A POTENTIAL FIELD METHOD FOR AUTONOMOUS LUNAR ROVER NAVIGATION IN 3D TERRAIN

Parth Nanadikar[1], Rahul Shome[2] and Ashish Dutta[1]

[1] Department of Mechanical engineering, Indian Institute of Technology Kanpur, Kanpur 208016, India. (e-mail: parthnan/adutta@iitk.ac.in)

[2] Department of Computer Science and Engineering, National Institute of Technology Durgapur, West Bengal 713209, India.

ABSTRACT

The development of planetary rovers is essential for the success of planetary missions. This paper discusses an algorithm based on the potential field method for navigation of a rover in an unknown 3D terrain containing obstacles. A 3D map of the terrain is generated using a structured light system, and the terrain is then divided into square grids having gradients. Assigning positive and negative gradients to the goal, obstacles, rover and grids we generate a potential field function. Using this function the rover finds the best path to reach a goal point. Unlike potential field functions in 2D this method works in 3D and also considers the rover kinematics.

Keywords: lunar rover, 3D path planning, potential field.

1. INTRODUCTION

In the last decade, with the launch of new planetary missions there are further developments being made in planetary rovers. The developments of new mechanical designs and software algorithms are aimed at enhancing the exploration capabilities of planetary rovers. The New generation planetary rovers make possible the performance of more tedious scientific experiments in situ on other planets. The rovers with rocker bogie mechanism are tried and tested designs, their success has been established by several Mars missions. This paper deals with development of path planning algorithm for the lunar rover, taking into consideration the kinematic constrains on lunar rover due to its mechanical design and 3D nature of ground profile of the terrain.

The surface of the moon has boulders acting as obstacles to the rover and the regolith with certain gradients which make maneuvering of the rover susceptible to obstruction or the over tipping. Hence there is a need to develop a path planning algorithm which will take care of kinematic constraints of the moon rover. The moon rover obtains the 3D local map from structured light based laser source system. The 3D map is converted into a 2D grid value map where each grid has a gradient and/or obstacles, based on which the path is generated. In the simulation developed the 2D grid value map with obstacles is manually entered to the simulator. Earlier potential field approaches considered flat terrains in 2D with obstacles for path planning. We propose a newer method in which the gradient information of the terrain as well as the kinematic constraints of the rover are also incorporated to find the best path.

The potential field methods are widely studied and implemented methods in robot path planning. Since O. Khatib [1] suggested the theory of potential field first time in 1986. Khatib proposed a method of real time obstacle avoidance approach for manipulators and mobile robots based on the artificial potential field concept. Much work has been done in the field since then. McFetridge and Ibrahim[2] have summarized the old potential field approach and the new hybrid adaptive Fuzzy – Potential field approach.Volpe and Khosle [3] suggested the potential function based on superquadrics which closely models a large class of object shapes. Shrikant Parakh et al. [4] discuss the motion of the rover on uneven terrain that can take into account the loss of contact of wheel with the ground. Their approach can be incorporated in the algorithm for field trials. Takeshi Ohki et al. [5,6] have demonstrated, in simulations and in actual field trials, the path planning method for autonomous robots considering the instability of attitude maneuvers on rough terrains. They generated more than one path using general graph search method. This paper also tries to address the same concern of instability of the mobile robot during attitude maneuvers, but the path is generated through potential field methods. But most of them discuss about methods of overcoming the local minima's once the robot is trapped in it. These methods put displacement criteria to identify the local minima problem. In these methods the rover is said to be trapped in a local minima if at any iteration its displacement is below a

certain critical value. We propose a method where we identify the local minima in advance and try to circumvent it without getting trapped in it. But most of these methods are for path generation in 2D. This paper describes the way to generate a path in 3D. The proposed method is discussed in section 2 in detail. The equations and the values of constants selected are shown in this section. A flowchart of the simulation is shown in this section.

Section 3 discuses implementation and evaluation of the proposed path planning algorithm. The assumptions made for the purpose of simulation are discussed in this section. Section 4 discuses the results of the simulation and the path extracted are shown for each result. Section 5 discusses the conclusion.

