GENERATION AND EXTENSION OF MAPS OBTAINED BY AUTONOMOUS ROBOTS

Maite López-Sánchez, Ramon López de Màntaras and Carles Sierra
Institut d'Investigació en Intel.ligència Artificial, IIIA-CSIC
Campus UAB, 08193 Bellaterra,
{maite, mantaras, sierra}@iiia.csic.es

Abstract

We present some improvements of a previous environment mapping work. Firstly, we relax a previous restriction about the environment orthogonality. And secondly, we increase the environment coverage by extending occupancy information in the map. Finally, some results show how that extension process yields to safer paths when extended maps are used for planning.

Key words: autonomous robots, possibility theory, map generation.

1 INTRODUCTION

The background of this paper is a previous mapping work [5] which consisted on two different steps: first, a troupe of small autonomous robots explore an unknown office-like orthogonal environment, and second, a host computer receives exploration information from the returning robots and uses it to build a map. Robots explore moving randomly in clear areas and following walls (or obstacle edges) when detected by their infrared (IR) sensors. The computer host models the environment generating a grid map in terms of degrees of possibility and necessity [2] of the position of the detected walls and obstacles. Since a total coverage cannot be guaranteed, the host also computes the shortest path between two positions so that a robot can use it as a guide to reach a less explored area by combining the path information with reactive behaviour.

This paper describes two improvements of the mapping: one is that the assumption about the orthogonality of the environment is partially discarded and the other is that detected segments are enlarged on the basis of implicit map information. This map extension process increases the map coverage of the real environment and gives more conservative and informed paths that reduce the robots use of reactivity when following them.

2 NON-ORTHOGONAL ENVIRONMENTS

Our mapping process only considers environmental features with edges long enough to be followed by the robots. Small obstacles, such as chair or desk legs, are not represented. In fact, since they can be avoided using reactivity, they are not crucial.

Considering only relatively-large environmental features, we can assume that office-like environments are highly orthogonal. On the one hand, walls are usually connected by right angles. And on the other hand, human made objects in an office such as bookshelves or drawers tend to have rectangular shapes. In case a robot follows a wall or an edge of an obstacle with a not exactly parallel trajectory, the orthogonal assumption is used to correct those slightly deviated segments. Oblique segments are allowed but not changed. This means a significant improvement to the mapping process because, as Yamauchi points out [10], it is still common to find in the literature systems that can only handle parallel or perpendicular walls.

3 MAP GENERATION

The space being explored by the robots is discretised by means of a grid. Cells in the grid represent a small area of the real environment and contain two values: the degree of possibility and the degree of necessity of the presence of obstacles. Initially, that is, before any exploration has taken place, each cell, represented by its co-ordinates (x_i, y_j), has a possibility value Π_{ij}(wall)=1 (i.e., it is completely possible that there is a wall or obstacle in the cell) and a necessity value N_{ij}(wall)=0 (i.e., there is no certainty at all that there is a wall or obstacle in the cell). These initial values correspond to a situation of total ignorance according to the theory of possibility [2]. As robots communicate the information gathered during their exploration to the host, the possibility and necessity values are modified in a way that depends on the detection, or not, of obstacles by the returning robot. The information gathered by each robot is the trajectory of the robot together with the position of the walls and singular points (that is wall ends and corners) that have been detected along it. Due to the odometry error, the position of the detected walls has an associated error. This error follows a Normal distribution [5] and has been approximated by a rectangle whose size increases with the travelled distance.

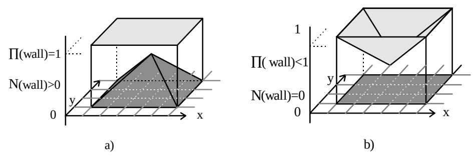

Figure 1: Π and N values assigned to cells corresponding to: a) wall detection, and b) free space.

When an error rectangle is associated to a position that belongs to a detected wall, the occupancy certainty degree (i.e., the certainty about the presence of an obstacle in that position) is expressed by means of necessity values in every cell that results covered by the error rectangle around that position. The necessity values decrease linearly (from the centre of the error rectangle) with the magnitude of the error and remains positive (N_{ij}(wall) > 0) for all the cells inside the error rectangle but gets the value 0 at the cells outside the limits of the rectangle. Figure 1 a) shows this case. Notice that according to the axioms of possibility theory, the possibility value is constantly equal to 1 in all the cells covered by the error rectangle.

On the other hand, robot trajectories supply information about free space so that the corresponding cells (covered by the error rectangle) have values of $\Pi_{ij}(\neg\text{wall}) = 1$ and $N_{ij}(\neg\text{wall}) > 0$. According to the axioms of possibility theory, these values are equivalent to $N_{ij}(\text{wall}) = 0$ and $\Pi_{ij}(\text{wall}) < 1$. Figure 1 b) shows this case, and the next subsection describes how these possibilities and necessities are computed.

