimate parameters for optimal asteroid transfer trajectories. Recently, the general problem of exploration has been framed as a POMDP, in analogy to terrestrial active SLAM. In [25] the POMDP is reduced to completely observable for designing an orbit selection policy. In [26] the POMDP is tackled using DRL, overcoming the simplifications in [25] and proposing a direct maneuvering policy, which can be risky for an on-board integration.

The focus of this work is on on-board autonomous decision-making during the mapping process for shape reconstruction. Autonomous operations would reduce the burden of routine navigational support and communication requirements on network services, thus decreasing the mission cost. Autonomy is desirable also to maximize the mission science return (high value data), enabling opportunistic science and real-time re-planning, otherwise impossible because of communications delays. The margin for improvement for autonomy is in first place related to the autonomous scheduling of images acquisition epochs. Nowadays, *images acquisition* policy is established on ground during operations [13]. Hundreds of thousands of images are collected during a mapping process. This work proposes a general approach, to be applied notwithstanding the mission orbit strategy and the asteroid body shape, accounting for autonomy challenges as the limited computational resources and data storage available on-board. DRL is the chosen method, since it offers both the advantages of a light implementation and generalizing capabilities. The main contribution of this work is the design and development of a DRL-based policy for autonomous decision making of the choosing image acquisition epochs, that increases the efficiency of asteroid mapping and enhances the shape model reconstruction. The actual benefits for the mapping process are evaluated by comparison between the DRL policy and benchmark policies. The efficiency of the proposed method is evaluated with performance indexes, and its applicability in a real operational scenario with navigation uncertainties is studied. Some preliminary work has been shown in [27], where the method benefits on body shape reconstruction image processing algorithms have been analysed by reconstructing the small body shape with the simulated collected images.

The paper is structured as follows. In Section 2 a description of the exploration planning problem in terms of a POMDP is provided, along with the adoption of DRL methods for its solution. In Section 3 the proposed DRL approach for images collection optimization during the body mapping operations is presented. Then, Section 4 deals with the training of the DRL policies. In Section 5 the presented results show a successful policy obtained for images selection, and in Section 6 the policy robustness and computational cost are evaluated, proving to be suitable for an on-board application. Finally, in Section 7 conclusions are drawn.

2. Autonomous Exploration Planning framework

This section deals with the problem of planning under uncertainty, providing the general framework under which small bodies autonomous mapping falls. In the robotics field, autonomous exploration of an unknown environment is typically formulated with an active SLAM approach, coupling the tasks of mapping, localization and planning. Active SLAM can be seen as an instance of POMDPs. The mathematical formulation of POMDPs is briefly introduced and the active SLAM problem is presented as a general model for robotic exploration. Finally, the adopted solution approach with DRL algorithms is detailed.

2.1. Partially Observable Markov Decision Processes

Markov Decision Processes (MDP) are based on Markov chains, i.e. stochastic processes with no memory. This means that the process randomly evolves from one state s_k to another s_{k+1} with a transition probability that depends only on the pair (s_k, s_{k+1}) and not on previous states. The decision-maker (called *agent*) can choose between several possible actions. The transition probability to the next state depends on the chosen action and can be associated to a scalar *reward*. The agent goal is to maximize the rewards over time, with an optimal *policy*.

The absence of memory in MDPs is defined by the *Markov property*: the next state depends only on the current state and action and not on past actions and states, hence the future is conditionally independent of the past, given the present state. This property is essential to many solution algorithms [28]. In real applications, the Markov property requirement can be difficult to meet: state information must be rich enough so that the observed state transition does not depend on additional historical information.

When the state is only partially observable, the problem can be defined as a POMDP. In this case the agent can have only a partial knowledge of the environment: the state is not observable but a signal stochastically related to it is. Hence a POMDP can be described as a tuple $\langle \mathcal{S}, \mathcal{A}, \mathcal{R}, \mathcal{T}, \Omega, \mathcal{O} \rangle$, where: $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $r : \mathcal{S} \times \mathcal{A} \to \mathcal{R}$ is the immediate reward (often named also cost function), and Ω is the space of possible observations. A transition probability $\mathcal{T}(s_{k+1}|\ a_k, s_k)$ governs the process by mapping a state-action pair to a probability distribution of states at the next time instant. $\mathcal{O}(o_{k+1}|\ a_k, s_{k+1})$ is the probability of making observation o_{k+1} at the next time step, given action a_k that leads to state s_{k+1}.

In the majority of applications POMDPs are computational intractable, therefore it is better to reduce them to find a computationally tractable solution. A POMDP can be reduced to a MDP including the agent history h as internal state. The history is composed by all past actions and observations, hence history at time step k will be $h_k =< a_0, o_1, a_1, ..., a_{k-1}, o_k >$. The problem is usually tackled with a less direct approach known as *belief-space* MDP. This formulation is a tuple $\langle \mathcal{B}, \mathcal{A}, \mathcal{R}_{\mathcal{B}}, \tau \rangle$, where:

- $\mathcal{B}$ is the belief space, with belief $b_k = p(s_k|h_k)$ equal to the probability of being in state s after history h.
- $\mathcal{A}$ is the action space as in the original POMDP.
- r_B is the expected immediate reward $\mathcal{B} \times \mathcal{A} \to \mathcal{R}_{\mathcal{B}}$
- $\tau(b_{k+1}|\ a_k, b_k)$ is the belief transition function, i.e. the probability of reaching the new belief b_{k+1}, starting from b_k and performing action a_k.

The optimal policy maximizes the reward in the long term, assuming to act according to that policy:

$$\pi^\star = \operatorname*{argmax}_{\pi} \mathbb{E}_\pi \left[\sum_{k=0}^{\infty} \gamma^k \mathcal{R}(a_k, b_k) \right] \tag{1}$$

This is called also *infinite horizon* problem, since the reward is maximized over the entire agent lifetime, considering a discount factor $\gamma \in [0, 1]$.

2.2. Active Simultaneous Localization And Mapping

SLAM consists in estimating a map of an unknown environment and simultaneously localizing - in the same environment - the moving object: indeed, localization is the task of estimating the robot position and orientation (pose) while moving in the environment [29, 30, 31].

SLAM problem can be formulated as follows. The environment map at time step k is made of a set of n landmarks $\mathbf{l_k} = \{\mathbf{l^1}, \mathbf{l^2}, \ \dots\ , \mathbf{l^n}\}$, where $\mathbf{l^i}$ is the position vector of the i-th landmark. The robot pose $\mathbf{x_k}$ changes while the robot moves under the control $\mathbf{u_k}$ and can be estimated through observations of landmarks location $\mathbf{z_k}$.

Active SLAM adds to the SLAM problem the planning task [14, 15, 17]. POMDP provides a framework to investigate the effects of actions and observations on the agent's environment perception, thus allowing designing policies that optimize the agent's interaction with the environment in some of its aspects.

