P. Hoppenot, E. Colle: "Real-time mobile robot localisation with poor ultrasonic data" - 3rd IFAC Symposium on Intelligent Component and Instrument for Control Application (SICICA), pp. 135-140, 9-11 June 1997.

REAL-TIME MOBILE ROBOT LOCALISATION WITH POOR ULTRASONIC DATA.

P. Hoppenot, PhD ; E. Colle, Professor.

CEMIF, Systèmes Complexes, 40 rue du Pelvoux, 91020 Evry Cedex (France)
Tel : Colle [(+33-1) 69-47-75-57], Hoppenot [(+33-1) 69-47-75-04]
email : Hoppenot, Colle,...@cemif.univ-evry.fr

Abstract : The research in autonomous mobile robot is enlarging upon low cost mobile robotics. This low cost constraint implies the use of a poor perception system and a low computing power. In such a context, algorithms have to be simple in order to be executed in real time and proof against the weaknesses of the sensing system. The solutions proposed for the localisation are based on the fact that the higher the localisation algorithm speed is the lower the error on the position and on the orientation due to the odometry is.

Keywords : Mobile robots, Ultrasonic transducers, Range data, Least-square algorithm, Position estimation, Position errors.

1.INTRODUCTION

The mobile robot displacement requires the correct knowledge of the position and the orientation given by the localisation function. The classical approach uses proprioceptive device such as an odometer for the relative localisation and a more complex exteroceptive system for regularly correcting the previous information. Many authors have studied the localisation problem in a known and structured environment with an advanced but expensive perception system such as a laser range finder or camera(s). Therefore new mobile robot applications will emerge on condition of a hard reduction of the cost.

It is the case of this application which deals with medical robotics aid and consists in providing a carriage and manipulation assistance to severely disable people. The system is composed of a mobile robot which plays the part of a manipulator arm carrier. The low cost constraint implies to drastically reduce and simplify the perception system. This concerns the sensors in a part and the computing power in the other part. Among the range finders, ultrasonic sensors respect low cost constraint but present several significant sensing problems. The specular reflection involves a no-orthogonal surface to the direction of acoustic propagation and are not detectable. Multiple reflections produce erroneous measures. Such a poor perception system requires algorithms proof against erroneous measures and low time-consumer.

This paper presents robust solutions for the localisation based upon real time and iterative algorithms. The methods are evaluated by experimental results.

2.THE LOCALISATION PROBLEM

Many works have been carried out on this subject. In fact, this problem is very close to the map building one. Most publications deal with both of them. The main issue, according to different authors, is to match measures with a known object of the environment. Indeed, once this matching is found, the only remaining work is to compute the best position of the robot with regard to these matchings.

Different techniques have been presented in the literature. In (Crowley, 1989 ; Kröse *et al.*, 1993) a measured point is associated with a known segment if the distance between them is less than a predefined threshold. The environment being built up during the robot movement, the new point is used to precise the position of the segment. In (Cox, 1991), the environment is a priori known. A measure is matched with the closest known segment. This approach doesn't need sensor model. The computation is not very complex and is made in real time. Another solution consists in using grids to represent the environment (Elfes, 1986 ; Elfes, 1990 ; Elfes and Matthies, 1988). At each measure is associated one cell of the grid and the matching with the environment is naturally made. In (Mandelbaum and Mintz, 1993), the environment is modelled with several segments and a grid. The grid is used for the matching. Each cell contains information (occupied or empty) for the obstacle avoidance for instance. A pointer to a list of features is associated to each cell. Higher level tasks as localisation need more precise information to be performed. A measure belonging to a cell is directly matched with the feature pointed by that cell. The drawback of the method is the high density of basic information to introduce in the model of the environment. In (Crowley and Schiele, 1994), another use of both segments and grids is based on several local maps and a global one. Each map can be represented by a grid or a set of segments. The main point is to match one local map with the global one. Four cases are encountered : segment-segment, segment-grid, grid-segment and grid-grid.