3.1 VALUE ASSIGNMENT AND COMBINATION

The height of the pyramids in Figure 1 are determined by the size of the error rectangle. The underlying idea is to establish a linear error-height relation such that, a null error implies the maximum allowed value (1) of height, and an error equal to the maximum allowed error (K) implies a zero height since the information is no longer reliable.

Let us represent the error rectangle by the tuple (x_c, y_c, $error_x$, $error_y$), where x_c, and y_c are the coordinates of the central cell of the rectangle and $error_x$ and $error_y$ are, respectively, half the length of the base and half the length of the height of the error rectangle measured in number of grid cells. Following this, the height of the pyramid is given by:

$$height = 1 - \frac{Max(error_x, error_y)}{K},$$

and the necessity value of having a wall at (x_i, y_j) is:

$$N_{ij}(wall) = height * min(1 - \frac{|x_i - x_c|}{error_x}, 1 - \frac{|y_j - y_c|}{error_y})$$

and the Π_{ij}(wall) is obtained through 1-$N_{ij}(\neg\text{wall})$ by using the same equations.

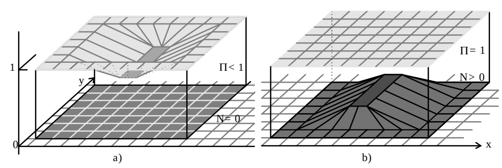

Figure 2: Segment representation corresponding to a) trajectories (possibility pyramids), and b) walls (necessity pyramids).

In considering now consecutive points along the trajectory of the robot or along a wall segment, since their corresponding pyramids overlap, some value combination is required. In the case of wall segments the values are necessities (increasing from 0) and are combined by using the *max* operation (figure 2 b)). In the case of trajectories these values are possibilities (decreasing from 1) and are combined by means of the *min* operator (Figure 2 a))

When the same portion of a wall has been followed more than once, we consider them as independent sources of evidence. Thus, the necessities are combined by means of the probabilistic sum, that is $S(x,y) = x+y-x\cdot y$, in order to reinforce the certainty about the location of the wall. Figure 3 shows a top view of this situation.

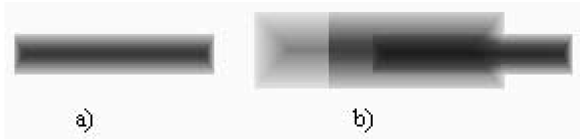

Figure 3: a) First partial detection of a wall, b) the same wall is detected during a different trajectory segment.

Although the definition of our possibility and necessity pyramids are based on a statistical analysis of the displacement error, the use of Possibility Theory allows us to define ignorance as well as to provide a simple value combination (and reinforcement).

4 MAP EXTENSION

Following a wall along a random distance implies that very often the robots leave the wall before reaching its end. This is done in order to increase the number of discovered features and to avoid that robots keep following the contour of a feature once it is detected. If it is the case that the end is reached, then the corresponding extreme of the segment is labelled as "singular point"). Since we can ensure that wall segments without singular points correspond to longer walls —or obstacle edges— they can be extended. However, we cannot know their real length and therefore we make the following decisions regarding their limits: robot trajectory segments limit the extension of wall segments, we stop extending a wall segment when it meets either a detected wall segment or another segment extension (in the former case they belong to the same wall, while the later represents perpendicular walls).

Extension is done locally by propagating small constant certainty values (they are just assumptions) at the ends of orthogonal wall segments. Since, oblique segments represent shorter features as doors, or other less predictable objects, they are not extended.

Wall extension has two main advantages: first, it increases the coverage of the real environment and second, the planning [8] over the resulting maps give more conservative and safer paths than the ones obtained considering just detected features. These paths are computed from a visibility graph of the free space and guide robots towards less explored areas, reducing the use of reactivity (to avoid non previously detected obstacles) when following the paths.

5 RESULTS

Five robots have been sent to explore an unknown environment (see Fig 4 a)). Since the host builds its map (Fig 4 b)) in an incremental way, the data in Table 1 shows the environment coverage each time one more robot delivers its exploration information to the host. Columns list the wall coverage percentage (respect a total length 30m.), first column stands for detected wall segments, second for correctly extended wall segments and the third represents the incorrect extensions.

Table 1: percentage of the environment coverage

# maps	*coverage*	*extension*	*incorrect ext.*
1	16.3%	32%	1.97%
2	28.2%	48.99%	1.97%
3	44.6%	58.7%	0.4%
4	58.8%	78.1%	-
5	65.2%	82.1%	-

The extension process is purely based on the application of heuristics and, therefore, it is hard to evaluate. It is also highly dependent on the structure of the environment as well as on the detections from random explorations.

