Low Level Navigation System for a POMDP based on WiFi and Ultrasound Observations

M. Ocaña, M. A. Sotelo, L. M. Bergasa and R. Flores

Department of Electronics
University of Alcala
Alcalá de Henares, 28805 Madrid, Spain

{mocana,sotelo,bergasa,flores}@depeca.uah.es

Abstract - **In this paper we present the low level navigation system carried out in a Partially Observable Markov Decision Process (POMDP) based on WiFi and ultrasound observations. We use an H-shape model for the corridor, obtained from ultrasound range sensor. This system leads the robot to follow a corridor while it's detecting transitions for each door in autonomous mode. We demonstrate that this system is useful as low level navigator in a POMDP for indoor environments with a real robot. Some experimental results are shown. Finally, the conclusions and future works are presented.**

Index Terms - indoor navigation system, low level robot control, POMDP, WiFi observation, autonomous system.

I. INTRODUCTION

The boom in wireless networks over the last few years has given rise to a large number of available mobile tools and their emerging applications are becoming more and more sophisticated by year. Wireless networks have become a critical component of the networking infrastructure and are available in most corporate environments (universities, airports, train stations, tribunals, hospitals, etc) and in many commercial buildings (cafes, restaurants, cinemas, shopping centres, etc). Then, new homes are slowly starting to add WiFi services in order to enable mobility to perform many routine tasks, in the known as intelligent houses. There are even emerging some projects about WiFi enabled cities as Paris, Barcelona, etc.

The localization, navigation and automatic map building are some of the principal problems in mobile robotics. There are several ad-hoc methods that they are only useful in singular applications.

The recent interest in location sensing for network applications and the growing demand for the deployment of such systems has brought network researchers up against a fundamental and well-known problem in the field of the robotics as is the localization. Determining the pose (position and orientation) of a robot from physical sensors is not a trivial problem and is often referred to as "the most important problem to providing a mobile robot with autonomous capabilities" [1]. Several systems for localization have been proposed and successfully deployed for an indoor environment. Examples include infrared-based systems [2], various computer vision systems [3], ultrasonic sensors and actuator systems [4], physical contact based actuator systems [5], laser range finder [6] and radio frequency (RF) based systems [7].

Many mobile robot platforms use wireless networking to communicate with off-line computing recourses, human-machine interfaces or others robots. Since the advent of inexpensive wireless networking, many mobile robots have been equipped with 802.11b wireless Ethernet. In many applications, a sensor from which position can be inferred directly without the computational overhead of image processing or the material expense of a laser is of great use. Many robotics applications would benefit from being able to use wireless Ethernet for both sensing position and communication without to add new sensors in the environment.

WiFi location determination systems use the popular 802.11b network infrastructure to determine the user location without using any extra hardware. This makes these systems attractive in indoor environments where traditional techniques, such as Global Positioning System (GPS) [8] fail. In order to estimate the user location, wireless Ethernet devices measure signal strength of received packets. This signal strength is a function of the distance and obstacles between wireless nodes and the robot. Moreover, the system needs one or more reference points (Access Points) to measure the distance from. Unfortunately, in indoor environments, the wireless channel is very noisy and the radio frequency (RF) signal can suffer from reflection, diffraction and multipath effect, which makes the signal strength a complex function of distance.

The authors demonstrated in [9] that the robot localization based on a WiFi signal strength measure is possible but this sensor is affected for different factors. An important factor is the small scale error; this error means that the signal is affected in several dBms when the robot is moved in a distance down to the wavelength. Another important factor comes from the error in the small orientation, this error means that the robot deviates in a few degrees from the ideal orientation.

Recently they have appeared probabilistic methods to give very useful solutions to the localization, navigation and automatic mapping using uncertainty information. These methods are based on Bayesian networks or Markov models.

For a global navigation system, in which the objective is the guidance to a goal room, a topological discretization is appropriate to facilitate the planning and learning tasks.

In this context, topological representation of the environment and observations using sensors with high uncertainty, the optimal methodology in order to build a robust navigation system is to use a Markov model known as POMDP.

