

# Spatio-Temporal Path Planning for Lunar Polar Exploration with Robustness against Schedule Delay\*

Hiroka INOUE<sup>1),2)†</sup> and Shuichi ADACHI<sup>2)</sup>

<sup>1)</sup>Institute of Space and Astronautical Science, Japan Aerospace Exploration Agency, Sagamihara, Kanagawa 252-5210, Japan

<sup>2)</sup>Department of Applied Physics and Physico-Informatics, Keio University, Yokohama, Kanagawa 223-8522, Japan

This paper presents a method to obtain a robust optimal path in an environment with time-varying safety features, such as in the lunar polar region. In designing the path for planetary exploration rovers, we must consider various safety conditions, such as terrain hazards, illumination, and communication visibility to the Earth. Some of the safety features are time-varying, and the optimal path should be searched for both the spatial direction and the temporal direction. In addition, there is no guarantee that all of the sequences will be successfully executed on time due to misoperation, failures, or trouble. Therefore, a path that is robust against the delay must be planned so as to guarantee safety even when behind schedule. The authors propose an algorithm called “Robust Spatio-Temporal Path Planner for the Planetary Rover (ROBUST-STP3R)” to obtain a path that is robust against schedule delay in a time-varying environment. This method defines a cost function that consists of the distance as well as the region type cost. To add robustness against schedule delays, the authors consider a weighted summation of the time-varying region type cost with regard to the temporal direction. The effectiveness of the proposed method is demonstrated through use in a simulated lunar polar exploration exercise.

**Key Words:** Guidance and Control, Spatio-temporal, Path Planning, Lunar Polar Region, Robustness against Schedule Delay

## 1. Introduction

Evidence of volatiles in the lunar polar region has been provided by remote-sensing data from a neutron spectrometer<sup>1)</sup> or observation results of the LCROSS mission.<sup>2,3)</sup> Volatiles (e.g., water ice) will be essential for future lunar missions as well as humans living on the Moon because they can be utilized as drinking water or energy resources. Additionally, the lunar polar region has advantages compared with other regions due to its long-term illumination. For these reasons, the lunar polar region has the potential to be a future Moon base.

Recently, many exploration missions targeting the lunar polar region have been planned worldwide. For example, the Japan Aerospace Exploration Agency (JAXA) is planning an exploration mission<sup>4)</sup> that will be executed using a combination of landers and rovers (see Fig. 1). To confirm the existence of a sufficient amount of volatiles for future lunar missions, their distribution and quantity must be investigated. This requirement is directly related to the necessity of in-situ scientific observations at many points; hence, JAXA plans to execute surface exploration missions in the lunar polar region.

For rover exploration missions, the path for the rover must be planned carefully.<sup>5,6)</sup> While the primary objective is to explore scientifically interesting regions, the rover must also avoid hazards such as rocks, steep slopes, and shadows. In a lunar polar mission, safety features such as illumination

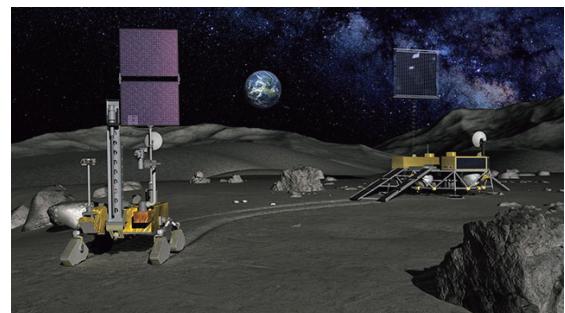


Fig. 1. Conceptual image of JAXA’s future lunar polar mission. (Courtesy of JAXA)

conditions drastically change. Therefore, mission plans should be more risk-sensitive regarding landing site selection<sup>7)</sup> and traversability analysis.<sup>8)</sup> In addition, when the rover is remotely operated, availability of communication windows must be considered.

One important point to be noted is that illumination and communication are time-varying. To avoid hazards in a time-varying safety environment, it is important to understand and manage the time-series data related to illumination and communications. In addition, the rover cannot necessarily execute all of the sequences in the mission period successfully. Since there can be misoperation, problems, or failures, the rover may be behind schedule. In the time-varying environment, the rover may visit an unsafe position due to a delay even if safety is guaranteed in the nominal plan. Hence, it is important to design a path robust against schedule delay.

### 1.1. Related work

Ono et al.<sup>9)</sup> proposed a rover operation tool that included

the machine learning-based terrain classification and a risk-aware path planner based on rapidly-exploring random graph (RRG) and the A\* search algorithms. Ono et al.<sup>10)</sup> proposed a traversability analysis method for Mars 2020 landing site selection. The method uses HiRISE images for the automated terrain classifier and DEM generation et al., finds optimal routes using a pseudo traveling salesman problem between multiple regions of interest (multi-ROI).

Path planning methods for time-varying environments were studied by Alvarez et al.<sup>11)</sup> and Lawrence et al.<sup>12)</sup> In previous work on path planning in the lunar polar region, Otten et al.<sup>13)</sup> planned a path that guarantees continuous illumination and a traversable slope. Additionally, Speyerer et al.<sup>14)</sup> proposed a method to optimize a path considering time-varying illumination as a safety condition. This work defined the cost as the summation of energy throughout the path from the start to the goal. Finally, Cunningham et al.<sup>15)</sup> conducted spatio-temporal path planning in the lunar polar region considering illumination, communication, obstacles, and slopes. In addition, Bai et al.<sup>16)</sup> conducted path planning considering slope and illumination conditions. In these articles, illumination or communication is usually treated as a hard constraint, and areas without them are regarded as no-go regions. However, it is not necessary to prohibit an agent from visiting a place without illumination or communication. In addition, the previous work mentioned above did not incorporate robustness against delays. Hence, when the schedule is behind, the rover may not have any viable action for visiting a safe region with illumination or communication.