2. PROPOSED METHOD

In this section path generation is described using the potential field methods with additional algorithms described in subsequent paragraphs. It is assumed that the rover is inscribed in a circle within a radius of 391 mm and this circle is called rover circle (figure.1). Circumference of rover circle and the surface of obstacles have like electrical charges. Hence they repel each other with force which is inversely proportional to the square of the shortest distance between them. The rover circle and the target have unlike charges hence they attract each other. The gradient cells in vicinity of the rover exerts a repulsive force depending upon their gradient. Each grid is of size 20x20mm. In each iteration the vector summation of all forces acting on the rover is calculated. We treat the rover path planning as problem as one in which the rover moves very slowly so only rover velocity is considered during simulation.

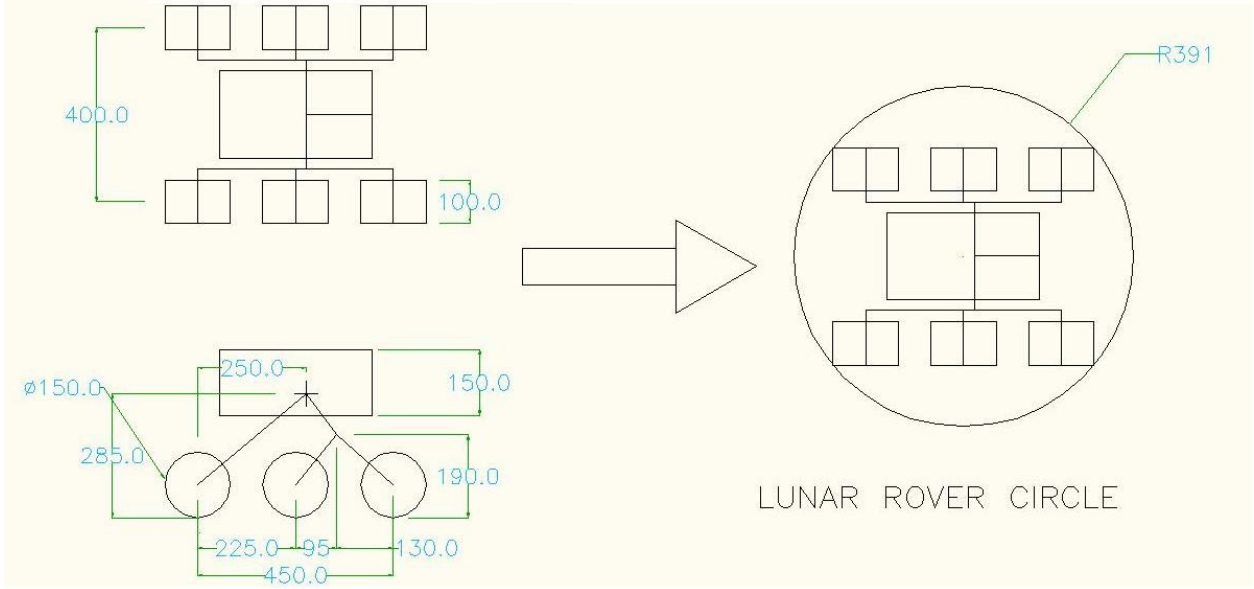


Fig. 1: The basic dimensions of the rover and the rover circle.

In the simulation there is a virtual positive electric charge on the rover of 10 units. The target is given a virtual negative electric charge of -1000 unit. The point on the obstacle surface, which is closest to the rover, is given a virtual positive charge of +1 unit. The value of the electrostatic constant is taken as 25. The charges on the rover and the target were so chosen that the velocity of the rover at the starting point is not less than 19.5 mm/ sec. The charge on the obstacle is taken as $\frac{1}{10}$ of the charge on the rover for our case.

The force acting between two electrical charges in the potential field method is given by :

$$F = \frac{k * k_1 * k_2}{d^2}$$

Where,

K is the electrostatic constant
k_1 is the first charge
k_2 is the second charge
d is the distance between the two charges.
Hence the virtual electrostatic attractive force acting in between the rover and the target is given by

$$\text{Fatt} = \frac{25*10*1000}{d^2}$$

And the virtual electrostatic repulsive force acting between the rover and the obstacles is given by

