Available online at www.sciencedirect.com
ScienceDirect
Robotics and Autonomous Systems 55 (2007) 132–145
Robotics and Autonomous Systems
www.elsevier.com/locate/robot

Low level controller for a POMDP based on WiFi observations

M.A. Sotelo, M. Ocaña*, L.M. Bergasa, R. Flores, M. Marrón, M.A. García

Department of Electronics, Escuela Politécnica Superior, University of Alcalá, Campus Universitario s/n, 28871 Alcalá de Henares, Madrid, Spain

Received 4 December 2005; received in revised form 26 June 2006; accepted 6 July 2006
Available online 22 August 2006

Abstract

This paper shows the results of a low level controller that has been applied to an autonomous robotic system using a WiFi-based Partially Observable Markov Decision Process (POMDP). These observations provide a clue for global robot localization from the first iteration of the POMDP algorithm. Due to the noise channel of WiFi measures, it becomes necessary to make observations at a well known location within the environment. Therefore, a robust local navigator is needed in order to place the robot in an optimal position for making the observation. The system has been tested with two Pioneer 2AT robots in the premises of the Department of Electronics at the University of Alcalá.

Keywords: WiFi-based robot navigation system; WiFi signal measure; Lateral control strategy; Partially observable markov decision process

1. Introduction

For a global navigation system design, in which the objective is the guidance to a destination room, the topological discretization is an appropriate representation to facilitate the planning and learning tasks [1]. In topological discretization the environment is divided into a priori known nodes. In this case, the POMDP models provide solutions to localization, planning, and learning in the robotic context. These models use probabilistic reasoning to deal with uncertainties, which is essential in the case of WiFi localization sensors, and a topological representation of the environment so as to reduce memory requirements and processing time. The robot needs a low level controller to move across the nodes and perform local navigation. The low level controller allows the robot to reach the next node with the lowest positioning error, in such a way that observations are obtained with minimum error.

Over the last few years the interest in wireless networks has increased, and a large number of available mobile tools as well as other emerging applications are becoming more and more sophisticated. 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.).

Many mobile robot platforms use wireless networking to communicate with off-line computing resources, human–machine interfaces, or others robots, since the advent of inexpensive wireless technology. These platforms usually have been equipped with 802.11 b/g wireless Ethernet, thus having a cheap sensor from which position can be directly inferred without the computational overhead required by image processing or the expensive solution provided by laser systems. Many robotics applications would benefit from being able to use wireless Ethernet for both sensing position and communication without having to add new sensors in the environment.

The recent interest in location sensing for a well-known network application and the rising demand on the deployment of such systems has made network researchers face a well-known problem in the field of robotics: localization.

Finding a robot pose (position and orientation) from physical sensors is not a trivial problem. In fact, it is often referred to as "the most important problem to provide a mobile robot with autonomous capabilities" [2]. Several systems for localization have been proposed and successfully deployed for an indoor environment. These systems are based on: infrared sensors [3], computer vision [4], ultrasonic sensors [5], laser [6] or radio frequency (RF) [7]. In some cases, it is possible to find authors

* Corresponding author. Tel.: +34 91 885 6573; fax: +34 91 885 6591.
E-mail addresses: michael@depeca.uah.es (M.A. Sotelo), mocana@depeca.uah.es (M. Ocaña), bergasa@depeca.uah.es (L.M. Bergasa), rflores@depeca.uah.es (R. Flores), marta@depeca.uah.es (M. Marrón), garrido@depeca.uah.es (M.A. García).

0921-8890/$ - see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.robot.2006.07.004

that combine two different technologies, such as ultrasonic sensors and RF [8], to improve the accuracy. In this work the authors use a combination of ultrasonic sensors and Bluetooth to achieve a high accuracy in the localization, but they need to add 5 beacons in an environment of 100 m^2 while in our work we don't need to add extra hardware in the environment. We use the APs or beacons used to establish the Wireless Local Area Network (WLAN), and only 7 APs are needed in a 3600 m^2 environment. In conclusion, the system of Ref. [8] achieves good localizations in small areas by adding a lot of beacons within the environment. This is not the purpose of this work where a peer to peer navigation in huge environments is enough. Within the RF group we can find the localization systems that use WiFi signal strength measures. WiFi location determination systems use the popular 802.11 b/g network infrastructure to determine the global device location without using any extra hardware. It makes these systems attractive for indoor environments where traditional techniques, such as Global Positioning System (GPS) [9], fail.

In order to estimate the robot location, wireless Ethernet devices measure the signal strength of received packets. The signal strength is a function of the distance and obstacles between wireless nodes and the robot. Moreover, the system needs one or more base stations, or Access Points (APs), to measure the distance from them to the device. A triangulation algorithm is then applied to infer the estimated position.

Unfortunately, the wireless channel is very noisy in indoor environments and the RF signal can suffer from reflection, diffraction, and multipath effects, which makes the signal strength a complex function of distance [7]. To solve this problem, several WiFi location determination systems use a priori radio map (wireless map) that captures the signature of each AP at certain points in the area of interest. These systems work in two phases: training and estimation. During the training phase, the system constructs the wireless map in a previous set-up, normally performed in manual mode. In the estimation phase, the vector of samples received from each access point is compared to the wireless map and the "nearest" match is returned as the estimated user location. As described in this work, accurate WiFi-based positioning can be used for autonomous navigation of mobile robots in indoor environments.

For a global navigation system, in which the objective is to guide a robot to a goal room in a semi-structured environment, a topological discretization is appropriate to facilitate the planning and learning tasks and it is specially indicated when the environment is very large. In this context, topological representation of the environment and observations using sensors with high uncertainty, like WiFi signal strength sensors, the optimal methodology in order to build a robust navigation system is to use a Markov model known as POMDP [1].

The POMDP is a mathematic model that permits us to characterize systems with noisy sensors or 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 of 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 it being the real state. The Belief Distribution must be updated whenever a new action or observation is carried out. In many applications, these two conditions are simultaneous. When the low level controller detects a transition a new observation is taken. To carry out 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. In this paper, we present the low level controller used in our POMDP based on WiFi and ultrasound observations for our robotic platforms.

With this separation an important advantage is achieved: the improvement of the low level controller does not affect the main tasks of localization, planning and learning that are carried out by the POMDP. In the same manner, the complexity of dealing with the localization, planning and learning in the topologic map does not affect the low level controller tasks.

