Thermal based Path Planning using Solar Orientation for a Lunar Micro Rover

By Hugo ZULIANI,[1)] Takuto OIKAWA,[1)] and Kazuya YOSHIDA[1)]

[1)]Department of Aerospace Engineering, Tohoku University, Japan, Japan

(Received June 21st, 2017)

In the context of the Google Lunar XPRIZE, team HAKUTO is developing a four wheel rover to explore the lunar surface. In order to prevent the electronic components from reaching dangerous temperatures, it is crucial to control the rovers exposition to solar radiations. A path planning algorithm able to take into account the rovers thermal behavior would help the pilot to take safe long-term decisions. A simple thermal model of the rover in the lunar environment was derived, based on the exposition of its different faces to the Sun and the potential apparition of shaded regions on the way. This model was incorporated in a path planning algorithm built around the particularities of the mission, and this algorithm was tested in different situations and with variations on the definition of an optimal path. This paper also presents a version of the algorithm taking into account the possibility of random slippage during the run.

Key Words: Path Planning, Thermal Control, Lunar Rover

1. Introduction

Unmanned robotic systems like rovers are an asset to planetary exploration as they represent a safer more cost efficient option than manned missions. However, companies wishing to enter the field often cannot afford heavy payloads to be launched from Earth, and minimizing the weight and size of a rover is a important task. Micro rovers generally need to sacrifice complex thermal systems that can be found in heavier planetary rovers to avoid exceeding their limited weight budget, and are therefore more likely to suffer from the extreme thermal conditions found outside of earth. The possibility to control a rover's thermal behavior without a complex heating or cooling system is thus crucial for its survival. This paper will present a path planning algorithm based on the thermal behavior of the pre-flight model of the rover from team HAKUTO,[1)] a participant to the Google Lunar XPRIZE.

Path planning for lunar rovers has already been explored in a few different perspectives, like terra-mecanics[7,8)] or energy generation from solar panels.[3)] However the field of path planning for vehicules in general is very wide and offers a great variety of options, like a genetic algorithm for robotic motion,[9)] a probabilistic algorithm for traffic simulation,[10)] or Dijkstra's algorithm applied to the control of a UAV.[11)]

This paper focuses on path planning guided by thermal simulations. It presents a model for those simulations, an algorithm designed to incorporate them in its decision process, and offers examples of the paths suggested by this algorithm. It is meant as an demonstration of the feasibility of such an algorithm, as well as a potential example on the design of a simple path planning algorithm based on a mathematical model.

2. Thermal Model of the rover

Figure 1 shows the pre-flight model of the HAKUTO rover and the different causes of temperature changes for

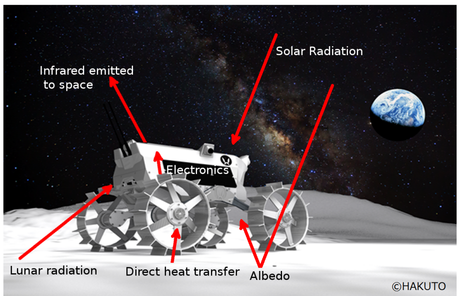


Fig. 1. Representation of the thermal exchanges between the rover and its environment.

our model.[6)] To account for the non uniformity of temperature in the different parts of the rover, it is decomposed into six parts, representing its different faces, as well as the components attached to them. Most of the avionics is considered to be part of the top (battery, most electronic boards) and the front (time of flight camera) of the rover, and the left and right sides are covered by solar panels. These electronic components are what define the temperature range each part can survive in, and therefore most of the focus will be put on the front and top of the rover. Table 1 displays the operational temperature range of the different hardware components.

At each time iteration of the algorithm presented in section 3 , each part of the rover undergoes a temperature difference proportional to the total amount of heat received during that time. This value for part "i" is calculated by taking the sum of the following components :

The solar radiations received $Q_{S,i}$:

$$Q_{S,i} = \alpha_i A_i P_S F_{S,i} \tag{1}$$

The radiations received from the lunar surface $Q_{L,i}$:

$$Q_{L,i} = \epsilon_i \epsilon_L A_i \sigma F_{L,i}(T_L^4 - T_i^4) \tag{2}$$

The radiations emitted back to space from the rover $Q_{A,i}$:

$$Q_{A,i} = \alpha_i A_i P_S F_{A,i} a \tag{3}$$

Table 1. Hardware Operational Temperature Range.