Robustness in path planning has been a central issue in mission planning, and this has been studied by Blackmore et al.,<sup>17,18)</sup> Aoude et al.,<sup>19)</sup> and Mahmoud et al.<sup>20)</sup> However, most previous works focused on planning a path that is robust against uncertainty associated with localization, disturbances, or modeling errors.

## 1.2. Contributions

In this paper, the authors propose an algorithm called the “Robust Spatio-Temporal Path Planning for the Planetary Rover (ROBUST-STP3R).” This method designs a path robust against the delays in a time-varying environment. A spatio-temporal map is constructed by combining time-invariant maps (e.g., obstacles and steep slopes) and time-varying maps (e.g., illumination and communication). All of the nodes are then classified into five types based on the

spatio-temporal map generated. We define a new cost function consisting of distance and region type representing the existence of obstacles, illumination, and communication. By defining the region type cost as the summation of nominal costs and an additional penalty, a penalty can be added to the agent for an unsafe action or when in an unsafe region. In addition, to provide robustness against schedule delay, the weighted summation of time-varying cost in regards to time is calculated, in which the weight is characterized by the negative binomial distribution. Finally, a traditional A\* algorithm is solved in the spatio-temporal map, the effectiveness of the proposed method is demonstrated in a randomly-generated synthetic environment and simulated lunar polar exploration.

## 2. Environment Data of the Lunar Polar Region

The lunar polar region is a severe environment for rovers because of dynamic changes in illumination and communication visibility. Figure 2 shows the pictures taken by the Lunar Reconnaissance Orbiter (LRO) using a narrow-angle camera (NAC) at Connecting Ridge (89.46° S, 137.49° W). Observe that the illumination condition changes dynamically over time.

To incorporate time-varying safety features in path planning for the lunar polar region, a map that includes information and communication availability must first be created. For path planning of future lunar polar exploration missions, an ephemeris of the Sun, Earth and Moon is created. Time-varying illumination and communications are then simulated.

The JAXA Lunar and Planetary Exploration Data Analysis (JLPEDA) group simulated illumination and communication conditions of the lunar polar region.<sup>21)</sup> This simulation is similar to those reported in Mazarico et al.,<sup>22)</sup> and Gläser et al.,<sup>23,24)</sup> which are based on digital terrain models (DTMs) and SPICE toolkits.<sup>25)</sup> We created DTMs using data from the lunar orbiter laser altimeter (LOLA) of the LRO and the terrain camera (TC) of JAXA’s orbiter, SELENE/Kaguya. The resolution of the created DTMs is 2 m/pixel.

Using the created DTMs and SPICE toolkit, retracing is conducted to obtain future illumination and communication visibility conditions. The illumination ratio map and the communication visibility map of the South Pole for one year (April 1, 2022 to March 31, 2023) are presented on the left

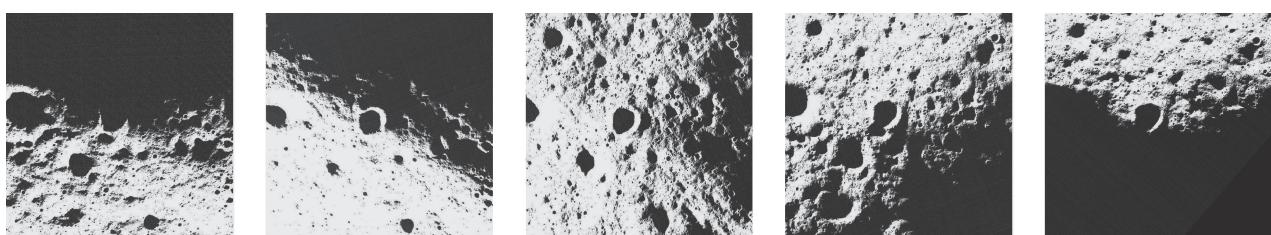


Fig. 2. A 3.0 × 3.0-km region around Connecting Ridge (89.46° S, 137.49° W) in the orthorectified NAC images. (M138264280, M140625419, M145354371, M180899951, M1100039834)

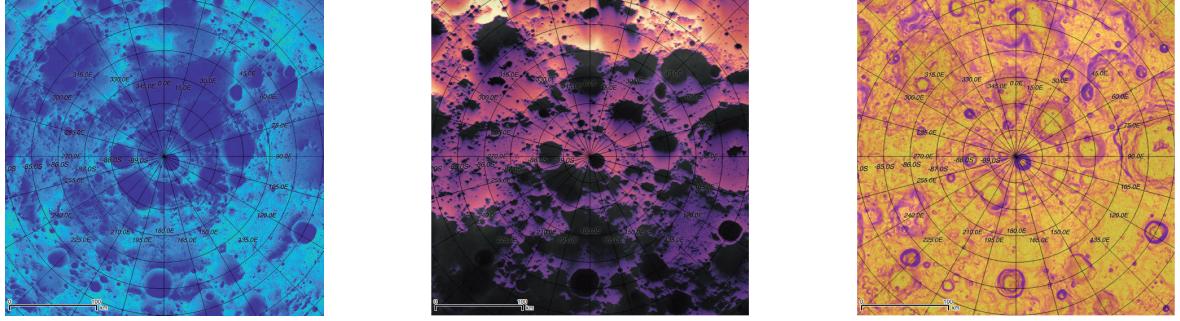


Fig. 3. Polar stereo graphic image of the lunar south polar region south of  $84^{\circ}$  S. (Left) The illumination ratio map. Lighter color indicates high illumination ratio. (Middle) Communication visibility ratio map. Lighter color indicates high communication visibility ratio. (Right) Slope map. Lighter color indicates low slope angles (i.e., safe regions).

