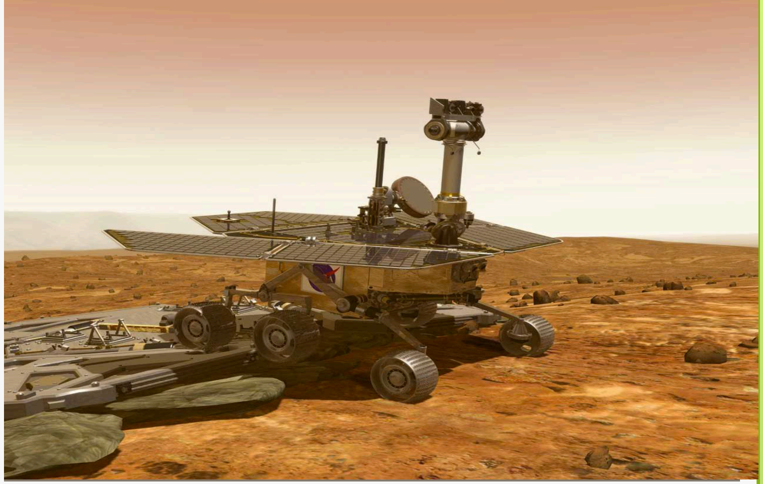


Global Path Planning on Board the Mars Exploration Rovers

Sepideh Hadadi, MSCV, semester 2, autonomous robotics Module, Centre Universitaire Condorcet, Le Creusot, France

May 2017

Topics:

1. **Introduction**
    - Remote navigation issue
    - Problem statement
2. **Autonomous navigation system**
    - On-board navigation Overview
    - shortcomings
3. **Global path planning**
    - D* algorithm
4. **Integration**
    - Cost map
    - Voting system for path selection
5. **Resource limitation**
    - Automatic re-centering
    - Coarse Resolution Cost Maps
    - Map Filtering
6. **Result and discussion**
    - Test-bench
    - Result
7. **conclusion**

Remote navigation issue

- In January 2004, NASA's twin Mars Exploration Rovers (MERs), *Spirit* and *Opportunity*, began searching the surface of Mars for evidence of past water activity
- Given the **latency** in sending commands from Earth to the Martian rovers (and in receiving return data), a **high level of navigational autonomy is desirable**
- GESTALT (Grid-based Estimation of Surface Trasversability Applied to Local Terrain) path planner
- The GESTALT is **susceptible to failure** when clusters of closely spaced, **non-traversable rocks** form extended obstacles.
- Carnegie Mellon University Field **D* global path planner** has been integrated into MER flight soft- ware A revised version of AutoNav was uploaded to the rovers during the summer of 2006.
- This paper describes how global planning was integrated into the MER flight software

Problem statement

- **Several problems might arise during mars rover navigation:**
  - It can take as long as **26 minutes** for a signal from Earth to reach Mars (and vice-versa).
  - line-of-sight and power constraints further complicate the situation
  - In general, before the rover shuts down for the night, it will send data back to Earth. This data is then used to plan activities for the following sol. Therefore high level of autonomy is required.
- **There are two main methods that can be used to reach autonomy goal.**
  1. The first and simplest is **the blind drive(no identify hazardous terrain).**
  2. autonomous navigation with **hazard avoidance** (AutoNav)

