## AUTONOMOUS SYSTEM FOR MOBILE ROBOT NAVIGATION

## Gheorghe LAZEA, Radu ROBOTIN, Cosmin MARCU, Levente TAMAS

Technical University of Cluj-Napoca, Department of Automation 26-28 Baritiu Str., Cluj-Napoca E-mail: {gheorghe.lazea; radu.robotin; cosmin.marcu; levente.tamas}@aut.utcluj.ro

Abstract: Cooperative mobile robots involve accurate robot localization, map-matching and information exchange and collision-free navigation. Autonomous robots must be able to learn and maintain models of their environments. Robots operating in a team must also exchange information and cope with the requirements of a dynamic world. The paper gives results for autonomous exploration, mapping and operation of a mobile robot in populated multi-room environments, part of a mobile team of exploratory robots.

Keywords: mobile robots, autonomous exploration, Kalman filters, sensor integration, navigation

## 1. INTRODUCTION

To efficiently carry out complex missions in indoor environments, autonomous mobile robots must be able to acquire and maintain models of their environments. The problem of acquiring models is difficult and far from being solved. The following factors impose practical limitations on a robot's ability to learn and use accurate models:

*Sensors.* Sensors often are not capable of directly measuring the quantity of interest. For example, cameras measure color, brightness and saturation of light, whereas for navigation one might be interested in assertions such as *"there is a door in front of the robot."* 

*Perceptual limitations.* The perceptual range of most sensors (such as ultrasonic transducers, cameras) is limited to a small range around the robot. To acquire global information, the robot has to actively explore its environment.

*Sensor noise.* Sensor measurements are typically corrupted by noise. Often, the distribution of this noise is not known.

*Drift/slippage*. Robot motion is inaccurate. Unfortunately, odometric errors accumulate over time. For example, even the smallest rotational errors

can have huge effects on subsequent translational errors when estimating the robot's position.

*Complexity and dynamics.* Robot environments are complex and dynamic, making it principally impossible to maintain exact models and to predict accurately.

*Real-time requirements.* Time requirements often demand that internal models must be simple and easily accessible. For example, accurate fine-grain CAD models of complex indoor environments are often inappropriate if actions have to be generated in real-time.

Recent research has produced two fundamental paradigms for modeling indoor robot environments: the *grid-based (metric) paradigm* and the *topological paradigm*. Grid-based approaches, such as those proposed by Moravec and Elfes (1985) and Borenstein and Koren (1991) and many others, represent environments by evenly-spaced grids. Each grid cell may, for example, indicate the presence of an obstacle in the corresponding region of the environment. Topological approaches, such a those proposed by Kuipers/Byun, Mataric or L. Kavraki and J.-C. Latombe (1994), represent robot environments by graphs. Nodes in such graphs correspond to distinct situations, places, or landmarks

(such as doorways). They are connected by arcs if there a direct path exists between them.



Fig. 1. Pioneer 2DX and Pioneer 3AT mobile robots

The robots used in our research are shown in Figure 1. All robots are equipped with an array of sonar sensors, consisting of 8 or 16 sonars. Sonar sensors return the proximity of surrounding obstacles, along with noise. One of these robots (Pioneer 3AT) is also equipped with a video camera, which detects proximity of nearby objects based on their color.

Throughout this paper, we will restrict ourselves to the interpretation of proximity sensors, although the methods described here have (in a prototype version) also been operated using cameras in addition to sonar sensors, using the image segmentation approach described in J. Buhmann *et al.* (1995)

## 2. SYSTEM ARCHITECTURE

This paper describes the three major components of our approach to building a robot system architecture that can cope with the requirements of a cooperative team:

*Interpretation and integration.* Sensor readings are mapped to occupancy values. Multiple sensor interpretations are integrated over time to yield a single, combined estimate of occupancy.

*Position estimation.* The position of the robot is continuously tracked and odometric errors are corrected.

*Exploration.* Shortest path through unoccupied regions are generated to move the robot greedily towards goal.



Fig. 2. System architecture.

A basic exploratory task implies that the robot has some sort of preliminary map, plans its movement strategy based on the available data and starts to explore the world. Most of the time the map may have inconsistencies or the robot discovers new information with its sensors. The gathered data must be correlated with the available one and most of the time the robot must modify its preplanned strategy. If the amount of information is significant and changes in the map are frequent, a brute-force replanner may be inadequate.