The POMDP is a mathematic model that permits to characterize systems with partial observability. In this work

two different kinds of partial observations are used: WiFi signal strength and ultrasound sensor.

A POMDP doesn't know its real state because the uncertainty of the observations. The POMDP maintains a belief distribution called *Bel*(*S*) or *Belief Distribution (Bel)* to solve it. This distribution assigns to each state *s* a probability; this probability indicates the possibility of being the real state.

The Belief Distribution must be updated whenever a new action or observation is carried out. In some many applications, these two conditions are simultaneous. When the low level controller detects a transition a new observation is taken.

To realize an action and reach the next state the POMDP needs a local navigation system that positions the robot. This local navigation system or low level controller is the system able to move the robot between the states and it contains the transitions detector for indicating to the POMDP the transition among nodes assuming some positioning and orientation errors. A deep explanation about the implemented navigation system using POMDP has been presented by the authors in [10].

In this paper, we present the low level navigating system used in our POMDP based on WiFi and ultrasound observations for our robotic platform. Firstly, we present the modelling of the corridor using an H-shape model and then we propose the non-linear lateral controller that leads the robot for the centre of the corridor. We experimentally demonstrate that the system performs well in real applications. Then, we present the results in real mode obtained from our Pioneer 2AT robot. Finally we extract conclusions about it and present the future works in this line.

II. LOW LEVEL NAVIGATION SYSTEM

A. Estimation of corridor model

The use of a model is crucial in order to robustly reconstruct the corridor geometry. In this case, the robot is to operate in straight corridors. Accordingly, a H-shape model is adequate to represent the real geometry of the corridor. For that purpose, the width of the corridor needs to be a priori known (based on the map) or on-line estimated. Based on the corridor width, a consistent H-shape corridor model is computed at each iteration using least squares techniques. The polynomial equation of the left wall is constrained by the polynomial equation of the right wall since they have to be parallel to each other. A H-shape polynomial model can then be calculated by taking advantage of this constraint. The adjustment of the H-shape model is computed in two steps. In a first approach, the parameters of the individual straight lines describing the left and right walls ($y_{left}=m_{left}x_{left} + b_{left}$, $y_{right}=m_{right}x_{right} + b_{right}$) are obtained based on ultrasound measures using least squares operators, yielding m_{left}, b_{left}, m_{right}, b_{right}. In the previous computation, variables y_{left}, x_{left}, y_{right}, x_{right} are relative to the robot centre of gravity. Based on these variables, a second step is realized in order to compute the H-shape constrained model. Let *m* denote the slope of the H-shape model. In a first approximative approach, *m* can be computed based on m_{left} and m_{right} by considering the orientation angle of the H-model to be the average of the

orientation angles of the individual straight lines that describe the left and right walls, this leads (1).

$$m = \tan \frac{\arctan m_{left} + \arctan m_{right}}{2} \quad (1)$$

Let E_{ms} denote the mean square error existing between the H-shape polynomial model and the measurements obtained by the ultrasound sensors, as expressed in (2).

$$E_{ms} = \sum_{n=1}^{n=N_{right}} (y_{ln} - y_{Hln})^2 + \sum_{n=1}^{n=N_{left}} (y_{rn} - y_{Hrn})^2 \quad (2)$$

Where N_{left} and N_{right} denote the number of points associated in the current iteration to the left and right walls, respectively, y_{ln} and y_{rn} represent the *y* coordinate of the nth measurement associated to the left and right wall, and y_{Hln} and y_{Hrn} stand for the *y* coordinate of the nth point corresponding to the left and right edges of the H-shape model that describes the corridor. Accordingly, the estimation of the appropriate parameters of the H-shape model is carried out by minimizing the derivative of E_{ms}. The error can be expressed as a function of the corridor orientation, represented by parameter *m*, and also as a function of parameter *b*, where *b* stands for the independent term of the straight line that describes the centre of the corridor as depicted in Figure 1.
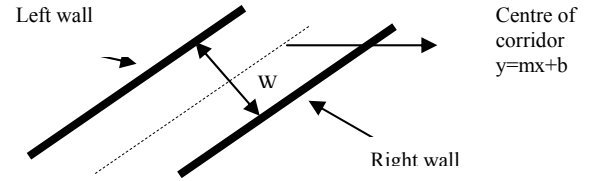