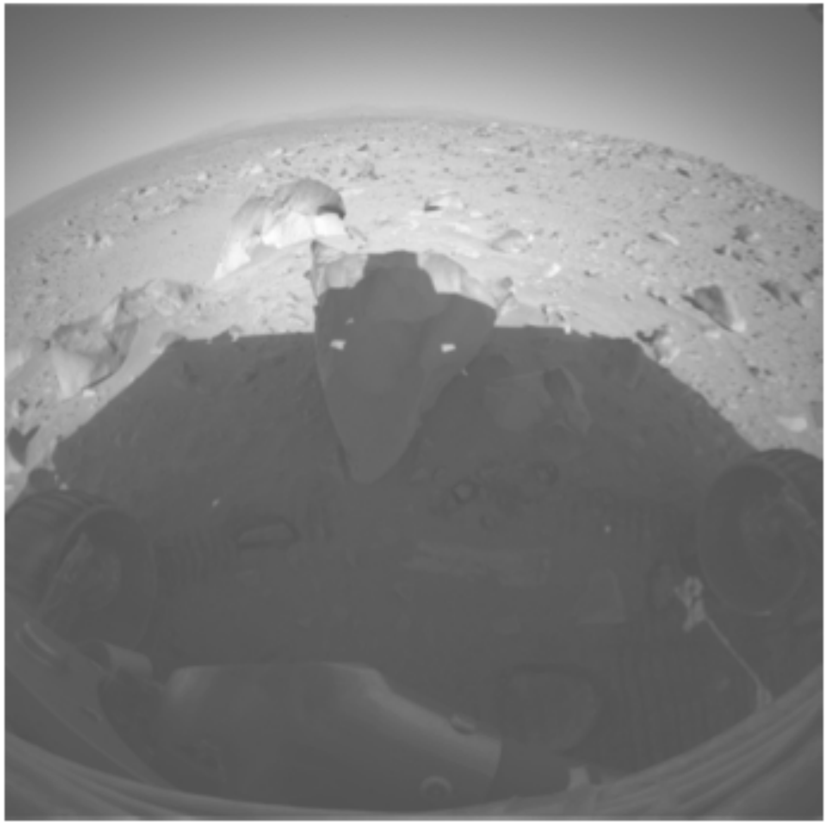
On-board navigation overview

Autonomous navigation system

- AutoNav is based on the GESTALT (Grid-based Estimation of Surface Trasversability Applied to Local Terrain) algorithm
- It uses stereo image pairs
- Part of this model is a goodness map
  - High goodness values: **easily traversable terrain**
  - low goodness values: **hazardous areas**
- **Waypoint navigation algorithm has following step:**

1. Once the terrain has been evaluated, a set of candidate short paths, from the current rover location to the target, is considered
2. The paths are evaluated based on 3 criteria:
   - avoiding hazards
   - minimizing steering time
   - reaching the goal
3. The three votes are then weighted and merged to generate a final vote for each path.

- This process is repeated until the goal is reached



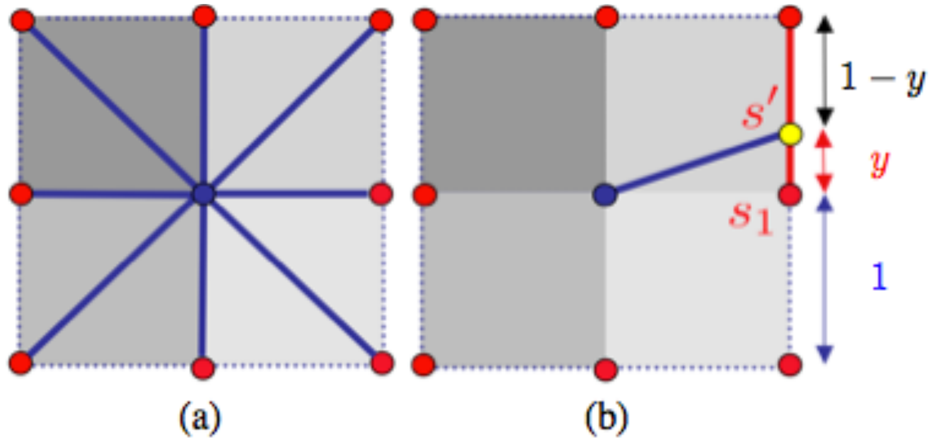
On sol 108, *Spirit* was unable to autonomously navigate to a goal location on the other side of this cluster of rocks. This image was captured by one of the front hazard avoidance cameras mounted on the body of the rover. *Cour- tesy NASA/JPL-Caltech.*