$$\text{Frep} = \frac{25*10*1}{d^2}$$

A particular trap situation is shown in figure 2. We neglect the ground profile for this example. The target is exactly behind the Obstacle and the obstacle surface facing the rover circle is exactly perpendicular to the line joining the rover circle centre and the target. Figure 2(b) shows the direction of net resultant force acting at each point on the map, if rover is placed at these points. The small dot like circles indicate the centre of rover circle and the normalized line originating from these centres indicate the direction of resultant force acting at those points. The figure 2(b) shows the trapped robot as a result at the local minima, as the attractive and repulsive forces are in line. Here the obstacle surface 'ab' which causes the trap situation is seen as a line view in top view. The surface has two end vertices, 'a' and 'b'. The algorithm checks which of these two end vertices are closer to the rover at every iteration. The vertex which is closer (in the case of figure 2(c) it is a) is joined to the centre of rover circle by a straight line. Then the reflection of the force of attraction is taken about this line. When the reflected force of attraction is used to calculate the resultant force it guides the rover away from the local minima. The figure 2(c) shows the method of reflection of forces applied to overcome such trap situation. In figure 2(d) the rover is able to avoid the local minima without getting trapped into it.

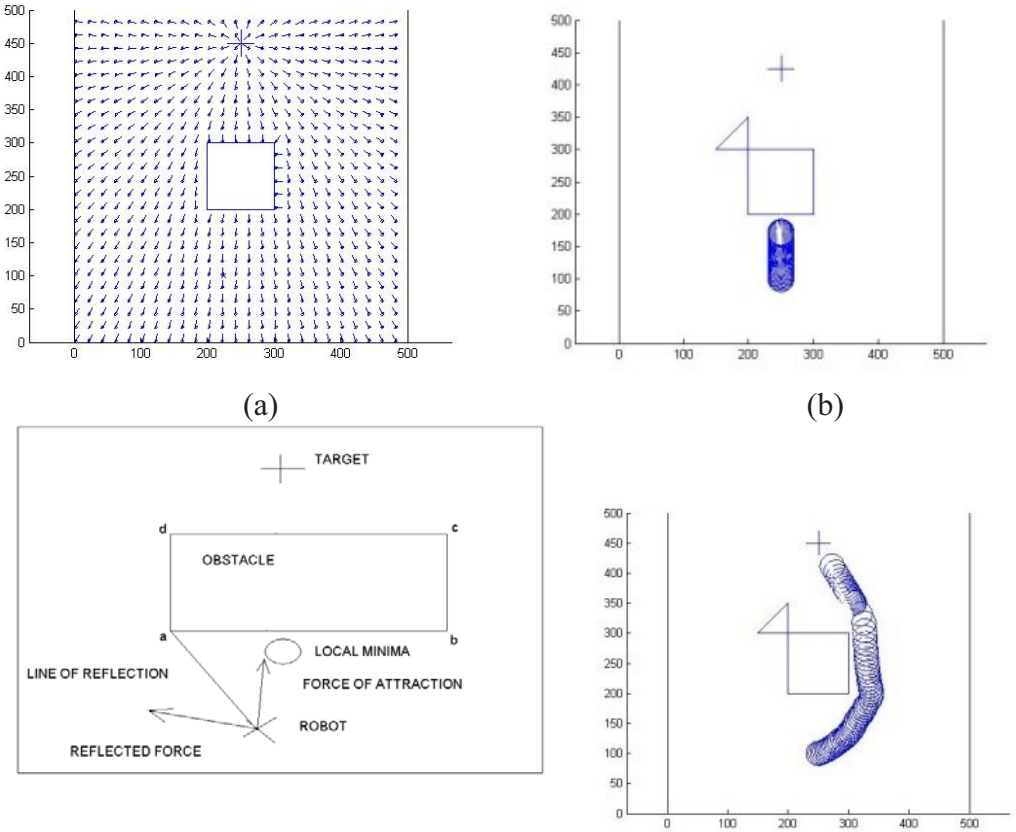


Fig. 2(a-d) . (a) The direction of net resultant force acting at each point in the map.
(b) As a result, the robot is trapped at the local minima (c) The method of reflection of forces implied to overcome such trap situation (d) The final path.

The rover should be guarded from over tipping. To achieve this, it is desirable that the path generated passes as far as possible through the cells which have lower radial gradients. In the figure. 3 it is shown how the rover will look at two different gradients.