The rest of the paper is organized in the following sections: Section 2 provides a description of the WiFi signal measure in indoor environments, as well as a description of the environment itself. Section 3 provides an introduction to the POMDP used in this work and the definition of observations. Section 4 provides a description of the low level controller and the corridor model estimation. Section 5 shows the system implementation and results. Finally, the conclusions and future work are described in Section 6.

2. WiFi signal measure

In this section we present the main causes that produce a variation of WiFi signal strength in indoor environments as well as the environment where the measurements have been made. We have carried out some real experiments in the environment with our Pioneer 2AT in order to obtain the constraints of the low level controller that are needed for the POMDP. In the following, some basic concepts about the test-bed environment and Wifi signal are presented.

2.1. Test-bed

The test-bed environment was established on the 3rd floor of the Polytechnic School building, in the Electronic Department at the University of Alcalá. The layout of this zone is shown in Fig. 1. It has a surface of 60 m × 60 m, with about 50 different rooms, including offices, labs, bathrooms, storerooms and meeting rooms.

Fig. 1(b) shows a detail of the two corridors where the localization and navigation tests have been carried out. We divide the environment in 67 cells placed 80 cm apart in order to build the radio-map.

Seven Buffalo Access Points (APs) (WBRE-54G) were installed at the locations indicated in Fig. 1(a). Five APs were

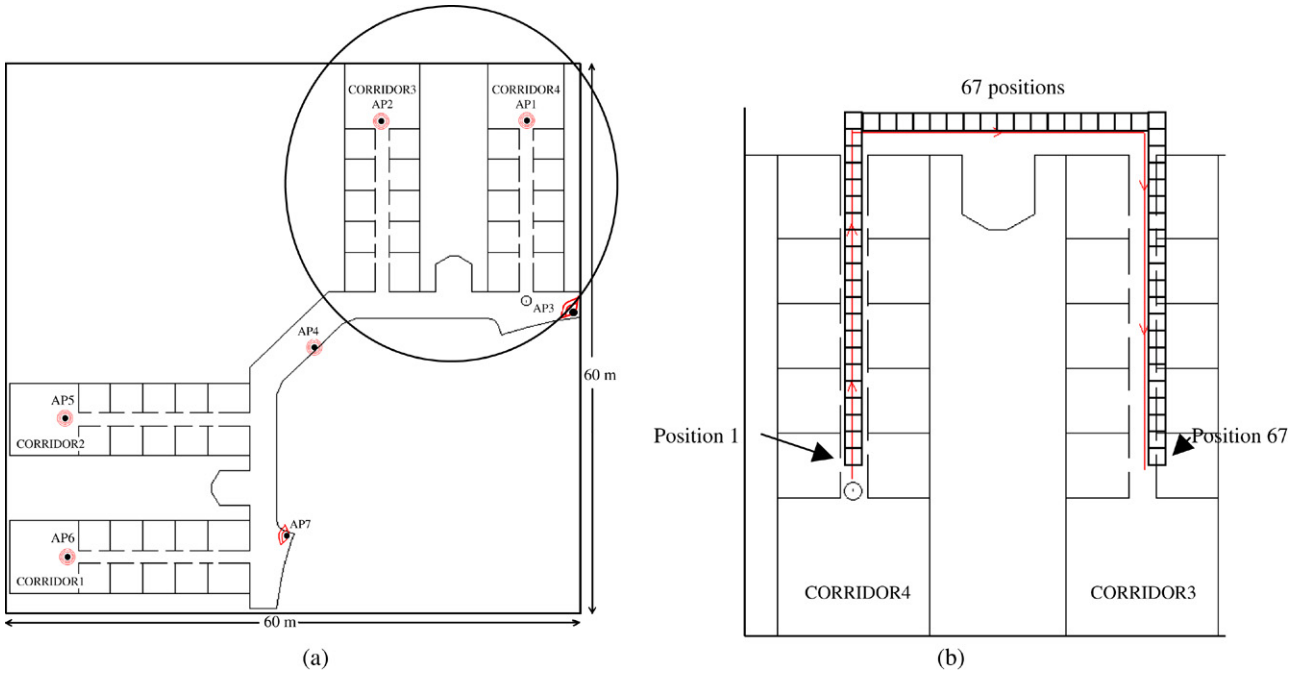

Fig. 1. (a) 3rd floor of the UAH electronics department. (b) Detailed view of the test-bed.

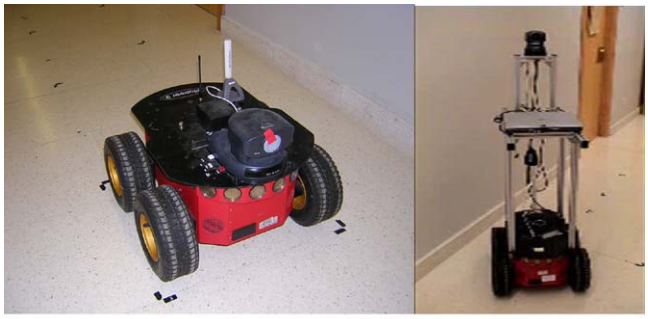

Fig. 2. Real prototypes used in the experimental setup. (a) BART, (b) SIRA.

connected to omnidirectional antennas and two APs (AP3 and AP7) were connected to antennas of 120° of horizontal beam-width. The APs act as wireless signal transmitters or base stations.

To obtain the measurements at the locations we have used two Pioneer 2AT robots, BART (Basic Agent for Robotic Tasks) and SIRA (Spanish acronym for Assistant Robotic System), which can simultaneously measure the WiFi signal from different APs using a scanning function. Fig. 2 depicts a picture of the robotic prototypes used in the experiments. The two robots have the same platform, but we have added a structure in the SIRA robot to carry a laptop and to increase the height of the camera.

2.2. Main WiFi signal variations

In this section we describe the main variations suffered by WiFi signals in indoor environments. In [10], the authors identify three main causes for the variation of the signal strength in an indoor environment:

(1) Temporal variations: variations standing at a fixed position for a long time.
(2) Large-scale variations: the signal strength varies over a long distance due to attenuation.
(3) Small-scale variations: these variations happen when the robot moves over a small distance and it is due to the signal wavelength (at 2.4 GHz the wavelength is 12.5 cm, then this effect will appear for distances below 12.5 cm).