shortcoming

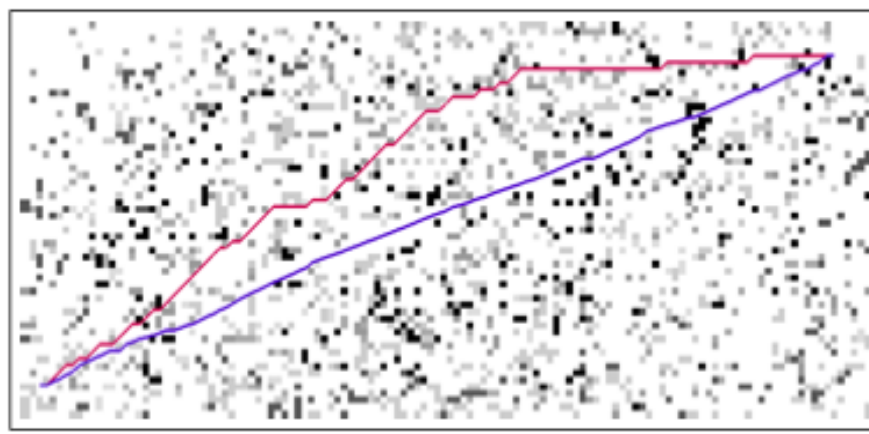
- AutoNav is very good at keeping the rover safe and usually gets the rover to the goal location. However, in some instances AutoNav can not reach to the goal.
- Example: *Spirit* spent approximately 105 minutes trying to get around a cluster of rocks (shown in the previous slide), but was unable autonomously do so. Forty-seven drive steps were taken during the attempt.
- The simple method used to construct the waypoint votes leads to this problem.
- **incorporating obstacle information** that *is* available into these global plans typically provides much **better estimates** than Euclidean distance.

D* algorithm

- **problem:** **classic grid-based planning algorithms involve unnecessary turning.**
- To use previously suggested improvement the AutoNav system has been extended to use the **Field D* algorithm** to generate these global paths.
- The Field D* algorithm removes corner-based restriction and allows paths to transit through any point on any neighboring grid cell edge, rather than just the neighboring grid cell corners or centers.
- Field D* algorithm used equation 1 to remove this restriction