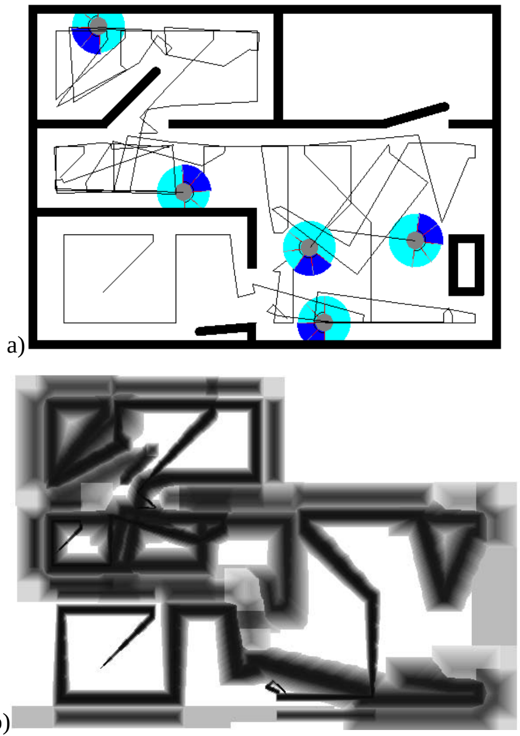

Figure 4: a) Robots explore an unknown 5.3×3.8 m. environment b) Extended map: wall detections in red, trajectories in blue, singular points in green and wall extensions in light red (darker colours mean more certainty).

In order to further illustrate the improvement, we have applied the same heuristics to the environment at Fig 5 a). In this case, the detections are shown in Fig 5 b) and cover 45.3% of the total walls. Fig 5 c) depicts the extensions, from which 70.1% are correct and 6.4% are incorrect. This second example has been also used to compare the planning in extended and non-extended maps. Fig 5 b) shows planned paths in the non-extended map and c) the paths in the extended map. Notice that paths planned on extended maps are named after those in the non-extended map, but a letter ‘e’ differentiates them. Obviously, they have the same initial and goal position.

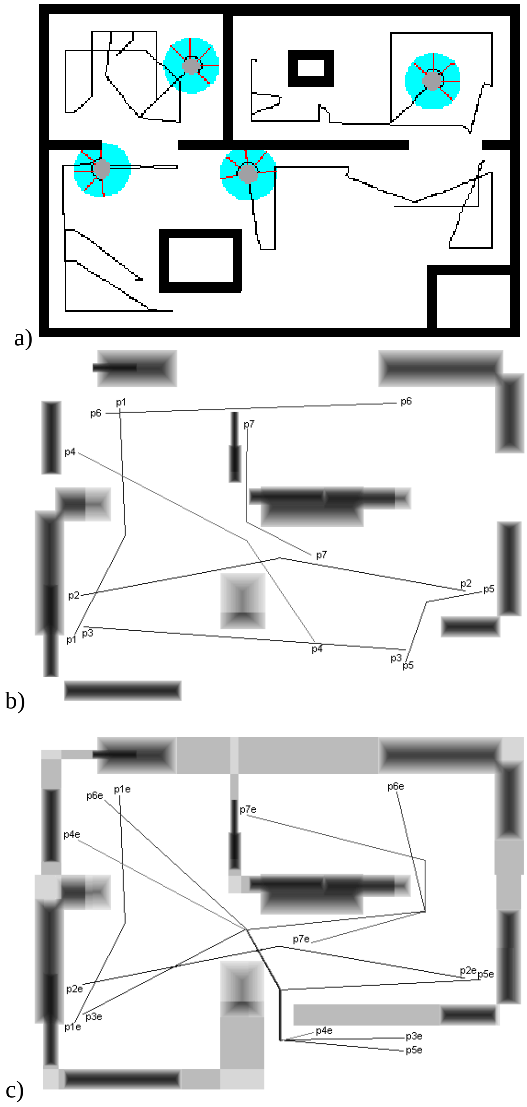

Figure 5: a) Robot exploration . Paths planned on the resulting map considering: b) wall detections and c) extensions.

The three images in Figure 6 depict the trajectories actually followed by the robots –some of them present common trajectories. Obviously, in areas with the same information both perform equally (see path p1 and p2). When one map is more accurate than the other, incorrectly extended information yields to paths longer than necessary, while non-detected walls imply a high use of reactivity. Sometimes, an incorrect extension closes narrow gaps that require the use of reactive capabilities to get through. In that way, an initially shorter path in the non-extended map can result in a trajectory whose length is in fact comparable to the length of a trajectory from the extended map (this is the case of p3 and p3e). Of course, this is not always the case: p4e avoids an actually non-existing piece of wall causing an extra displacement of 0.37m. (p4 is a 6.45m long trajectory). The worse case of avoiding a non-existent wall is p5e, which turns around the corner where a previous robot went. Although this tendency of going over previous robot trajectories can be a disadvantage, already explored free space is also the source of safer paths. Finally, when extension is correct, resulting paths are more informed and therefore reactivity is less often needed (see p6 & p6e).