Hardware	Temperature (deg C)	
	Minimum	Maximum
Motor	-20	100
Camera	0	45
Time of Flight Camera	-40	85
Ethernet Switch	-40	85
Motor Controller	-20	70
Power Interface Board	-40	125
Cube-Sat PDU	-40	85
Battery in Discharge	-20	60
Battery in Charge	0	45
IMU	-40	100
Micro-controller	-40	90
Arm based CPU	-20	70

The radiations from the Sun reflected back to the rover, or solar albedo[2] $Q_{C,i,j}$:

$$Q_{C,i,j} = \alpha_i A_i P_S F_{A,i} a \tag{4}$$

The heat conduction from part i to part j $Q_{C,i,j}$:

$$Q_{C,i,j} = c_{i,j}(T_j - T_j) \tag{5}$$

The heat generated by the electronic components $Q_{E,i}$.

Nomenclature

α	:	Absorptivity
A	:	Area
P_S	:	Solar constant
F	:	View factor between two heat sources
ϵ	:	Emissivity
σ	:	Steffan-Boltzmann constant
T	:	Temperature
a	:	Albedo factor of the lunar surface
c	:	Heat transfer coefficient

Subscripts

i, j	:	Part i or j
L	:	Lunar surface
S	:	Sun
Sp	:	Outer space
A	:	Albedo

The heat equation for each part of the rover can be seen as a first order differential equation. Most terms are constant but two sets of variable have to be considered : the view factors of the faces to the Sun, and their temperature. This implies that the position of the Sun in the lunar sky, the orientation of the rover and its thermal conditions at the previous position are required to calculate those temperatures after an incremental movement.

3. Path Planning Algorithm

3.1. Choice of the Algorithm

Choosing amongst the wide variety of existing path planning algorithms requires to understand the specificities of the problem at hand.

Firstly the goal of this algorithm is to evaluate the thermal behavior of the rover during a theoretical run to assess its viability. Therefore, it is required to take parameters like time, position and temperature of the rover as factors in this evaluation. This criterion excludes the use of methods like the Bug Algorithm or Cell Decomposition. Furthermore, the previously mentioned factors will vary after each movement, rendering a Potential Field based method ineffective.

Thermal control is a critical issue for the survival of the rover and thus it is preferable to obtain for a given problem the exact best solution available. However the algorithm is designed to help the pilot make decisions over a long period of time, and therefore does not require to favor calculation speed over the exactitude of the solution. Sampling based methods should therefore be avoided.

The method chosen will therefore be Dijkstra's algorithm, for its exactitude and its use of a customizable cost function.[5]

3.2. Thermal Path Planning

The algorithm uses an altitude map of the region to traverse, with the knowledge of the current position of the rover and its objective. The altitude and direction of the Sun, based on the time of the lunar day are also calculated and updated after each movement. The temperature of the rover at the beginning of the run is assumed to be in the equilibrium state corresponding to its initial situation.

The altitude map and the rover's movement offer the knowledge of its angular position at each point in time. From this, the view factors between its faces and the Sun, as well as the albedo, are calculated. Then, knowing the previous thermal state of the rover, temperature differentials are evaluated on a incremental time step corresponding to an estimation of the time the rover would need to clear the distance between to points. The rover's speed is considered constant, although regular poses may be required during the mission, as the calculations consider long distances and times, making irregularities in the rover's speed negligible.

Due to the possible existence of obstacles like rocks, craters and hills on the trajectory, some regions of the map might be shaded, particularly during the lunar morning. As the position of the Sun changes throughout the run, the location of these shadows must be updated regularly. Shade needs to be taken into account during the calculations as solar radiations are a crucial source of heat for the rover.

The steps to calculate the temperature differentials for any theoretical movement is therefore :

- Update the time, position of the Sun and temperature of the lunar surface.
- Locate the shaded regions.
- Determine the view factors between the rover and the Sun as well as the solar albedo.
- Calculate the sum of the heat intake in each section of the rover.

From there the new temperatures of the rover are used to assess the viability of the movement through the cost function.

