AUTONOMOUS NAVIGATION AND MAPPING SYSTEM FOR MRPTA ROVER

*V. Polotski[1,3], F.J. Ballotta [2] and J. James[3]

[1]Ecole de Technologie Superieure
1100 Notre Dame
Montreal, Canada, H3C1K3
*(*Corresponding author vladimir.polotski@gmail.com)*

[2]Software professional currently working at Bloomberg

[3] Cohort Systems Inc
900 Greenbank, # 525
Nepean, Canada, K2J4P6

AUTONOMOUS NAVIGATION AND MAPPING SYSTEM FOR MRPTA ROVER

ABSTRACT

Initiated and financed by the Canadian Space Agency, Micro-Rover Platform with Tooling Arm (MRPTA) project has targeted the development of a robotic system with remote control and autonomous navigation capabilities for testing a large variety of analogous planetary missions. Although developed for planetary exploration, the navigation system is highly suitable to such areas of application as construction and in mining. Autonomous navigation system is capable of moving the platform to predefined position(s)/orientation(s) while continuously mapping the terrain, assessing its traversability and choosing most appropriate path to follow. Designed according to the JAUS (Joint Architecture for Unmanned Systems) framework, the system consists of several interconnected components. Following components are of particular importance: Pose Estimator (PE), Terrain Evaluator (TE), Map Manager (MP), and Path Planner (PP). The PE uses a minimal sensor configuration consisting of an azimuth gyroscope, inclinometer and wheel odometry. The PE provides reliable pose estimates in spite of pronounced slippage by employing extensive use of fused data. The TE uses a nodding laser scanner which continuously sweeps the area in front of the platform and constructs a "traversability grid" (TG) of the surrounding area. Multiple TGs are constantly constructed along the platform motion path are combined into a terrain map maintained by the MP. The PP uses this map to compute a motion path Obtained path is executed by the motion controller and remains current until it is contradicted by the most recent map – at which point it is re-planned. This paper addresses integration issues, lessons learned and also outlines the possible applications for construction and mining.

KEYWORDS

Autonomous navigation, planning, control, mapping, planetary exploration, data fusion

INTRODUCTION

Planetary exploration in general, and rover navigation in particular, have been a subject of continuous interest to the mobile robotic community over the last two decades. Examples of early works worth mentioning are (Matthies et al., 1995) and (Howard & Seraji 2000). More recent results are often related to the development and analysis of the Mars missions by (Biesiadecki & Maimone, 2006). The Planetary Exploration Group at the Canadian Space Agency (CSA) has also been very active investigating long term autonomous navigation activities (Dupuis et al., 2005).

Starting in 2009 CSA initiated and financed the series of projects devoted to various aspects of planetary exploration activities. This paper describes an onboard navigation system developed to guide the Micro-Rover Platform with Tooling Arm (MRPTA) designed for CSA in 2010-2012. MRPTA rover's main goal is to help astronauts remotely explore unknown environments within short to medium range proximity (approximately 100m) from their location. The typical mission consists of ''scouting'' - moving autonomously to specified locations, taking video recordings, making scientific measurements or collecting samples, then returning back to the starting point. The work described in this paper has been accomplished by Cohort Systems Inc. under a subcontract to Engineering Services Inc (ESI) according to CSA contract F028-090480. Electromechanical design and low–level software design was the responsibility of the ESI engineering team and are not addressed in this paper. We focus on the high level architecture for planning, control and sensor fusion, as well as software design and lessons learned along the testing and delivery of the rover.

AUTONOMOUS NAVIGATION – ARCHITECTURE

The rover navigation software is designed to support autonomous and tele-operated operation modes. At a high level it is partitioned into 3 layers: (1) a deliberative or mission execution layer (DL), (2) a reflexive or navigation layer (RL), and (3) a platform control or low level layer(PL). In order to provide the human operator a way to interact with the rover in autonomous mode, and to execute the tasks requiring tele-operation, an Operator Control Unit (OCU) was developed. The low-level platform control is tightly connected to the hardware development and is not addressed here; we limit our discussion to the deliberative and reflexive layers, and shortly address the OCU structure.