#### Algorithm 1 ROBUST-STP3R Algorithm

- 1: **Inputs:** Region  $\mathcal{R}_{\text{all}}$ , Obstacle region  $\mathcal{R}_{\text{ob}}$ , Illumination region  $\mathcal{R}_{\text{il}}$ , Communication region  $\mathcal{R}_{\text{com}}$ , Current node  $n_i$ , Next node  $n_j$
- 2: Create Merged Map
- 3:  $n \leftarrow (x, y, t)$
- 4:  $d(n_i, n_j) \leftarrow$  Euclidean distance between  $\mathcal{P}(n_i)$  and  $\mathcal{P}(n_j)$
- 5:  $r(n_i, n_j) \leftarrow$  Equation (6)
- 6:  $\hat{r}(n_i, n_j) \leftarrow$  Equation (7)
- 7: Calculate  $c(n_i, n_j)$  by

$$c(n_i, n_j) = d(n_i, n_j) + \alpha \cdot \hat{r}(n_i, n_j)$$

- 8: Solve A\* algorithm on the three dimensional merged map (2D for state space and 1D for time) as follows:

$$\min \sum c(n_i, n_j)$$

and in the middle of Fig. 3. In these maps, the lighter color indicates a high illumination/communication ratio. These conditions change dynamically with time. In addition, a slope map was obtained using DTMs (right image in Fig. 3).

### 3. ROBUST-STP3R

#### 3.1. Overview of the proposed method

Algorithm 1 outlines the authors' proposed method called ROBUST-STP3R. This method enables obtaining the minimum cost path with robustness against delay in a time-varying environment.

The following procedure shows how to obtain the robust optimal path using ROBUST-STP3R.

- Construct a merged map of the three dimensions and classify the region (Section 3.2)
- Define the region type cost (Section 3.3)
- Add the robustness against schedule delay (Section 3.4)
- Plan a path using the merged map and the cost function (Section 3.5)

Intuitively, the space being considered is first extended to three dimensions (i.e., two-dimensional for space and one-dimensional for time), as shown in Fig. 4. Specifically, while conventional path planning methods store only spatio infor-

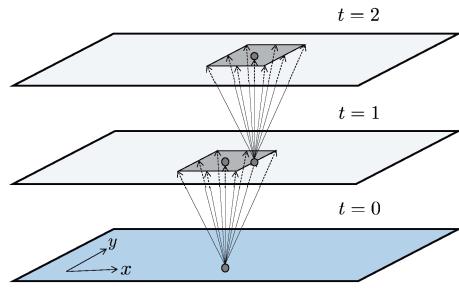


Fig. 4. Motion model: An agent has nine choices for the next action. In this figure, “ $t$ ” represents the time step.

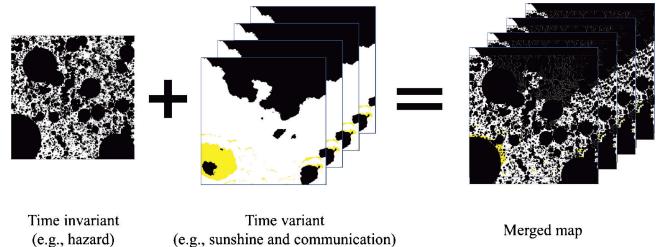


Fig. 5. Merged map. By combining the time-invariant map (e.g., obstacles or slope angle) and time-varying map (e.g., illumination or communication conditions), several maps were merged together and regions were classified into five types.

mation in each node, the proposed method additionally stores temporal information. Therefore, even if the rover stops at the same position, it will move to a different node in the algorithm as time passes. By managing time, path planning can be robust to schedule delays. Next, the time-varying cost functions are modeled. The cost function is characterized by the distance and region type cost; the latter is associated with environmental safety features in order to obtain a reasonable path. Then, to plan a path robust against schedule delay, the weighted summation of the time-varying cost in regards to time direction is calculated. The weight is characterized by the negative binomial distribution, which makes it possible to incorporate how likely the rover is to be late.

#### 3.2. Merged map construction and region classification

In the problem setting for this study, the authors consider

three types of maps: obstacle, illumination, and communication.<sup>‡</sup> These maps were synthesized into a merged map (see Fig. 5). The obstacles are detected by hand using orthorectified NAC images. In the merged map, the whole state space is classified into five region types (i.e., types A through E). A brief explanation of how the state space is classified into five region types using obstacle, illumination, and communication data follows.

In this research, each node has information on the 2D position and time. A node is denoted as  $n$  and a set of nodes as  $\mathcal{N}$ . Here two functions are defined. The first function, denoted as  $\mathcal{T}$ , is to return time on  $n$ . The other function, denoted as  $\mathcal{P}$ , is to return 2D position on  $n$ . For  $n$  with the position of  $(\bar{x}, \bar{y})$  on time  $\bar{t}$ , then  $\bar{t} = \mathcal{T}(n)$  and  $(\bar{x}, \bar{y}) = \mathcal{P}(n)$  hold.

First, sets of nodes are defined using obstacle ( $\mathcal{R}_{ob}$ ), illumination ( $\mathcal{R}_{il}$ ), and communication ( $\mathcal{R}_{com}$ ). Each set is expressed as:

$$\mathcal{R}_{ob} := \{n \in \mathcal{N} \mid \text{An obstacle exist at } \mathcal{P}(n)\}, \quad (1)$$

$$\mathcal{R}_{il} := \{n \in \mathcal{N} \mid \text{Illumination is available at } n\}, \quad (2)$$

$$\mathcal{R}_{com} := \{n \in \mathcal{N} \mid \text{Communication is feasible at } n\}. \quad (3)$$

Note that the obstacle map is time-invariant, unlike the illumination and communication maps. Therefore,  $\mathcal{R}_{ob}$  is not associated with  $\mathcal{T}(n)$ .