Fig. 1 H-shape corridor model

The left and right edges of the corridor can then be constrained by parameters *m*, *b*, and the corridor width, denoted by *W*. This allows to compute the derivative of E_{ms} yielding an expression that depends only on parameter b, assuming that *m* is already known as described by (3).

$$\frac{dE_{ms}}{db} = \sum_{n=1}^{n=N_{left}} 2 \cdot (y_{ln} - mx_{ln} - b) \cdot (-1) + a_i \quad (3)$$

$$a_i = \sum_{n=1}^{n=N_{right}} 2 \cdot (y_{rn} - mx_{rn} - b + W \cdot (\sin \arctan m - m \cos \arctan m)) \cdot (-1)$$

Where x_{ln} and x_{rn} represent the x coordinate of the nth measurement associated to the left and right wall, respectively. The minimization of (3) with respect to parameter *b* yields the value expressed in (4).

$$b = \frac{\sum_{n=1}^{n=N_{left}} (y_{ln} - m \cdot x_{ln}) + \sum_{n=1}^{n=N_{right}} (y_{rn} + W(\sin \arctan m - \cos \arctan m) - m \cdot x_{rn})}{N_{left} + N_{right}} \quad (4)$$

Upon computation of parameters *m* and *b*, the estimation of the H-shape model that describes the corridor is complete. The corridor model is computed in relative coordinates with respect to the robot centre of gravity.

Accordingly, it provides the basis to measure the lateral and orientation errors of the robot with regard to the centre of the corridor. Figure 4 depicts a graphical example of corridor model estimation during on-line operation. The measurements associated to the model are represented by green points, while the corridor model is represented by a couple of straight lines, as observed in Figure 2. To avoid perturbations such as doors and moving obstacles, the distance between the measured points and the previous corridor model must be bellow 5 cm in order to validate the measurement and include it in the estimation process.
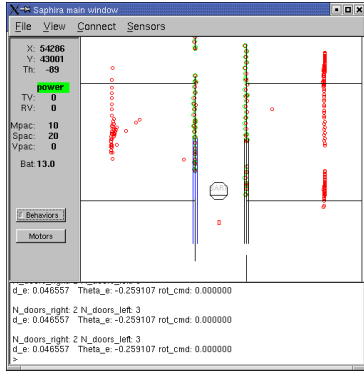

Fig. 2 Online estimation of corridor model

The validation process allows to easily detect doors in the corridor walls. Considering the 5cm validation band, all doors (either open, close, or ajar) will be detected as an open gap in the wall model, as depicted in Figure 2. Doors can be robustly detected even in the presence of sporadic moving obstacles (people walking around) by taking advantage of a priori knowledge about a door typical width. In case the corridor is overcrowded with moving obstacles, the current door detection process could fail but in these cases the high level navigation system based on POMDP would be able to recover the localization of the robot. From (4), variables b_L and b_R for the left and right edges of the wall, respectively, can be demonstrated to be as expressed in (5).

$$b_L = b - \frac{W}{2}\sqrt{m^2 + 1}$$
$$b_R = b + \frac{W}{2}\sqrt{m^2 + 1} \qquad (5)$$

Based on (4) and (5) the lateral and orientation error of the robot with regard to the centre of the corridor can be computed as in (6).

$$\theta_e = \tan^{-1} m$$
$$d_e = \frac{b_R}{\sqrt{1+m^2}} - \frac{W}{2} \qquad (6)$$

Where d_e represents the lateral error, and θ_e stands for the orientation error. A graphical illustration of d_e and θ_e can be observed in Figure 3.