3.3. Cost Function

The cost function is in Dijkstra's method what evaluates if a route is optimal, and its definition is crucial in the design of the algorithm. To ensure the safety of the rover, the function should penalize route leading to potentially dangerous situations. In this perspective, two strategies are proposed :

- The Stable Temperature Strategy : The cost function penalizes sudden changes in the rover's temperatures, mostly at its front and top. With this strategy, the algorithm will tend to suggest routes where the rover avoids sudden heating or cooling. The cost function is defined as follows :

$$C = d(1 + \sum_i A_i \Delta T_i) \tag{6}$$

- The Optimal Temperature Strategy : A temperature is defined as a goal, and the cost function penalizes a movement that leads towards a different thermal state. The algorithm will therefore suggest routes where the rover mostly stays near this temperature. The cost function is defined as follows :

$$C = d(1 + \sum_i B_i (T_i - T_0)^2) \tag{7}$$

C	:	Cost function
d	:	Distance
$AandB$	:	Wighing coefficients
T_i	:	Temperature of face i
T_0	:	Optimal temperature

Both strategies present positive and negative aspects, which will be presented in case studies in the following section.

4. Simulation Results

In order to validate the design of the algorithm and understand both strategies suggested previously, simulations of the algorithm in theoretical situations were conducted. In all cases, the focus will be put on the path obtained as well as the temperatures calculated by the algorithm for this path.

4.1. Cases Studied

The first mission of the HAKUTO rover is scheduled to happen between around two earth days after the lunar sunrise, or lunar morning, and lunar noon. The location of the landing should be situated at a latitude of about 45°. All cases studied will correspond to two extreme situations : the beginning of the mission, where the Sun is low in the sky and the lunar surface temperature is at around -150°C; and lunar noon, where the Sun is at its zenith and the lunar surface temperature reaches 80°C.

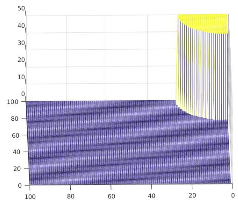
The simulations are performed in three types of environment :

- A flat map with only one large obstacle casting a shadow between the rover and its objective. This simulation is run during lunar noon and is designed to verify the ability of the algorithm to avoid shade that may create a dangerous drop in its temperature.

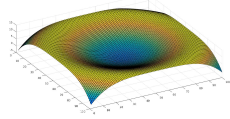
- A crater like map where the rover and its objective are at higher altitude than most of the terrain. This simulation is also run during lunar noon and therefore almost no shadow appears. This case is used to verify the algorithm's ability to provide an efficient route when confronted to a simple problem.

- A randomly generated map where the rover has to go through a series of hills and valleys to reach its objective. This simulation is run during lunar morning, and the hills create shade on large portions of the terrain, creating a particularly dangerous environment. This case tests the algorithm's ability to generate a route as safe as possible in a complex situation where the rover's survival is highly at risk.

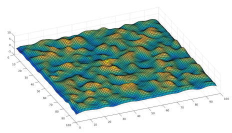
Figure 2 presents the altitude maps of the different terrains while Fig. 3 displays the shade appearing on the flat at the beginning of the run.



(a) Altitude map of the flat terrain with a single obstacle.



(b) Altitude map of the crater like terrain.



(c) Altitude map of the randomly generated terrain

Fig. 2. Altitude maps of the different terrains used for simulations.

4.2. Simulation Results

4.2.1. Flat Shaded Terrain in Lunar Noon

The temperatures of the front and top of the rover when going through a straight line are displayed in Fig. 4. The top appears to drop under -20°C, a situation dangerous for the survival of the battery. Avoiding the shadow appears to create this temperature drop.

The temperatures obtained when using both strategies previously introduced, as presented in Fig. 5 are maintained between 0°C and 30°C. With those routes, the rover stays entirely safe, even with the battery charging. These simulations proved the algorithm's ability to detect and avoid a potentially dangerous situation.

4.2.2. Crater Like Terrain in Lunar Noon

In this situation, going through a straight line from start to finish is a perfectly safe solution, as shown in Fig.6, and therefore appears as the most efficient route to take.