We have measured two additional parameters that are directly related with the low level controller. These parameters are:

(4) Large orientation variations: these are the variations suffered by the measurement when the robot is positioned at the same node, but in the four basic orientations (North–South–East–West).
(5) Small orientation variations: the signal strength varies when the robot is positioned at the same node but with a slight difference with respect to the reference orientation.

In [11], the authors perform different tests to get these variations in the test-bed environment. To test the temporal variations we collected samples over a complete day (Friday to Saturday) from the two nearest Access Points (AP1 and AP2) and at a fixed robot position near AP1. The sampling rate was 1 s. The signal strength obtained from AP1 (mean $= -56.8$ dBm, $\sigma = 4.5$ dBm) was larger than the one obtained from AP2 (mean $= -70$ dBm, $\sigma = 3.7$ dBm). The reason for this lies in the fact that AP1 is closer to the robot than AP2. Additionally, the signal coming from AP2 has to get through two walls with the corresponding attenuation. The other conclusion is that the standard deviation of AP1 signal was larger than the one from AP2. This is due to the fact that the effect of secondary paths has less influence for AP2 than for

AP1. Then, the signal received from AP2 is mainly due to the direct path, while the signal received from AP1 exhibits a high influence from multi-path fading.

For testing large-scale variations, the signal strength from all APs was collected with the robot positioned across the 67 positions shown in Fig. 1(b). We took 300 samples for each position and then we obtained the average value. The variation of the average signal strength over a distance of 18 m was about 20 dBm. Moreover, there is not a linear variation of the signal with the distance due to the multi-path effect. This is the reason why it is very difficult to build a propagation model for indoor environments.

For demonstrating small-scale variations we took several measures from all APs at different points across the environment, separated by a short distance (<12.5 cm). Then we generated a histogram for each case. Variations up to ±2 dBm were measured in a distance smaller than $\frac{1}{2}\lambda$ (6.25 cm) with different profiles for the histograms. The highest variations, up to ±6 dBm, were measured in distances between $\frac{1}{2}\lambda$ and $\frac{3}{4}\lambda$. We also analysed the effect of the robot orientation in our environment. For this purpose, we took 300 samples of WiFi signal strength and then obtained their histograms in the four basic orientations in order to observe that it is possible to obtain the orientation, and not only the position of the robot. The histogram profile is different for the test orientations, having a maximum difference in the average signal for the test cases of 8 dBm.

For measuring the small orientation variations we placed the robot in several nodes of the environment. We then obtained the histograms for all APs with 300 samples at different small orientations, with a variation of 3° with regard to the reference orientation. Variations up to ±1 dBm were measured in orientations smaller than 9°.

To conclude this section, we can state that the large scale and orientation variations are useful for robot localization while the small scale and orientation variations are pernicious for this system. Therefore, if the small scale error is related to the lateral error of the low level controller, and the small orientation error is related to the orientation error of the low level controller, we need to design a low level controller that meets the following constraints:

(1) The lateral error must be smaller than $\frac{1}{2}\lambda$ (6.25 cm).
(2) The orientation error must be smaller than 9°.

3. Description of the POMDP

While in a Markov Decision Process (MDP) the environment observation is free of uncertainty, in real robotic systems there are some uncertainties associated to their sensor observations, especially when the sensors are WiFi signal strength detectors. The MDP considers that only the effect of the actions has uncertainty.

When a MDP performs a series of execution steps along states $(s_0, s_1, \ldots, s_n)$, executing actions $(a_0, a_1, \ldots, a_n)$, the probability of being in state s_{t+1} in the $t+1$ execution step is given by Eq. (1).

$$p(s_{t+1} \mid s_0, a_0, s_0, a_0, \ldots, s_t, a_t) = p(s_{t+1} \mid s_t, a_t). \quad (1)$$

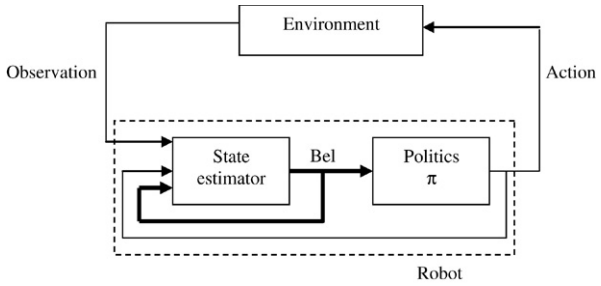

Fig. 3. POMDP structure.

This expression indicates that the current state (s_{t+1}) depends only on the previous state (s_t) and the previous action (a_t). This is known as the *Markov Property*.

When a noisy sensor such as the WiFi signal strength is used, observations with uncertainty are obtained from the sensors. This case is known as a partially observable process.

The POMDPs are mathematical models that permit us to characterize this type of system. A POMDP is defined by the same elements as a MDP: S (states set), A (actions set), T (transition function), R (recompense function); as well as by the following elements: O is the observations set ($o \in O$) and υ is the observation function [12–14].

A POMDP does not provide the real state because the observation has uncertainty. A POMDP maintains a belief distribution called Bel(S) or *Belief Distribution (Bel)* over the states to solve it. This distribution assigns to each state s a probability that indicates the possibility of being the real state. This is the main reason to divide the control stage of a POMDP in two blocks, as can be seen in Fig. 3:

(1) *State estimator*: the input of this block is the current observations and its output is the Bel function. This block obtains the probability over all possible states.
(2) *Politics*: the input of this block is the current Bel and its output is the action to perform. This block obtains the optimal action to perform in the next execution step to maximize the recompense (R).

The Belief Distribution must be updated whenever a new action or observation is carried out. When an action a is executed and a new observation o is taken, the new probabilities are obtained as in (2)

$$\mathrm{Bel}_t(s') = \eta \times p(o \mid s') \times \sum_{s \in S} p(s' \mid s, a) \times \mathrm{Bel}_{t-1}(s), \forall s' \in S. \quad (2)$$

In the context of robot navigation, the states of the Markov model are the localizations (or nodes) of the topological representation of the environment. Actions are local navigation behaviours that the robot executes to move from a state to another, and observations are perceptions of the environment that the robot can extract from its sensors. In this case, the Markov model is partially observable because the robot may never know exactly which state it is in. To solve the POMDP model the EM algorithm is used.

