

SIMULATION METHODS FOR TRAFFIC CONTROL AND POSITION ESTIMATION OF MOBILE ROBOTS

Miklós VOGEL, Ferenc VAJDA and László VAJTA

Department of Control Engineering and Information Technology
Budapest University of Technology and Economics
H-1117 Budapest, Hungary

Received: Nov. 30, 2000

Abstract

In this paper we present a simulation approach in the field of landmark-based mobile robot navigation. A method is shown to involve statistical traffic analysis for determining critical positioning accuracy limits in certain points of the environment. We also describe an algorithm to evaluate spatial uncertainties assuming a navigation strategy. This strategy uses dead-reckoning frequently updated by absolute position measurements. We have implemented a simulation software to check the effectiveness of the new algorithms. The methods and simulation results are explained through simple examples.

Keywords: mobile robot, navigation, traffic flow, landmark, multi-agent, dead-reckoning error.

1. Introduction

There are several optimisation problems in the field of landmark based autonomous robot navigation. The fields of optimal trajectory planning and optimal landmark arrangement can be considered as problems with multiple variables and parameters. To create new algorithms and test their goodness, simulation methods should be used. Simulation is also essential to test navigation strategies of concurrent multi-agent systems which are much more complex than those for single robots.

In this paper we present methods for traffic analysis and simulation of landmark based positioning. Planning landmark arrangements in a given environment is the field where both methods can be effectively used. The required position accuracy at a certain point of the environment may depend on the traffic load. The landmark arrangement and the current speed have to accord to the determined error limits.

We also present simple examples to show how the effects of modified parameters can be analysed.

2. Introducing the Statistical Analysis of Traffic Flow

One of the most important questions in the course of mobile robots' research is navigation of multi-agent systems. One of its main fields is the communication between

robots. They communicate directly or by the use of a central control-system. The other focus is a route-control based on traffic load statistics examined in navigation space. An additional problem is the optimal landmark arrangement. Solving this problem we have the possibility to measure more accurate position and orientation values. There is a detailed literature available concerning communication tasks. In our paper we discuss traffic statistics of mobile robots in simplified working environments.

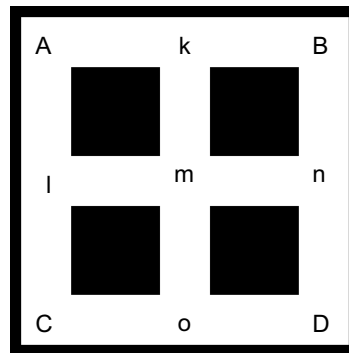


Fig. 1. The workspace of analysed traffic flow

To get closer to this question on traffic statistical basis, we should start with a reduced system. We examine a traffic flow first where our robots are moving only between two docking nodes. Our workspace can be seen in *Fig. 1*. There are four docking points (A, B, C, D). Let's choose all the possible routes $A - C$ and $A - D$ analysing their traffic loads. If we had only a single-agent system, we would use always $A - l - C$ route. In the case of more robots working concurrently sometimes we have to use longer routes. The statistical probability of choosing a longer path is less than that of the shortest one, of course.

There are some additional parameters of road-sections. The speed of movement can be limited by the width of the road-section; it may have smaller priority in crossings so the robot has to slow down or stop etc. All these limitations and lengths can be included in the weight values of a weighted graph. The probability distribution can be examined using these values.

Let's consider the segment between two nodes as an edge of a graph weighted with the length of the route-section and additional limitations. We can get the whole way's length adding the weights along the edges of the graph. Of course, we would choose the shortest of them, but if it is over-loaded, we have to choose another way to reach our target. If we can't go on the shortest way, we have to use the shortest of the others. So we use long ways quite rarely.

Our first criterion is to get the shortest way. A graph $G = (V, E)$ is defined, where V is the set of all nodes and E is the set of all edges. There are two preferred nodes $s, t \in V$, where s is the starting and t is the target node. In the case when the

edges aren't weighed – all the road-sections have the same length –, searching for the shortest way is a simple problem which can be solved by a BSF (Breadth First Search) algorithm. The solution is more difficult in the case of weighed edges. For example, we try to get from s to t as shown in the following figure.