Both strategies suggest paths keeping the rover in safe temperatures. However, the stable temperature strategy opts for a straight line while the optimal temperature strategy offers a complex trajectory that extends significantly

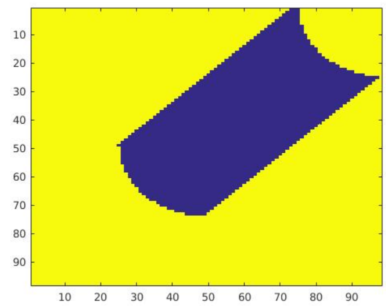


Fig. 3. Lit (yellow) and shaded (blue) regions on the flat terrain.

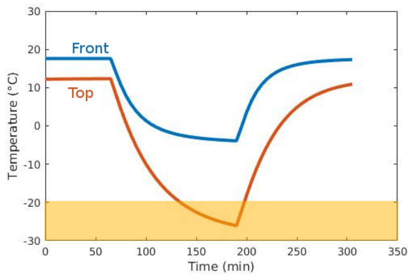
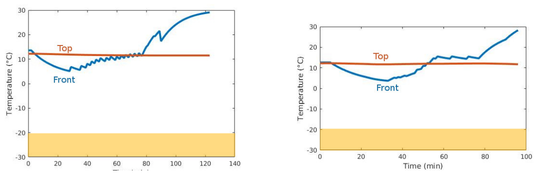


Fig. 4. Temperatures of the rover going through a straight line on the flat terrain.



(a) Temperatures obtained by the stable temperature strategy.
(b) Temperatures obtained by the optimal temperature strategy.

Fig. 5. Temperatures of the top and front of the rover on the flat terrain using the two strategies proposed.

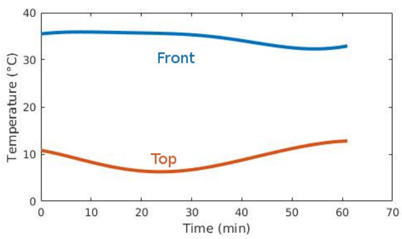


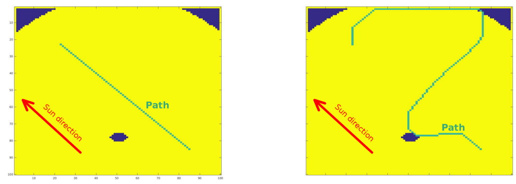
Fig. 6. Temperatures of the rover going through a straight line on the crater like terrain.

the distance to run, as shown in Fig.7.

This case points out a flaw with the optimal temperature strategy. It will tend to favor more complex solutions if they allow the rover's temperature to approach the optimal value, even when this precaution is not justified.

4.2.3. Random Terrain in Lunar Morning

In this situation, the rover is at high risk from the original position as the top part is under -20°C, a critical situation for the battery. Furthermore, the low Sun altitude creates large unavoidable shaded regions that threaten to lower the rover's temperatures lower than it is. The path planning algorithm will here have to provide a path that



(a) Temperatures obtained by the stable temperature strategy.
(b) Temperatures obtained by the optimal temperature strategy.

Fig. 7. Temperatures of the top and front of the rover on the flat terrain using the two strategies proposed.

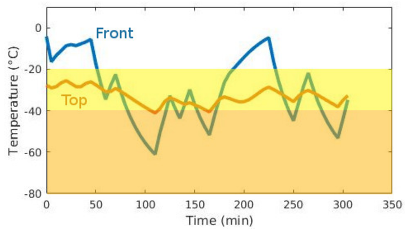
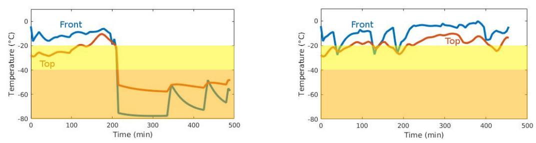


Fig. 8. Temperatures of the rover going through a straight line on the randomly generated terrain.

maximizes the heat intake in order to keep the rover safe as much as possible. As seen in Fig.8, a simple straight line maintains both the top and front of the rover mostly under -20°C, sometimes reaching below -40°C.