Based on the Eqs. (1)–(3), we classify all of the nodes into five types (i.e., types A through E) depending on the safety features. This concept is presented in Table 1. Let  $\mathcal{S}_A$ ,  $\mathcal{S}_B$ ,  $\mathcal{S}_C$ ,  $\mathcal{S}_D$ , and  $\mathcal{S}_E$  denote the set of nodes belonging to types A through E. We formally label the set of nodes as:

$$\begin{aligned} \mathcal{S}_A &:= \bar{\mathcal{R}}_{ob} \cap \mathcal{R}_{il} \cap \mathcal{R}_{com}, \\ \mathcal{S}_B &:= \bar{\mathcal{R}}_{ob} \cap \{\mathcal{R}_{il} \setminus (\mathcal{R}_{il} \cap \mathcal{R}_{com})\}, \\ \mathcal{S}_C &:= \bar{\mathcal{R}}_{ob} \cap \{\mathcal{R}_{com} \setminus (\mathcal{R}_{il} \cap \mathcal{R}_{com})\}, \\ \mathcal{S}_D &:= \bar{\mathcal{R}}_{ob} \setminus (\mathcal{R}_{il} \cup \mathcal{R}_{com}), \\ \mathcal{S}_E &:= \mathcal{R}_{ob}, \end{aligned}$$

where  $\bar{\mathcal{X}}$  is the complementary set of a set  $\mathcal{X}$ .

### 3.3. Region type cost

Region type cost is determined using the illumination and communication conditions of a pair of adjacent nodes  $n_i$  and  $n_j$ . The nominal cost is defined using the visited node, and then some penalty is added for special cases with regard to the action. For example, this cost function definition criteria makes it possible to encourage an agent to visit an illuminated region while making the agent stay when communication is not available.

#### 3.3.1. Nominal cost depending on the terrain type

The nominal part of the region type cost is defined here. This cost is simply allocated according to the type of the next visited node, which is defined as:

Table 1. Classification of regional space.

Type	Obstacle	Illumination	Communication	Preferable action
A	None	Good	Good	Can visit without cost
B	None	Good	Bad	Stay
C	None	Bad	Good	Weakly discouraged
D	None	Bad	Bad	Discouraged
E	Exist	—	—	Must not visit

$$r_{\text{nom}}(n_i, n_j) := \begin{cases} 0 & \text{if } n_j \in \mathcal{S}_A \\ r_B & \text{if } n_j \in \mathcal{S}_B \\ r_C & \text{if } n_j \in \mathcal{S}_C \\ r_D & \text{if } n_j \in \mathcal{S}_D \\ \infty & \text{if } n_j \in \mathcal{S}_E, \end{cases} \quad (4)$$

where  $r_B, r_C, r_D \in \mathbb{R}$ , and  $r_B, r_C, r_D \geq 0$  are the cost function values. When discouraging an agent from visiting a region without illumination and/or communication visibility,  $r_B, r_C$ , and  $r_D$  should be large values. In addition, an agent must not visit type E when the cost is defined as  $\infty$ . Note that  $r_{\text{nom}}$  is determined only by  $n_j$ .

#### 3.3.2. Additional penalty for moving without communication availability

An agent that is remotely controlled by human operators may need to stay at the current position until it recovers communication with the ground stations. In order to simulate such behavior, an action (i.e., state and action pair) must be taken into account; that is, the nominal cost function defined previously is insufficient for this purpose. As a consequence, an additional penalty related to the actions is defined.

For clarity of exposition, the additional penalty for cases of no communication is explained. Suppose the rover is prohibited from moving to another position without communication. For every  $n_i \in \mathcal{S}_B \cup \mathcal{S}_D$ , the additional cost is defined as:

$$r_{\text{ad}}(n_i, n_j) := \begin{cases} 0 & \text{if } \mathcal{P}(n_i) = \mathcal{P}(n_j), \\ r_{\text{move}} & \text{otherwise,} \end{cases} \quad (5)$$

where  $r_{\text{move}} \in \mathbb{R}$ ,  $r_{\text{move}} \geq 0$  is the additional penalty for the *not stay* action under the condition of no communication availability.

#### 3.3.3. Overall region type cost

Based on the previously defined nominal cost,  $r_{\text{nom}}$ , and the additional penalty,  $r_{\text{ad}}$ , the overall region type cost  $r$  is formally stated as:

$$r(n_i, n_j) = r_{\text{nom}}(n_i, n_j) + r_{\text{ad}}(n_i, n_j). \quad (6)$$

### 3.4. Robustness against the schedule delay

Planetary rovers that explore uncertain environments may well have problems or accidents. Take, for example, NASA's Mars rovers *Spirit* and *Curiosity* experienced accidents in which the rover's wheels became embedded in deep sand. If a rover encounters such accidents, the whole schedule gets delayed. A path that was planned for a nominal schedule may become unsafe due to the delays; that is, we should thus consider the possibility of the delays for planning

<sup>‡</sup>For the simplicity, we assume that the obstacle includes a steep slope. More concretely, in our simulation, we define a hill with a slope angle greater than 25 degrees as an obstacle.