The deliberative layer continuously evaluates a best sequence of actions that will achieve mission goals and elaborates these action sequences into an ordered list of commands that are performed by the reflexive layer. Individual commands are presented to the RL for execution and continuously monitored. The DL decides on appropriate parallel or subsequent actions based on the execution state reported by the reflexive software (*in process, success, failure, aborted,* or *cancelled*). Development of the deliberative layer follows the methodology of "robotic autonomy" proposed in (Alami et al 1998). This layer also contains the Deliberative-Reflexive Interface - a communication portal between two layers used to transfer commands and status messages. The interface module performs protocol conversion between the layers: it translates message contents to a form required by the receiving component and dispatches the translated content to the intended receiver using an appropriate communication protocol. The DL, in particular, initiates and monitors the execution of the complementary tasks such as scooping, scientific measurements, or taking panoramic views according to the goals set in the mission plan.

MRPTA inter-component communication and the full reflexive layer implementation follow the guidelines of the Joint Architecture for Unmanned Systems (JAUS) and the OpenJAUS standards in particular. JAUS has evolved from an experimental development initiative to a current set of SAE – compliant standards maintained by the JAUS Working Group. JAUS is a message-based architecture organized in a three-level hierarchal network (subsystem, node and component) with *component* being a main functional element. All modules in the reflexive layer and the OCU are implemented as JAUS components. Components collaborate by exchanging well-defined messages. Each JAUS component is a self-contained entity that provides services to other components. Each service is implemented as a messaging interface that uses a standard message protocol to interact with other components. Each component is defined as a state machine with 6 standard JAUS states: *Initialization, Standby, Ready, Emergency, Failure and Shutdown*. The state machine within each component operates at a specified frequency (the default is 10 Hz). State transitions are initiated by internal logic or by external messages.

AUTONOMOUS NAVIGATION – MAIN FUNCTIONALITY

Autonomous navigation is a complex task requiring pose estimation, assessment of the environment and motion execution. Below we briefly address the sensing equipment then describe the functionality of the most important components involved in autonomous navigation (there are ~30 components in total, and their full description is impossible due to space and scope limitations).

Sensing Equipment

MRPTA pose is reconstructed from the data provided by an absolute inclinometer, an azimuth gyro, and an *odometer* composed of encoders attached to the left and right driving motors. Since the rover may be used in several driving configuration, an effective wheel radius for each configuration is stored onboard, and the configuration is a part of mission definition. AMD900-TW (Applied Geomechanics) absolute inclinometer provide pitch and roll measurements and a KVH DSP-3000 fiber-optic gyro provides azimuth angle increments. These sensors were chosen for their accuracy and robustness. For operations on high slopes (beyond the range of the AMD900-TW) a small IMU is used, which also serves as a backup sensor. A laser range finder manufactured by Hokuyo (UT-30LX) is used to collect environment surface data. This sensor is mounted on a servo controlled tilt unit in order to increase the sensing area. Each sensor has a dedicated software component - a *server* responsible for providing sensor data to client components.

Pose Estimator

The pose estimation algorithm (PE) first takes data from azimuth gyro and inclinometer to estimating platform rotational pose (yaw/pitch/roll) and then combines the results with incremental displacements obtained from an odometer. The PE developed for the MRPTA rover has two additional features: it estimates the gyro bias when the platform is not moving along a certain period of time, and it performs best possible estimates of wheel slippage. Because the rover is skid-steered in both wheeled and tracked configurations slippage-induced errors can significantly affect pose estimation. The ability to eliminate or reduce slippage-induced error has proven to be very important (Biesiadecki.& Maimone, 2006). Slippage correction relies on redundant platform rotation estimates: those obtained from differential odometry and those from the azimuth gyro. Careful monitoring and correction of rotational error lets the PE provide consistent fusion-based displacement and rotation estimates.