In the route given by the stable temperature strategy, the rover first struggles to maintain itself above -20°C, and the temperatures then suddenly plummet below -40°C and never comes back to a safe state, as seen in Fig.9(a). The reason for this problematic decision is the following : due to the nature of the terrain and the low Sun altitude, the rover will inevitably during its course enter a shaded region. When it does, the temperature suddenly drops. However, once the temperatures reach very low values, the stable temperature strategy will encourage the rover to remain in this state and therefore continue in the shadows as much as possible.



(a) Temperatures obtained by the stable temperature strategy.
(b) Temperatures obtained by the optimal temperature strategy.

Fig. 9. Temperatures of the top and front of the rover on the randomly generated terrain using the two strategies proposed.

For the case of the optimal temperature strategy, the path suggested mostly manages to keep the rover in a safe state, as shown in Fig9(b). by minimizing the passage in shaded regions.

4.3. Conclusion on the Strategies

The simulations presented in this section proved that both definitions suggested for the cost function work efficiently in some cases but can encounter some difficulties.

The stable temperature method works properly in simple problems and offer efficient solutions, but because of its definition it can also maintain the rover in a dangerous position when the situation becomes more complex.

The optimal temperature strategy is able to ensure the rover's safety in more dangerous situation, but can sacrifice efficiency for unnecessary security.

5. Probabilistic Path Planning

5.1. Justification

In the algorithm presented thus far, the assumption that the rover would be able to follow exactly the path designed was made. However, with the possibility of slippage on the lunar terrain and potential errors in the self-localization of the rover, this assumption may not be realistic and could put the rover into harm.

Probabilistic path planning aims to take into account the possibility of errors in the rover's trajectory and avoid situations where they might me dangerous.

5.2. Implementation

If we define A the starting point of a movement and u the command for the rover to move from A to one direction, the rover may arrive in different possible positions B_i with a possibility p_i for each.[4] In this algorithm we define the cost function at the point A $C(A)$ to be a function of the costs in the positions B_i, $C(B_i)$, and the cost of the movements from A to B_i, $C(A, B_i)$, as well as the probabilities p_i :

$$C(A) = \sum_i p_i(C(B_i) + C(A, B_i)) \tag{8}$$

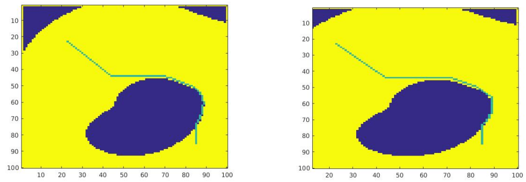
Here,contrary to the common Dijkstra method, the cost function at a point is calculated from the next points in the trajectory. Therefore, the calculations are made in the opposite order, and the temperatures of the rover cannot be anticipated when assessing the cost of a movement, but instead need to be assumed. In the rest of this section, the temperature at each point will be considered as the the equilibrium temperature in this situation, and the cost function for the movement from A to B_i will be defined, imitating the optimal temperature strategy, as follows :

$$C(A, B_i) = d_i(1 + \sum_j D_j(T_{i,j} - T_0)^2) \tag{9}$$

$T_{i,j}$ is the temperature of the part j of the rover in A in order for this part to reach its equilibrium temperature in B_i.

5.3. Simulation results

Simulations were run with this algorithm, in the conditions of the lunar morning and in a crater like terrain presented in the previous section. In this case, the low Sun altitude creates a large shadow on the way. The rover had a possibility of slipping to the left or right during a movement. For several values of slip ratio, two paths were generated, one not considering the possibility of slippage, presented in Fig. 10(a) and one taking it into account presented in Fig. 10(b) for a 2.5% slippage probability.



(a) Path obtained with a non probabilistic approach. (b) Path obtained with a probabilistic approach.

Fig. 10. Comparison of the paths obtained with or without a probabilistic approach to thermal path planning.

For each path generated and each value of the slippage probability, one thousand theoretical were launched. Each time, the minimal temperature reached by the front of the rover was saved and an average minimal temperature was calculated per path and slippage probability. Figure 11 displays those values.

Although high slippage probability proves to be a great source of hazard for the rover even when taking it into account when determining the path to take, a probabilistic approach to thermal path planning reduces significantly this risk and suggest safer routes at low slippage. Indeed, the front of the rover remains over -40°C in average with twice as much slippage with this approach than with the traditional deterministic algorithm.