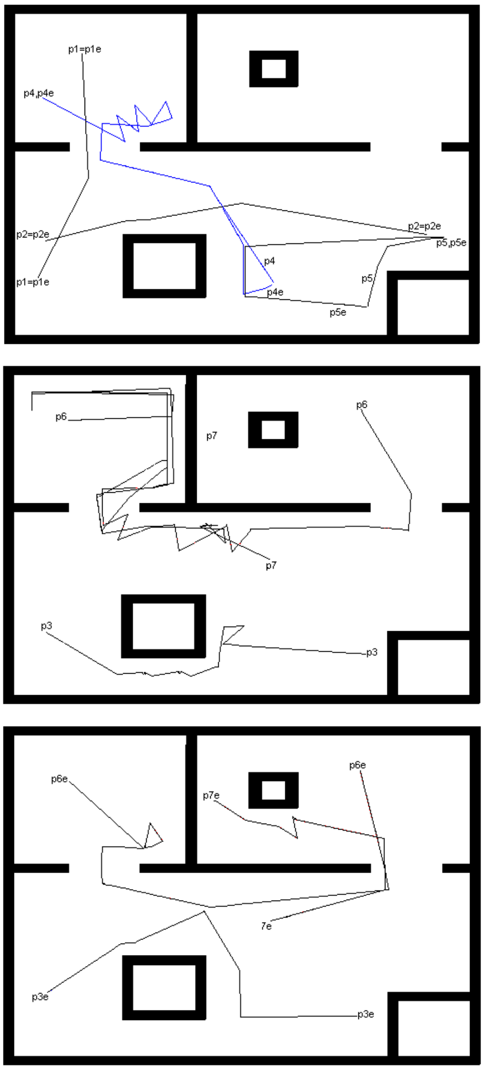

Figure 6: Performance of the robots following the paths at Figure 5.

Reactivity is less efficient and can yield the problem that appears in p7, in which the reactivity makes the robot to take so many wrong decisions that the robot stops before reaching the target (the robot position error grows so much due to the many wrong displacement movements that the area it covers already includes the target co-ordinates even when the robot is still quite far from the target).

5 RELATED WORK

Probabilistic techniques are commonly used for map representation, for example [1] defines landmarks in natural environments assuming a Gaussian certainty distribution of their positions and [4,6,7] estimate the probability of cell occupancy in certainty grids. Probabilistic techniques need a huge amount of data and assume known distributions, Fuzzy Set theory is a good alternative when these conditions are not met. For example [3] uses fuzzy numbers to model the uncertainty of the parameters of geometric primitives and of co-ordinate transformations used to describe natural environments and [9] uses fuzzy sets to assign to each point in the grid map a degree of being empty and a degree of being occupied. This last work is the closest to ours, but our use of infrared sensors in short distances gives better results than their imprecise information coming from ultrasonic sensors.

References

[1] Betgé-Brezetz, Hébert, Chatila, Devy, Uncertain Map Making in Natural Environments, Proc *IEEE: Int. Conf. on Robotics and Automation*, 1048-1053, 1996.

[2] Dubois and Prade, *Possibility Theory*, Plenum Press, 1988.

[3] Kim, Ko and Chung, Uncertain robot environment modelling using fuzzy numbers, *Fuzzy Sets and Systems*. Vol 61, pp 53-62, 94.

[4] Lim, and Cho, Physically Based Sensor Modelling for Sonar Map in Specular Environment, Proc *IEEE Conf on Robotics and Automation*, pp 1714-1719, 1992

[5] López-Sánchez, López de Màntaras and Sierra, Incremental map generation by low cost robots based on possibility/necessity grids, P. *Uncertainty in Artificial Intelligence*, pp. 351-357, 1997.

[6] Movarec and Elfes, High Resolution Maps from Wide Angle Sonar, P. *Int. Conf. on Robotics and Automation*, pp 116-121, 1985.

[7] Pagac, Nebot and Durrant-White, An evidential approach to probabilistic map-building, Proc *IEEE Int Conf on Robotics and Automation*, pp 745-750, 1996.

[8] Pearl, *Heuristics: Intelligent Search Strategies for Computer Problem Solving*, Addison-Wesley. 1984.

[9] Poloni, Ulivi and Vendittelli, Fuzzy logic and autonomous vehicles: Experiments in ultrasonic vision, *Fuzzy Sets and Systems*. Vol 69, pp15-27, 1995.

[10]Yamauchi, B. Frontier-based exploration using multiple robots, Proc *Autonomous Agents*, 47-53,1998.