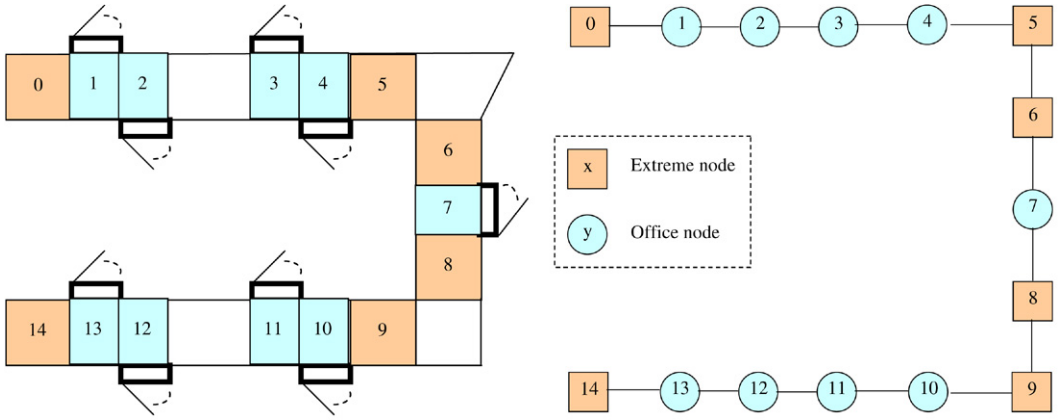

Fig. 4. (a) Example of environment discretization and (b) its topological representation.

Table 1
Actions set

Action	Symbol	Function	States where it is possible to execute
Follow corridor	a_F	Continue through the corridor to the next state	Office nodes
No operation	a_{NO}	Used as a directive in the goal state	Office and extreme nodes
Turn around	a_T	Change the navigation direction	Extreme nodes
Turn right	a_R	Turn 90° to the right	Extreme nodes
Turn left	a_L	Turn 90° to the left	Extreme nodes

3.1. *Design of the* POMDP *navigation system*

In this section we describe the design of our navigation system using a POMDP based on WiFi and ultrasound observations. This design includes: the environment representation, the states set, the observations type selection, the possible actions of the robot, and the transition and observation matrix.

3.1.1. *Environment representation*

In a topological representation of the environment, the discretization degree is the most important parameter to select because the processing time depends directly on it. In the topological map, the nodes should be useful for the robot patrol application. In this case the robot must be able to navigate in an autonomous mode inside the corridors. The robot must be able to stop in front of all the office doors in order to get into the rooms. This last maneuver will be carried out in teleoperated mode.

Fig. 4(a) shows an environment example, where the corridors are discretized into coarse-grained regions (nodes) of variable size. The limits of these nodes correspond to any changes in lateral features of the corridor (door, opening or piece of wall). We have selected two types of nodes: office nodes (nodes that are in front of office rooms) and extreme nodes (nodes that are at the end of the corridor). The extreme sensor can have a connection with a corridor in the left or in the right, or with an ending room. In Fig. 4(b) the topological representation is shown, where the extreme nodes are represented by squares and office nodes are represented by circles.

Local navigation in corridors is carried out based on ultrasound range measurements. The low level controller keeps the robot in the centre of the corridor while navigating, indicating to the POMDP the occurrence of a transition, such as a door or a corridor ending.

3.1.2. *States set*

With this topology, the states of the Markov model are directly related to the nodes of the topological graph. Two states are assigned to each corridor node, one for each of the two main orientations (forward, backward) that the robot can adopt during corridor following. We denote as *forward* direction the direction from node 0 to node 14 in Fig. 4(a), and as *backward* direction the opposite one.

3.1.3. *Actions set*

The actions set has been selected to establish correspondences between transitions from one state to another and local navigation behaviors of the robot. We assume imperfect actions. Thus, the effect of an action can be different from the expected one (this is modeled by the transition model T).

The action set is very simple in our application due to the configuration of the states and the local navigation system. The actions set, depicted in Table 1, is executed by the local navigation system.

3.1.4. *The observations*

We select two kinds of observations in our model: the novel WiFi signal strength measurement observation (*obswifi*) and the ultrasound observation (*obsus*).

The *obswifi* is obtained as the average value of 60 samples of the signal strength, received in the WiFi robot interface from all APs. This filtering is carried out in order to minimize the high noise of the WiFi signal measures. The number

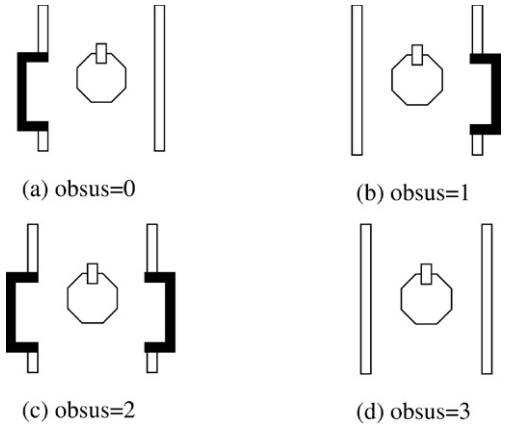

Fig. 5. Possible combinations for *obsus*: (a) door on left side, (b) door on right side, (c) door on both sides and (d) no doors detected.

of samples has been obtained in a experimental way for optimal localization. The obswifi is then divided in N different observations ($obswifi_{AP1}, obswifi_{AP2}, \ldots, obswifi_{APN}$), one for each access point.

The mean signal is then rounded to the closest integer value in order to obtain a discrete space of values. The possible values that can be obtained from the WiFi interface range from 0 to −99 dBm. We have added an offset to the measure so as to obtain a positive value that can serve as an index in the observation matrix.

The *obsus* is obtained from the ultrasound sensors. Four different observations are established: door on the left, door on the right, door on both sides, and wall on both sides. This way, the possible values are discrete and useful to index the observation. Fig. 5 shows the combinations for the *obsus* and the associated codified values. This observation is easily obtained from the ultrasound sensors, and leads to a robust local observation that can be fused with global WiFi observations, as we explain later on in this paper.

3.1.5. Sensor fusion

Observations from WiFi and ultrasound sensors are complementary. The first one obtains an estimation of the global localization while the second one obtains a precise estimation of the local environment. The fusion of these observations will produce a good observability of states.