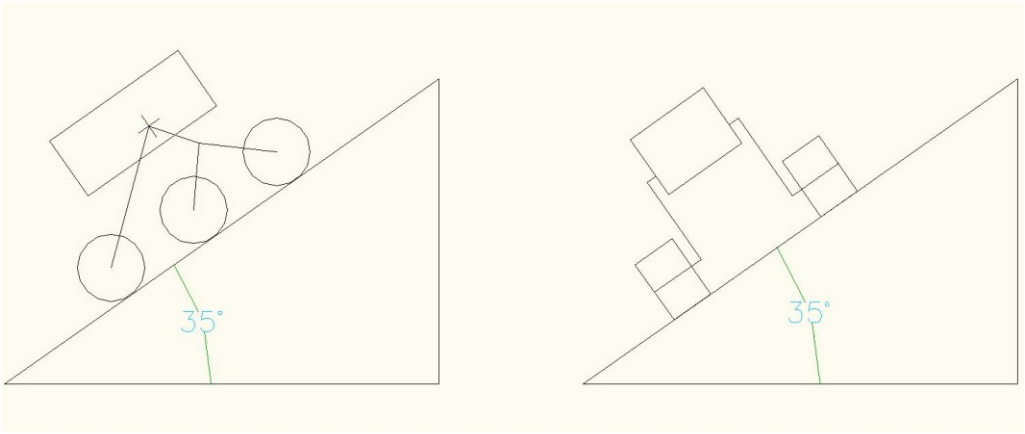


Fig 3: Maximum longitudinal and lateral slopes allowed in each grid considering stability

The map is now divided into cells. The resolution is decided by the laser scanning capacity of the rover. The structured light used is capable of generating a grid of 2 square cm up to 2m distance. The algorithm has adaptability to different resolutions, but higher resolution means higher number of cells hence more computational time. Each cell has gradient value in radial outward direction from the rover. A cell exerts the repulsive force on the rover circle from its centre which is given by,

$F_{rep\text{-}cell} = K_{cg} * K_{ad} * F^{*}_{att}$ (1)

Where,

$F_{rep\text{-}cell}$ is repulsive force due to a single grid cell.

K_{cg} is cell gradient factor for the particular cell

K_{ad} is the angle distance factor for the particular cell

F^{*}_{att} is the maximum force of repulsion acting on the rover from the centre of the gradient cell. This factor is proportional to, F_{att} by relation :

$$F^{*}_{att} = k' * F_{att} \; ; \; 0 < k' < 1 \qquad (2)$$

F_{att} is force of attraction by target acting on The rover if it were placed at the centre of the cell

The different factors are calculated as follows,

$$K_{cg} = \frac{\text{gradient /}}{\tan(\text{max slope allowed in degree})};$$

in case of the rover maximum allowed slope is taken 35°

hence K_{cg} becomes,

$$K_{cg} = \frac{\text{gradient}}{0.7002} \quad (3)$$

$K_{ad} = 0.75$ if 20°=< cellcentreangle < 138° and if 150 mm < cellcentrerobodistance <150 mm + cellwidth

Where, cell centre angle is the angle made by the line joining the centre of a cell and the centre of rover circle with the direction of advancement of the rover. The relative position of gradient cells which have non zero value of K_{ad} is shown in figure. (4). The cell which have their centres inside the hatched area have non zero value of K_{ad} . Roborad is the radius of the rover circle and cell width is the width of a cell.

The net force acting on the rover due to the goal, obstacle, rover and grid gradient at any iteration is summed by equation :