Since the environment is stochastic, the problem can be described in probabilistic terms according to the belief-space MDP formulation presented in the previous section 2.1. The state vector is composed by robot pose and landmark locations $\mathbf{s_k} = (\mathbf{x_k}, \mathbf{l_k})$ and its belief is $\mathbf{b_k} = p(\mathbf{s_k}|\mathbf{z_{0:k}})$. The actions that the agent can take coincide with the control $\mathbf{a_k} = \mathbf{u_k}$. Current belief is estimated from past control, past belief, and current observations: $\mathbf{b_{k+1}} = \tau(\mathbf{u_k}, \mathbf{b_k}, \mathbf{z_{k+1}})$. The reward is usually modeled in terms of an objective function. For instance, a planner can have multiple objectives like maximizing either coverage or map accuracy while minimizing navigation duration, motion cost or resources utilization. Several criteria exist to formulate the objective function to be optimized by the planning policy [32, 15]. In particular, the exploration problem consists in choosing the sensing trajectory to obtain the best map.

2.3. Deep Reinforcement Learning

POMDPs are usually computationally intractable. Hence, a reduction of state, action and policy spaces is needed. An Artificial Neural Network (ANN) is used to approximate the Q-value, i.e. the expected return over time. Therefore the optimal policy is the one that maximizes the Q-value:

$$\pi^\star = \underset{\pi}{\operatorname{argmax}}\, Q_\pi(\mathbf{a_k}, \mathbf{s_k}|\theta_\pi) \tag{2}$$

where θ_π are the ANN weights and biases. Different approaches can be applied to formulate the problem with a neural network. This work investigates two alternatives, NFQ and DQN.

The NFQ algorithm scheme is reported in Fig. 1. NFQ sees its major difficulty in the training data collection: the problem must be suited to be solved with a random policy that allows the agent-environment interaction and the collection of the state-action-state triples. If a random policy poorly performs the environment exploration, the net will be trained only on a subset of the different situations it could encounter. This approach is model-free, stable, data efficient and simple to implement.

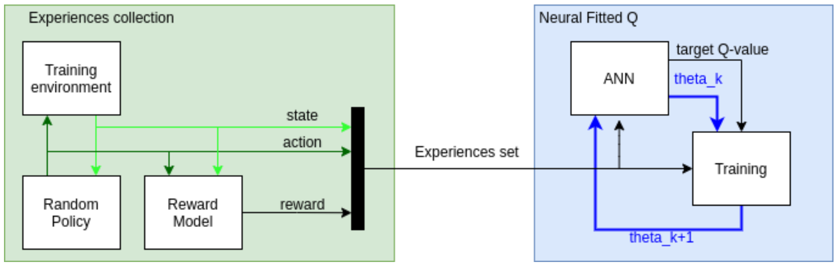


Figure 1: NFQ scheme.

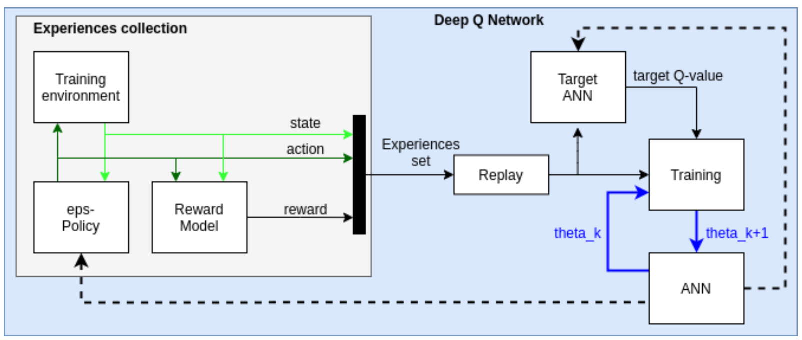


Figure 2: DQN scheme.

The DQN algorithm scheme is reported in Fig. 2. Experiences are collected playing many episodes, during which actions are chosen according to an ϵ-*greedy* policy. This means that with probability ϵ the action is random and with probability $(1 - \epsilon)$ the action is the one that maximizes the current Q-function. The choice of the greedy parameter can be critical to correctly collect transitions, as also exploration of unknown regions of the state space is important.

3. Images collection planning with Deep Reinforcement Learning

The autonomous DRL-based decision making proposed approach is presented in this section. Then, the detailed definition of the POMDP reduced

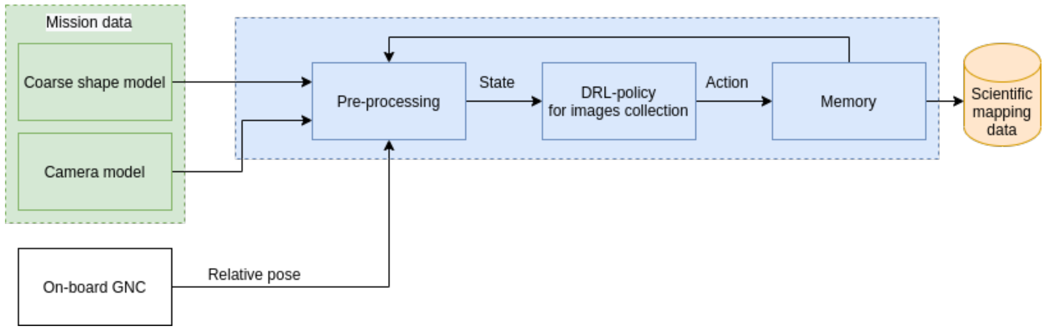


Figure 3: Autonomous DRL-based planning and shape reconstruction validation method.

spaces is provided, in terms of reward $\mathcal{R}$, belief state space $\mathcal{B}$ and action space $\mathcal{A}$.

3.1. Proposed architecture

This paper develops a method to autonomously plan the timing of observations during the mapping of an unknown small body, with particular application to imaging for SPC. The goal is to define a policy that improves mapping quality, while both limiting the amount of images to downlink and fastening the mapping process. The planning framework is defined as a POMDP, proposing a novel problem architecture focused on data collection. DRL is exploited to design the planning policies. A scheme of the proposed architecture for small bodies imaging and shape reconstruction is schematized in Fig. 3.

Algorithm architecture is designed to be mission-independent and computationally light to cope with limited on board resources. In particular, the algorithm needs information on the camera Field of View (FoV), the relative pose between camera and the target, the illumination conditions and the body low resolution polyhedral shape model, which is already available during the considered mission phases. Then, data are pre-processed along with history information of already collected images. The next block is related with the autonomous decision making: if the current observation epoch is worth, then the image is taken and actions are recorded. The decision making problem is solved by means of DRL, dealing with the challenge of optimizing images collection for small bodies shape reconstruction. The spacecraft acts according to a policy for next best step selection, i.e. for selecting the most proper time instant for collecting a new image, based on the relative pose between camera and body and on the illumination conditions.

3.2. Reward definition

In this section the reward space $\mathcal{R}$ is defined. SPC benefits from images with large illumination variation and small viewing angle variation. Scores related to the photometric angles (see Fig 4) can be defined to assess the quality of the taken pictures for the shape reconstruction process [25],[27].

The overall score S^i associated to the i-th facet of the polyhedral shape model is given by the weighted sum of five different contributions:

$$S^i = w_1 S^i_i + w_2 S^i_e + w_3 S^i_{\Delta e} + w_4 S^i_{\Delta\alpha} + w_5 S^i_{\Delta\beta} \tag{3}$$

where S^i_i is the incidence score, S^i_e the emission one, $S^i_{\Delta e}$ the emission variation score and $S^i_{\Delta\alpha}$ and $S^i_{\Delta\beta}$ the solar and spacecraft azimuth angle scores. Such scores are dependent on the spacecraft position and orientation, which also determines which facets are in view. The mapping resolution is not considered here, assuming to apply the policy at each mapping stage, thus not significantly varying the distance from the body. They are defined as follows, starting from previous studies on the SPC mapping quality [10], [25].