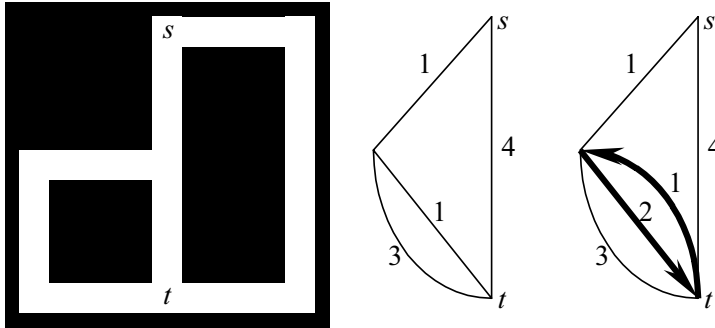


Fig. 2. Searching for shortest way – weighed graphs

If we have limitations depending on priorities at crossings, we have to use directed graphs with different weights from x to y , and vice versa (figure above).

To solve this problem we should use the algorithms of Ford or Dijkstra. We assign a length d to all nodes in the graph, which determines the shortest way to this point from s . The initial condition of the algorithms:

$$d(s) = 0, \quad d(u \neq s) = \infty. \quad (1)$$

The principle of both algorithms is the following:

if there is a way e from x to y and

$$d(y) > d(x) + l(e), \quad (2)$$

then

$$d(y) = d(x) + l(e),$$

where $l(e)$ is the length of edge e . To solve the shortest way problem, both algorithms (Ford, Dijkstra) are suitable. To choose the better we have to know the number of nodes and edges.

Algorithm of Dijkstra
step 0: $S \leftarrow \{s\}, T \leftarrow V - \{s\}, (1), u_0 = s$
step 1: (2) for edges e from u_0 to $x \in T$
step 2: $u_0 \leftarrow x \in T, d(x)$ minimal, $S \leftarrow S + \{u_0\}, T \leftarrow T - \{u_0\}$
step 3: if $T = \{\}$ STOP, else step 1
Algorithm of Ford
step 0: numbering edges $(1 - e), (1), i \leftarrow 1$
step 1: (2) for all edges as numbered
step 2: $i \leftarrow i + 1$, if $i > v$ STOP, else step 1

Now we can determine the shortest way to our target. What can we do if we want to choose another one? We can go on another road section and at the next node we search for the shortest way. We can evaluate statistical probabilities in all nodes, so we can choose a way. We have to calculate the shortest way in the direction of all road-sections.

At a given node u we have to decide on which way to go. We have to determine the length of road-sections to all neighbour nodes ($K(u)$), and also the shortest way to target from $K(u)$. If we add them we can get $d_{u,i} = d(u, t, x_i \in K(u))$, where $i \in 1 - n(u)$, the number of neighbour nodes of u .

$$d_{u,i} = ssw(x_i \in K(u), t) + l(e_{u,x_i}),$$

where ssw is the function to search for the shortest way. We can compute the statistical probabilities of choosing road-sections according to

$$p_{u,i} = \frac{\frac{1}{d_{u,i}}}{\sum_{j=1}^{n(u)} \frac{1}{d_{u,j}}}.$$

These probabilities have to be multiplied to determine the probability of getting from s to t on a given way:

$$P_{\mathbf{v}} = \prod_{i=1}^{v(\mathbf{v})} p_{u_i, l_{u_i}} = \prod_{i=1}^{v(\mathbf{v})} \left(\frac{\frac{1}{d_{u_i, l_{u_i}}}}{\sum_{j=1}^{n(u_i)} \frac{1}{d_{u_i, j}}} \right).$$

Now we can determine the traffic load of a given node or an edge using the above result. We get it so that we add all the statistical probabilities on this node or

edge. The traffic load of the start and target point is 1, of course. The traffic load probability of a node is given with the following equation:

$$\xi_u = \sum_{\mathbf{v}_{u,k}} P_{\mathbf{v}_{u,k}} = \sum_{\mathbf{v}_{u,k}} \left[\prod_{i=1}^{v(\mathbf{v})} \left(\frac{1}{\frac{d_{u_i,1_{u_i}}}{n(u_i)} \sum_{j=1}^1 \frac{1}{d_{u_i,j}}} \right) \right],$$

where $\mathbf{v}_{u,k}$ are the ways from s to u . We can get the traffic load probability of an edge s , determining all the $P_{\mathbf{v}u}$ probabilities to the two neighbouring nodes multiplied with the probability of the edge after \mathbf{v} . Adding these probabilities we get:

$$\xi_e = \sum_{\mathbf{v}_{u_1,k}} \pi_{u_1,k} P_{\mathbf{v}_{u_1,k}} + \sum_{\mathbf{v}_{u_2,k}} \pi_{u_2,k} P_{\mathbf{v}_{u_2,k}}.$$