POMDP provides a natural way for using multisensorial fusion in their observation models ($p(\vec{o} \mid s)$) by means of Bayes' rule. Because these are independent observations, the observation model can be simplified in the following way:

$$\begin{aligned} p(\vec{o} \mid s) &= p(obswifi_{AP1}, \ldots, obswifi_{APN}, obsus \mid s) \\ &= p(obswifi_{AP1} \mid s) \times \cdots \times p(obswifi_{APN} \mid s) \\ &\quad \times p(obsus \mid s). \end{aligned} \tag{3}$$

In Eq. (3) $\vec{o}$ is a vector composed of two types of observations: obswifi (provided by the WiFi signal strength) and obsus (provided by the ultrasound sensors).

3.1.6. Actions uncertainty model

The actions uncertainty model represents the real errors or failures in the execution of actions. The transition function T incorporates this information to the POMDP. In our case, T is a matrix that represents the probability of reaching state s_{t+1} when the robot is in the state s_t and it has executed action a_t. The matrix dimensions are $n\_est \times n\_est$, where n_est is the total number of states in the topological map.

If the action has no uncertainty, the robot executes an F action ("Forward"). Consequently, the robot advances to the next state, which is the ideal case. In a real situation there may appear some errors that introduce uncertainty in the actions. Some of them are described in the following:

(1) If a person is in the hollow of the door and he is blocking the depression of the door, the local navigator shall detect a depression with a width less than a door. As a consequence of this, the local navigator will not validate it as a door. The error in this case is called FF ("Forward–Forward"), i.e. the robot goes two states forward instead of one. This case can get repeated for two or more doors, then the error will be called FFF (for two blocked doors), FFFF (for three blocked doors) and so on.
(2) If somebody leaves an object in a corridor with the same width as a door, the local navigator shall validate the object as a door. This error is called NO ("No-Operation"). In this case, the robot does not reach the next door.

3.1.7. Observation uncertainty model

The observation uncertainty model represents the real errors or failures of the sensor systems (ultrasound ring and WiFi interface). The observation function υ incorporates this information in the POMDP. In this work, υ is a matrix for each observation (seven for WiFi observations and one for ultrasound observations). The matrix dimensions are $n\_est \times obs\_values$, where obs_values is the possible observation values in the current state.

The ultrasound observation uncertainty is bound to the same cases as in the actions uncertainty. For the WiFi observations, the error sources are more complex. In indoor environments the WiFi signal is affected by different factors:

1. Reflections, refractions, and diffractions: that in indoor environments is known as multi-path fading.
2. Water resonant frequency: WiFi technology operates at 2.4 GHz, which is the water resonant frequency. Therefore, all persons in the environment can modify the signal strength received.
3. Free band frequency: we also remark that this frequency is in the free band frequency where several applications operate, such as Bluetooth technology, very common in wireless keyboards and mice.

Due to these factors the signal strength measurement can be largely modified with respect to the ideal value, and this variation changes as a function of time.

3.1.8. Training method for obtaining the transition and observation matrices

In many real systems using POMDPs, the values of the transition and observation matrices are obtained with a simple deduction or using a priori expertise. In our case, we use the

ability of our low level controller to build an autonomous learning system. The robot navigates in autonomous mode using ultrasound information alone. The initial state is known beforehand. Actions are stored in the training action set. Then, the robot obtains the WiFi and ultrasound observations at each transition. This information represents the training observations set. After that, in an off-line stage, we carry out SLAM, based on a Baum–Welch algorithm (EM algorithm), using the training sets in order to yield the transition and observation matrices. We have denoted this technique as WSLAM (Wifi-SLAM).

This process constitutes a learning phase in which the robot learns the transition and observation matrices. Once this phase has finished, a tracking phase is executed to track the robot using the previous matrices.

3.1.9. Politics π

There are different algorithms to solve the selection of the ideal action to execute in each state. In a POMDP the problem is more complex than in a MDP because the current state is unknown. Only a belief distribution is maintained in a POMDP.

In this work we use the *Most Likely State* (MLS) method to select the optimal action, because the global observation provided by the WiFi sensor normally obtains a belief distribution that exhibits a maximum at the real state. This method selects the optimal action associated to the most probable state of the Belief Distribution as shown in (4)

$$a = \pi_{\mathrm{MLS}}(\mathrm{Bel}) = \pi * (\arg\max_{S} \mathrm{Bel}(s)). \tag{4}$$

4. Low level controller

The local navigation system has three main goals: execute the action commanded from the POMDP, inform it when a state transition is detected and place the robot in the optimal location to measure the WiFi signal. To carry out the state transition detection, the low level controller uses ultrasound range measurements. The local navigation system is able to navigate a robot in a corridor, whatever the state of its doors and the position of persons walking around it. For this purpose, the detection of doors in the side of the corridors and the ending corridor detection are used as key information for local navigation purposes. Using a door detection procedure instead of relying on dead-reckoning leads to more robust performance, as demonstrated in practice, especially for large corridors. Door detection accuracy can be highly improved by using a model of a corridor. Thus, deviations from the model can be regarded as doors, as will be described later on.

When the robot enters a corridor it needs to get the appropriate orientation in parallel with respect to the corridor walls. After that, a model of the corridor is estimated based on ultrasound range measurements. The robot's local position (lateral and orientation error with regard to the centre of the corridor) is measured based on the model of the corridor. Then, a lateral controller leads the robot to the centre of the corridor while navigating. When the robot reaches the end of the corridor a state transition is issued.

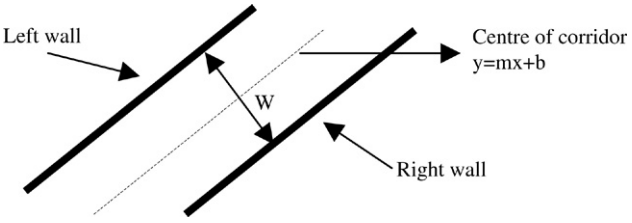

Fig. 6. H-shape corridor model.