6. Conclusion

The lunar environment is dangerous for a small rover for many reasons, and the wide temperature differentials can cause irreversible damage to its hardware components. However, a model of the thermal behavior of such a rover can be derived fairly accurately due to the absence of unpredictable factors such as those caused by Earth's atmosphere. Such a model was used here to find temperature-wise safe paths in an environment solely from the Sun's position in the sky and the geometry of the ground.

This algorithm can be used to guide any type of rover that can be considered as a set of plane surfaces. It is highly customizable and can also then be accommodated to include more path planning criteria like slope or obstacle avoidance and slippage control.

Two strategies for thermal path planning were suggested, and each was introduced with its own merits and flaws. The choice of which approach to chose or how to combine them is left to the designer of the algorithm or the pilot, depending on the situation.

To improve the rover's safety by taking the possibility of errors in its movement, a probabilistic approach to thermal path planning as introduced. Although it did not prevent entirely dangerous situations to occur because of random movement errors, it was able to limit the damages to a significant extent with just a simple model of slippage.

This algorithm is not designed for an entirely automated rover, but rather as a long term decision making tool for

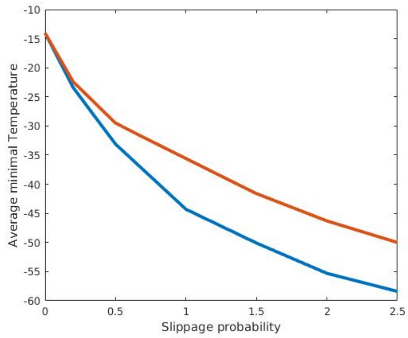


Fig. 11. Average minimal temperature of the front of the rover with or without a probabilistic approach.

the pilot to avoid perilous situations, which are hard to predict for a human due to the wide differences between the Earth and the Moon's environment, particularly in a thermal perspective.

References

1) Yoshida, K., Britton, N. and Walker, J.: Development and field testing of moonraker: a four-wheel rover in minimal design, "ICRA13 Planetary Rovers Workshop, 2013.
2) Williams, D.R.: Moon Fact Sheet, http://nssdc.gsfc.nasa.gov/planetary/factsheet/moonfact.html (accessed December 15, 2016).
3) Wettergreen, D., Dias, B., Shamah, B., Teza, J., Tompkins, P., Urmson, C.,Wagner, M. and Whittaker, W. : First Experiment in Sun-synchronous Exploration, in "Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conferenceon," volume 4 ,pp.3501–3507, IEEE, 2002.
4) Thrun, S., Burgard, W. and Fox, D. : Probabilistic Robotics, The MIT Press, 2005.
5) Siciliano, B. and Kathib, O. : Springer Handbook of Robotics, Springer Science and Business Media, 2008.
6) Oikawa, T., Walker, J. and Yoshida, K. : Experimental Evaluation of Thermal Simulation Model for Lunar Exploration Rover, in "System Integration (SII), 2015 IEEE/SICE International Symposium on," pp. 882–887, IEEE, 2015.
7) Kubota, T., Kuroda, Y., Kunii, Y., and Yoshimitsu, T. : Path planning for newly developed microrover, in "Proceedings of the 2001 IEEE International Conference on Robotics and Automation", 2001.
8) Ishigami, G., Nagatani, K. and Yoshida, K. : Path planning for planetary exploration rovers and its evaluation based on wheel slip dynamics, in "Robotics and Automation, 2007 IEEE International Conference on Robotics and Automation," pp. 2361–2366, IEEE, 2007.
9) Hu, Y. and Yang, S.X. : "A knowledge based genetic algorithm for path planning of a mobile robot in" Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on Robotics and Automation, "volume 5, pp.4350–4355, IEEE, 2004.
10) Galán-García, J., Aguilera-Venegas, G., Galán-García, M. and Rodríguez-Cielos, P. : A new Probabilistic Extension of Dijkstra's Algorithm to simulate more realistic traffic flow in a smart city in Applied Mathematics and Computation, volume 267: p.780–789, 2015
11) Economou, J.T., Kladis, G., Tsourdos, A. and White, B.A. : UAV optimum energy assignment using Dijkstra's Algorithm in "Control Conference (ECC), 2007 European" pp.287–292, IEEE, 2007.