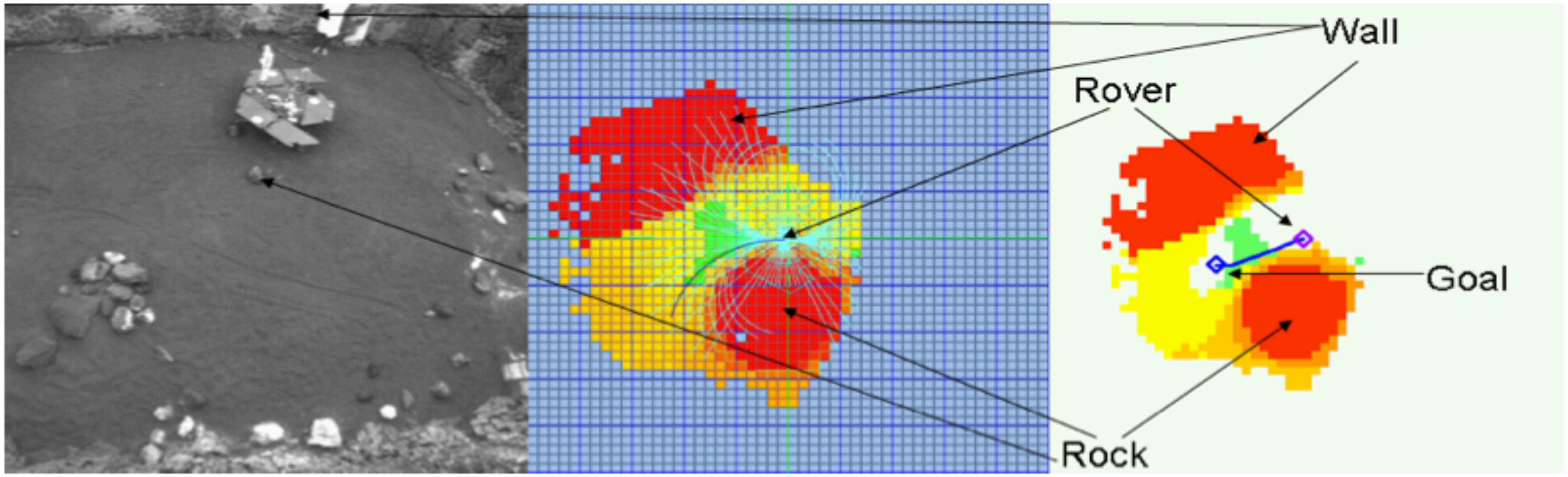


$$PathCost(s') \approx y \cdot PathCost(s_2) + (1-y) \cdot PathCost(s_1) \quad 1$$



Field D* is able to provide much more direct(blue), less-costly (red)

Cost map



using Field D* to improve AutoNav involves two main tasks:

1. providing terrain information to Field D* in a form it can utilize
2. Field D* to generate steering recommendations in a form that AutoNav can understand

Field D* uses a uniform grid which is very similar to the goodness map. the Field D* map is fixed to the environment and does not move along with the rover.

Blue cells in the above map have unknown Trasversability. All other cells are colored based on a gradient between green (high goodness/low cost) and red (low goodness/high cost).

voting system for path selection

- The path with the lowest cost of traversal to the goal is the best and the one with the highest cost is the worst. Numerical vote values are assigned using a weighted sum of v_{scale} and v_{close}, v_{max} is the maximum possible vote, c_{max} and c_{min} are the maximum and minimum traversal costs for the current path set evaluation, and c_i is the traversal cost for a given path.

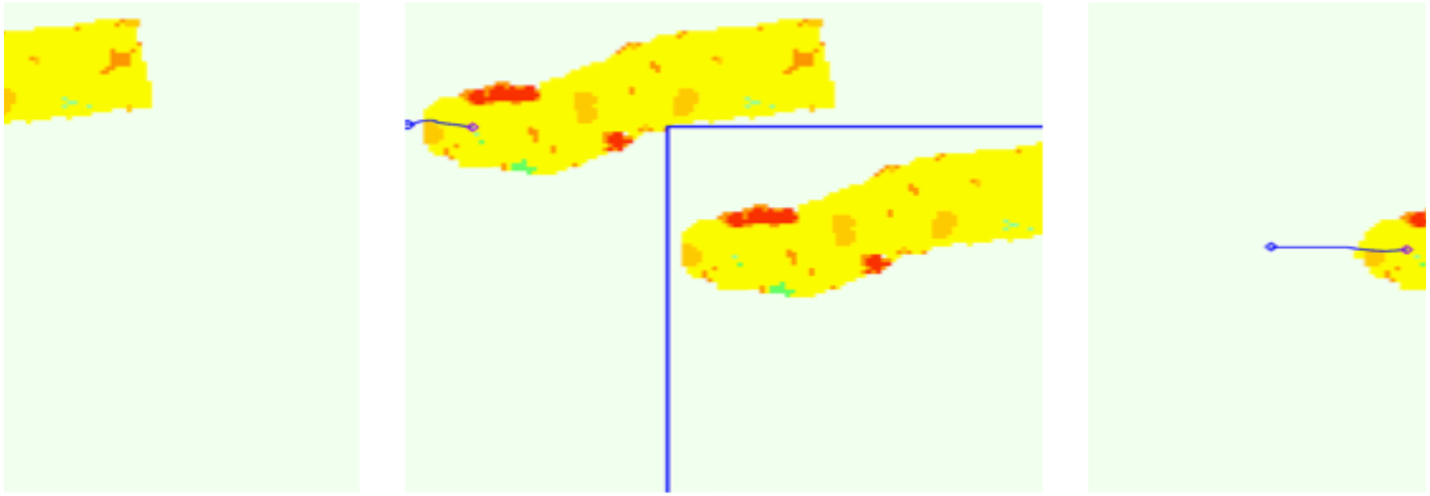
$$\begin{aligned} v_{scale_i} &= v_{max} * (c_{max} - c_i)/(c_{max} - c_{min}) \\ v_{close_i} &= v_{max} * c_{min}/c_i \end{aligned}$$

Once these votes have been constructed, they replace the way- point votes constructed by GESTALT. They are then combined with steering bias and hazard avoidance votes in order to select the arc that will be followed.