In this article, the approach is based on Cox's (1991) work. Each measure is matched with the nearest segment of the environment. If the minimum distance found in higher than a predefined threshold (here 25 cm) the measure is not matched. This pre-processing simplifies highly the following of the algorithm.

The method allows the real time calculation, the iterative implementation and the taking into account of the odometry. The main hypothesis is that the higher the localisation algorithm speed is the less the position given by the odometer erroneous is. The mobile robot position being refreshed regularly, the absolute localisation algorithm only has to correct the low odometric errors. Due to the low cost constraint imposed by the application the perception system is composed of an eight ultrasonic sensor ring. Contrary to the quite good Cox's laser measures, ultrasonic sensor provides many erroneous measures among a set of data due to specularity, multiple echoes and large solid angle (Wilkes *et al.*, 1993). In short the localisation algorithm has to respect real time constraint depending on the maximum robot speed and has to present a high insensitivity to erroneous and inaccurate measures.

3.THE DIFFERENT ALGORITHMS

3.1.Pin-point algorithm

Presentation of the algorithm. Position (x, y, θ) is given by the odometry. Distances are measured by the 7 forward sensors of the ring (Fig. 2). When the robot moves in a known environment the best robot position minimises the sum of the square measured distances to the walls. The function to minimise is complex if all the possible robot positions are considered. Every measure can be matched with walls belonging to the known environment. The computing complexity is high.

Here, the matching choice simplifies the problem. Indeed, each measure is matched with the nearest wall of the environment, the odometry being considered as not too bad. That leads to the following algorithm :

1- measuring the seven distances
2- finding the impact
3- matching impact with the environment
4- computing the gradient
5- correcting the position
6- estimating the cost function F
  6-1-if $F < \varepsilon$, or $\partial F < \varepsilon'$ or $n > N$ goto 7
  6-2-else goto 2.
7-if $n > N$, giving back the original position.
  7-1-if $\partial F / \partial x < \varepsilon'$ using x correction
  7-2-if $\partial F / \partial y < \varepsilon'$ using y correction
  7-3-if $\partial F / \partial \theta < \varepsilon'$ using θ correction
8- else using the global correction

where F : cost function,
∂F : $\partial F / \partial x$ or $\partial F / \partial y$ or $\partial F / \partial \theta$,
n : iteration number,
N : maximum of iteration.

Cost function. Let $P(x, y, \theta)$ be any position of the robot. Let $M_i(x_i, y_i, \theta_i)$ be any impact of ultrasonic sensor number i where θ_i direction of shooting in the robot reference. So

$$\begin{cases} x_i = x + d\cos(\theta + \theta_i) \\ y_i = y + d\sin(\theta + \theta_i) \end{cases} \quad (1)$$

where d is the measured distance.

Let S, supported by the straight line of equation $ax + by + c = 0$, be the segment matched with M_i. Let $S1(x_1, y_1)$ and $S2(x_2, y_2)$ be the two extremities of S. Let $H(x_h, y_h)$ be the orthogonal projection of M_i on the straight line supporting S. So

$$\begin{cases} x_h = \left(b^2 x_i - aby_i - ac\right) \big/ \left(a^2 + b^2\right) \\ y_h = \left(a^2 y_i - abx_i - ac\right) \big/ \left(a^2 + b^2\right) \end{cases} \quad (2).$$

Then F is defined as follows :

- if H belongs to S then $F_i = \dfrac{\left(ax_i + by_i + c\right)^2}{\left(a^2 + b^2\right)}$ (3)

which is the square distance between M and H,

- else $F_i = \min \begin{cases} \left(x_i - x_1\right)^2 + \left(y_i - y_1\right)^2 \\ \left(x_i - x_2\right)^2 + \left(y_i - y_2\right)^2 \end{cases}$ (4)

which is the minimum square distance between M and S_1 or S_2.