B. Lateral Control

The main goal of the lateral controller is to ensure adequate corridor tracking by correctly keeping the robot in the centre of it with appropriate orientation (parallel to the corridor walls). This goal is very important in our system in order to minimize the WiFi small scale and orientation errors. This constraint can be formulated into the minimization of the lateral and orientation errors. Thus, a model describing the dynamic behavior of d_e and θ_e is needed in order to design a stable lateral controller.

1) *Kinematic model*: the robot configuration space is composed of the global position and orientation variables described by (x,y,θ), while the robot angular velocity T and linear velocity v are the variables of the robot actuation space. Mapping from the actuation space to the configuration space can be solved by using the so-called dead reckoning equations as expressed in (7).

$$\dot{x} = v \cdot \cos\theta$$
$$\dot{y} = v \cdot \sin\theta \qquad (7)$$
$$\dot{\theta} = \omega$$

Let's remark that v represents the velocity of the robot centre of gravity, denoted as control point. The kinematic model in terms of global position and orientation of the robot is converted into a kinematic model in terms of relative coordinates. As observed in Figure 3, d_e is defined as the distance between the robot control point and the closest point (x_d, y_d) along the desired trajectory. This implies that d_e is perpendicular to the straight line that describes the centre of the corridor at (x_d, y_d). The tangent of the slope of the central line is denoted by θ_d. It represents the desired robot orientation. Based on this, d_e and θ_e suffice to precisely characterize the location error between the robot and the centre of the corridor, as described in (8).

$$d_e = -(x - x_d) \cdot \sin\theta_d + (y - y_d) \cdot \cos\theta_d$$
$$\theta_e = \theta - \theta_d \qquad (8)$$

The derivative of d_e and θ_e with respect to time are demonstrated in (9), while the complete non-linear kinematic model for d_e and θ_e is formulated in (10).

$$\dot{d}_e = -\dot{x}\sin\theta_d + \dot{y}\cos\theta_d = v\sin(\theta - \theta_d) = v\sin\theta_e$$
$$\dot{\theta}_e = \frac{d(\theta - \theta_d)}{dt} = \dot{\theta} - \dot{\theta}_d = \dot{\theta} = \omega \qquad (9)$$

$$\dot{d}_e = v\sin\theta_e$$
$$\dot{\theta}_e = \omega \qquad (10)$$

2) *Non-linear lateral controller*: the lateral error d_e and the orientation error θ_e must be minimized in order to lead the robot along the centre of the corridor. For simplicity, the robot linear velocity v is assumed to be constant. The design of the control law is based on general results of the so-called chained systems theory [11]. The use of the popular tangent linearization approach is avoided since it is only locally valid around the configuration chosen to perform the linearization, and thus, the initial conditions may be far away from the reference trajectory. On the contrary, some state and control variable changes are posed in order to

convert the non linear system into a quasi-linear one, termed as chained form. Thus, the non linear model for d_e and θ_e can be transformed into chained form using the state diffeomorphism and change of control variables shown in (11).

$$Y = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \Theta(X) = \begin{pmatrix} d_e \\ \tan\theta_e \end{pmatrix}$$

$$W = \begin{pmatrix} \omega_1 \\ \omega_2 \end{pmatrix} = \Gamma(U) = \begin{pmatrix} v\cos\theta_3 \\ \dfrac{\omega}{\cos^2\theta_e} \end{pmatrix} \qquad (11)$$

These transformations are invertible whenever the robot linear velocity is different from zero. Based on (11), the kinematic model can be rewritten as in (12), considering y_1 and y_2 as the new state variables.

$$\dot{y}_1 = v\sin\theta_e = \omega_1 y_2$$
$$\dot{y}_2 = \frac{d(\tan\theta_e)}{dt} = \frac{1}{\cos^2\theta_e}\cdot\dot{\theta}_e = \frac{\omega}{\cos^2\theta_e} = \omega_2 \qquad (12)$$