The accuracy of the metric map depends crucially on the alignment of the robot with its map. Unfortunately, slippage and drift can have devastating effects on the estimation of the robot position. Identifying and correcting for slippage and drift (odometric error) is therefore an important issue in map building. Maps are the result of exploiting and integrating two sources of information:

*Wheel encoders.* Wheel encoders measure the revolution of the robot's wheels. Based on their measurements, odometry yields an estimate of the robot's position at any point in time. As can be seen from Figure 3 and 4, odometry is very accurate over short time intervals, but inaccurate in the long run.

*Map matching.* Whenever the robot interprets an actual sensor reading, it constructs a "local" map (on Pioneer robots these are stored in LPS – Local Perceptual Space, along other relevant information see ActivMedia (2002, 2006)). The *correlation* of the local and the corresponding section of the global map is a measure of their correspondence. Obviously, the more correlated both maps are, the more alike the local map looks to the global one, hence the more plausible it is.

# 2.1 Odometric error correction using extended Kalman filters.

The location of a robot is defined in a global coordinate system and consists of the x and y coordinates of the center of the robot, the center point, and the orientation  $\theta$  of the robot in that general coordinate system at a certain time k.

$$x_{k} = \begin{bmatrix} x_{[x],k} \\ x_{[\nu],k} \\ x_{[\theta],k} \end{bmatrix}$$
(1)

The position and orientation are updated based on the relative linear and angular robot displacement u, depending on raw data d from the wheel encoders:

$$\begin{cases} x_{[x],k} = f_x \left( x_{k-1}, u_{k-1} \right) = x_{[x],k-1} + u_{[\Delta D],k-1} \cdot \cos \left( x_{[\theta],k-1} + \frac{u_{[\Delta \theta],k-1}}{2} \right) \\ x_{[y],k} = f_y \left( x_{k-1}, u_{k-1} \right) = x_{[y],k-1} + u_{[\Delta D],k-1} \cdot \sin \left( x_{[\theta],k-1} + \frac{u_{[\Delta \theta],k-1}}{2} \right) \end{cases}$$

$$x_{[\theta],k} = f_{\theta} \left( x_{k-1}, u_{k-1} \right) = x_{[\theta],k-1} + u_{[\Delta \theta],k-1}$$

The prediction equations refer to previous state  $\hat{x}_k^$ and uncertainty  $P_k^-$ :

$$\hat{x}_{k}^{-} = f\left(\hat{x}_{k-1}^{+}, \hat{u}_{k-1}\right) + \overline{w}_{k-1}$$
(3)

$$P_{k}^{-} = A_{x,k} P_{k-1} A_{x,k}^{T} + A_{u,k} U_{k-1} A_{u,k}^{T} + Q_{k-1}$$
(4)

with  $W_{k-1}$  an estimate of the noise,  $P_{k-1}, Q_{k-1}$  covariance matrix and  $A_{x,k}, A_{u,k}$  the Jacobian matrix.

The correction equations for previous estimates are given by:

$$\hat{x}_k^+ = \hat{x}_k^- + K_k \left( z_k - h \left( \hat{x}_k^- \right) \right)$$
(5)

$$P_{k}^{+} = \left(I - K_{k}H_{k}\right)P_{k}^{-}(1)$$

$$K_{k} = P_{k}^{-}H_{k}^{T}\left(H_{k}P_{k}^{-}H_{k}^{T} + R_{k}\right)^{-1}$$
(6)

z being the measurement, h() the measurement function and H the Jacobian.

#### 2.2 Occupancy grid and ultrasonic sensors

Sonar interpretations must be integrated over time to yield a single, consistent map. To do so, it is convenient to interpret the network's output for the  $t^{th}$  sensor reading (denoted by  $s^{(t)}$ ) as the *probability* that a grid cell  $\langle x, y \rangle$  is occupied conditioned on the sensor reading  $s^{(t)}$ :

$$\operatorname{Prob}\left(occ_{x,y} | s^{(t)}\right)$$

A map is obtained by integrating these probabilities for all available sensor readings, denoted by  $s^{(1)}, s^{(2)}, ..., s^{(T)}$ . In other words, the desired occupancy value for each grid cell  $\langle x, y \rangle$  can be written as the probability:

$$\operatorname{Prob}\left(occ_{x,y} | s^{(1)}, s^{(2)}, ..., s^{(T)}\right)$$

which is conditioned on *all* sensor readings. A straightforward approach to estimating this quantity is to apply Bayes's rule - see A. Elfes. (1989) for more details. To do so, one has to assume independence of the noise in different readings.