It is easy to see that F_i is continuous everywhere in x_i and y_i. As x_i and y_i are continuous in x, y and θ, F_i is continuous everywhere in x, y and θ.

About the derivatives of F_i only the junction between the two parts has to be verified. Let calculate the derivative of each form of F_i with regard to x_i when H equals S_1. The first form (3) gives $\dfrac{\partial F_i}{\partial x_i} = \dfrac{2a\left(ax_i + by_i + c\right)}{\left(a^2 + b^2\right)}$ and the second (4) gives $\dfrac{\partial F_i}{\partial x_i} = 2\left(x_i - x_1\right)$. Now, as H equals S_1, (2) gives $\begin{cases} x_1 = \left(b^2 x_i - aby_i - ac\right) \big/ \left(a^2 + b^2\right) \\ y_1 = \left(a^2 y_i - abx_i - ac\right) \big/ \left(a^2 + b^2\right) \end{cases}$.

So $\left(x_i - x_1\right) = \dfrac{a\left(ax + by + c\right)}{a^2 + b^2}$ and F_i is derivable everywhere with regard to x_i. The derived forms show that $\dfrac{\partial F_i}{\partial x_i}$ is continuous too. The same result is true for $\dfrac{\partial F_i}{\partial y_i}$.

The final point is to derive F_i with regard to x, y and θ. Now x_i and y_i are continuously derivable in regard x, y and θ (1). So F_i is continuously derivable with regard to x, y and θ.

As $F = \sum_i F_i$ it is continuously derivable with regard to x, y and θ.

The minimisation method. Using F and all its derivatives, the minimisation method is a classical least square algorithm. Here, $\varepsilon = \varepsilon' = 5.10^{-3}$, and $\alpha_x = \alpha_y = \alpha_\theta = 10^{-1}$ in the beginning (backpropagation coefficients).

3.2.Pin-point algorithm with memory effect

In this case the last ten measures of the seven sensors of the ring are memorised in order to build segment features. Then the segments are matched with the modelled environment.

How to build the segments ? The first step is to choose which segment measure belongs to. For example in (Crowley, 1989) and (Kröse *et al.*, 1993) a new segment is built when three points are aligned with a certain predefined tolerance. A new point belongs to a segment if the distance to it is smaller than a predefined threshold. In (Mc Kerrow, 1993), an ultrasonic measure is stored as a circle arc. Two different measures come from the same plane if there is a common tangent to the two arcs. In both cases computation is quite complex and time-consuming.

As in the previous method a sensor measure is assigned to a segment of the environment taking odometric position into account. When a set of points is associated to the same segment of the environment, a segment is computed with a linear regression. So two segments are available : the known one in the model of the environment and the computed one. It is very important to point out that each measured segment has an associated segment in the known environment : the matching problem is solved. In that method segments are represented according to the Crowley's formalism (Crowley, 1989).

The correction is performed in two steps. First, only the orientation is corrected. It is the sum of the differences between the calculated segments orientations and the known ones divided by the number of used segments. Then the (x,y) position is corrected by minimising the sum of the distances between the measured segment middle and the known segment of the environment. Here, the same least square algorithm than above is used.

P. Hoppenot, E. Colle: "Real-time mobile robot localisation with poor ultrasonic data" - 3rd IFAC Symposium on Intelligent Component and Instrument for Control Application (SICICA), pp. 135-140, 9-11 June 1997.