In order to get a velocity independent control law, the time derivative is replaced by a derivation with respect to ς, the abscissa along the direction parallel to the centre of the corridor, as graphically depicted in Figure 3.
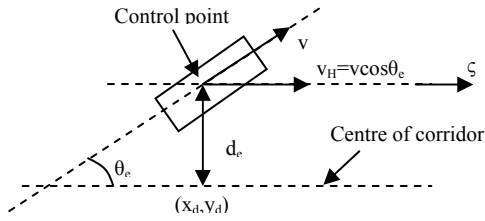

Fig. 3 Graphical description of variable ς

Analytically, ς is computed as the integral of linear velocity v_H, measured along axis ς. The expression of ς is provided in (13).

$$\varsigma = \int v_\varsigma dt = \int v\cos\theta_e dt \Rightarrow \dot{\varsigma} = \frac{d\varsigma}{dt} = v\cos\theta_e = \omega_1 \qquad (13)$$

The time derivative of state variables y_1 and y_2 is expressed as a function of ς in (14).

$$\dot{y}_1 = \frac{dy_1}{dt} = \frac{dy_1}{d\varsigma}\cdot\frac{d\varsigma}{dt} = y'_1\cdot\dot{\varsigma}$$
$$\dot{y}_2 = \frac{dy_2}{dt} = \frac{dy_2}{d\varsigma}\cdot\frac{d\varsigma}{dt} = y'_2\cdot\dot{\varsigma} \qquad (14)$$

Where y'_1 and y'_2 stand for the derivative of y_1 and y_2 with respect to H. Solving for y'_1 and y'_2 leads to (15).

$$y'_1 = \frac{\dot{y}_1}{\dot{\varsigma}} = \frac{v\sin\theta_e}{v\cos\theta_e} = \tan\theta_e = y_2$$

$$y'_2 = \frac{\dot{y}_2}{\dot{\varsigma}} = \frac{\omega}{v\cos^3\theta_e} = \omega_3 \qquad (15)$$

As observed in (15), the transformed system is linear and thus, state variables y_1 and y_2 can be regulated to zero by using the control law proposed in (16).

$$\omega_3 = -K_d y_2 - K_p y_1 \qquad (K_d, K_p) \in \Re^{+2} \qquad (16)$$

Using (15) and (16), the dynamic behavior of y_1 with respect to ς is demonstrated to be linear, as shown in (17).

$$y''_1 + K_d y'_1 + K_p y_1 = 0 \qquad (17)$$

This implies that variables $y_1=d_e$ and $y_2=\tan\theta_e$ tend to zero as variable ς grows. The previous statement is analytically expressed in (18).

$$\lim_{\varsigma\to\infty} d_e = \lim_{\varsigma\to\infty} \theta_e = 0 \qquad (18)$$

Accordingly, variable ς must always grow so as to ensure that the lateral and orientation error tend to zero. This condition is met whenever v>0 and $-B/2 < 2_e < +B/2$. In other words, the robot must continuously move forward and the absolute value of its orientation error should be below B/2 in order to guarantee proper trajectory tracking. Thus, the non linear control law is finally derived from (15) and (16) and formulated in (19).

$$\omega = \arctan\left[-v\cos^3\theta_e\left(K_d\tan\theta_e + K_p d_e\right)\right] \qquad (19)$$

III. EXPERIMENTAL RESULTS

First of all we describe the test-bed used and then we present some experimental results to validate the proposed local navigation system with the real robot.

A. Test-bed

The test-bed was established on the 3rd floor of the Polytechnic School building, in the Electronic Department, at the University of Alcala. The layout of this zone is shown in Figure 4. It has dimensions of 60 m by 60 m with about 44 different rooms, including offices, labs, bathrooms, storerooms and meeting rooms.
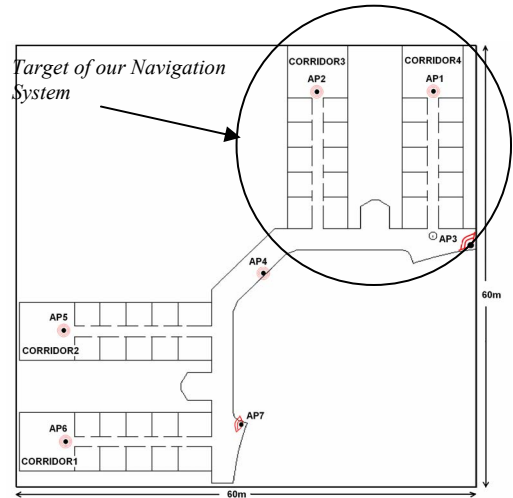