Terrain Evaluator

Based on the data collected by the LIDAR the terrain evaluator (TE) determines traversability values for the area around the rover's location and stores this data in a centred-on-platform toroidal grid (T-grid). The T-grid is continuously updated with the new LIDAR data and updated T-grid data is regularly sent to the Map Manager where the world model is developed and maintained. If there are not enough data-points in the area in front of the rover the TE issues commands to the LIDAR Tilt Controller to achieve a minimum point density over contiguous grid cells immediately in front of the vehicle. The TE attempts to provide cell coverage within a ~6 meters horizon in front of the rover and to guarantee a minimal point density over a shorter, configurable distance.. Currently a the TE assigns point data to a (121x121) T-grid with cell size of 0.15 meters. Traversability assessment uses an approach proposed in (Solanki 2007) and is based on the combination of three characteristics: (1) slope estimate *S*, (2) roughness estimate *R,* and (3) neighbourhood estimate *N*. Traversability values are scaled between 1 and 14 - higher means better traversability, 7 is neutral and 2 or less is not traversable. The LIDAR pose needed for computations is obtained from the Tilt controller. The slope estimates uses the best fitting plane through the data points in each cell. The roughness estimate is represented by the variance of the elevation of the data points within a cell. The neighborhood estimate is based on the assessment of the travel from a cell to the center of the grid (note that T-grids are centred-on-platform). The rationale is given below.

Let us consider a cell at the row *i* and column *j* from the T-grid center. The cell height $h(i,j)$ is computed first by averaging elevations of all points in a cell. Then the weight factors c1, c2, and c3 are defined as follows:

$$c1 = \frac{|i|}{C}; c2 = \frac{|j|}{C}; c3 = 1 - c1 - c2;\ here\ C = \left(|i| + |j| + \frac{|i| + |j|}{\sqrt{2}}\right)$$

Using the heights of the cell and of its neighbors on each side, and the weight factors the neighbourhood estimate $N(i,j)$ is computed as follows:

$$N(i,j) = h(i,j) - c1 * h(i-1,j) - c2 * h(i-1,j-1) - c3 * h(i,j-1)$$

This formula is for a cell in the first quadrants (*i, j>0*), other cases are similar.
The final traversability estimate $T(i,j)$ is then calculated as: $T = \min(S + R/2\,, N)$.

Map Manager

Map Manager (MM) maintains a *world model* containing all spatial information used by the navigation system. At start-up it reads in map data from external sources if available or initialises its *world model* as a flat terrain. During operation it updates the *world model* using position/orientation data received from PE and terrain information obtained from TE and provides map data to other navigation components. The MM maintains a traversability map and an elevation map where *elevation* is identical to *Z(X,Y)* in JAUS world coordinates. Information from the TE is received in a form of a T-grid (121x121, 0.15m cell size). With information from the PE, MM registers the central cell of the T-grid with the rover's pose, it monitors the rovers displacement and adjusts the T-grid to ensure T-grid shifts (cell entering and falling off the grid). Obtained terrain traversability map differs from the elevation map and is used only by path planning components.

Figure (1.a) illustrates the elevation map constructed dynamically during a MRPTA experimental runs. A "castle" – type structure shown in figure (1.b) was built artificially for testing the rover's ability to estimate positive and negative heights. The height is coded with the colors (red corresponds to >40cm height, blue to >10cm deep). One can see that the area encircled in figure (1.a) reflects the terrain structure.

(a) (b)

Figure 1 – (a) Map constructed by the MRPTA along the exploratory run; (b) Picture of an artificial obstacle (castle) in the Cohort testing site taken by the external camera

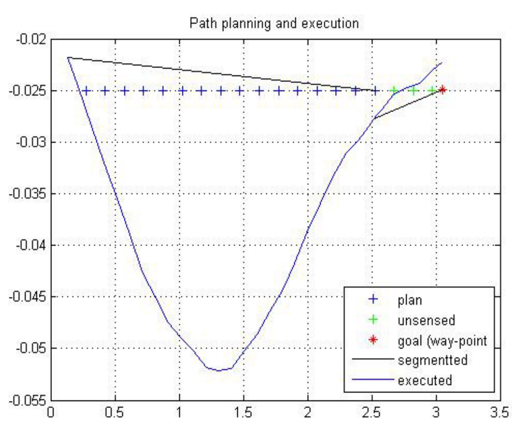


Figure 2 – Planning, Segmentation and Execution Figure 3 – MRPTA - short track configuration

Path Planner and Segment Driver

Path planning is based on the terrain traversability map constructed by TE and MM components. The planning algorithm computes the path to the goal considering the area that has not been sensed yet as being neutral (flat). The planned path consists of the sequence of cells to be visited. This information is passed to the Segment Driver (SD). The SD is responsible for defining the portion of the path containing cells that have been sensed and to divide this portion into a sequence of straight line segments. SD also verifies that the directions of consecutive segments are sufficiently close (a 10 degree threshold was applied), and inserts the *in-situ* rotation between neighboring segments if needed. A final set of segments (and *in-situ* rotations) is presented to the Motion Controller for execution. Motion control along the straight lines and the execution of *in-situ* rotation are routine procedures and their description is omitted.