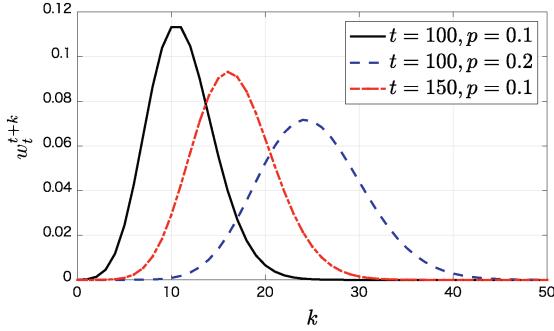


Fig. 6. Characteristics of the weight function,  $w_t^{t+k}$ , which represents the probability of being delayed by  $k$  time steps until time  $t+k$ . As  $t$  and  $p$  increases, the mode also increases.

a path a priori.

To add robustness against delays, the weighted summation of the region type cost in relation to time is incorporated. More specifically, the region type cost is replaced as follows:

$$\begin{aligned} \hat{r}(\cdot, n_j) := & w_t^t \cdot r(\cdot, n_j) + w_t^{t+1} \cdot r(\cdot, \{\mathcal{P}(n_j), \mathcal{T}(n_j) + 1\}) \\ & + w_t^{t+2} \cdot r(\cdot, \{\mathcal{P}(n_j), \mathcal{T}(n_j) + 2\}) + \dots, \end{aligned} \quad (7)$$

where  $w$  is a weight coefficient. Intuitively, Eq. (7) calculates a kind of expected value of region type cost depending on how late the rover is likely to be. *How then, should  $w$  be decided?*

The authors propose a method to determine the weight coefficient using the negative binomial distribution. A weight is assigned by considering the possibility of being behind schedule. If the rover reaches a position by executing  $t$  actions at time  $t+k$ , it means that the rover has  $k$  time delays till time  $t+k-1$  and successfully executes an action at time  $t+k$ . Hence, the weight  $w_t^{t+k}$  is interpreted as the probability of a rover being delayed  $k$  times for reaching a position at time  $t+k$ , which is expressed as:

$$\begin{aligned} w_t^{t+k} = & {}_{t+k-1}C_k \cdot p^{k-1} \cdot (1-p)^t \cdot p \\ = & {}_{t+k-1}C_k \cdot p^k (1-p)^t, \end{aligned} \quad (8)$$

where  $p \in [0, 1]$  is the probability of being delayed for each time step, and  ${}_{t+k-1}C_k$  represents the number of combinations to choose  $k$  elements from  $t+k-1$  elements.

The characteristics of the weights  $w_t^{t+k}$  for various  $p$  are shown in Fig. 6. The figure represents the probability of being delayed by  $k$  time steps until time  $t+k$ . The characteristic of this weight coefficient meets expectations. When  $p$  increases, the possibility of being behind the schedule also gets larger; hence, the peak of  $w$  shifts to the right. In addition, for a larger  $t$  (i.e., greater cumulative elapsed time of the mission), the peak of  $w$  shifts to the right as well. Note that the weight  $w$  converges to zero, which means that the appropriate number of horizons is automatically decided.

### 3.5. Path planning using the created map and cost function

Let  $d : \mathcal{N} \times \mathcal{N} \rightarrow \mathbb{R}$  be the Euclidean distance. The overall cost function  $c : \mathcal{N} \times \mathcal{N} \rightarrow \mathbb{R}$  is defined as:

$$c(n_i, n_j) = d(n_i, n_j) + \alpha \cdot \hat{r}(n_i, n_j), \quad (9)$$

where  $\alpha \in \mathbb{R}$  is a weight coefficient.

Finally, the optimization problem to solve is formally stated as

$$\min \sum c(n_i, n_j). \quad (10)$$

By solving the optimization problem Eq. (10), an optimal path robust against delays in an environment with spatio-temporal safety features is obtained. Note that Eq. (10) can be solved using conventional methods such as the A\* algorithm.<sup>26)</sup>

## 4. Experiments

In this section, the results of simulations using two types of data are presented. One uses a grid world with randomly changing terrain types, and the other uses real data of the Moon.

### 4.1. Synthetic data

Consider a time-varying environment where a three-dimensional (3D) map is formed using  $24 \times 11 \times 11$  rectangular grids, and the mission period is changed by delays. This environment is randomly created to include all region types. At every point (except the boundary), the agent can take one of nine actions, as indicated in Fig. 4.

The effectiveness of the proposed ROBUST-STP3R is shown by comparing the path length and safety with 1) the shortest path, 2) the hard constraints method reported by Otten et al.<sup>13)</sup> and Bai and Oh,<sup>16)</sup> and 3) STP3R (i.e., not including robustness against schedule delay). The hard constraints method passes through safe regions that have good illumination and communication conditions for the entire mission period; that is, the parameters are set as  $r_A = 0$ ,  $r_B = r_C = r_D = r_E = \infty$ . STP3R optimizes the path without considering robustness against delays. In other words, for STP3R, the cost function is defined as  $c = d + \alpha \cdot r$ . As a measure for evaluation, the number of visits to each region type is used. For every method tested, the cost for distance is  $d = 0$  for stay action,  $d = 1$  for actions to move horizontally or vertically, and  $d = \sqrt{2}$  otherwise. Using STP3R and ROBUST-STP3R, for the nominal part of the region type cost,  $r$ ,  $r_A = 0$ ,  $r_B = r_C = 0.5$  and  $r_D = r_E = \infty$ . The additional penalty for region type cost,  $r_{ad}$ , is set as  $r_{move} = 10$  for types B and D, which means that a large penalty is added for not staying when communication is unavailable.