4.1. Estimation of corridor model

As previously mentioned, 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, an H-shaped 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 ($y_{\mathrm{left}} = m_{\mathrm{left}}x_{\mathrm{left}} + b_{\mathrm{left}}$, $y_{\mathrm{right}} = m_{\mathrm{right}}x_{\mathrm{right}} + b_{\mathrm{right}}$) are obtained based on ultrasound measures using least squares operators, yielding $m_{\mathrm{left}}, b_{\mathrm{left}}, m_{\mathrm{right}}, b_{\mathrm{right}}$, that stand for the slope (m) and bias (b) of the straight lines representing the left and right walls, respectively, with respect to a non-inertial reference frame located at the robot centre of gravity. In the previous computation, variables $y_{\mathrm{left}}, x_{\mathrm{left}}, y_{\mathrm{right}}, x_{\mathrm{right}}$ are also 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, as depicted in Fig. 6. 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 to Eq. (5)

$$m = \tan\left(\frac{\arctan m_{\mathrm{left}} + \arctan m_{\mathrm{right}}}{2}\right). \tag{5}$$

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 Eq. (6)

$$E_{\mathrm{ms}} = \sum_{n=1}^{n=N_{\mathrm{right}}} (y_{\mathrm{ln}} - y_{\mathrm{H\,ln}})^2 + \sum_{n=1}^{n=N_{\mathrm{left}}} (y_{\mathrm{rn}} - y_{\mathrm{Hrn}})^2 \tag{6}$$

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

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

Box I.

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 Fig. 6.

The left and right edges of the corridor can then be constrained by parameters m, b, and the corridor width, denoted by W. This permits us 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 Eq. (7)

$$\begin{aligned}\frac{\mathrm{d}E_{\text{ms}}}{\mathrm{d}b} = &\sum_{n=1}^{n=N_{\text{left}}} 2 \cdot (y_{\text{ln}} - mx_{ln} - b) \cdot (-1) \\ &+ \sum_{n=1}^{n=N_{\text{right}}} 2 \cdot (y_{\text{rn}} - mx_{\text{rn}} - b \\ &+ W \cdot (\sin\arctan m - m\cos\arctan m)) \cdot (-1) \qquad (7)\end{aligned}$$

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 Eq. (7) with respect to parameter b yields the value expressed in Box I. 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. Fig. 7 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 Fig. 7. To avoid perturbations such as doors and moving obstacles, the distance between the measured points and the previous corridor model must be below 5 cm in order to validate the measurement and include it in the estimation process.

The validation process allows us to easily detect doors in the corridor walls. Considering the 5 cm validation band, all doors (either open, closed, or ajar) will be detected as an open gap in the wall model, as depicted in Fig. 7. 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's typical width. In case the corridor is overcrowded with moving obstacles, the current door detection process could fail and further sensors, such as vision cameras, should be needed for robust door detection. From the equation given in Box I, variables b_L and b_R for the left and

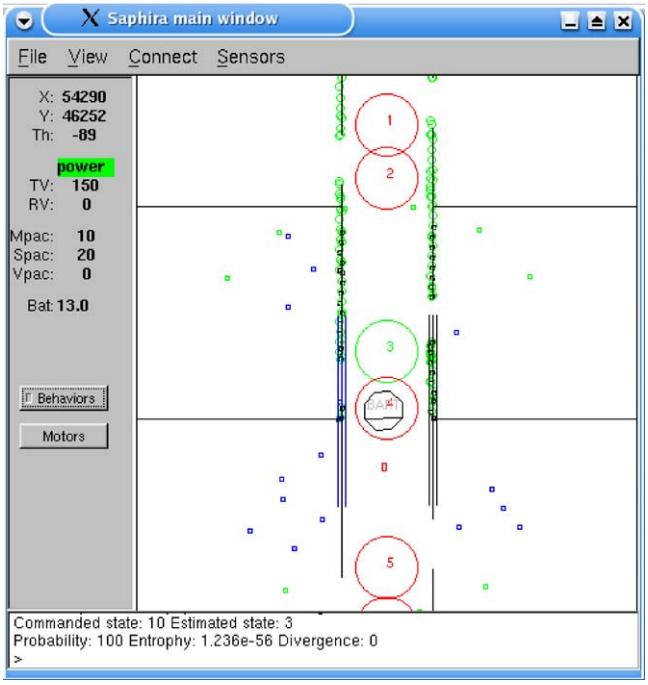

Fig. 7. Online estimation of corridor model.

right edges of the wall, respectively, can be demonstrated to be as expressed in Eq. (8)

$$\begin{aligned} b_L &= b - \frac{W}{2}\sqrt{m^2+1} \\ b_R &= b + \frac{W}{2}\sqrt{m^2+1} \end{aligned} \qquad (8)$$

where W stands for the corridor width. Based on the equation given in Box I and Eq. (8) the lateral and orientation error of the robot with regard to the centre of the corridor can be computed as in Eq. (9)

$$\begin{aligned} \theta_e &= \tan^{-1} m \\ d_e &= \frac{b_R}{\sqrt{1+m^2}} - \frac{W}{2} \end{aligned} \qquad (9)$$

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 Fig. 8.

4.2. 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 constraint can be formulated into the minimization of the lateral and orientation errors. Thus, a model describing the

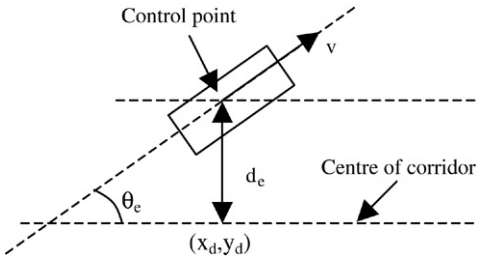

Fig. 8. Lateral and orientation errors (d_e, θ_e) with respect to the centre of the corridor.

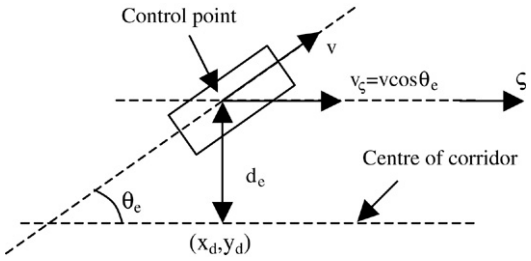

Fig. 9. Graphical description of variable ς.

dynamic behavior of d_e and θ_e is needed in order to design a stable lateral controller.