Automatic re-centering

The Mars rovers are constrained by very limited computational resources. The onboard computer uses a radiation hardened RAD6K processor running at 20 MHz, and has 128 Mbytes of DRAM. **This resources are shared among the 97 tasks (including AutoNav)**
the **path cost minimization** step of the algorithm **is pre-computed**, and the results are stored in a lookup table that is then accessed at runtime. This **significantly decreases the computation time** required for planning

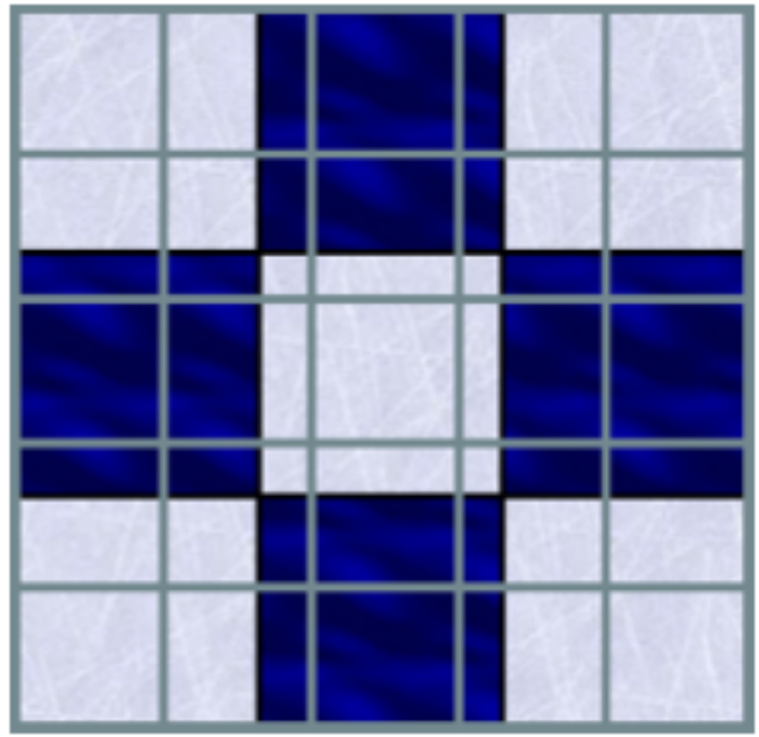
Field D* is efficient for automatic re-centering because it does not have to re-plan from scratch when new costs are discovered.

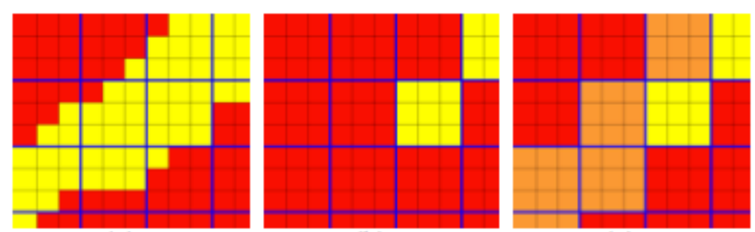


(a) (b) (c)

Coarse resolution cost map

- Another way to manage limited memory resources is to change the resolution of the cost map grid cells.
- because the Field D* planner is able to compute paths that are not restricted to transitioning between grid cell centers or corners, it can be used to plan direct.
- In the figure shown at the bottom, Red represents obstacle, yellow is traversable, and orange is the maximum traversable cost. Coloring is by goodness value in (a) and cost value in (b) and (c). The cost map in (b) is produced by using the minimum goodness value in each cost cell.