Fig. 4 Test bed environment. 3rd Floor of the Electronic Department

With a circle we remark our target test-bed. Results obtained in this area could be applied to all environment, because the building and the WiFi Access Points (APs) are symmetrically distributed.

Seven Buffalo APs (WBRE-54G) were installed at the locations indicated in figure 4, five APs were connected to omnidirectional antennas and 2 APs (AP3 and AP7) were connected to antennas of 120 degrees of horizontal beam-width. The APs acts as the wireless signal transmitters or base stations.

As mobile robot we have used a Pioneer 2AT of Activmedia robotics with the following configuration: an embedded computer with a Pentium III 850MHz, a 16 ultrasound sensor ring, one Orinoco PCMCIA Gold wireless card with a omnidirectional Buffalo antenna placed above the robot, the operating system is Linux Red Hat 9.0. We modified the wireless tools of Jean Tourrilhes [12] to obtain a valid observation for this application and the patch of Moustafa A. Youssef for the Orinoco driver [13]. Figure 5 shows the robot Pioneer 2AT that we have used.
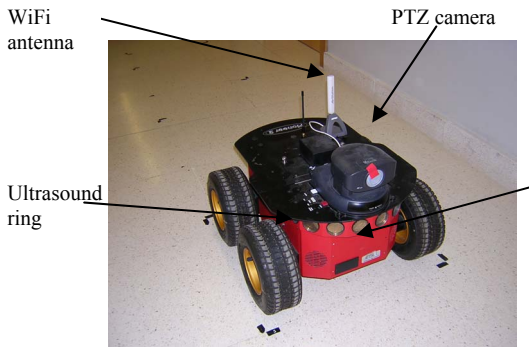

Fig. 5 Real robot used to test the navigation system developed

B. Lateral and orientation errors

We have obtained the lateral and orientation error of the proposed low level navigator with the real robot. These errors are directly related with the *small scale error* and *small orientation error* of the WiFi sensor as we have explained. It's desirable that navigator leads the robot with a small lateral and orientation errors to take WiFi observations with low uncertainty.

We have programmed different targets and we have obtain the lateral and orientation error of the robot.

Figures 6 and 7 show the percentage of lateral and orientation errors respectively, obtained from 40 different targets. The errors have been divided by ranges, referenced to the wavelength for the lateral error and to the degrees for the orientation error.
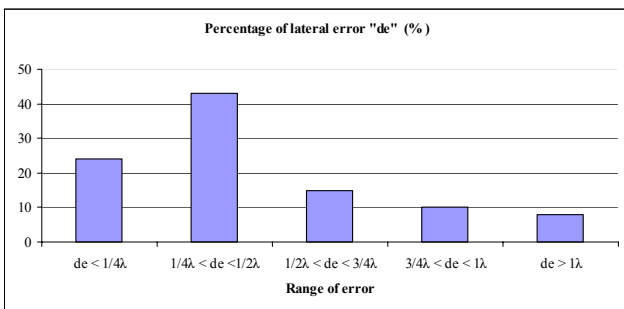

Fig. 6 Results from 40 experiments in lateral error (%)

As can be seen in Figure 6, the accumulated probability that the lateral error is under the 1/2 of the WiFi wavelength is 70%.
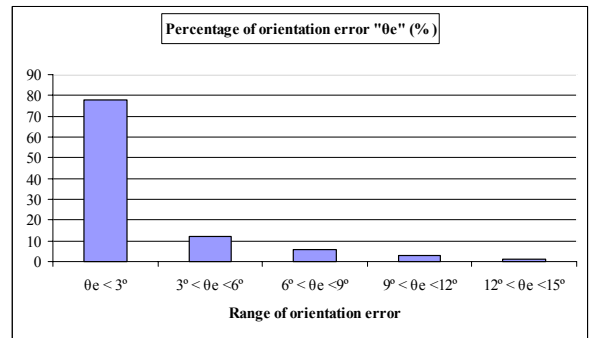

Fig. 7 Results from several experiments in orientation error (%)

In the same way in Figure 7, we can see that the accumulated probability that the orientation error is under 3º is 78%.