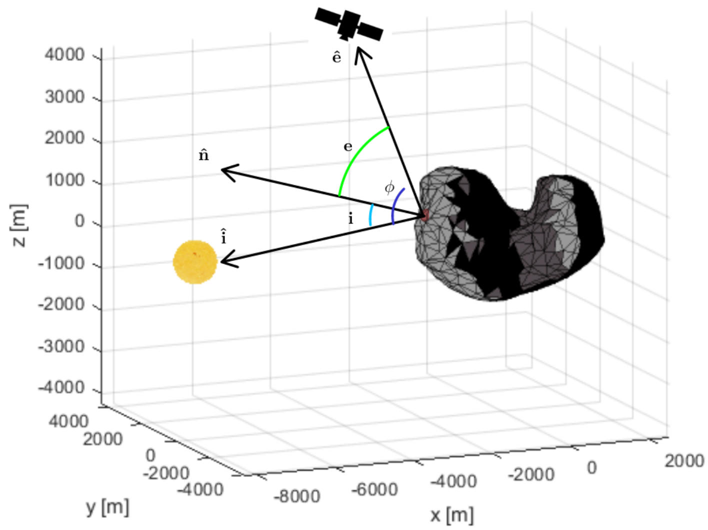


Figure 4: Photometric angles: emission angle e, phase angle ϕ, inclination angle i.

Incidence score. The incidence angle i should be kept between $20° - 60°$ to avoid shadows and excessive brightness, that won't allow the extraction of useful information. Let's define the incidence score S_i^i:

$$s_i = \begin{cases} 1 & \text{if } 20° \leq i \leq 60° \\ \frac{1}{10}i - 1 & \text{if } 10° \leq i \leq 20° \\ -\frac{1}{10}i + 7 & \text{if } 60° \leq i \leq 70° \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

$$S_i^i = \mu_j(s_i) \tag{5}$$

where μ_j is the mean performed over all the n_{img} taken pictures that contain the facet.

$$\mu_j(x) = \frac{1}{n_{img}} \sum_{j=1}^{n_{img}} (x_j) \tag{6}$$

Emission score. The emission angle should be kept between $10° - 50°$. Hence in a similar manner the emission score S_e^i is defined as follows:

$$s_e = \begin{cases} 1 & \text{if } 10° \leq e \leq 50° \\ \frac{1}{5}e - 1 & \text{if } 5° \leq e \leq 10° \\ -\frac{1}{10}e + 6 & \text{if } 50° \leq e \leq 60° \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

$$S_e^i = \mu_j(s_e) \tag{8}$$

Emission variation score. Also, a large variation of emission angles is considered beneficial, therefore the emission variation score is:

$$S_{\Delta e}^i = \mu_j \left(\max_k \frac{2\Delta e_{jk}}{\pi} \right) \tag{9}$$

where $\Delta e_{jk} = |e_j - e_k|$. Hence for each emission angle e_j under which the i-th facet is seen, the maximum difference between the considered angle e_j and all the other angles e_k under which the facets was observed is computed. Then, all the maximum differences are normalized of $\frac{\pi}{2}$, i.e. the maximum possible emission variation, and the mean is performed.

Solar and spacecraft azimuth score. Finally, the variation of solar azimuth angles α should be large and the one of spacecraft azimuth angles β small. The respective scores are computed in a similar fashion.

$$S_{\Delta\alpha}^i = \mu_j \left(\max_k \frac{\Delta\alpha_{jk}}{\pi} \right) \tag{10}$$

$$S_{\Delta\beta}^i = 1 - \mu_j \left(\max_k \frac{\Delta\beta_{jk}}{\pi} \right) \tag{11}$$

Please note that in this case the normalizing value is π.

According to the images history, the facet mapping index m^i is defined for the i-th facet:

$$m^i = S^i \min\left(1, \frac{n_{img}}{N_{img}}\right) \tag{12}$$

where n_{img} and N_{img} are respectively the number of taken images and the number of ideally necessary images (equal to at last 3 for SPC). The index m^i can assume values in the interval $[0, 1]$ and in particular the maximum value represents an ideal perfect mapping.

The immediate reward depends on both states and actions:

$$r_k = r_k(s_k, a_k) \tag{13}$$

If no action is taken the reward is null. Whenever an image is collected in a forbidden state $s \in S^-$, a negative reward equal to -1 is returned to the agent and the image is not accounted for in the successive mapping. Forbidden states correspond to situations in which either the image is in complete shadow or the ideal number of images is overcome, occurrence which might potentially cause problems in on-board data storage. The ultimate goal is to maximize the mapping index, therefore if the picture is taken in allowed states the reward is:

$$\tilde{r} = \mu_m \left(\frac{m_k^i - m_{k-1}^i}{m_k^i} \right) \tag{14}$$

where m_k^i is the mapping index of facet i at time k and μ_m stands for the mean over all the facets in the current frame.

Summarizing, the overall reward is:

$$r_k = \begin{cases} -1 & \text{if } a_k = 1 \text{ and } s_k \in S^- \\ 0 & \text{if } a_k = 0 \\ \tilde{r} & \text{otherwise} \end{cases} \tag{15}$$

If the agent immediately takes all photos allowed to be sent on ground, along the successive time steps it will be forced to accept either a zero or a negative reward. On the other hand, the long term reward will be higher if images are collected only when it is worth, hence smoothly distributed in time.

3.3. Action space

In this section the action space $\mathcal{A}$ is defined. The agent interacts with the environment only by choosing its sensing locations, hence by collecting images, without controlling its relative pose with respect to the body surface. The action at time step k is boolean:

$$a_k = \begin{cases} 0 & \text{if no picture is taken} \\ 1 & \text{otherwise} \end{cases} \tag{16}$$

The number η of pictures to be ideally taken in a certain storage time $T_{storage}$ is fixed. After this storage time images are downlinked and therefore the memory is empty again. The discrete time steps at which an action can be taken are defined in number equal to the ideal number of images times a control parameter Δ_c. Ideally with a large control parameter the final performance would be better, but the number of decisions to be taken would be too high, entailing a longer and more difficult learning process. Hence a trade off between performance and learning must be done for the choice of Δ_c.

3.4. State space

In this section the belief state space $\mathcal{B}$ is defined. States have been designed to synthesize only the information necessary and useful for decision making. In particular, the state is constituted by the *memory state*, *map state* and *angles state*. A total of 12 states are defined.

Memory state. The memory state provides information on the time lapse and number of collected images. The idea is that in a certain time interval T_{storage} pictures can be stored in the on-board memory before being sent to ground. The ideal number of images to communicate at every time interval is η. In particular, the percentage of time spent in the current storage interval and the number of pictures taken n with respect to the ideal number η are fed to the net. The parameters T_{storage} and η can be tuned depending on mission constraints without affecting the algorithm. These inputs help in evaluating how the collection of a new image would impact on data storage.