4.EXPERIMENTAL CONDITIONS

4.1.environment and robot

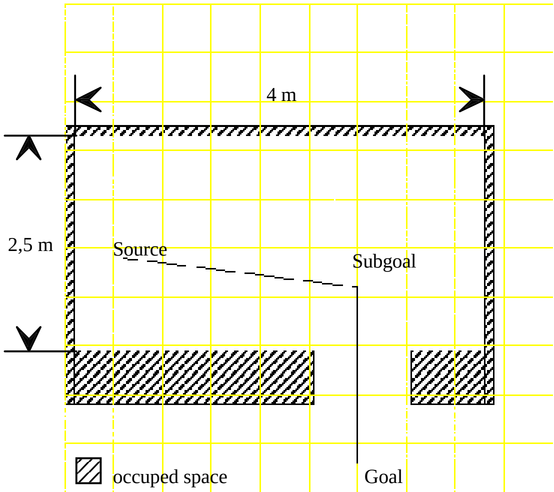

Fig. 1. Planned path

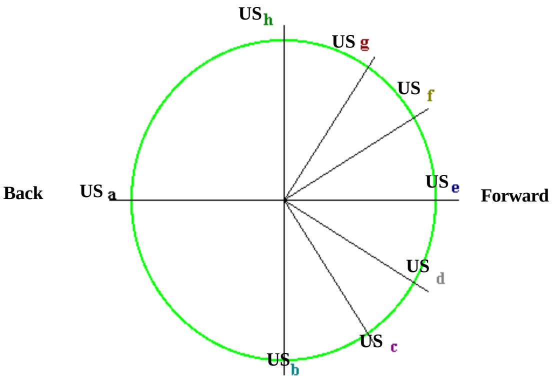

Fig. 2. Ultrasonic sensors layout

The known environment (Fig. 1) is composed of a room with a door aperture and a smooth ground. The task to perform is a movement from the source to a goal cross sub-goals. The robot called RMI (French abbreviation for Intelligent Mobile Robot), is a two wheeled circular robot. The perception system integrates a ring of eight Polaroid ultrasonic sensors (Fig. 2) and an odometric device.

To compare localisation algorithms the performed experimental protocol is :
1- move real robot and memorise odometric and ultrasonic data
2- degrade odometric data
3- simulate robot movement with real ultrasonic and modified odometric data
4- execute localisation algorithms at each burst of sensors measurements
5- draw the path followed by the real robot, the modified odometric robot and the modified odometric robot after the position correction.

In those experimental conditions the assumption that the odometer provides correct information about the robot position is verified.

4.2.odometric data degradation

A constant bias is applied to the wheel diameter. The deviation taken into account at each position computing ends in the complete bewilderment of the robot. Two cases have been treated :
- dissymmetrical inflation e.g. the left tyre is over-inflated and the other sub-inflated
- symmetrical but incorrect inflation

5.RESULTS

All results are presented for the same set of odometric and ultrasonic data. Each algorithm act is presented first without odometry degradation in order to show that the correction doesn't deteriorate a correct localisation given by the odometry too much. Then degradation is introduced to display the method limits. Finally, a table of maximum errors following x and y axes allows the comparison of the different methods.

5.1.Pin-point algorithm

Without odometric degradation. Ultrasonic sensors have a great angle (30 degrees for Polaroid transducers) and a smaller distance measurement accuracy typically 3 centimetres. So using them to localise the robot when the odometry is correct introduces an error. This error must not be important too and must be compared with the improvement when the odometry is disturbed.

In Fig. 3, two trajectories are presented. It is important to notice that the little orientation error in the doorway cannot be corrected. Indeed, from this point, there is no more sensor information because the room is not modelled after the doorway. The distance error goes far up to 15 centimetres at the end because of the unknown environment. This error at the trajectory end will be noticed in each case.

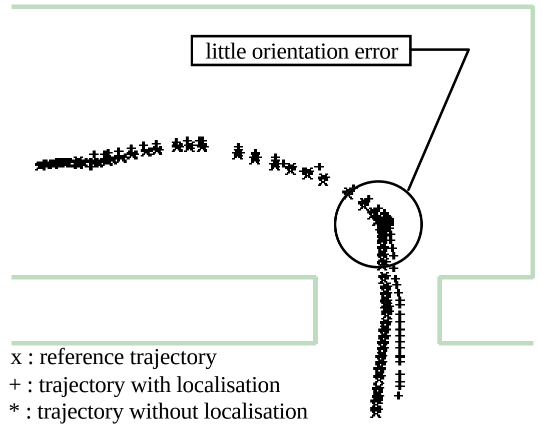

Fig. 3. Robot trajectories without *degradation*.