$$F_{total} = F_{att} + F_{rep} + \sum_{grid} F_{rep-cell} \quad (4)$$

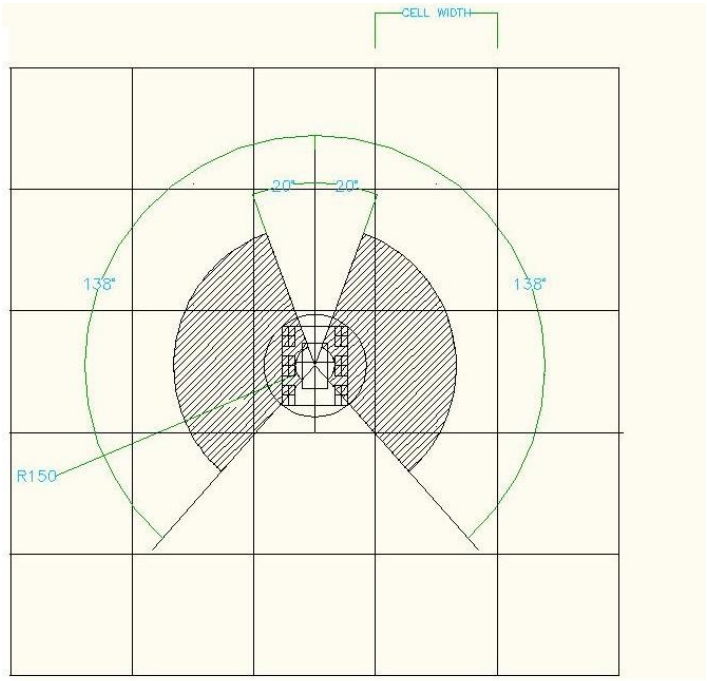


Fig. 4: Area around the rover considered for the calculation of K_{ad}

The above algorithms are combined and implemented in a Matlab based simulation. The constraints and variable values are determined based on kinematic design of the rover as follows

Steering angle or turn angle: The rover cannot turn at any step more than 0.34906 radian to either left or right. The path planning algorithm has to take into account the turning limitation on the rover.

Speed: The rover is not designed to move with higher speeds and the speed limit put on the rover is 100 mm/sec.

Over tipping: Rolling and yawing angle is 0.61086 radian, while the cell with ground slopes above this value is treated as an obstacle.

3. IMPLEMENTATION AND EVALUATION

a. Conversion of rovers 3D environment into grids for path planning

First, before path planning the 3D map of the rover environment is developed by a structured light system mounted on the rover and converted into gradient grids. A line laser mounted on the rover is rotated at intervals of 1.8^o. At each instant a camera captures the laser profile and at the end of the scan combines all the laser profiles to generate a 3D map. The laser calibration is done with respect to a flat ground and vertical gradient. The deflections from such a reference laser line of the actual perceived straight line gives the measure of the height or depression of the surface at that point. 100 such readings at angular steps of 1.80^o are used to generate a contour estimation of the 180^o field of view in front of rover. Every reading consists of 480 discrete point data along the sensed laser line. Thereby, we obtain the height data about 48000 discretely identifiable points in front of the rover. The distribution of these 48000 points depends upon the radial distance from the rover as well as the nature of the surface. The nature of the surface also determines the regions the points represent. Since the points pertain to the points on the laser line, and deflection of the laser line necessitates the absence of the data at regions where the laser line cannot go, e.g. the shadows behind obstructions to the laser line, etc. Hence the output of the structured light analysis is a set of 48000 discrete, scattered data points with corresponding estimated heights. An experiment with obstacles is shown in Fig 5 and the corresponding maps are shown in Fig. 5

Fig 5. Experimental Set-up.

The processing of 48000 points, or even more if greater accuracy is desired, for the purpose of terrain estimation is thereby not feasible to be performed at every step. Therein lies the need to reduce the amount

of data used to represent the environment, with minimum loss of useful information, as mentioned before.

These problems have been resolved by reducing the discrete, scattered point data into a cell grid , representing the entire sensed environment. Currently there are 50 cells in the X direction and 25 cells in the Y direction, of dimensions of about 10cm*10cm, subject to the calibration and range of the structure light set-up. Every grid cell contains the following information:

$$[x_{max}, y_{max}, z_{max}, x_{min}, y_{min}, z_{min}, z_{avg}, n, \text{cell state}, sd]$$

$x_{max}, y_{max}, z_{max}$ are the coordinates of the point in the cell with the maximum height.
$x_{min}, y_{min}, z_{min}$ are the coordinates of the point in the cell with the minimum height.
z_{avg} is the average height for all points in the cell.
n is the number of points detected in the cell.
cell state ε (0,1,2), 0 when the cell is free, 1 when the cell has obstacle, 2 when the cell cannot be sensed.
sd is the standard deviation of all the points that belong to the cell about z_{avg}.