We have to give additional limitations by determining loads of edges and nodes. We can specify that the robot should not go back on the same way. It is also practical to specify not to use the same road-section twice. This would be a loop without any sense. If we do not take these restrictions in account, the evaluation of probabilities is also much more complicated, because of the endless loops. Then we should use a recursive solution, where we define an ε probability value as a limit for stopping the evaluation.

Considering the restrictions we confront with other problems. We have to redefine the method of the shortest way calculation. The result of ssw will be the shortest way, which goes only on road-sections not used previously. The easiest way to solve this problem is to modify the graph of ssw , and to remove the used edges. It is important not to remove any of the nodes.

Let us examine the problem discussed above. In the simplest case we navigate between two docking points. Let us see the load distribution in two cases. In the first figure we represent the way $A - C$. The shortest way has the highest load, of course, which can be seen in the figure. Another important fact is that the loads are higher at nodes.

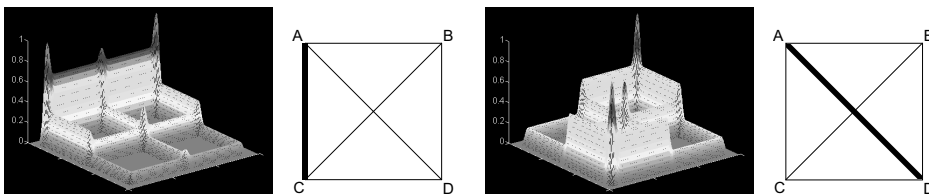


Fig. 3. Result of analysis (two docking points)

Let us see the behaviour of the loads in such environments. Robots can move from any docking nodes to any other one. The probabilities of moving from a node

to another are equal. We can realise that the highest load is concentrated at the middle crossing.

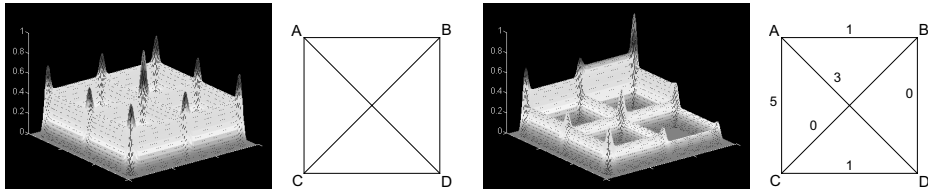


Fig. 4. Result of analysis (four docking points)

The statistic distribution is not necessarily equal between two docking nodes. Let us suppose that there is a manufacturing node in *A* which produces mostly for *C*, but its products are used also in *B* and *D*. There is a small traffic also between *C* and *D*. The statistic distribution can be seen in *Fig. 4*.

As mentioned above, the statistical analysis of traffic loads gives the possibility to determine the critical points where position should be evaluated more accurately. The landmark density at a given position must be higher at such critical points. For example, now we can state that the landmarks should be placed in crossings first of all. Summarised, we have to place landmarks considering the statistical probability of the traffic load. In the following section we present a method to simulate the behaviour of position uncertainties depending on the locations relative to the landmarks.

3. Simulated Position Uncertainties along a Planned Path

An autonomous mobile robot has to evaluate its current position in its working environment. Most of the autonomous vehicles are equipped with several internal sensors (gyroscopes, odometers, accelerometers...) for relative position measurements. The navigation strategy, which can be realised based on these relative measurements is called 'dead-reckoning'. This strategy assumes that the initial point of the navigation is definite and the position error remains acceptable until the robot reaches the next point with known coordinates (goal or sub-goal) in the environment. The main characteristic feature of relative positioning methods based on internal sensors is the accumulating position error. The navigation system has to evaluate the error of position estimation and if it exceeds a certain limit, it has to cancel it by some kind of absolute measurement. A practical method to do this is to update the position by landmark measurements. In this case the robot is provided with a landmark database which helps to detect and identify the available landmarks. Not only artificial markers but also features of the environment detectable by the robot's sensors can be used as landmarks. Landmark based positioning systems can be very different in physical realisation but the position evaluation is usually carried out using triangulation or trilateration. The more landmarks around the current