A simulation using samples of randomly generated was carried out. It is not always possible to accurately predict the failure rate a priori; that is, the failure rate assumed (i.e.,  $p$ ) is not necessarily the same as the true failure rate. Let  $p^*$  denote the true failure rate. Hence, for each algorithm, three simulations in total were conducted with the true failure rate  $p^*$  as 0, 0.1, and 0.5, while the assumed failure rate (i.e.,  $p$ ) of ROBUST-STP3R was fixed as 0.5.

The simulation results obtained are summarized in Table 2. ROBUST-STP3R plans a safer path for various true

Table 2. Comparison of ROBUST-STP3R with three baselines in terms of the average path length and the percentage of the number of visits to each region type.

(a)  $p^* = 0.0$

Path length	Percentage of the regions					
	A	B	C	D	E	
Shortest path	5.7	20	55	20	5	0
Hard constraints	11	100	0	0	0	0
STP3R	7.7	30	50	20	0	0
ROBUST-STP3R	<b>11</b>	<b>95</b>	<b>0</b>	<b>5</b>	<b>0</b>	<b>0</b>

(b)  $p^* = 0.1$

Path length	Percentage of the regions					
	A	B	C	D	E	
Shortest path	5.7	24	48	24	5	0
Hard constraints	—	—	—	—	—	—
STP3R	7.7	29	52	14	5	0
ROBUST-STP3R	<b>11</b>	<b>90</b>	<b>5</b>	<b>5</b>	<b>0</b>	<b>0</b>

(c)  $p^* = 0.5$

Path length	Percentage of the regions					
	A	B	C	D	E	
Shortest path	5.7	9	41	25	25	0
Hard constraints	—	—	—	—	—	—
STP3R	7.7	9	44	22	25	0
ROBUST-STP3R	<b>11</b>	<b>78</b>	<b>22</b>	<b>0</b>	<b>0</b>	<b>0</b>

Table 3. Comparison of ROBUST-STP3R using three baselines in terms of the average path length and the percentage of the number of visits to each region type.

(a)  $p^* = 0.0$

Path length	Percentage of the regions					
	A	B	C	D	E	
Shortest path	55	92	8	0	0	0
Hard constraints	—	—	—	—	—	—
STP3R	55	92	8	0	0	0
ROBUST-STP3R	<b>67</b>	<b>92</b>	<b>8</b>	<b>0</b>	<b>0</b>	<b>0</b>

(b)  $p^* = 0.1$

Path length	Percentage of the regions					
	A	B	C	D	E	
Shortest path	55	83	17	0	0	0
Hard constraints	—	—	—	—	—	—
STP3R	55	83	17	0	0	0
ROBUST-STP3R	<b>67</b>	<b>83</b>	<b>17</b>	<b>0</b>	<b>0</b>	<b>0</b>

(c)  $p^* = 0.5$

Path length	Percentage of the regions					
	A	B	C	D	E	
Shortest path	55	60	14	1	26	0
Hard constraints	—	—	—	—	—	—
STP3R	55	60	14	1	26	0
ROBUST-STP3R	<b>67</b>	<b>61</b>	<b>37</b>	<b>0</b>	<b>2</b>	<b>0</b>

failure rates when compared to the shortest path and STP3R. The hard constraints method can plan a safe path only within a pre-arranged period. In addition, in an environment such as the lunar polar regions, where the mission duration can be long and the timing of the local loss of illumination is unknown, path planning using the hard constraints method can often result in no solution. As evidence, in Table 2(c), the hard constraints baseline failed to find a viable solution.

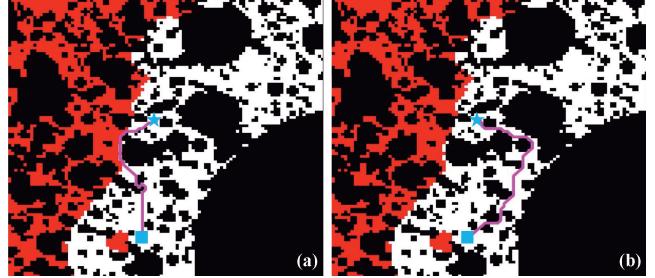


Fig. 7. (a) The shortest path. (b) ROBUST-STP3R.

The path that is obtained using ROBUST-STP3R is relatively longer, but tries to guarantee safety while considering delays. Additionally, ROBUST-STP3R can provide some answers in any environment. When used in a real mission, the ability to obtain several solutions by adjusting parameters such as the proposed method is a great advantage.

#### 4.2. Lunar observation data

Next, the effectiveness of ROBUST-STP3R in a simulated Moon surface exploration scenario is demonstrated. The map of the environment includes information on time-invariant safety features (i.e., obstacles and slope angles) and time-variant features (i.e., illumination and communication availability). Real data of a region covering 200 m × 200 m around the de Gerlache rim (88.664° S, 68.398° W), which is of the landing site candidates for a future lunar polar mission, is used.

The resolution of the map is 2 m, so assume grids 100 times 100 in size. There are 70 figures, and the total mission period is assumed to be three days. The time step between figures is 1 h, and the total number of nodes is thus  $100 \times 100 \times 70 = 7.0 \times 10^5$ . Additionally, the velocity of the rover is assumed to be between 1.0 and 3.0 m/h.

The same baselines used in the first experiment are applied here. Cost functions are defined as follows. Cost for distance is  $d = 0$  for stay action,  $d = 1$  for actions to move horizontally or vertically, and  $d = \sqrt{2}$  otherwise. For region type cost,  $r$ , parameters are set as  $r_A = 0$ ,  $r_B = 10$ ,  $r_C = \infty$ ,  $r_D = \infty$ , and  $r_{\text{move}} = 10$ . This cost function is based on the requirement that an agent must not visit the shadow region.