Figure 2 illustrates the described algorithm with experimental data. The planned path is marked with crosses (the sensed portion with blue crosses, the un-sensed portion - with green crosses, the goal-point - with a red cross). Black solid line corresponds to a computed segment. An offset between starting points of the planned path and the segment are due to the fact that planned points are the centers of the cells, but the segment starts at the rover's location. The solid blue line corresponds to the executed trajectory (PE estimates). After execution of the segment the new path is planned. By that time, previously un-sensed points have been sensed and a new segment starting close to the previous segment is computed and executed, and so forth. Note the very different scale factors along X-axis (0.5 m) and Y-axis (5 mm).

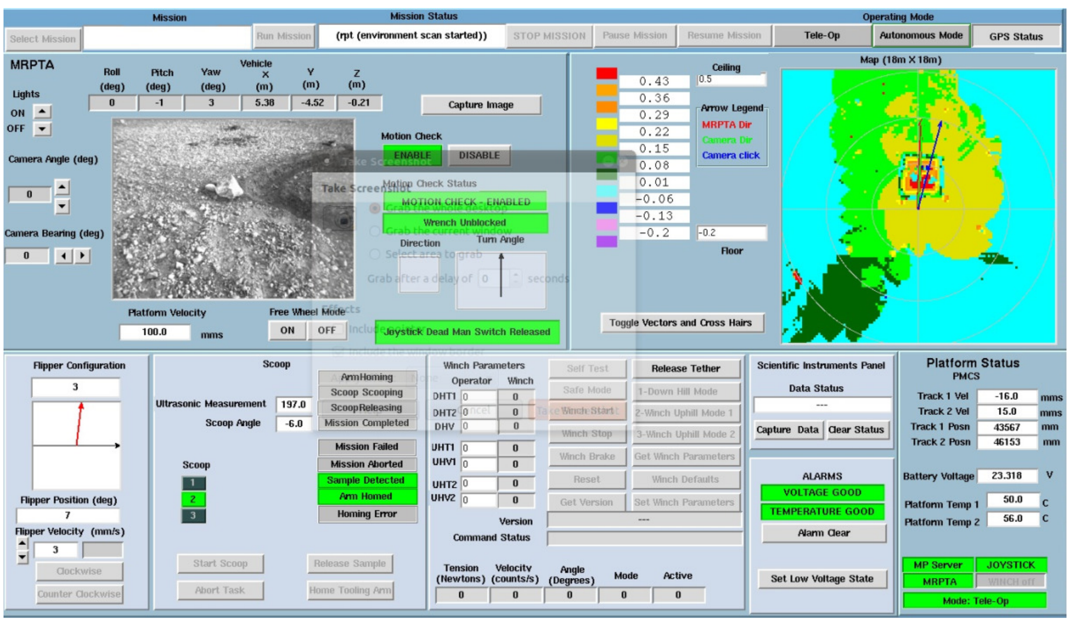


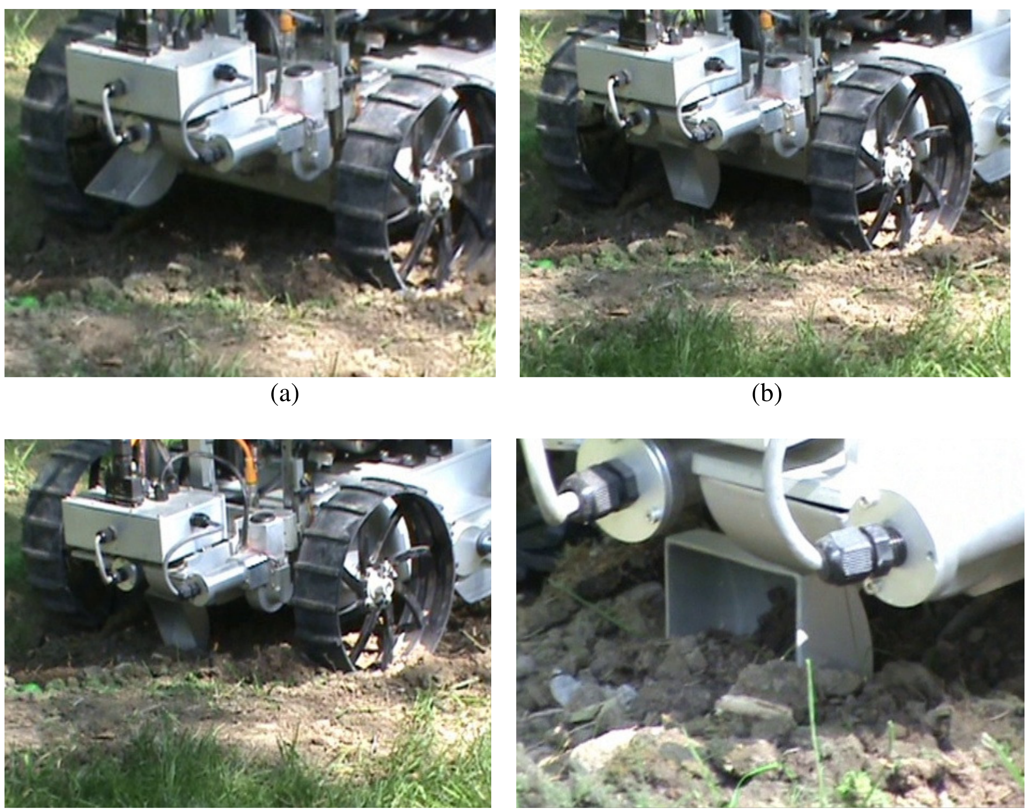
Figure 4 – OCU snapshot along an experimental run on the Cohort testing site.

Operator Control Unit (OCU)

The OCU is mainly a graphical interface providing a human operator with the possibility to control the system. The OCU continuously processes the events triggered by interface signals and incoming JAUS messages. It also manages various logging activities and stores data for future analysis. Figure 4 illustrates the OCU showing (1) constructed map in the upper right corner with the elevations coded by colours (as shown be a panel on the left of the map), (2) camera view ensuring operator situational awareness, (3) platform status parameters in bottom right corner. The OCU is used in both autonomous and tele-operation modes. It contains POP-UP menu for mission definition (not shown). In tele-operation mode the operator uses the joystick and may switch from platform driving to controlling various pieces of equipment, including a *flipper* – a variable length arm used to change the track shape (flipper position is shown in the bottom left corner). The flipper arm is visible in figure 3 where MRPTA is in short track configuration (the triangular shape) while the long track configuration used for long range missions is presented in figure 6.