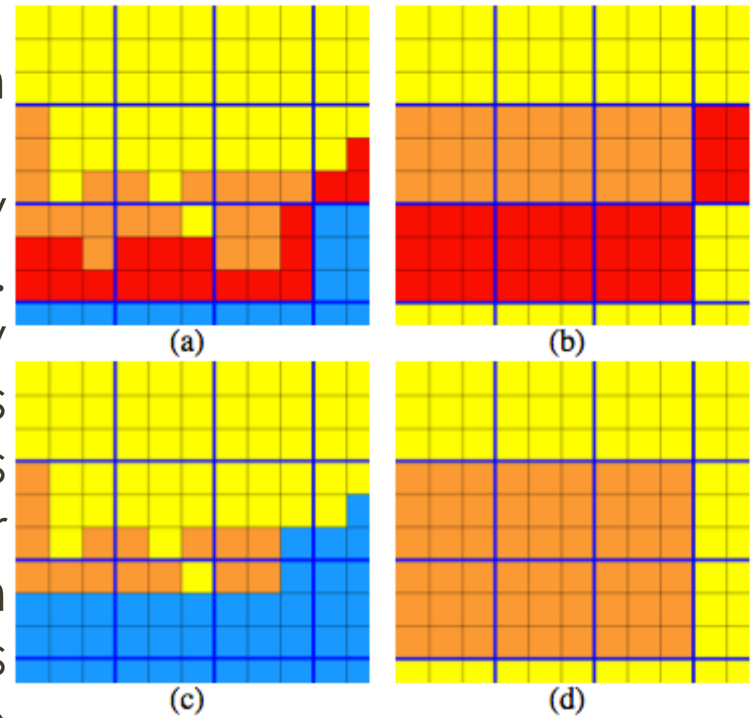


(a) (b) (c)

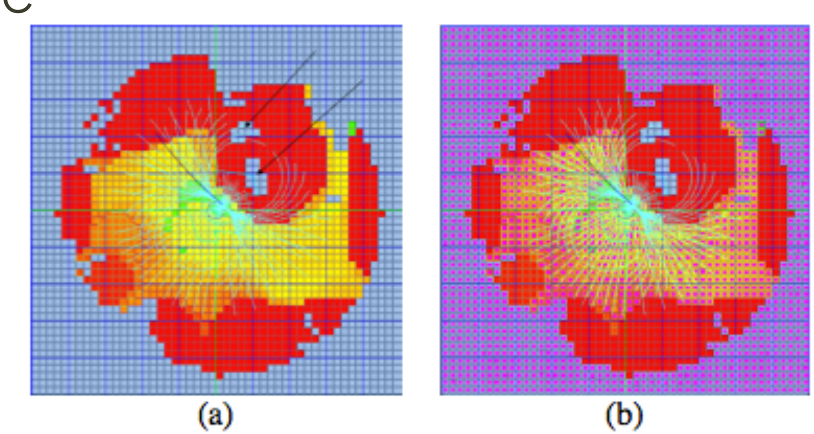
Resource limitation

Map filtering

- Obstacle cells are shown in red. The goodness map in (a) contains regions completely surrounded by obstacle cells. Planning time is greatly increased when arc endpoints fall in these regions. All cells reachable from the rover location are shaded pink in (b). Note that the regions surrounded by obstacle are not marked as reachable.



(a) (b) (c) (d)



(a) (b)