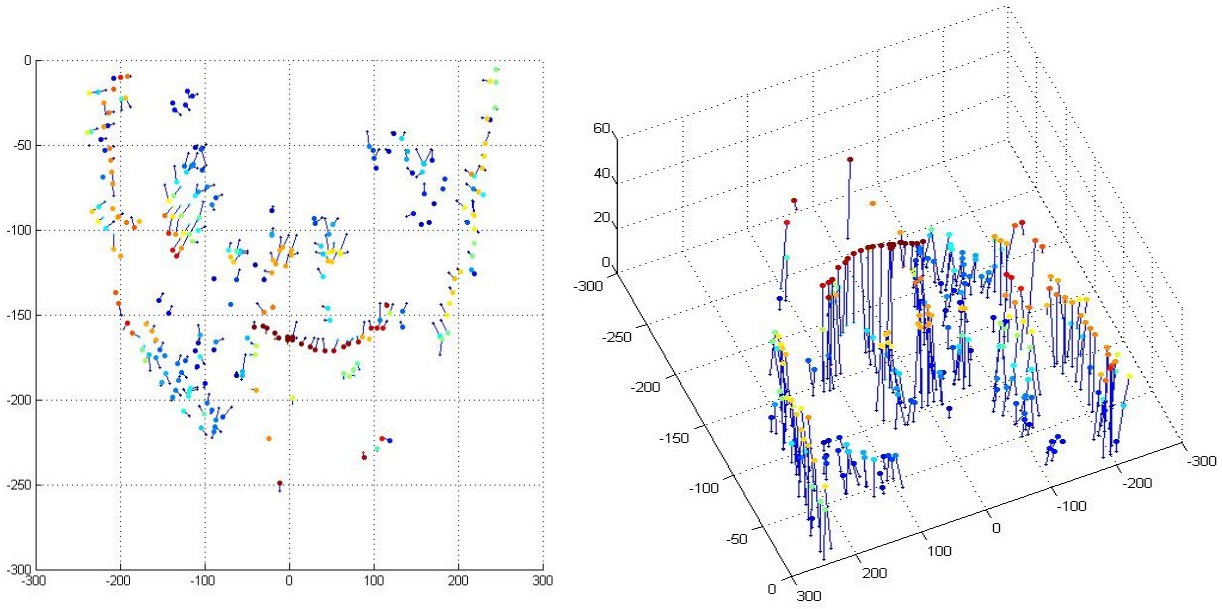


Fig. 6 Cellular gradient data, top view and 3D perspective.

b. Simulator

The flow chart of the simulation model is shown in the figure. 7. The functional blocks shown inside the dotted boundary are shown implemented here in this paper. For simulation purpose, we assume we have the map of surrounding available to us at every iteration of the path planning algorithm. If the map data, obstacles or gradients, is not changed during the path planning it is assumed that no new information about surrounding is available to the rover. The algorithm has provision to include new surrounding data in continuous manner while the path planning is being done. But no new data is updated during simulation run.