4.2.1. 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 ω 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 Eq. (10)

$$\begin{aligned} \dot{x} &= v \cdot \cos\theta \\ \dot{y} &= v \cdot \sin\theta \\ \dot{\theta} &= \omega. \end{aligned} \tag{10}$$

Let's remark that v represents the velocity of the robot centre of gravity, denoted as the 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 Fig. 8, 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 Eq. (11)

$$\begin{aligned} d_e &= -(x - x_d) \cdot \sin\theta_d + (y - y_d) \cdot \cos\theta_d \\ \theta_e &= \theta - \theta_d. \end{aligned} \tag{11}$$

The derivatives of d_e and θ_e with respect to time are demonstrated in Eq. (12), while the complete non-linear kinematic model for d_e and θ_e is formulated in Eq. (13)

$$\begin{aligned} \dot{d}_e &= -\dot{x}\sin\theta_d + \dot{y}\cos\theta_d = v\sin(\theta - \theta_d) = v\sin\theta_e \\ \dot{\theta}_e &= \frac{\mathrm{d}(\theta - \theta_d)}{\mathrm{d}t} = \dot{\theta} - \dot{\theta}_d = \dot{\theta} = \omega \end{aligned} \tag{12}$$

$$\begin{aligned} \dot{d}_e &= v\sin\theta_e \\ \dot{\theta}_e &= \omega. \end{aligned} \tag{13}$$

4.2.2. 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 [15]. An excellent example on this topic can be found in [16,17]. 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 proposed in order to convert the non-linear system into a quasi-linear one, termed as a chained form. Thus, the non-linear model for d_e and θ_e can be transformed into a chained form using the state diffeomorphism and change of control variables shown in Eq. (14)

$$\begin{aligned} Y &= \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \Theta(X) = \begin{pmatrix} d_e \\ \tan\theta_e \end{pmatrix} \\ W &= \begin{pmatrix} \omega_1 \\ \omega_2 \end{pmatrix} = \Gamma(U) = \begin{pmatrix} v\cos\theta_3 \\ \dfrac{\omega}{\cos^2\theta_e} \end{pmatrix}. \end{aligned} \tag{14}$$

These transformations are invertible whenever the robot linear velocity is different from zero. Based on Eq. (15), the kinematic model can be rewritten as in Eq. (15), considering y_1 and y_2 as the new state variables.

$$\begin{aligned} \dot{y}_1 &= v\sin\theta_e = \omega_1 y_2 \\ \dot{y}_2 &= \frac{\mathrm{d}(\tan\theta_e)}{\mathrm{d}t} = \frac{1}{\cos^2\theta_e} \cdot \dot{\theta}_e = \frac{\omega}{\cos^2\theta_e} = \omega_2. \end{aligned} \tag{15}$$

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 Fig. 9.

Analytically, ς is computed as the integral of linear velocity v_ς, measured along axis ς. The expression of ς is provided in Eq. (16)

$$\begin{aligned} \varsigma &= \int v_\varsigma \mathrm{d}t = \int v\cos\theta_e \mathrm{d}t \Rightarrow \dot{\varsigma} \\ &= \frac{\mathrm{d}\varsigma}{\mathrm{d}t} = v\cos\theta_e = \omega_1. \end{aligned} \tag{16}$$

The time derivative of state variables y_1 and y_2 is expressed as a function of ς in Eq. (17)

$$\begin{aligned} \dot{y}_1 &= \frac{\mathrm{d}y_1}{\mathrm{d}t} = \frac{\mathrm{d}y_1}{\mathrm{d}\varsigma} \cdot \frac{\mathrm{d}\varsigma}{\mathrm{d}t} = y_1' \cdot \dot{\varsigma} \\ \dot{y}_2 &= \frac{\mathrm{d}y_2}{\mathrm{d}t} = \frac{\mathrm{d}y_2}{\mathrm{d}\varsigma} \cdot \frac{\mathrm{d}\varsigma}{\mathrm{d}t} = y_2' \cdot \dot{\varsigma} \end{aligned} \tag{17}$$

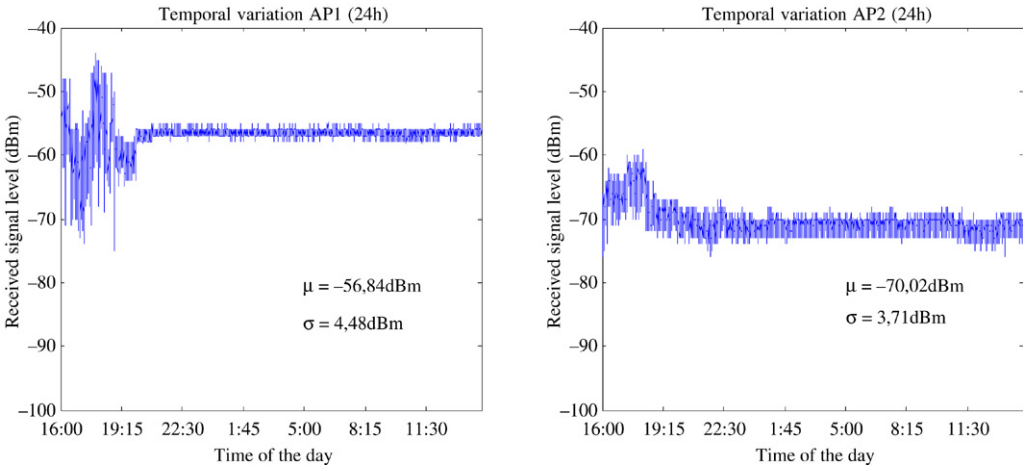

Fig. 10. Time invariance of the WiFi signal.

where y_1' and y_2' stand for the derivative of y_1 and y_2 with respect to ς. Solving for y_1' and y_2' leads to Eq. (18)

$$y_1' = \frac{\dot{y}_1}{\varsigma} = \frac{v \sin\theta_e}{v \cos\theta_e} = \tan\theta_e = y_2$$
$$y_2' = \frac{\dot{y}_2}{\varsigma} = \frac{\omega}{v \cos^3\theta_e} = \omega_3. \tag{18}$$

As observed in Eq. (18), 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 Eq. (19)

$$\omega_3 = -K_d y_2 - K_p y_1 \quad (K_d, K_p) \in \Re^{+2}. \tag{19}$$

Using Eqs. (18) and (19), the dynamic behavior of y_1 with respect to ς is demonstrated to be linear, as shown in Eq. (20)

$$y_1'' + K_d y_1' + K_p y_1 = 0. \tag{20}$$

This implies that variables $y_1 = de$ and $y_2 = \tan\theta_e$ tend to zero as variable ς grows. The previous statement is analytically expressed in Eq. (21)

$$\lim_{\varsigma\to\infty} d_e = \lim_{\varsigma\to\infty} .\theta_e = 0. \tag{21}$$

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 $-\pi/2 < \theta_e < +\pi/2$. In other words, the robot must continuously move forward and the absolute value of its orientation error should be below $\pi/2$ in order to guarantee proper trajectory tracking. Thus, the non-linear control law is finally derived from Eqs. (18) and (19) and formulated in Eq. (22)

$$\omega = \arctan\lfloor -v \cos^3\theta_e (K_d \tan\theta_e + K_p d_e)\rfloor. \tag{22}$$

The control law can be modified by a sigmoidal function to account for physical limitations in the robot wheels' turning angle and prevent actuator saturation. On the other hand, the use of sigmoidal functions preserves the system stability [18]. Finally, the tuning of constants (K_d, K_p) can be properly done by following the method proposed in [16].