Map state. The map state provides general information on the mapping campaign advancement. The fraction of area in light of the surface portion in view, which relates to the area percentage whose knowledge will actually be improved by a new picture, can be roughly computed as the ratio between the image facets in light and the total number of facets visible in the image. The map state also includes the mean of the mapping index and its standard deviation over the surface in view and the same quantities computed over the whole body. These data are useful to make a decision on whether the exploration of the area under exam is worth from the coverage point of view.

Angles state. The angles state gives local information about photometric angles under which the facets in view and in light are seen at the specific epoch. In particular, the angle state includes inclination and emission scores mean, over all the facets in view and in light for the angles of the current time instant only. While the other states are the facets mean of the maximum variation of current Sun azimuth, spacecraft azimuth and emission angles with respect to the angles of already take pictures. These inputs concur in evaluating the possible improvement of SPC for what concerns stereo angles and illumination conditions.

The use of statistical quantities (mean and standard deviation) is the only solution that allows keeping constant the number of observed states despite of the change of number of facets in view. Moreover, to understand how the mapping campaign is proceeding, the whole history of past actions should be part of the states as well. Of course to include the whole history in the states observation is unfeasible, but anyway the POMDP is reduced by making part of the history observable. The POMDP is also simplified by assuming the belief equal to the actual state $\mathbf{b_k} \simeq \mathbf{s_k}$. As a future improvement to overcome the drawbacks of classical DQN, the prioritized experience reply could be employed as in [33].

3.5. Neural Network architecture

The architecture of the ANN used to approximate the Q-value function is kept light to achieve a low on-board computational time: a multi-layer perceptron with 2 hidden layers of 10 neurons each is adopted. The network graph is shown in Fig. 5. Such network can be defined a Deep Neural Network [34].

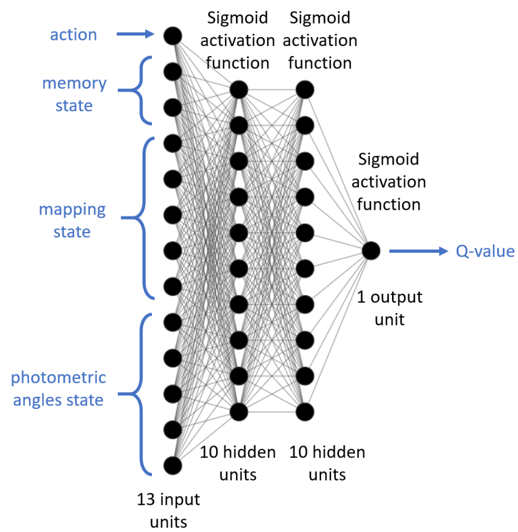


Figure 5: Neural Network architecture.

The ANN architecture is the same for both NFQ and DQN and is kept as simple as possible, with a 13 elements input vector and a scalar output, in accordance to the necessities of the planning model. Weights and biases of the ANN are changed during the learning process with resilient back-propagation (RPROP) steps [35].

4. DRL policy training

In this section the learning environment is described and the training results are shown.

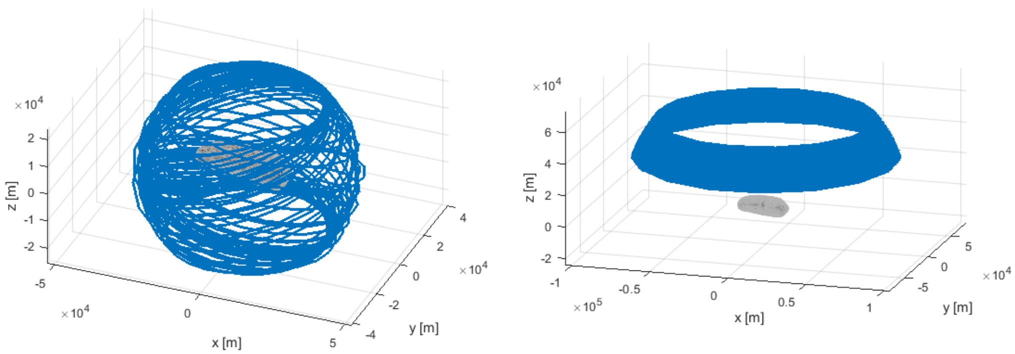
4.1. Training environment

To properly define the environment with which the agent interacts during the learning is fundamental for the learning success. The set of experiences should be complete, i.e. it should be an exhaustive collection of all possible cases that the agent may encounter. Please note that the state is defined to be independent from the asteroid, orbit and camera characteristics. Therefore, a complete training set is not a set built considering several asteroids and orbits, but a set of examples that sufficiently explores the state space and includes relevant experiences to get to the final goal. Ideally it should contain a whole *mapping stage*, from the beginning to the end, in which all pictures are taken with the same instrument and at about a constant distance from the asteroid. Since the resolution of the maps to be created is finer than that already achieved at larger distances, the stage can be considered independent from past stages for what concerns the coverage of the asteroid. In a few words, the learning environment should allow the agent to collect both experiences in prohibited states S^-, to identify them and avoid them in the future, and to make very successful actions for mapping. For these reasons, to enhance the learning process, a somewhat unrealistic situation is selected as learning environment:

- Non-keplerian orbit around asteroid Eros in Figure 6.
- Camera FoV of $10°$.

Such scenario allows a great variation of the spacecraft-Sun-body relative geometry.

The chosen asteroid is Eros being one of the few shape models publicly available in databases and because its elongated shape allows imaging different percentages of the body surface, keeping the distance fixed. In fact, the percentage of surface in view varies between 6.4 % and 0.7 % with a mean of the 2.4 % along the orbit.



(a) Spacecraft position - body-fixed frame.

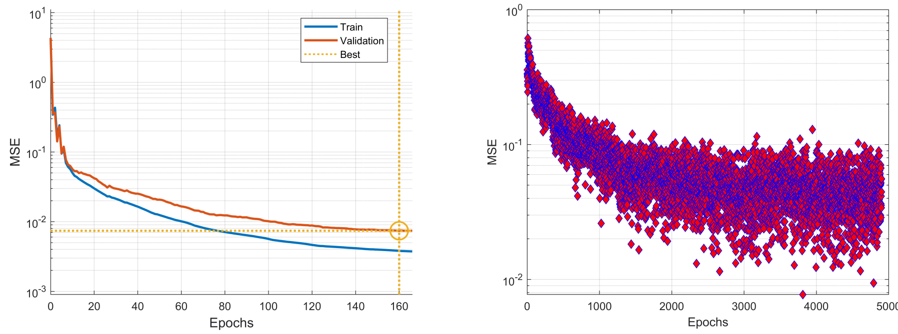
(b) Sun direction - body-fixed frame.

Figure 6: Spacecraft and Sun position in training environment.

The trajectory has been obtained considering spherical harmonics perturbations, starting from an initial condition corresponding to osculating orbital parameters of null eccentricity, 45° inclination and radius twice the asteroid maximum one. For what concerns the body illumination, some areas remains always in shadow, as it can be seen from Sun direction in the body-fixed frame shown in Figure 6. The reason is that the body spin axis inclination with respect to the ecliptic north is larger than Eros orbit inclination. This allows to frequently collect also negative experiences for the body mapping, which need to be learned and avoided.