More specifically, given the true occupancy of a grid cell  $\langle x, y \rangle$  the conditional probability  $\operatorname{Prob}(s^{(t)} | occ_{x,y})$  must be assumed to be independent of  $\operatorname{Prob}(s^{(t)} | occ_{x,y})$  if  $t \neq t'$ . This assumption is not implausible—in fact, it is commonly made in approaches to building occupancy grids. It is important to note that the conditional independence assumption does not imply the independence of  $\operatorname{Prob}(s^{(t)})$  and  $\operatorname{Prob}(s^{(t')})$ . The latter two random variables are usually dependent.

The desired probability can be computed in the following way:

$$\operatorname{Prob}\left(occ_{x,y} | s^{(1)}, s^{(2)}, ..., s^{(T)}\right) = \\1 - \left(1 + \frac{\operatorname{Prob}\left(occ_{x,y} | s^{(1)}\right)}{1 - \operatorname{Prob}\left(occ_{x,y} | s^{(1)}\right)} \prod_{\tau=2}^{T} \frac{\operatorname{Prob}\left(occ_{x,y} | s^{(\tau)}\right)}{1 - \operatorname{Prob}\left(occ_{x,y}\right)} \frac{\operatorname{Prob}\left(occ_{x,y}\right)}{1 - \operatorname{Prob}\left(occ_{x,y}\right)}\right)^{-1}$$

The update formula follows directly from Bayes's rule and the conditional independence assumption, with  $\operatorname{Prob}(occ_{x,y})$  the prior probability for occupancy (which was set to 0.5, and can be omitted in this equation).

Notice that, as V. Vapnik (1982) showed last equation can be used to update occupancy values *incrementally*, i.e., at any point in time it suffices to memorize a single value per grid cell:

$$\operatorname{Prob}\left(occ_{x,y} | s^{(1)}, s^{(2)}, ..., s^{(T)}\right)$$

Technically, local maps are computed in local, robotcentered coordinates, whereas global maps are computed in a global coordinate system. As a result, each grid cell of the global map that overlaps with the local map overlaps almost always with exactly four grid cells of the local map. Let  $\langle x, y \rangle$  be the coordinates of a cell in the global map which overlaps with the local map, and let  $\langle x', y' \rangle$  denote the corresponding coordinates in the local map. Let  $\langle x_i, y_i \rangle$  with i = 1...4 denote the coordinates of the four grid points in the local map that are nearest to  $\langle x', y' \rangle$ . The global occupancy  $occ_{x,y}$  is then matched with the local occupancy value obtained using the following interpolation:

$$\frac{\sum_{i=1}^{4} |x - x_i| |y - y_i| lococc_{x_i y_i}}{\sum_{i=1}^{4} |x - x_i| |y - y_i|}$$

where  $lococc_{x_iy_i}$  denotes the local occupancy grid. In

other words, the coordinate of a global grid cell is projected into the local robot's coordinates, and the local occupancy value is obtained by interpolation. The interpolating function is similar in spirit to Shepard's interpolation (1968).

## 2.3 Goal acquisition methods

Our research was focused on global planner algorithms derived form graph search theory. Several experiments were conducted in order to find the best trade-off between path efficiency (in terms of distance traveled) and requirements of a dynamic environment. We have conducted several experiments using a greedy,  $A^*$  and  $D^*$  algorithms for global navigation.

To autonomously acquire maps, the robot has to explore. The idea for (greedy) exploration is to let the robot always move on a minimum-cost path to the nearest unexplored grid cell. The cost for traversing a grid cell is determined by its occupancy value. Initially, all unexplored cells are 0 while the explored ones have a large value (*inf*). Next the values of all explored grid cells are updated by the value of their best neighbors, plus the costs of moving to this neighbor. Cost is here equivalent to the probability that a grid cell is occupied. To determine where to explore next, the robot generates a minimum-cost path to the unexplored.

It is not the aim of this paper to present the algorithms or their implementation. More detailed information about A\* and D\* and their implementation can be found in Stentz (1995, 1996). Our experiments focused on comparison between different planners implemented on a real robot in terms of number of states processes, planning time and replanning time.

#### 3. TESTS AND RESULTS

Figures 3 and 4 give examples that illustrate the importance of position estimation in grid-based robot mapping. For example, in Figure 3a the position is determined solely based on dead-reckoning and obviously, the resulting map is too erroneous to be of practical use.

In another run, after approximately 15 minutes of robot operation, the position error is approximately 11.5 meters while in the case of odometric error correction, after approximately 15 minutes of robot operation the position error was about 3.5 meters. Though this error may still seem large, it proves the advantages of using Kalman filters over raw sensor data.