Test-bench

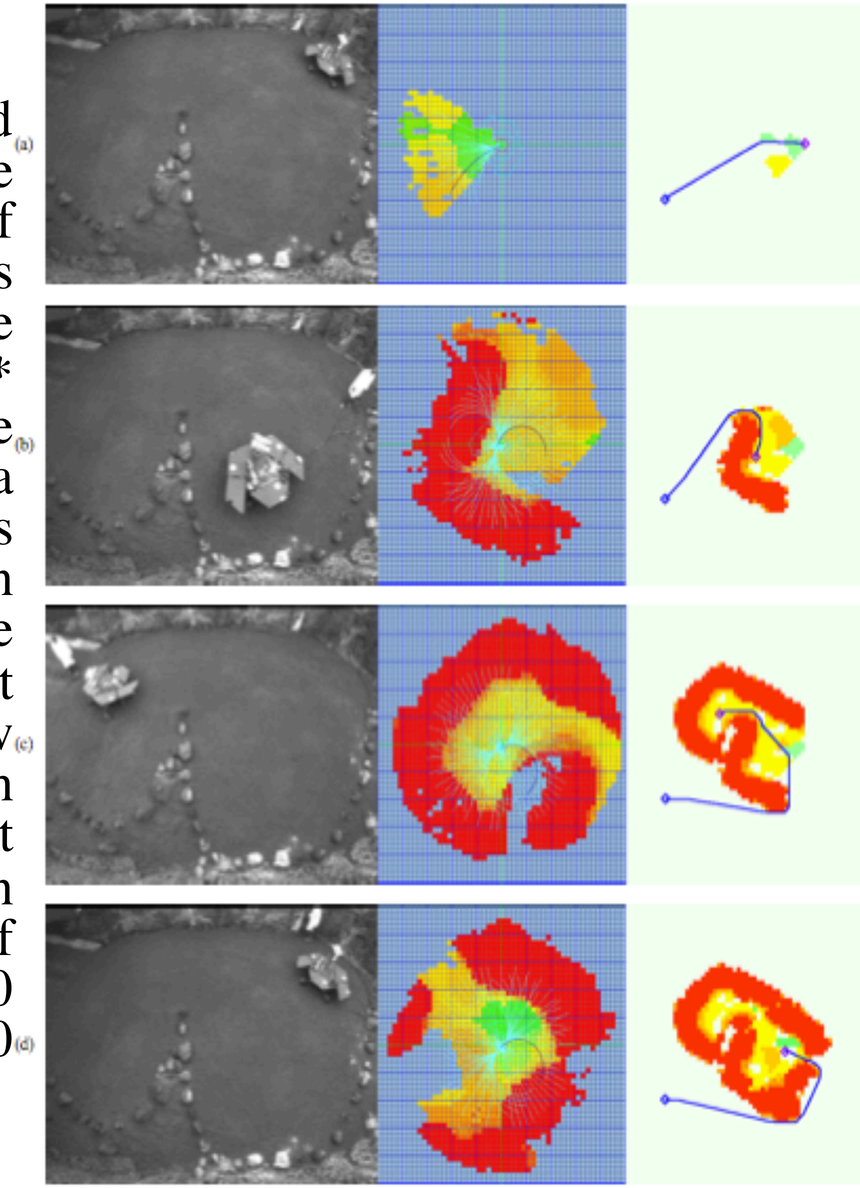
- The MER Surface System TestBed (SSTB) was used to extensively test flight software modifications.
- The SSTB is a high- fidelity engineering model of the Mars Exploration Rovers
- The SSTB is housed in an indoor sandbox approximately 9 meters wide and 22 meters long
- Testing was limited to this environment for several reasons.

MER Surface System Test-Bed rover

Result and discussion

result

- Field D* assisted hazard avoidance using the SSTB. The left image is an overhead view of the sandbox. The middle image is the local goodness map, and the image on the right is the Field D* cost map. Note that the entire goodness map is shown, but only a portion of the cost map is included. Blue cells have unknown Trasversability. All other cells are colored based on a gradient between green (high goodness/low cost) and red (low goodness/high cost). The blue line on the cost map is the path planned between the rover and the goal. The size of each goodness cell is 20 cm x 20 cm. Each cost cell is 40 cm x 40 cm.



conclusion

- An state of ART autonomous robot navigation was presented for mars rover.
- The path planning algorithm discussed and shown how D* algorithm helps the rover to follow his path when uncertainty is involved.
- The resource allocation discussed and the test setup was presented.
- The result achieved by the rover was illustrated and the analysis of the results was presented.