With a constant odometric degradation.

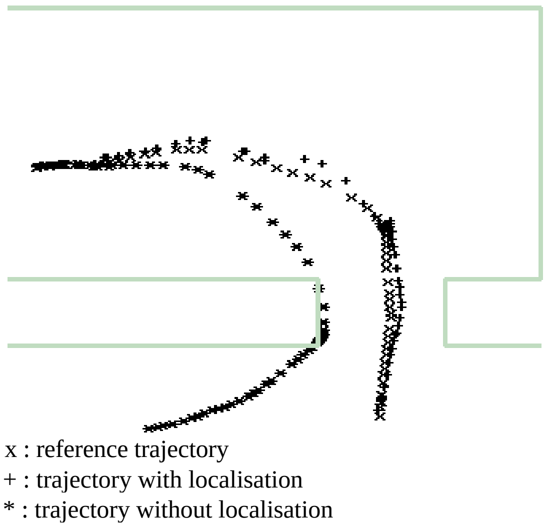

Fig. 4. Robot trajectories with an 8% *degradation*.

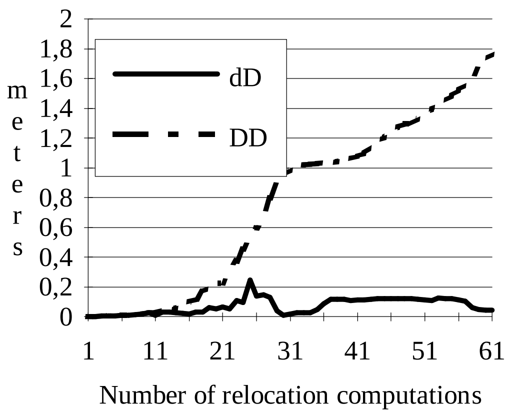

Fig. 5. Errors evolution with an 8% degradation.

Fig. 5 shows a result with a constant 8% sub-inflated wheel. DD is the cumulated error without correction and dD with the correction. The error of the corrected trajectory is not more than 20 centimetres after a 4 meters trip when the error of the non-corrected trajectory goes up to 1.8 meters. The aim is not to locate the robot very precisely but not to be lost. So this result is satisfying (Fig. 4). With a constant 9% sub inflated wheel the corrector act is insufficient. Until the thirtieth step the correction is good. But, if the error is over than 20 centimetres, the matching algorithm doesn't work anymore. So there is no more localisation and the error increases.

with a partially known environment. In the sequel the environment is composed of the same room and an unknown obstacle. That poses a matching problem between measures and known modelled segments of the environment. The trajectories are presented with the sensor shots and the unknown obstacle.

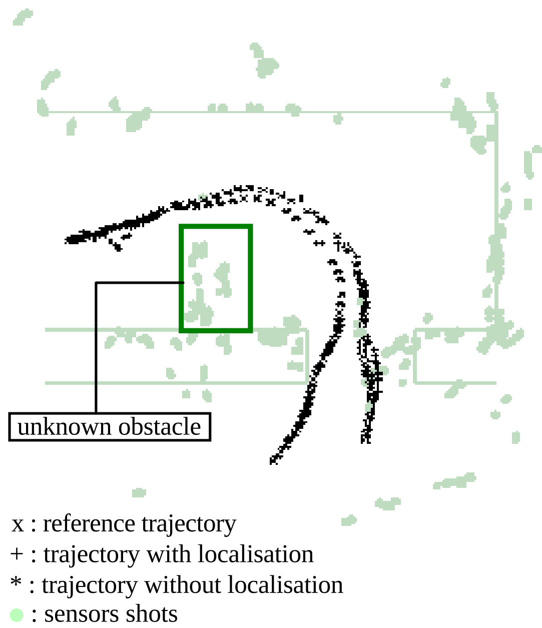

Fig. 6. Robot trajectory with a 3% degradation.