The training simulation environment also assumes that the ideal number of images during one episode is 500, with an ideal frequency of 1 picture per hour. The data downlink and control parameters are:

- $T_{\text{storage}} = 10$ h
- $\eta = 10$



(a) Training Mean Square Error for NFQ network.

(b) Training Mean Square Error for DQN network.

Figure 7: DRL policies learning process.

- $\Delta_c = 3$

In practice, the orbit is discretized so that the number of points in the storage time is three times the number of photos allowed. Hence the control interval between one action and the next is quite coarse. This interval can be refined in future works.

4.2. Learning process

RPROP is selected as training algorithm because of it robustness. In particular, batch learning is preferred to incremental learning, because the training set has a low dimension (500 ideal images and $\Delta_c = 3$ lead to a total number of 1500 experiences). Input and output scaling is performed on the whole experiences set.

NFQ learning. The Mean Square Error between network outputs and targets is reported for one training iteration in Figure 7a, where one epoch corresponds to a weights update step on the entire set of experiences. As it can be observed, the Mean Square Error smoothly decreases until the validation check is met, i.e. the validation error stops decreasing. As expected, the training error is lower than the validation error.

DQN learning. The Mean Square Error evolution during the learning is reported in Figures 7b. In this case one epoch corresponds to one RPROP step on the mini-batch. As it can be noticed, Mean Square Error decreases but it is much less stable with respect to the NFQ case. This is a consequence of the different batch sizes used in the two algorithms.

5. DRL policy performance

In this section, the performance of the DRL policy, trained with DQN and NFQ methods, is presented. The two DRL methods are compared to benchmark policies in different mission scenarios to verify their generalizing capability, which is of great importance when exploring an unknown environment.

5.1. Benchmarks and performance metrics definition

Benchmarks. A first numerical validation is performed by comparing the DRL-based algorithm with two different simple benchmarks: a policy that takes pictures at regular intervals (UNI) and another that randomly selects the image acquisition instants (RAND). UNI takes a picture every Δ_c time steps. For the RAND strategy if $n_k > n_{k,UNI}$ the image is discarded and all presented results are the mean over 100 runs. For UNI, NFQ, and DQN only 1 run is needed, since they are deterministic policies.

Performance indexes. Often DRL results are compared just with the numerical final score obtained during the episode. In such a way however, to critically analyze how policies actually behave is hard. Being the design and learning procedure highly based on engineering judgment, test results are presented not with final reward scores, but by means of some complementary indexes that facilitate the performance understanding:

1. Final number of collected images I_n, (the lower the better).
2. Final mapping index $I_{map} = \mu_M(m^i_{k_{end}})$, where μ_M is the mean over all the body facets (the higher value the better).
3. Integral mapping index over the campaign $I_{sum} = \frac{1}{\Delta_c} \sum_k \mu_M(m^i_k)$, (the higher the better).

Such parameters quickly allow verifying whether the modeled reward actually leads to an improvement of the proposed tasks: data reduction and mapping enhancement and fastening.

5.2. Test cases definition

Test cases have been chosen to cover all the relevant aspects for the algorithm application. In particular, four different bodies are considered: Eros, on which the training has been performed, Itokawa, that presents an elongated shape, Bennu, with diamond shape, and 67P-CG, with two-lobes shape. Small bodies are assumed on Keplerian orbits around the Sun, with constant spin axis orientation and rotational period. The camera is assumed to have a conical FoV of 3° and a shape model of 1000 facets is considered available on-board. The length of each test episode is set to 1500 time steps, with $\Delta_c = 3$, and an ideal final images number of $1500/\Delta_c = 500$.

Sensitivity analysis run for the above mentioned small bodies by varying the following quantities:

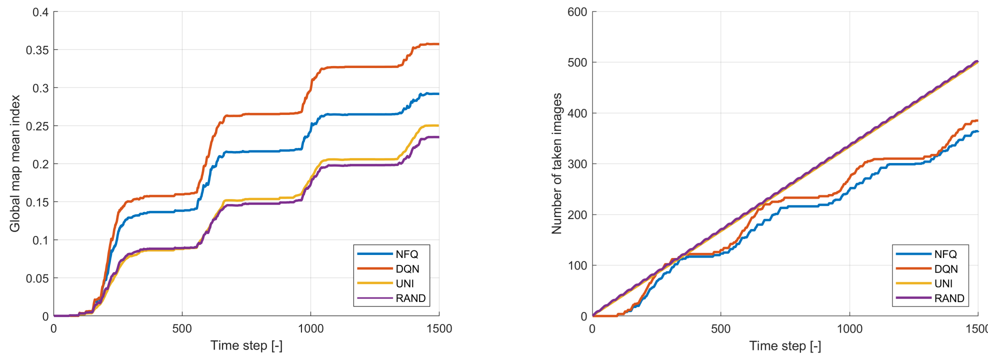
- The distance from the body, that affects both relative dynamics and percentage of surface in the camera FoV. In particular, the quantity here referred to is the *interest ratio*, i.e. the ratio between distance from the body center and maximum body radius.
- The body rotational period T, that influences illumination conditions variation and again relative pose.
- The orbit inclination i, that changes the surface portion object of the mapping.

5.3. Detailed results for 67P test case

For the sake of brevity, results are presented in this section for the 67P scenario only, among the ones above mentioned. In fact, it well represents challenges linked to an extremely irregular body shape, as self-shadowing and self-occlusion, and it can be assumed as a worst-case example for the achieved performances. Please note that the presented case differs from the case exploited for the learning process.

The basic simulation scenario for comet 67P has the following parameters:

- Rotational period $T_{rot} = 12.4\,\mathrm{h}$.
- Circular polar orbit, with interest ratio equal to 6.
- Percentage of surface in view in the range $0.3\,\%$ to $2.9\,\%$.



(a) Mapping performance (b) Number of collected images

Figure 8: Polar orbit at 67P. Comparison of the different strategies.

The evolution of the mapping quality index and the number of collected images are respectively shown in Fig. 8 for a circular polar orbit at 67P, with interest ratio 6. For both NFQ and DQN, not only the amount of collected data is equal or less with respect to RAND and UNI strategy, but also the mapping index is higher.

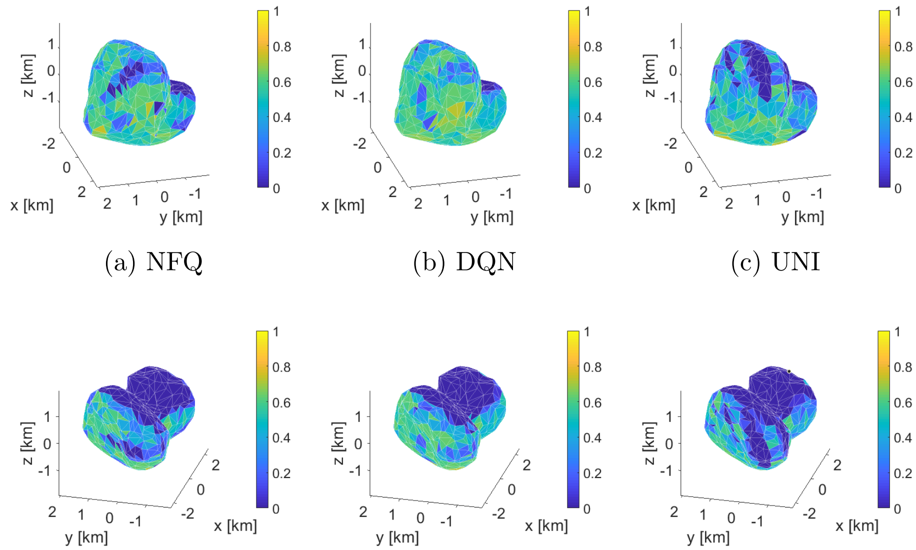
The final mapping index is shown for NFQ, DQN and UNI strategies in Fig. 9. In this case the mapping is hindered by Sun illumination but also by the significant self-shadowing and self-occlusion.