In both cases the accumulated percentage between the explained margins allows to obtain WiFi observations with low uncertainty.

As an example, Figure 8 depicts the lateral error and the orientation error evolution during online operation in a couple of real robot missions, in which the robot was located at the beginning of a corridor.

The robot linear velocity was set to 20 cm/s and the previously explained lateral controller led the robot along the centre of the corridor.
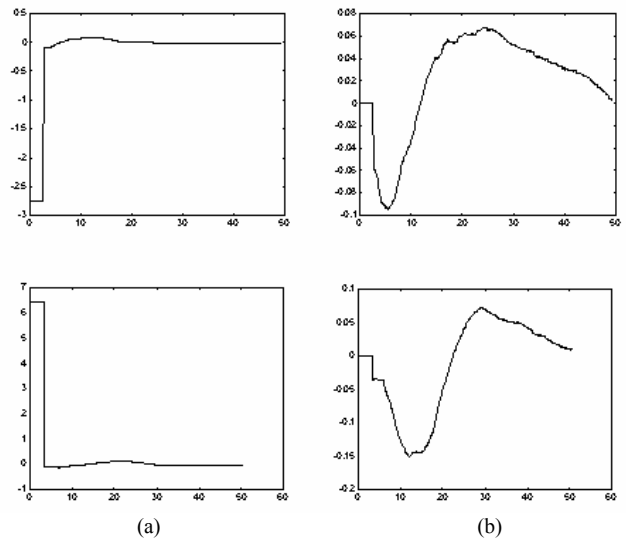

(a)   (b)

Fig. 8 Results from two different experiments: (a) lateral error (in cm) and (b) orientation error (in degrees)

In both experiments, the orientation error didn't exceed 1 degree while the lateral error was under the 1/2 of the wavelength.

As observed in Figure 8, both d_e and θ_e tend to zero as the robot moves forward, this indicates that the controller is stable.

C. Navigation errors

To check that the local navigation system is useful for the global navigation system we programmed a several

different chained targets. The robot was navigating during 3 hours with normal daily work activity and obtaining some few localization errors as can be seen in Table I.

TABLE I
RESULTS IN SEVERAL DIFFERENT CHAINED TARGETS

		Number	Percentage
Successful	Direct path	26	65%
	Indirect path	12	30%
Failures	Incorrect target	1	2,5%
	Loops	1	2,5%

"Direct path" means that the target is reached following the ideal trajectory while "Indirect path" means that the target is reached after recovering maneuver.

D. Transitions detection errors

Another important characteristic in the low level navigation system is the fact of obtaining failures in the transitions detection. When the low level doesn't detect a transition this introduces uncertainty in the action model.

To measure these transitions detection failures, we programmed different chained targets. The robot was navigating during 1 hour with different conditions of people in the environment and then we obtained the transitions detection errors as can be seen in Figure 9.
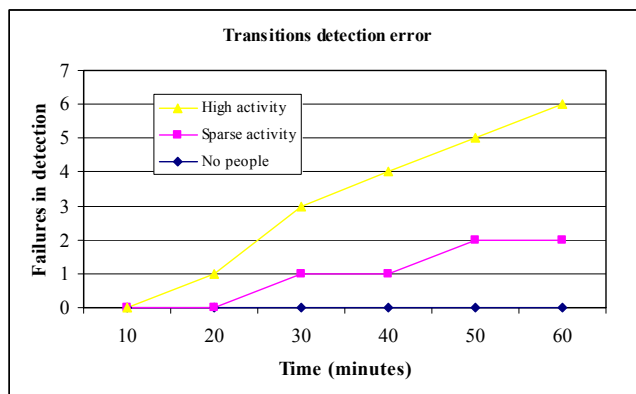

Fig. 9 Results of transitions detection error

When the activity in the environment is null the failures are zero, while when people in the environment is raising the probability of having a failure also rises.