The same degradation percentage cannot be reached. In fact, with this very simple method, it is possible to correct only a 3% degradation (see Fig. 6).

One can observe how the robot sees the obstacle. This one occults some important segments in the environment ; so the robot computes a false orientation without being well localised. Then, the following measurements are too erroneous and the robot is lost.

5.2.pin-point algorithm with memory

More information are available than in the first case. The last ten bursts of measurements so 70 measures are used. Consequently, several of them can be eliminated in the matching algorithm. It seems to be very important especially in a partially known environment where a large number of measurements are not matched with one of the known walls. A specific ultrasonic sensor characteristic is used : the opening cone of measurement. A measure is associated to a segment if and only if the difference between the direction of the shot and the normal to the wall is less than 20 degrees. This can not be used in the pin-point algorithm because of the too small number of resulting matchings.

Without odometric degradation. The localisation does not disturb the trajectory. The large number of measurements explains the better behaviour of the method. The error grows up at the end for the same reason that is previously stated : no more information is available (after the doorway the environment is no more modelled). But here the error appears only at the very end because of the memory effect.

With a constant odometric degradation. With this method an 11% degradation can be corrected in the worst case. The resulting trajectory is shown on Fig. 7. The corrected trajectory follows quite well the reference one (Fig. 8).

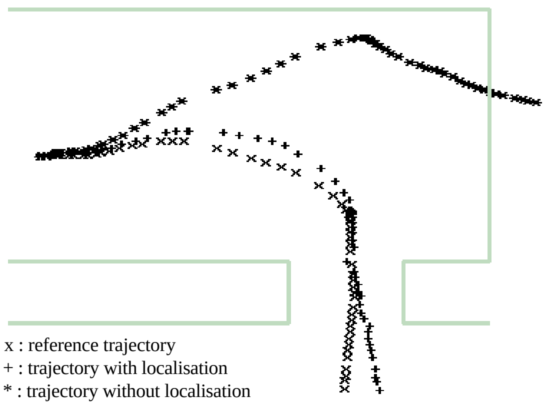

Fig. 7. Robot trajectory with an 11% degradation.

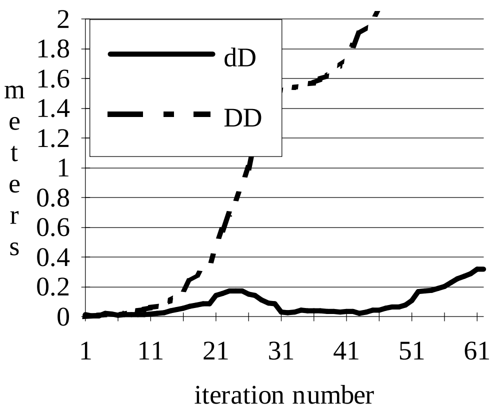

Fig. 8. Error evolution with an 11% degradation.

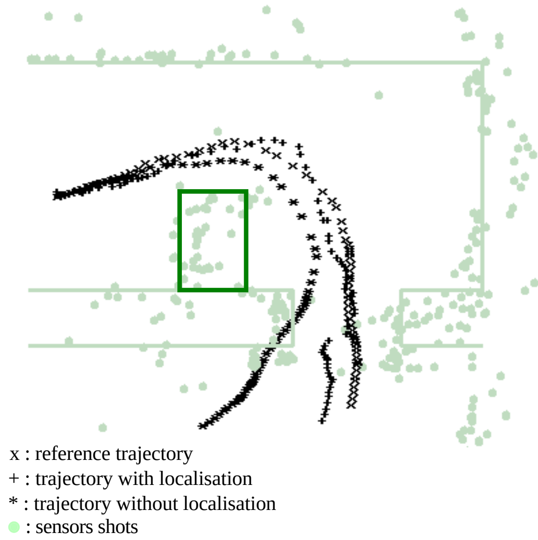

Fig. 9. Robot trajectory with a 4.75% degradation.