INTEGRATION AND TESTING

MRPTA project required integration of several pieces of sophisticated equipment. In figure 3 one can see the Hokuyo LIDAR mounted on the tilt unit, a stereo camera mounted on pan-tilt unit and a scientific instrument attached to the tooling arm. The sequence of pictures in figure 5 illustrates the autonomous execution of “scooping” – taking samples of the soil using a small scoop specifically designed for MRPTA and attached to the tooling arm. The platform is in “wheeled” configuration (as compared to figure 3 where it is on “short tracks”). One can observe the scoop gradually opening, reaching the ground and finally collecting the soil. Figure 6 shows the MRPTA in long track configuration during acceptance testing at the CSA Mars Yard. Hundreds of hours have been devoted to verification and integration tests and kilometres of experimental runs have been executed. For scouting type missions, the accuracy of MRPTA observed at the return to the starting point was ~ 2% of the distance travelled for the missions of ~100 meters, executed at average speed of 0.1 m/sec. Obtained accuracy is better than reported by most of the systems relying on dead-reckoning and inertial sensors.



(a) (b)

(c) (d)

Figure 5 – Testing the MRPTA Scoop in the Cohort testing site: (a) - opening the scoop, (b) - start going down, (c) - start cutting the soil, (d) - taking a soil sample.

Mapping capabilities provided by MRPTA, although not as accurate as its positioning, are on a very acceptable level. The traversability estimates are quite sensitive to terrain characteristics and required a lot of tuning; different parameter sets are required for significantly different terrains. The JAUS messaging system proved to be stable and facilitated modular development and testing. We noted that sometimes the JAUS framework would fail to start cleanly, but attributed this to a flaw in our process supervisor implementation and not an issue inherent to JAUS.