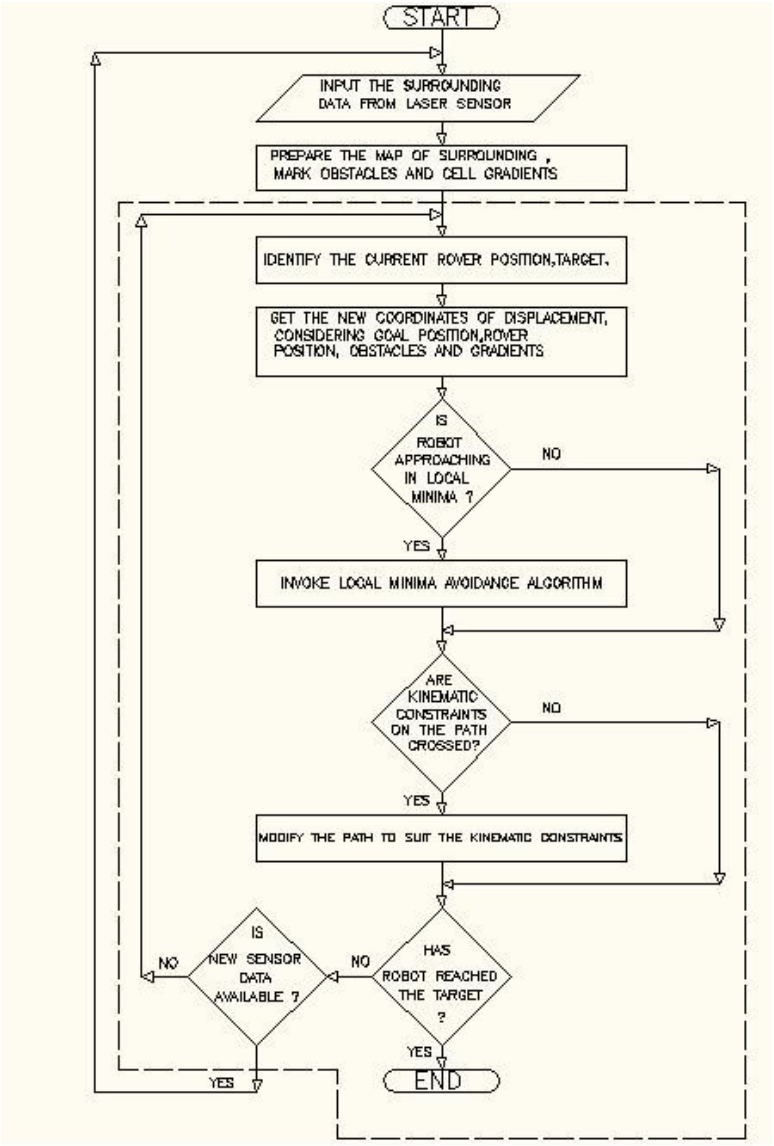


Fig. 7: flowchart of the algorithm.

c. Colour code for cells

The cells in the simulation plan are given a color code as follows:

GRID SLOPE (in radian)	COULOUR
Slope <0.1745	**White**
0.1745 < slope < 0.3490	**Green**
0.3490 < slope < 0.5235	**Yellow**
0.5235 < slope < 0.6108	**Red**

It should be noted that the two cells in the plan with same color may have different slopes hence they may exert repulsive forces which are different in magnitude. The simulation starts, with The rover placed at starting point. The successful continues until the rover reaches the target. The figures below show the final result of the simulation.

4. RESULTS

After running the simulation for different types of obstacles and ground profile combinations we get the following results. In the results the grid size is 20 x 20 mm each.

4.1 Overcoming trap situation

First it is demonstrated how the algorithm to avoid trap situation works without including the gradients. The figure 8 shows the final output of the simulation. There are two obstacles and the target is shown by a cross. Such type of trap may or may not be always present in given obstacle formation.

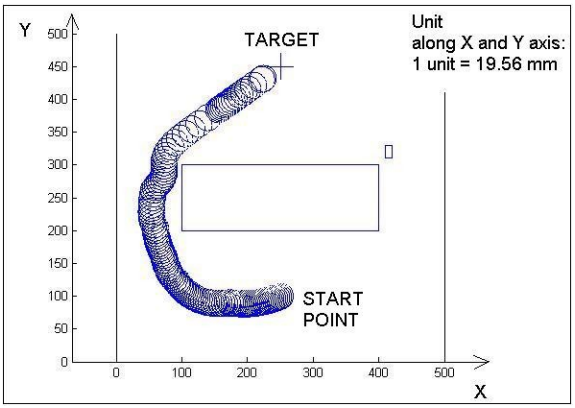
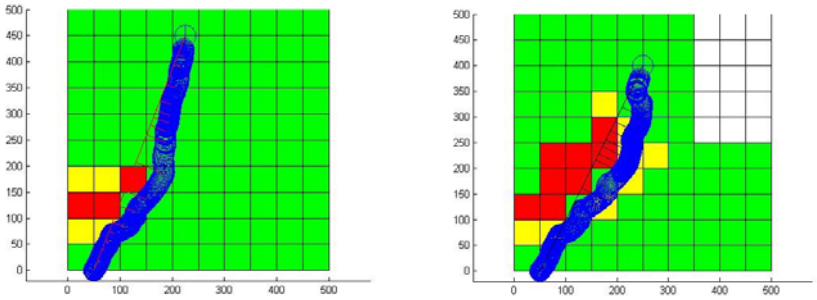


Fig. 8 Avoiding onbstacles wihtout gradients.