Fig. 3. Odometric error for a 2000x2000 square run.



Fig. 4. Results in the case of odometric error correction.

Second sets of tests involve the planning/replanning strategy. Two types of algorithms were under tests: A\* and D\*. Although in simulator implementation the two of them will produce optimal paths in terms of distance traveled, in real-time implementation on a Pioneer mobile robot D\* gains a little advantage. The tests were conducted on map consisting of 20x20cm cells, having similar configurations.

The first thing we will compare is the planning time. In figure 5 can be seen how the planning times evolves for both algorithm, depending on the surface. The planning time for  $A^*$  is much lower than the planning time for  $D^*$ , that leads to a first conclusion: that in known maps is better to use  $A^*$  algorithm.

In figure 6 is presented the comparison between the replanning times. As one can see,  $D^*$  is very fast in the replanning part, and that  $A^*$  doesn't fit for unknown maps from replanning point of view.

From the number of states processed one can see the comparison in figure 7, showing that  $D^*$  is faster compared to  $A^*$ .  $D^*$  processed more states during the planning time, while during replanning the number of states processed was lesser. This is especially important in real-time operation since changes in the map according to sensor data interpretation may dramatically change the configuration of a map.



Fig. 5. Comparison between planning times.



Fig. 6. Comparison between the replanning times.



Fig. 7. Comparison between the number of states processed and evolution of replanning with surface.

### 4. CONCLUSIONS AND FUTURE WORK

The aim of our research is to develop a navigation system for mobile robots capable of coping with the requirements of cooperative robot team. This paper focused on three of the key problems in such an approach: odometric error correction, sensor data integration and planning algorithms. We have tested error correction using extended Kalman filters and the results are encouraging, however the presence of position errors leads to the conclusion that further work in this field is required (landmark detection, better map-matching using stereovision or trinocular systems). Our future plans also include data fusion from multiple sensorial systems (laser rangefinder, ultrasonic sensors, vision system) and better mapmatching approaches.

## REFERENCES

- Borenstein, J. and Koren. Y. (1991). The vector field histogram – fast obstacle avoidance for mobile robots. *IEEE Journal of Robotics and Automation*, 7(3):278–288, June.
- Buhmann, W J.. Burgard, A. B. Cremers, D. Fox, T. Hofmann, F. Schneider, J. Strikos, and S. Thrun. (1995). The mobile robot Rhino in *AI Magazine*, vol 16(1).
- Elfes, A. (1989). Occupancy Grids: A Probabilistic Framework for Robot Perception and Navigation. PhD thesis, Department of Electical and Computer Engineering, Carnegie Mellon University.
- Engelson, S. and D. McDermott. (1992). Error correction in mobile robot map learning. In *Proceedings of the 1992 IEEE International Conference on Robotics andAutomation*, pages 2555–2560, Nice, France.
- Kalman, R.E. (1960). A new approach to linear filtering and prediction problems. *Trans. ASME, Journal of Basic Engineering*, Vol 82 pages.35– 45
- Kavraki, L. and Latombe J.-C. (1994). Randomized preprocessing of configuration space for fast path planning. In *IEEE International Conference* on Robotics& Automation, pages 2138–2145, San Diego.
- Kortenkamp, D. and Weymouth, T. (1994). Topological mapping for mobile robots using a combination of sonar and vision sensing. In *Proceedings of the Twelfth National Conference* on Artificial Intelligence, pages 979–984, Menlo Park, AAAI, AAAI Press/MIT Press.
- Moravec, H. P. and Elfes, A. (1985). High resolution maps from wide angle sonar. In *Proc. IEEE Int. Conf. Robotics and Automation*, pages 116–121.
- Shepard, D. (1968). A two-dimensional interpolation function for irregularly spaced data. In 23rd National Conference ACM, pages 517–523.
- Stentz, A. (1995). The Focussed D\* Algorithm for Real-Time Replanning, International Joint Conference on Artificial Intelligence.
- Stentz, A. (1996). Map-Based Strategies for Robot Navigation in Unknown Environments, AAAI Symposium on Planning with Incomplete Information for Robot Problems.
- Vapnik, V. (1982). *Estimations of dependences based* on statistical data. Springer Publisher.
- \*\*\* ActivMedia, (2002) Pioneer 2 mobile robots, Operations Manual.
- \*\*\* ActivMedia, (2006) Pioneer 3 mobile robots, Operations Manula.