A merged map of the lunar polar region was created using two time-invariant maps (obstacles and slope angle) and two time-varying maps (illumination and communication). In this simulation, the slope angle is regarded as the binary function; that is, if the slope angle is more than 25 deg, the state is regarded an obstacle (i.e., cost is infinity); otherwise it is regarded as safe. By combining four maps for each time step, a merged map was created and the region space classified into the five types indicated in Table 1.

The ROBUST-STP3R algorithm was ran for the merged map. Figure 7 shows the shortest path and the path obtained from ROBUST-STP3R. The path (magenta line) extends from the start (cyan square) to the goal (cyan star) while avoiding the no-communication region (red). Observe that the shortest path proceeds left to the destination. On the other hand, the

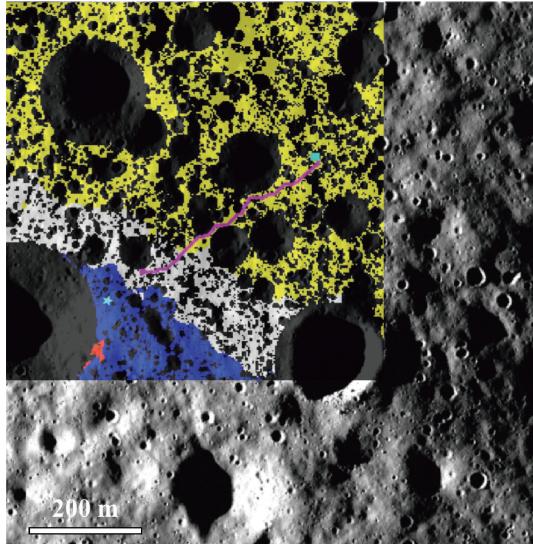


Fig. 8. Magenta line represents the path obtained using the ROBUST-STP3R algorithm in a simulation applying real lunar data. The base map colors represent the region type as described in Table 1.

proposed method turns proceeds to the destination in order to avoid the shadow looming from the left side of the figure by sacrificing path length. If the robustness against schedule is not incorporated, the rover cannot consider the risk of future shadow in the case of stay the location by delay. Table 3 shows that ROBUST-STP3R tends to have a larger path length than the other baselines, but successfully chooses a safer route in the case of schedule delays. In the actual rover operation, the proposed method is expected to work more effectively since the rover conducts the long-distance travel on the lunar surface by repeating the short-term offline path planning many times. While, the shortest path and STP3R are reasonable paths in the cases of Table 3(a) and (b), the hard constraints baseline failed to find a viable solution. This indicates that there is no path in the targeted lunar region with good illumination and communication conditions throughout the mission period. This situation can occur frequently in lunar polar regions if the mission duration is somewhat long.

For this simulation a path with soft constraints using the ROBUST-STP3R framework is planned. However, depending on how the region type cost is defined, the shortest path or a path determined using hard constraints can be obtained. This flexibility in the ROBUST-STP3R algorithm is considered to be an advantage.

The ROBUST-STP3R was also tested using a larger map with  $250 \times 250$  grids. The experimental results are shown in Fig. 8. This shows that the computational cost of the proposed algorithm is small enough to use in a real lunar exploration mission.

## 5. Conclusions

The authors have proposed a novel method, ROBUST-STP3R, obtain a robust path in time-varying environments. First, the entire region is classified into five types with regard to the various kinds of safety features. The cost function is

then modeled as the summation of the distance and the region type cost; the latter consists of the nominal cost and an additional penalty. The proposed method then takes into account the robustness against schedule delay, which leverages the negative binomial distribution. The effectiveness of ROBUST-STP3R was demonstrated through two simulations, including one that applied real observation data.

Future work includes reducing computational complexity and solving the path-planning problem for a larger regional space. For example, it would be interesting to obtain an optimal solution with low computational cost using a sample-based method such as RRT<sup>27)</sup> or RRT\*.<sup>28)</sup>

## Acknowledgments

We wish to JAXA's lunar polar exploration mission members for useful discussions. We are grateful to Mitsuo Yamamoto for assistance with the numerical simulations. Finally, we thank JLPEDA for sharing their dataset with us.