5.4. Sensitivity analysis results

Detailed results of the sensitivity analysis are here reported for the Eros and 67P scenarios, respectively in Table 1 and Table 2. For most test cases in the four bodies scenarios DQN proves to be the best policy. In some cases, I_{map} has a similar value for the three strategies, but the goal is achieved faster by NFQ and DQN, that have a larger I_{sum}. A general trend observed is that with NFQ the number of collected pictures is lower than DQN and UNI, but

Table 1: Eros, sensitivity analysis.

	i = 30 deg			**i = 60 deg**			**i = 90 deg**		
	$\mathbf{I_n}$	$\mathbf{I_{map}}$	$\mathbf{I_{sum}}$	$\mathbf{I_n}$	$\mathbf{I_{map}}$	$\mathbf{I_{sum}}$	$\mathbf{I_n}$	$\mathbf{I_{map}}$	$\mathbf{I_{sum}}$
UNI	500	0.21	66.87	500	0.22	69.10	500	0.22	69.43
NFQ	430	0.25	87.64	431	0.26	87.58	404	0.26	86.50
DQN	498	0.24	83.30	493	0.26	84.91	434	0.29	89.67
	Interest Ratio = 6			**Interest Ratio = 8**			**Interest Ratio = 10**		
	$\mathbf{I_n}$	$\mathbf{I_{map}}$	$\mathbf{I_{sum}}$	$\mathbf{I_n}$	$\mathbf{I_{map}}$	$\mathbf{I_{sum}}$	$\mathbf{I_n}$	$\mathbf{I_{map}}$	$\mathbf{I_{sum}}$
UNI	500	0.33	114.38	500	0.38	142.34	500	0.41	160.01
NFQ	381	0.33	120.64	377	0.37	143.31	488	0.39	150.86
DQN	317	0.37	135.90	320	0.38	151.13	326	0.39	159.87
	T = 2 h			**T = 5 h**			**T = 12 h**		
	$\mathbf{I_n}$	$\mathbf{I_{map}}$	$\mathbf{I_{sum}}$	$\mathbf{I_n}$	$\mathbf{I_{map}}$	$\mathbf{I_{sum}}$	$\mathbf{I_n}$	$\mathbf{I_{map}}$	$\mathbf{I_{sum}}$
UNI	500	0.21	57.78	500	0.22	61.80	500	0.20	58.97
NFQ	409	0.25	79.00	402	0.25	84.18	415	0.25	74.06
DQN	459	0.28	81.41	446	0.29	86.20	469	0.25	71.76



(a) NFQ (b) DQN (c) UNI

(d) NFQ (e) DQN (f) UNI

Figure 9: 67P, facets final mapping index.

with DQN a larger mapping performance is achieved, sometimes even with a lower number of pictures.

As visible in Table 1 and Table 2, by increasing the interest ratio the two DRL strategies become less efficient: even if I_n is reduced, I_{map} and I_{sum} are comparable to the UNI strategy. This trend has been observed for all considered small bodies and may be due to two reasons: the percentage of surface in view is out of training interval and of typical mission values; or when a large portion of the body is imaged it is more difficult to have control on the viewing conditions of all facets in the frame. In fact 1 % of the surface means to consider about 10 facets, while 10 % corresponds to 100: very different viewing conditions may be present in the same picture. Please note that in any case the number of pictures for DQN and NFQ is largely

Table 2: 67P-CG, sensitivity analysis.

	i = 30 deg			i = 60 deg			i = 90 deg		
	I_n	I_{map}	I_{sum}	I_n	I_{map}	I_{sum}	I_n	I_{map}	I_{sum}
UNI	500	0.27	79.67	500	0.27	72.15	500	0.25	67.01
NFQ	356	0.28	95.70	342	0.30	95.24	365	0.29	91.55
DQN	476	0.30	94.65	417	0.36	112.69	385	0.36	111.34
	Interest Ratio = 8			**Interest Ratio = 10**			**Interest Ratio = 12**		
	I_n	I_{map}	I_{sum}	I_n	I_{map}	I_{sum}	I_n	I_{map}	I_{sum}
UNI	500	0.33	102.68	500	0.41	125.66	500	0.40	130.36
NFQ	310	0.32	114.30	290	0.40	126.50	279	0.40	126.52
DQN	285	0.38	134.45	285	0.41	143.01	314	0.41	133.63
	T = 2 h			**T = 5 h**			**T = 12 h**		
	I_n	I_{map}	I_{sum}	I_n	I_{map}	I_{sum}	I_n	I_{map}	I_{sum}
UNI	500	0.21	50.94	500	0.25	62.84	500	0.24	65.59
NFQ	355	0.31	87.28	354	0.29	93.19	379	0.26	84.03
DQN	398	0.36	98.93	381	0.36	110.49	427	0.28	95.00

reduced in spite of a small difference in I_{map} and I_{sum}.

DRL-policy has been trained in a completely different scenario, concerning both body shape and orbit, and has been designed to be easily employed into a wide variety of different mission scenarios. Therefore, an optimal behaviour is actually not expected to be reached, even is a significant enhancement in scientific mapping and data collected, compared to simpler acquisition strategies, is sought.

Sensitivity analyses results show that the proposed solutions are capable to deal with far-off different scenarios and outperform the UNI and RAND benchmarks. The presented algorithm can work independently of the relative dynamics between spacecraft and small body, proving to be highly flexible: in all considered small bodies scenarios the policy for selection of the observation times actually enhances the efficiency of data collection.

6. Compatibility with on-board application

The applicability of the proposed architecture for an on-board application is analysed in this Section. In particular, two relevant aspects are studied: the robustness of the DRL policy to uncertain inputs coming form the on-board navigation system, and the computational effort required by the architecture.

6.1. DRL policy robustness to uncertainty

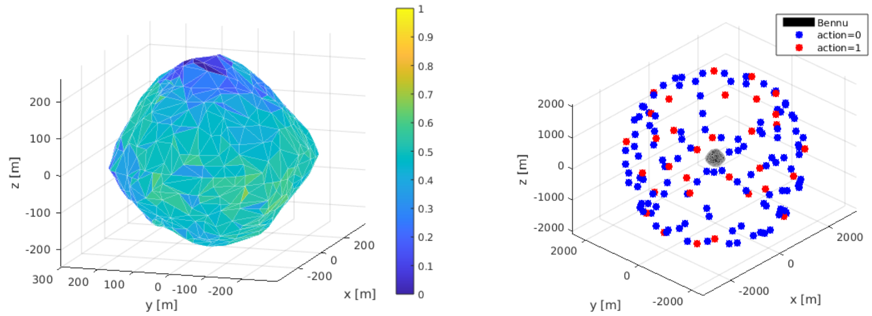
During close proximity operations, the on-board knowledge of the relative pose with respect to the body surface is limited by the navigation accuracy. This aspect directly influences the inputs to the DRL-policy and may lead to a behaviour different from the expected, since uncertain inputs were not considered at all during the training. For these reasons, some tests are performed on the DRL-policy, introducing errors in the knowledge of the relative pose of the spacecraft.