position, the better for the robot to recognise its location. Because of the limitation of processing power on board the robot, it is not always possible to measure plural landmarks simultaneously, even though there are many landmarks available. In this situation the system has to select a landmark to measure out of the landmarks available from the current position. A control system using dead-reckoning updated with landmark measurements usually integrates the following tasks:

- position estimation based on sensor parameters and kinetic model,
- evaluation of estimation error,
- selection among the available landmarks,
- position evaluation based on landmark measurements,
- velocity optimisation to obtain the required positioning accuracy.

We developed a simulation-software to study the behaviour of systems based on the above principles. Our simulation results gave valuable information about such systems in general although we chose our kinetic and sensor models according to the latest developments in our laboratory. The sensor system in our model consists of an odometer as internal sensor and a special laser range-finder – the LABrador – as global positioning sensor. The Landmark Based Random Deflected Optical Range-finder (LABrador) is a positioning sensor developed at the Mobile and Microrobot Group of the Department of Control Engineering and Information Technology (BUTE). This onboard sensor is able to detect, capture and track reflective landmarks on the way. It can continuously measure the distance of the selected landmarks.

Our model assumes that the robot tracks two point-type landmarks and measures their distances with a certain sample period. We used the simulation to estimate the position uncertainties along the theoretical path in function of current location relative to the landmarks. We analysed the behaviour of position uncertainties at different velocities if the sample interval of the distance measurement remains constant.

The simulation can give help to optimise landmark arrangements and estimate critical velocities along the planned trajectory. The critical values of acceptable position error limits should be determined during the trajectory planning.

3.1. Estimation of Dead-Reckoning Errors

Dead-reckoning methods can be of many kinds. The most widely used navigation method for mobile robot positioning is odometry. It provides good short-term accuracy and allows very high sampling rates at low prices. Odometry means integration of incremental motion information over time, which leads to accumulation of errors. These errors increase proportionally with the travelled distance. The errors can be of systematic and non-systematic types. The errors caused by unequal wheel diameters or misalignment, for example, can be considered as systematic errors. Main non-systematic errors are caused by wheel slippage and by travel over

unexpected objects. Systematic errors accumulate constantly and on most smooth indoor surfaces non-systematic errors can be ignored on lower distances [2]. In our simple model of dead-reckoning performance we made simplifications and we took only predictable systematic errors into account. We rewrite here the well-known equations for odometry [KLARER 1988; CROWLEY and REIGNER 1992]:

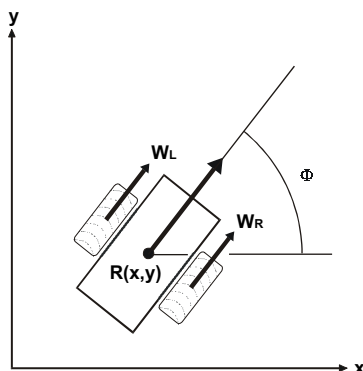


Fig. 5. Differential drive

c = conversion factor: lin. displacement/encoder pulses

D = nominal wheel diameter

g = gear ratio

The incremental travel distances for left and right wheel:

$$\Delta W_L = cN_L,$$

$$\Delta W_R = cN_R.$$

The displacement of the robot's center-point is calculated according to:

$$\Delta R = (\Delta W_L + \Delta W_R)/2.$$

The incremental change of orientation:

$$\Delta \Phi = (\Delta W_R - \Delta W_L)/d,$$

where d is the wheel-base of the vehicle. The robot's new co-ordinates and the value of current orientation can be computed according to

$$\Phi_i = \Phi_{i-1} + \Delta \Phi_i,$$

$$x_i = x_{i-1} + \Delta R_i \cos \Phi_i,$$

$$y_i = y_{i-1} + \Delta R_i \sin \Phi_i.$$

The system in our model is equipped with a landmark-based absolute positioning device. The frequency of landmarks is determined based on the worst-case

systematic errors of the odometry. Many researchers have developed algorithms that estimate the worst-case position uncertainty of a moving robot [KOMORIYA and OYAMA 1992]. In these approaches the computed position is surrounded by a characteristic ‘error ellipse’. This ellipse grows with the travel distance until an absolute measurement resets its size.