With a partially known environment. The best result is a correction of a 4.75% degradation in the worst case (Fig. 9). It is 50% better than the first method.

5.3.Discussion

Only the worst case is presented. It corresponds to the asymmetrical inflation of the tyres which induces localisation errors both on x, y and θ .

With regard to real time, algorithm timing is at least 10 times less than sensors data acquisition timing which is 0.5s. The mean execution time is 2.5 ms for the pin-point algorithm and 50 ms for that with memory effect. In the latter method, it is interesting to point out that 10% of the time is used for the matching operation, 70% for the linear regression calculation to build up the calculated segments and 20% for the localisation computation. So 90% of the time is taken by the least square algorithm (which is used in both linear regression and localisation computation).The execution time of that algorithm must be improved.

The pin-point method with a memory effect presents the possibility to adjust different parameters. It is possible for example to adjust the memory depth, the association distance, the maximum angle of measures and the localisation correction rate. Thanks to that adjustment ability, the same algorithm operates both in known and partially known world. The dynamic modification of parameters can be driven by the rate of recognition of the environment.

Table 1 presents comparative results of the two algorithms. Generally, authors estimate a real odometric error around few percents on a smooth ground.

Table 1 results presentation.

	number 1	number 2
without obstacles	8%	11%
with obstacles	3%	4.75%

The second method is better than the first one and also presents a larger margin of evolution.

6. CONCLUSION

The perception system used to perform a localisation task is composed of an odometer and an ultrasonic transducer ring.

The proposed algorithms are simple in order to be executed in real time and to have a very low odometric error at each step. It is the matching technique that permits this simplicity. Then the improving of the least square algorithm will probably reduce more the time consuming.

More the algorithms present a high insensitivity to erroneous and inaccurate ultrasonic measures due to specularity, multiple echoes and large solid angle.

Up to now the approach doesn't take into account the bad knowledge of the orientation position and the position of the robot at the beginning point of the task.

The addition of a memory effect to the pin-point method gives the ability dynamically to adjust the parameters of the algorithm to the type of environment.

That method allows a correction of the odometry until an 11% degradation in a known environment and a 4.75% in a partially known environment. These values have to be compared to the few percent error typically attributed to the odometry.

One of the main job to do now is to develop the global correction algorithm in which the robot must find his location only with the sensor measurements.

7. REFERENCES

Cox, J. (1991). Blanche - an experiment in guidance and navigation of an autonomous robot vehicle. IEEE Trans on Rob and Aut, vol 7, n°2.

Crowley, J.L. (1989). World modelling and position estimation for a mobile robot using ultrasonic ranging. IEEE Int. Conf. on Robotics and Automation.

Crowley, J.L., B. Schiele (1994). A comparison of position estimation techniques using occupancy grids. IEEE International Conference on Robotics and Automation, San Diego.

Elfes, A. (1986). A sonar-based mapping and navigation system. IEEE International *conference* on robotics and automation.

Elfes, A., L. Matthies (1988). Integration of sonar and stereo range data using a grid-based representation. IEEE Int Conf on Rob and Aut, pp 727-733.

Elfes, A. (1990). Occupancy grids : a stochastic spatial representation for active robot perception. IEEE Proc 6th Conf on Uncertainty in Al.

Kröse, B.J.A., K.M. Compagner and F.C.A. Groen (1993). Accurate estimation of environment parameters from ultrasonic data. Robotics and Autonomous systems 11, pp221-230.

Mandelbaum, R., M. Mintz (1993). Active sensor fusion for mobile robot exploration and navigation. 130 SPIE vol 2059, sensor fusion VI.

Mc Kerrow, P.J. (1993). Echolocation - from range to outline segments. Robotics and Autonomous systems 11, pp205-211, El Sevier.

Wilkes, D., G. Dudek, M. Jenkin. Multi-transducer sonar interpretation. IEEE Int. Conf. on Robotics and Automation, pp 392-397.