The considered testing scenario consists in a quasi-Keplerian circular orbit at asteroid Bennu, at 2.5 km distance and with an inclination of 45°. The assumed camera has a FoV of 10°. In particular, 6 tests are performed considering an increasing uncertainty separately for position and pointing, which are perturbed by additive Gaussian white noise with standard deviation σ. Finally, test 7 considers both effects, with the uncertainty expected for the study case, i.e. 100 m for the relative position and 1° for the pointing. To have a fair term of comparison, in each simulation the spacecraft ground-truth pointing and position history is the same and only the state belief is different. Each test has been run with 100 simulations; mean value and standard deviation of the mapping index and acquired frames are reported in Table 3. The mapping index is reported as a percentage with respect to the complete mapping, which is impossible to achieve, and the acquired frames are reported as percentage of the imposed maximum capability of data storage and communication.

Table 3: DRL-policy performance with uncertainties on relative state.

Test ID	**Position σ [m]**	**Pointing σ [°]**	**Mapping index [%] (σ)**	**Frames acquired [-] (σ)**
ALL	0	0	48.3 (-)	150 (-)
0	0	0	45.9 (-)	66 (-)
1	100	0	43.9 (1.4)	68 (4)
2	250	0	37.1 (4.7)	67 (5)
3	500	0	20.9 (5.7)	70 (3)
4	0	1	45.8 (0.5)	71 (5)
5	0	3	47.4 (0.3)	82 (7)
6	0	5	47.9 (0.4)	92 (8)
7	100	1	44.1 (0.1)	70 (4)

Table 3 shows that the DRL-policy is quite robust to both kinds of uncertainties and still achieves a good performance within the typical navigation



(a) Mapping index

(b) Acquired frames

Figure 10: Bennu, DQN performance without uncertainties (test 0).

uncertainties (test #7), but the performance decreases when the state belief is very far from the real state.

Test ALL is performed to assess the maximum ideally achievable mapping without any constraint on the data storage (UNI policy applied with high sampling frequency), while test #0 assesses the behaviour of the DRL policy with a perfect knowledge of the state (see Fig. 10).

Tests #1-3 show that the number of frames acquired is comparable to the test-0, but the actual mapping performance decreases with a higher position uncertainty. Since the DRL-policy assumes to point Bennu centre of mass from the belief of its relative position, the information regarding to the illumination of the surface in view is affected, leading to a lower quality mapping.

Tests #4-6 highlight a different behaviour of the net: a larger number of images is collected, increasing the mapping quality thanks to the larger amount of data. In test #6 almost all the images are collected: the belief of the current mapping is worse than the actual one because part of the body is believed to exit the FoV; thus the policy continues collecting data to complete the body coverage even if they are not necessary.

In test #7 the policy confirms to be robust to combined uncertainties in the state estimation, leading to a good mapping of the object, but with a rise of data acquired with respect to absence of uncertainties. The considered uncertainties are in line with a realistic scenario and the AI-policy is verified to outperform a classical UNI scheduling in terms of amount of data and in relation to images quality.

6.2. Computational cost preliminary assessment

A computational analysis is addressed to determine the feasibility and limits of a possible on-board implementation of the algorithm. The algorithm is implemented in MATLAB and all tests are run on an Intel® Core™ i7-5500U CPU, clocked at 2.4 GHz, paired to a 16 GB DDR3 memory.

Computational time for a global mapping case. The computational time to take a single decision is evaluated over 500 runs - i.e. the ideal number of images for an episode - considering a 1000 facets spherical shape model and a typical case in which 1.5 % of the surface is in view. Results in Table 4 and Fig. 11 show that a low computational time is required, with a mean value of 33.5 ms.

Surface in view: percentage variation. Another analysis was performed, to examine the computational cost trend with respect to the percentage of surface in view. All other parameters are kept as before. The mean time linearly increases with the surface portion, as shown in Fig. 12, where each point is the mean computational time over 500 decisions. The mean time can be

Table 4: Computation time to take a single decision, with 1.5% of the surface in view, 500 runs.

	Time [ms]
Average	33.5
Minimum	7.4
Maximum	132.5

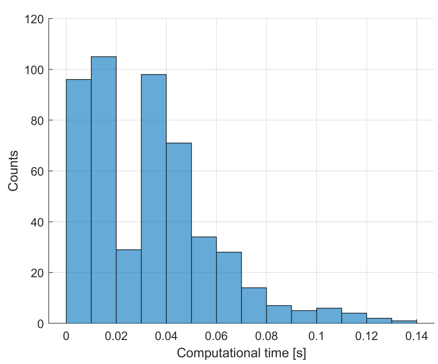


Figure 11: Computation time to take a single decision. Time with 1.5% of the surface in view, 500 runs.

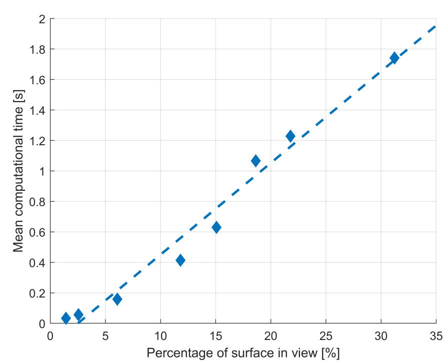


Figure 12: Computation time to take a single decision, varying the surface portion in view.

about two orders of magnitude larger when the surface portion in view is above the 30 %.

Implementation on flight hardware would corresponds to increased computation time, potentially introducing a certain delay in the decision making. The effect of such a delay would be the imaging of a slightly different area with respect to the expected one, due to the target rotation and the spacecraft dynamics. For a hypothetical fast rotating spherical body with 2 h rotational period, the surface displacement in 1 s is of about the 0.1 % of the characteristic dimension. If the surface portion in view is large, such displacement is likely not significant; while for a small area in view the computational time is minimum. Similar considerations hold considering the specific spacecraft dynamics.

7. Conclusion

In conclusion, an AI-based planning policy for enhancing the mapping of an asteroid or comet is here proposed. Achieved results of the presented approach reveal the methodology to be a promising step forward in autonomous operations, helping in decreasing the human effort during the mapping phases of unknown small bodies and increasing imaging exploitation efficiency with a simple and flexible scheme. The merits of the proposed architecture are the decoupling of the decision-making process from spacecrafts dynamics, the autonomy improvement with very low risks for the mission, and the general validity of the planning framework, which is mission-independent and does not require learning during operations. The DRL-based strategies generalizing capabilities are verified through numerical simulations, obtaining promising results. Two results are consequent to the application of the DRL-based policy: an increased performance mapping efficiency and a correct handling of memory storage during the mapping. The strategy robustness to uncertain inputs coming from the on-board navigation is tested, confirming its suitability for a realistic scenario. Future work will further investigate the effectiveness of the proposed techniques with more challenging benchmarks cases.