We can modify the odometry equation involving the error derived from the wheel speed (ignoring other errors):

$$\begin{aligned}\bar{X}_{i+1} &= f_i(\bar{X}_i, \bar{W}_i + \bar{E}_i), & \bar{E}_i &= \text{Error}, \\ \bar{X}_i &= \begin{bmatrix} x_i \\ y_i \\ \Phi_i \end{bmatrix}, \\ \bar{W}_i &= \begin{bmatrix} W_R \\ W_L \end{bmatrix}.\end{aligned}$$

The distribution of estimation error with certain existing probability forms an ellipse in the $X - Y$ plane [1]. It grows its size and changes its shape as the robot moves.

We simulate this error increasing only the standard deviation of the position uncertainty between two points of absolute measurement. In these points we update the robot position with the information obtained by measuring landmarks. The shape of the simulated error distribution is kept constant until the next measurement, but its standard deviation is incremented proportionally with the travelled course. The shape of the distribution is calculated again at the next control point.

3.2. Simulated Landmark Measurements

Let us assume a two-dimensional environment where we use point type landmarks. Our theoretical system is able to measure the distances of two landmarks at the same time (or almost the same time). The relative orientation of the landmarks is used only for verification but not directly for position evaluation. In this case we consider only two distance values. We also assume that the error of distance measurement is quasi distance-independent. This condition is almost true for a laser range-finder.

The distance measurements are performed equidistantly in time. As a result we get two values after each measurement. The statistical parameters of the measurement can be given if they are known. During our simulations normal distribution was supposed.

A circular boundary can be given around both selected landmarks. The intersection of these boundaries gives the estimated location of the robot. To each point of this area we can order a value, which corresponds to the probability that the robot is located there. The simulation software calculates the co-ordinates of four intersection points with a given worst case error and interpolates between these marginal points applying a two-dimensional distribution function (in our case it is a 2D normal distribution).

The planned path is stored as a spline. After giving the control points of the path and the co-ordinates of the landmarks the time parameter should be determined. As the sample period of the landmark measurement is constant, the adequate points of the trajectory should be defined where these measurements are performed. As the first step the statistical distribution of the position is evaluated in these points. The next operation is the interpolation considering the estimated odometer errors. The result is presented in form of a map matrix. The elements of the matrix give the probability of crossing the corresponding area unit with a realised trajectory. It can be visualised easily as a 2D image.

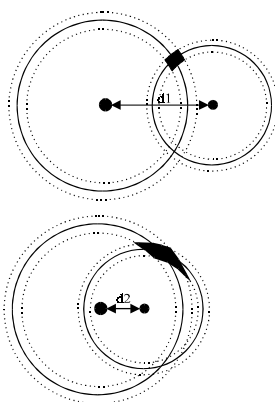


Fig. 6. Effect of two markers not far enough from each other

3.3. Simulation Example

In this paper we present a simple example. The robot travels through a constriction caused by two obstacles. Two landmarks are placed at corners of the mentioned obstacles.

In Fig. 7 we have displayed only the distribution of robot positions in each point where measurements were performed. In the figure we can see how the shape of the spatial uncertainty changes along the course. The width of the ellipses perpendicular to the robot motion decreases approaching the obstacles. Besides this the radius in the direction of motion increases. In real systems a smaller radius in all directions of the ellipse (smaller standard deviation) is one of the criteria for the landmark selection. The tolerance perpendicular to the course is usually tighter for the robot motion than in the direction of the motion. The effective width in each point can be easily calculated determining a criterion for triggering landmark measurements.