CONCLUSIONS AND FUTURE WORKS

The autonomous navigation system developed for the MRPTA rover has been described including software architecture, sensing equipment and navigation algorithms on the level of details allowed by the short length of the paper. Although designed primarily for planetary exploration, our system, due to its modularity and high level of abstraction from the actual hardware, is easily adaptable to the tasks that may arise in such applications areas as construction and mining. Our system does not require GPS positioning and therefore can be used underground or on obstructed construction sites. As compared to the system developed for mining applications in (Bakambu & Polotski, 2007), MRPTA uses more advanced positioning and planning algorithms and has more mapping capabilities.

Adapting the system to the mining or construction environments some of the assumptions made in the context of planetary exploration have to be revisited. Our system, for example, does not consider overhead obstacles (that might appear from suspended objects). Also MRPTA mapping is based on 2 D representation of the environment (with an added height component). Mapping of the underground mines requires more generic 3D representation as discussed in (Huber & Vandapel, 2006) and (Artan at al 2011). Modularity of our system greatly simplifies the development and integration of such additional functionalities. One particular useful system feature is the ability to reset the system position and/or orientation from an external authoritative source. In construction or mining sites it may allow integration with an existing RF-tag, visually augmented or other type of infrastructure-based positioning environment.

Possible system enhancements we would like to consider the integration of stereo–camera-based scene reconstruction (visual odometry) with pose estimation in order to eliminate the effects of platform longitudinal slippage in long range missions. Further development of map representation that is more appropriate for the future use also seems necessary. A possible solution that was not implemented due time constraints might be based on the concept of *atlas* (Lisien et al., 2005; Marshall et al., 2008).

Figure 6 – MRPTA moving across the CSA Mars Yard along the delivery demonstrations

REFERENCES

Artan, U., Marshall, J., & Lavigne, N. (2011). *Robotic mapping of underground mine passageways.* Mining Technology, pp. 18-24.

Alami, R., Chatila, R., Fleury, S., Ghallab, M., & Ingrand, F. *An Architecture for Autonomy.* International Journal of Robotics Research (Special Issue on `Integrated Architectures for Robot Control and Programming', Vol 17, N° 4, Apri1 1998).

Bakambu, J.N., & Polotski, V. (2007) Autonomous *System for Navigation and Surveying in Underground Mines*. Journal of Field Robotics, Vol. 24 (10), pp.829-847.

Biesiadecki, J.& Maimone, M. (2006) *The Mars Exploration Rover Surface Mobility Flight Software: Driving Ambition*, Proceedings of the IEEE Aerospace Conference. March, Montana, USA.

Dupuis, E., Allard, P., Bakambu, J., Lamarche, T., Zhu, W.-H., & Rekleitis , I. (2005) *Towards Autonomous Long-Range Navigation*,"ISAIRAS05" Conference, Munich. Germany.

Howard, A., & Seraji, H. (2000) *Real-Time Assessment of Terrain Traversability for Autonomous Rover Navigation,* Proceedings of the IEEE Int. Conference on Intelligent Robots and Systems.

Huber, D., & and Vandapel, N. (2006) *Automatic Three-dimensional Underground Mine Mapping*, International Journal of Robotics Research, Vol. 25, No 1, pp. 7-17.

Lisien,B., Morales, D., Silver, D., Kantor, G., Rekleitis, I., & Choset, H. (2005) *The Hierarchical Atlas*, IEEE Transaction on Robotics, Vol. 21, No. 3, pp. 473-481.

Marshall, J., Barfoot. T. & Larssson J. (2008) *Autonomous Underground Tramming for Center-Articulated Vehicles*, Journal of Field Robotics, Vol.25 (6-7) pp.400-421.

Matthies, L., Gat, R., Harrison, R.,Wilcox, B., Volpe, R., & Litwin., T. (1995) *Mars microrover navigation: Performance evaluation and enhancement.* Autonomous Robots Journal, Special Issue on Autonomous Vehicle for Planetary Exploration, Vol. 2 No. 4.

Solanki, S. (2007) *Development of Sensor Component for Terrain Evaluation and Obstacle Detection for an Unmanned Autonomous Vehicle* (Doctoral dissertation). University of Florida, (www.cimar.mae.ufl.edu/CIMAR/pages/thesis).