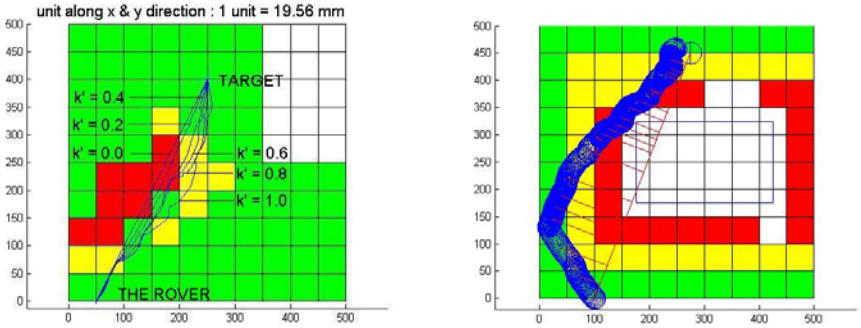
4.1 Overcoming gradient situation

Figure 9(a), figure 9(b) show how a gradient is dealt with by the algorithm. The cell with a gradient is different from an obstacle. The rover cannot enter an obstacle. But the rover can pass through the cell which is not treated as an obstacle regardless of its gradient. The straight line(red in fig 9(a) and black in fig 9(b)) shows the straight path starting from initial rover position and ending at the target. The line helps to observe the deviation of path from the straight line. Figure 10(a) shows the different path traced by the rover for different values of k' in equation (2).



(a) (b)

Fig. 9 (a-b) The rover dealing with only gradients of two different types.



(a) (b)

Fig. 10: (a) The path traced for different values of k' in case of fig. 7(b) and (b) path with obstacles and gradients.

Table 1: The results of the case Figure 9(b)

case	Value of k'	Length of path (m)	Percentage path over white zone	Percentage path over green zone	Percentage path over yellow zone	Percentage path over red zone
7 (b)	0	8.18	0	45.87	13.72	40.41
	0.2	8.20	0	51.11	15.76	32.26
	0.4	8.26	0	57.04	26.03	16.93
	0.6	8.59	0	63.06	27.21	9.39
	0.8	9.04	0	60.33	29.36	10.31
	1.0	9.21	0	59.66	40.33	00.00

4.2 Overcoming obstacle and gradient situation simultaneously

Thirdly it is shown how the path is generated for the rover taking into account the obstacles present as well as the gradient of the ground profiles. Figure 10(b) shows the path traced by the rover in a map where both ground gradient and the obstacles are present.

5. CONCLUSION

It can be concluded that this method is capable of path generation for the lunar rover considering both the kinematic constraints of the rover and gradients of the terrain. The method reduces the amount of tilting the rover would undergo compared to the algorithm where the ground profile is not taken into consideration. This in turn helps safety of the rover from over tipping. We are currently experimentally evaluating the rover path on different terrain conditions.

6. REFERENCES

[1] 0. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots", Proceedings of the IEEE International Conference on Robotics & Automation, 1985.

[2] McFetridge and M. Yousef Ibrahim, "New Technique of Mobile Robot Navigation Using a Hybrid Adaptive Fuzzy-potential Field Approach", Computers & Industrial Engineering, Vol. 35, Issues 3-4L., December 1998.

[3] R. Volpe and P. Khosla, "Manipulator Control with Superquadric Artificial Potential Functions: Theory and Experiments", IEEE Trans. on System, Man and Cybernetics, Vol. 20, No.6, 1990.

[4] Parakh S., Pankaj Wahi, Dutta, A, "Velocity kinematics based control of rocker-bogie type planetary rover", TENCON 2010 - 2010 IEEE Region 10 Conference

[5] Takeshi ohki, keiji Nagatani and Kazuya Yoshida, "Safety Path Planning for Mobile Robot on Rough Terrain Considering Instability of Attitude Maneuver", IEEE/SICE International Symposium on System Integration (SII), 2010

[6] Tomoaki Yoshida, Kejii Nagatani, Eiji Koyanagi, Yasushi Hada, Jazunori Ohno, Shoichi Maeyama, Hidehisa Akiyama, Kazuya Yoshida and Satoshi Tadokoro, "Field experiment on Multiple Mobile Robots conducted in an Underground Mall", Springer Tracts in Advanced Robotics, 2010, Volume 62/2010, 365-375.