In Fig. 8 we show the effect of dead-reckoning errors. The speed is constant on the whole path. The uncertainty grows with a certain rate until the moment of

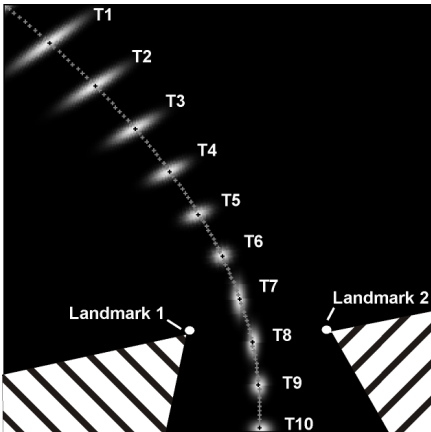


Fig. 7. Position uncertainties

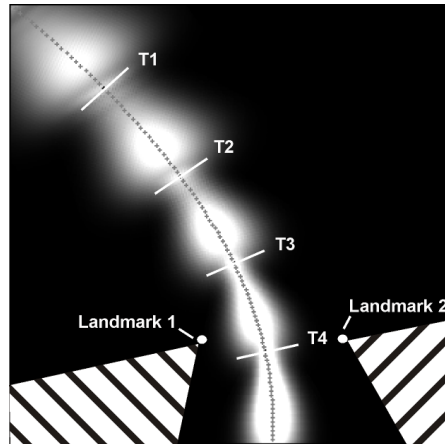


Fig. 8. Dead-reckoning

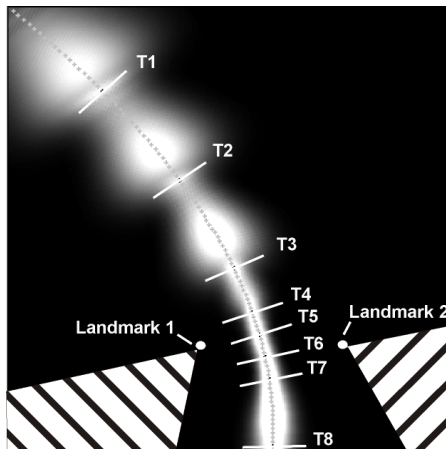


Fig. 9. Effect of slowing down

the next absolute measurement. It's obvious that the sample rate might be critical at narrow places. If the sample rate is relatively slow – but the sampling can be triggered – optimal points of measurement should be determined using continuous uncertainty estimation.

The third figure shows the effect of slowing down. The velocity parameter offers a good opportunity to minimise uncertainties if sample rate is limited. In an inverse interpretation we can determine velocity limits not to exceed a given limit of position error.

4. Conclusion

The results of this paper are summarised as follows:

A method is proposed to involve statistical traffic analysis for determining critical positioning accuracy limits in certain points of different working environments.

An approach is described to evaluate spatial uncertainties assuming a navigation strategy, which uses dead-reckoning updated frequently by landmark measurements.

Both methods were implemented in simulation software to check the effectiveness of proposed algorithms. Some of the simulation results are also presented.

Acknowledgement

Support for the research is provided by the Hungarian National Research Program under grant No. OTKA T 029072.

References

- [1] KOMORIYA, K. – OYAMA, E. – TANI, K., Planning of Landmark Measurement for the Navigation of a Mobile Robot, *Proc. of IEEE /RSJ International Conference on Intelligent Robots and Systems*, 1992.
- [2] BORENSTEIN, J. – EVERETT, H. R. – FENG, L., Where am I? (Sensors and Methods for Mobile Robot Positioning), University of Michigan, 1996.
- [3] WATANABE, Y. – YUTA, S., Estimation of Position and its Uncertainty in the Dead Reckoning System of the Wheeled Mobile Robot, *Proc. of 20th ISIR*, 1990.
- [4] ARATÓ, P. – VAJTA, L. – FELSO, G., Visual Sensors for Microrobot Systems, *Proc. of World Multiconference on Systemics, Cybernetics and Informatics*, 7–11.06.1997, Venezuela.
- [5] VAJTA, L., The Realisation of a Hierarchical Sensor System for Mobile Robot Positioning, in the *Proc. of the TEMPUS INTCOM Symposium*, Miskolc, 1998.
- [6] VAJDA, F. – VAJTA, L. – VOGEL, M., LABrador – a Hierarchical Sensor System for Mobile Robot Navigation, *Proc. of the INES'99 International Conference on Intelligent Engineering Systems*, 1999, Slovakia.
- [7] KOLLER, S., *Traffic Techniques and Transport Planning*, Budapest, 1986.