References

[1] R. W. Gaskell, O. S. Barnouin-Jha, D. J. Scheeres, A. S. Konopliv, T. Mukai, S. Abe, J. Saito, M. Ishiguro, T. Kubota, T. Hashimoto, J. Kawaguchi, M. Yoshikawa, K. Shirakawa, T. Kominato, N. Hirata, H. Demura, Characterizing and navigating small bodies with imaging data, Meteoritics and Planetary Science 43 (2008) 1049–1061.

[2] B. Giese, J. Oberst, R. Kirk, W. Zeitler, The topography of asteroid ida: A comparison between photogrammetric and two-dimensional photoclinometric image analysis, International Archives of Photogrammetry and Remote Sensing 31 (1996) B3.

[3] A. Capannolo, F. Ferrari, M. Lavagna, Families of bounded orbits near binary asteroid 65803 didymos, Journal of Guidance, Control, and Dynamics 42 (2019) 189–198.

[4] S. B. Broschart, G. Lantoine, D. J. Grebow, Quasi-terminator orbits near primitive bodies, Celestial Mechanics and Dynamical Astronomy 120 (2014) 195–215.

[5] D. J. Scheeres, Orbit mechanics about asteroids and comets, Journal of Guidance, Control, and Dynamics 35 (2012) 987–997.

[6] D. Scheeres, B. Sutter, A. Rosengren, Design, dynamics and stability of the osiris-rex sun-terminator orbits, Advances in the Astronautical Sciences 148 (2013) 3263–3282.

[7] C. Circi, A. D'Ambrosio, H. Lei, E. Ortore, Global mapping of asteroids by frozen orbits: The case of 216 kleopatra, Acta Astronautica 161 (2019) 101–107.

[8] Y. Zhang, X. Zeng, F. Zhang, Spacecraft hovering flight in a binary asteroid system by using fuzzy logic control, IEEE Transactions on Aerospace and Electronic Systems 55 (2019) 3246–3258.

[9] X. Zeng, S. Gong, J. Li, K. T. Alfriend, Solar sail body-fixed hovering over elongated asteroids, Journal of Guidance, Control, and Dynamics 39 (2016) 1223–1231.

[10] T. A. Pavlak, S. B. Broschart, G. Lantoine, Quantifying mapping orbit performance in the vicinity of primitive bodies (2015).

[11] S. B. Broschart, D. J. Scheeres, Control of hovering spacecraft near small bodies: application to asteroid 25143 itokawa, Journal of Guidance, Control, and Dynamics 28 (2005) 343–354.

[12] T. Wen, X. Zeng, C. Circi, Y. Gao, Hop reachable domain on irregularly shaped asteroids, Journal of Guidance, Control, and Dynamics 43 (2020) 1269–1283.

[13] R. P. de Santayana, M. Lauer, Optical measurements for rosetta navigation near the comet, in: Proceedings of the 25th International Symposium on Space Flight Dynamics (ISSFD), Munich.

[14] H. J. S. Feder, J. J. Leonard, C. M. Smith, Adaptive mobile robot navigation and mapping, The International Journal of Robotics Research 18 (1999) 650–668.

[15] H. Carrillo, I. Reid, J. A. Castellanos, On the comparison of uncertainty criteria for active slam, in: 2012 IEEE International Conference on Robotics and Automation, IEEE, pp. 2080–2087.

[16] T. Kollar, N. Roy, Trajectory optimization using reinforcement learning for map exploration, The International Journal of Robotics Research 27 (2008) 175–196.

[17] A.-a. Agha-mohammadi, S. Chakravorty, N. M. Amato, FIRM : Sampling-based Feedback Motion Planning Under Motion Uncertainty and Imperfect Measurements, The International Journal of Robotics Research 33 (2003) 268–304.

[18] D. Izzo, M. Märtens, B. Pan, A survey on artificial intelligence trends in spacecraft guidance dynamics and control, Astrodynamics 3 (2019) 287–299.

[19] B. Gaudet, R. Furfaro, R. Linares, Reinforcement learning for angle-only intercept guidance of maneuvering targets, Aerospace Science and Technology 99 (2020) 105746.

[20] Y. Zhou, E.-J. van Kampen, Q. Chu, Incremental model based online heuristic dynamic programming for nonlinear adaptive tracking control with partial observability, Aerospace Science and Technology 105 (2020) 106013.

[21] M. Riedmiller, Neural fitted q iteration – first experiences with a data efficient neural reinforcement learning method, in: J. Gama, R. Camacho, P. B. Brazdil, A. M. Jorge, L. Torgo (Eds.), Machine Learning: ECML 2005, Springer Berlin Heidelberg, Berlin, Heidelberg, 2005, pp. 317–328.

[22] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, et al., Human-level control through deep reinforcement learning, Nature 518 (2015) 529.

[23] S. Li, P. Cui, H. Cui, Autonomous navigation and guidance for landing on asteroids, Aerospace science and technology 10 (2006) 239–247.

[24] H. Shang, X. Wu, D. Qiao, X. Huang, Parameter estimation for optimal asteroid transfer trajectories using supervised machine learning, Aerospace Science and Technology 79 (2018) 570–579.

[25] V. Pesce, A.-a. Agha-mohammadi, M. Lavagna, Autonomous navigation & mapping of small bodies, in: 2018 IEEE Aerospace Conference, IEEE, pp. 1–10.

[26] D. M. Chan, A.-a. Agha-mohammadi, Autonomous imaging and map-

ping of small bodies using deep reinforcement learning, in: 2019 IEEE Aerospace Conference.

[27] M. Piccinin, M. R. Lavagna, Deep reinforcement learning approach for small bodies shape reconstruction enhancement, in: AIAA Scitech 2020 Forum, p. 1909.

[28] R. S. Sutton, A. G. Barto, Reinforcement learning: An introduction, MIT press, 2018.

[29] H. Durrant-Whyte, T. Bailey, Simultaneous localization and mapping: part i, IEEE Robotics Automation Magazine 13 (2006) 99–110.

[30] T. Bailey, H. Durrant-Whyte, Simultaneous localization and mapping (slam): part ii, IEEE Robotics Automation Magazine 13 (2006) 108–117.

[31] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, J. J. Leonard, Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age, IEEE Transactions on Robotics 32 (2016) 1309–1332.

[32] L. Mihaylova, T. Lefebvre, H. Bruyninckx, K. Gadeyne, J. De Schutter, A comparison of decision making criteria and optimization methods for active robotic sensing, in: International Conference on Numerical Methods and Applications, Springer, pp. 316–324.

[33] J. Jiang, X. Zeng, D. Guzzetti, Y. You, Path planning for asteroid hopping rovers with pre-trained deep reinforcement learning architectures, Acta Astronautica 171 (2020) 265–279.

[34] A. Géron, Hands-on machine learning with Scikit-Learn and TensorFlow: concepts, tools, and techniques to build intelligent systems, O'Reilly Media, Inc., 2017.

[35] M. Riedmiller, H. Braun, A direct adaptive method for faster backpropagation learning: The rprop algorithm, in: IEEE international conference on neural networks, IEEE, pp. 586–591.