It's important to note that even with these transitions detection failures in the low level navigation system, the global navigation system is able to recover the correct position and it's able to reach its target in the majority cases.

IV. CONCLUSIONS AND FUTURE WORKS

In this paper, we have presented a local navigation system for indoor environment that is used as a low level navigator in a POMDP navigation system based on WiFi and ultrasound observations Then, it works as a robust transitions detector for the POMDP topological map.

With this structure we have obtained an autonomous navigation system very robust and that it is able to recover its position in spite of the sensor uncertainty.

This low level controller is specially indicated to use in this POMDP navigation system because it has a small scale and orientation errors and it contributes to take WiFi observations with a low uncertainty.

In the future, we will try to enhance the algorithm to be faster than actual, and then we will plan to take WiFi observations only in the interesting states, such as the end of the corridors, obtaining a faster and robust algorithm for real applications.

REFERENCES

[1] I. Cox, "Blanche-an experiment in guidance and navigation of an autonomous robot vehicle". IEEE Transactions on Robotics and Automation, 7(2): 193-204, 1991.
[2] R. Want, A. Hopper, V. Falco and J. Gibbons, "The Active Badge location system". ACM Transactions on Information Systems, 10, 91-102, January 1992.
[3] J. Krumm, S. Harris, B. Meyers, B. Brumitt, M. Hale, and S. Shafer, "Multi-camera, Multi-Person Tracking for Easy Living". 3 rd IEEE International Workshop on Visual Surveillance, pp. 3–10, Piscataway, NJ, 2002.
[4] N. B. Priyantha, A. Chakraborthy, and H. Balakrishnan, "The Cricket Location Support System". Proceeding of the 6th ACM MobiCom, pp:155-164, July 2002.
[5] Abowd, "The Smart floor: A Mechanism for Natural User Identification and Tracking". Proceedings of the Conference on Human Factors in Computing Systems, pp. 1–6, Hague, Netherlands, April 2000.
[6] R. Barber, M. Mata, M.J.L. Boada, J.M. Armingol, M.A. Salichs, "A Perception System based on Laser Information for Mobile Robot Topologic Navigation", 28th Annual Conference of the IEEE Industrial Electronics Society, pp. 2779-2784. 2002.
[7] P. Bahl, and V. N. Padmanabhan, "RADAR: A, In-building RF-based User Location and Tracking System". Proceedings of the IEEE Infocom 2000, vol.2, pp. 775–784, Tel Aviv, Israel, March 2000.
[8] P. Enge and P. Misra, "Special issue on GPS: The Global Positioning System". Proceeding of the IEEE, pages 3-172, January 1999.
[9] M. Ocaña, L.M. Bergasa, M.A. Sotelo, "Robust Navigation Indoor Using WiFi Localization", Proceedings of the 10th IEEE Internacional Conference on Methods and Models in Automation and Robotics, pp. 851-856, Miedzyzdroje, Poland, 30 August-2 September 2004.
[10] M. Ocaña, L.M. Bergasa, M.A. Sotelo, R. Flores, "Indoor Robot Navigation using a POMDP based on WiFi and Ultrasound observations", IEEE/RSJ International Conference on Intelligent Robots and Systems IROS2005. 2005. (to appear)
[11] M. A. Sotelo, "Lateral Control Strategy for Autonomous Steering of Ackerman-like vehicles". Robotics and Autonomous Systems, 45 pp. 223-233. 2003.
[12] http://www.hpl.hp.com/personal/Jean_Tourrilhes/
[13] http://www.cs.umd.edu/~moustafa/mapi/mapi.html