## References

- Mitrofanov, I. G., Sanin, A. B., Boynton, W. V., Chin, G., Garvin, J. B., Golovin, D., Evans, L. G., Harshman, K., Kozyrev, A. S., Litvak, M. L., et al.: Hydrogen Mapping of the Lunar South Pole Using the LRO Neutron Detector Experiment Lend, *Science*, **330** (2010), pp. 483–486.
- Gladstone, G. R., Hurley, D. M., Rutherford, K. D., Feldman, P. D., Pryor, W. R., Chaufray, J., Versteeg, M., Greathouse, T. K., Steffl, A. J., Throop, H., et al.: LRO-LAMP Observations of the LCROSS Impact Plume, *Science*, **330** (2010), pp. 472–476.
- Colaprete, A., Schultz, P., Heldmann, J., Wooden, D., Shirley, M., Ennico, K., et al.: Detection of Water in the LCROSS Ejecta Plume, *Science*, **330** (2010), pp. 463–468.
- Hoshino, T., Wakabayashi, S., Ohtake, M., Karouji, Y., Hayashi, T., Morimoto, H., et al.: Lunar Polar Exploration Mission for Water Prospection – JAXA's Current Status of Joint Study with ISRO, *Acta Astronautica*, **176** (2020), pp. 52–58.
- Gennery, D. B.: Traversability Analysis and Path Planning for a Planetary Rover, *Autonomous Robots*, **6** (1999), pp. 131–146.
- Singh, S., Simmons, R., Smith, T., Stentz, A., Verma, V., Yahja, A., and Schwehr, K.: Recent Progress in Local and Global Traversability for Planetary Rovers, IEEE International Conference on Robotics and Automation, 2000.
- De Rosa, D., Bussey, B., Cahill, J. T., Lutz, T., Crawford, I. A., Hackwill, T., et al.: Characterisation of Potential Landing Sites for the European Space Agency's Lunar Lander Project, *Planet. Space Sci.*, **74** (2012), pp. 224–246.
- Heldmann, J. L., Colaprete, A., Elphic, R. C., Bussey, B., McGovern, A., Beyer, R., et al.: Site Selection and Traverse Planning to Support a Lunar Polar Rover Mission: A Case Study at Haworth Crater, *Acta Astronautica*, **127** (2016), pp. 308–320.
- Ono, M., Fuchs, T. J., Steffl, A., Maimone, M., and Yen, J.: Risk-aware Planetary Rover Operation: Autonomous Terrain Classification and Path Planning, IEEE Aerospace Conference, 2015, pp. 1–10.
- Ono, M., Rothrock, B., Almeida, E., Ansar, A., Otero, R., Huertas, A., and Heverly, M.: Data-driven Surface Traversability Analysis for Mars 2020 Landing Site Selection, IEEE Aerospace Conference, 2016, pp. 1–12.
- Alvarez, A., Caiti, A., and Onken, R.: Evolutionary Path Planning for Autonomous Underwater Vehicles in a Variable Ocean, *IEEE J. Oceanic Eng.*, **29** (2004), pp. 418–429.
- Lawrance, N. R. and Sukkarieh, S.: Path Planning for Autonomous Soaring Flight in Dynamic Wind Fields, IEEE International Conference on Robotics and Automation, 2011.
- Otten, N. D., Jones, H. L., Wettergreen, D. S., and Whittaker, W. L.:

- Planning Routes of Continuous Illumination and Traversable Slope Using Connected Component Analysis, IEEE International Conference on Robotics and Automation, 2015.
- 14) Speyerer, E. J., Lawrence, S. J., Stopar, J. D., Gläser, P., Robinson, M. S., and Jolliff, B. L.: Optimized Traverse Planning for Future Polar Prospectors Based on Lunar Topography, *Icarus*, **273** (2016), pp. 337–345.
  - 15) Cunningham, C., Amato, J., Jones, H. L., and Whittaker, W. L.: Accelerating Energy-aware Spatiotemporal Path Planning for the Lunar Poles, IEEE International Conference on Robotics and Automation, 2017.
  - 16) Bai, J. H. and Oh, Y. J.: Global Path Planning of Lunar Rover under Static and Dynamic Constraints, *Int. J. Aeronaut. Space Sci.*, **21** (2020), pp. 1105–1113.
  - 17) Blackmore, L., Li, H., and Williams, B.: A Probabilistic Approach to Optimal Robust Path Planning with Obstacles, American Control Conference, 2006.
  - 18) Blackmore, L., Ono, M., Bektassov, A., and Williams, B. C.: A Probabilistic Particle-control Approximation of Chance-constrained Stochastic Predictive Control, *IEEE Trans. Robotics*, **26** (2010), pp. 502–517.
  - 19) Aoude, G. S., Luders, B. D., Joseph, J. M., Roy, N., and How, J. P.: Probabilistically Safe Motion Planning to Avoid Dynamic Obstacles with Uncertain Motion Patterns, *Autonomous Robots*, **35** (2013), pp. 51–76.
  - 20) Mahmoud Zadeh, S., Powers, D. M. W., Yazdani, A., Sammut, K., and Atyabi, A.: Differential Evolution for Efficient AUV Path Planning in Time Variant Uncertain Underwater Environment, *arXiv*, (2016).
  - 21) Inoue, H., Otake, H., Otake, M., Yamamoto, M., Hoshino, T., Shimada, T., et al.: Landing Site Analysis for Future Lunar Polar Exploration Missions, *Icarus*, **211** (2018), pp. 1066–1081.
  - 22) Mazarico, E., Neumann, G. A., Smith, D. E., Zuber, M. T., and Torrence, M. H.: Illumination Conditions of the Lunar Polar Regions Using LOLA Topography, *Icarus*, **211** (2011), pp. 1066–1081.
  - 23) Gläser, P., Scholten, F., De Rosa, D., Figuera, R. M., Oberst, J., Mazarico, E., et al.: Illumination Conditions at the Lunar South Pole Using High Resolution Digital Terrain Models from LOLA, *Icarus*, **243** (2014), pp. 78–90.
  - 24) Gläser, P., Oberst, J., Neumann, G. A., Mazarico, E., Speyerer, E. J., and Robinson, M. S.: Illumination Conditions at the Lunar Poles: Implications for Future Exploration, *Planet. Space Sci.*, **162** (2018), pp. 170–178.
  - 25) Acton, C. H., Jr.: Ancillary Data Services of NASA's Navigation and Ancillary Information Facility, *Planet. Space Sci.*, **44** (1996), pp. 65–70.
  - 26) Hart, P. E., Nilsson, N. J., and Raphael, B.: A Formal Basis for the Heuristic Determination of Minimum Cost Paths, *IEEE Trans. Systems Sci. Cybernetics*, **4** (1968), pp. 100–107.
  - 27) LaValle, S. M.: Rapidly-exploring Random Trees: A New Tool for Path Planning, (1998).
  - 28) Karaman, S. and Frazzoli, E.: Sampling-based Algorithms for Optimal Motion Planning, *Int. J. Robotics Res.*, **30** (2011), pp. 846–894.

Zheng Hong (George) Zhu  
Associate Editor