5. Implementation and results

The local navigation system described in this paper was tested on the two prototypes shown in Fig. 2, BART and SIRA. The robots have the following configuration: Orinoco PCMCIA Gold wireless card, Linux Red Hat 9.0 operating system, wireless tools by Jean Tourrilhes [19], Orinoco driver patch by Moustafa A. Youssef, a 16 ultrasound sensor ring and a SONY pan–tilt–zoom camera.

Practical experiments were conducted in the 3rd floor of the Electronics Department in the University of Alcalá. This section is organized as follows: first, the WiFi signal is measured to obtain the constraints of the low level controller, then several experiments are carried out to test it as a valid low level controller for this application, and finally several statistics will be obtained.

Fig. 10 shows the result of measuring the WiFi signal during one complete day at a 1 second period in a stationary position of the environment. As we mentioned in Section 2, this test provides an idea about the time invariance of the signal.

It is important to note that the signal varies much more in AP1 than in AP2 during the operation time (from 16:00 to 21:30). This figure shows the presence of a multi-path effect in the WiFi signal measurement, as well as the pernicious effect of the people and wireless devices, such as Bluetooth keyboards and mice.

An example of large scale variations is shown in Fig. 11. The robot was located at the 67 positions shown in Fig. 1(b). We took 300 samples for each position and then we obtained the mean value. Fig. 11 represents these mean values for access points AP1 and AP2. The figure shows the difficulty in building a propagation model for indoor environments. This is the justification that supports the use of a previous radio map for WiFi localization systems in indoor environments.

The small scale variations are related to the lateral error of the low level controller as mentioned in Section 2. After several experiments in our environment we conclude that the maximum small scale variations that we obtained were in $\frac{3}{4}\lambda$ (approximately 9 cm). This distance represents the maximum lateral error that the controller can assume.

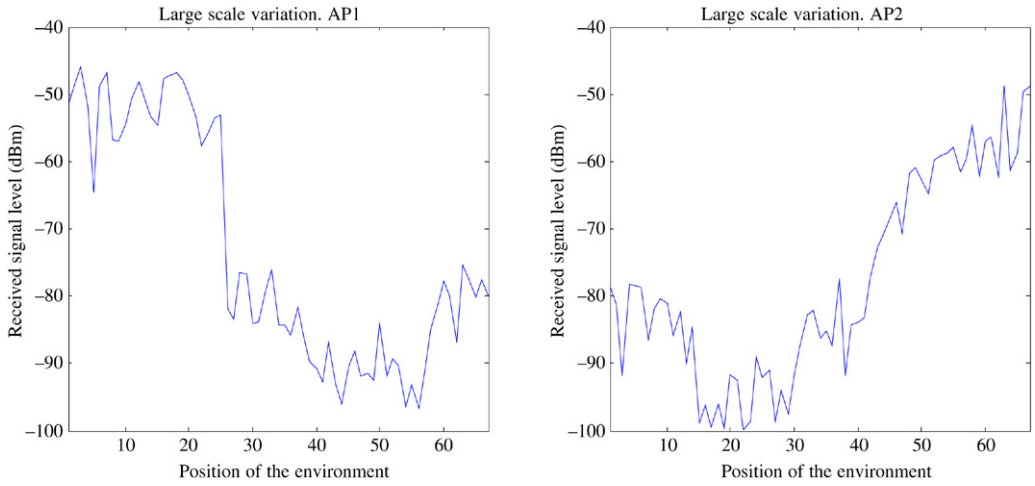

Fig. 11. Large scale variations in AP1 and AP2 WiFi signal measure.

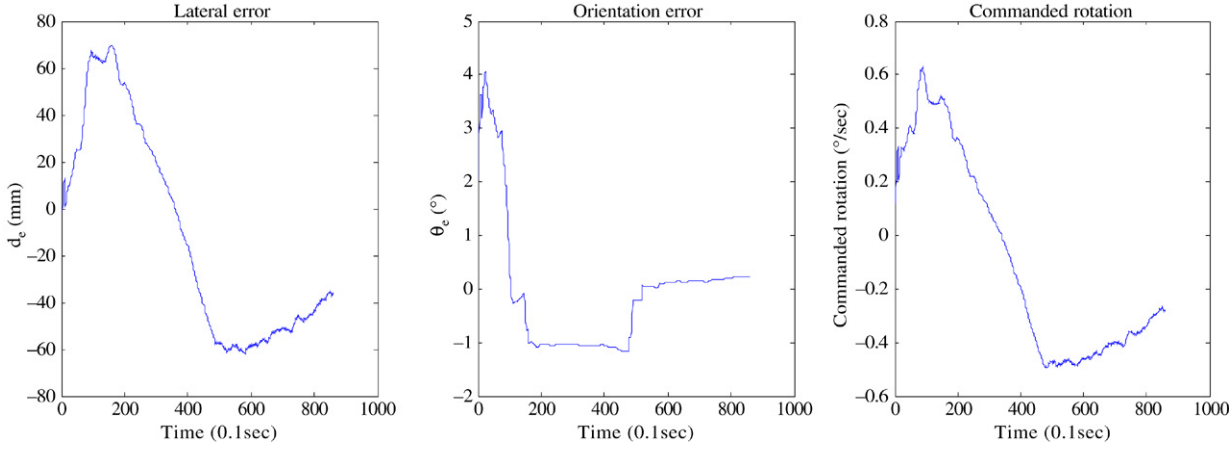

Fig. 12. Real results for a complete route in one corridor with $v = 20$ cm/s and initial orientation $= 3°$.

Another important variation that suffers the WiFi signal measure is the large orientation. For testing this variation we carried out a lot of experiments to extract some conclusions. For this purpose, the histograms of all visible APs at each position were obtained, concluding that these are useful for obtaining the four basic orientations. In this manner, we can extract the basic orientation of the robot. This is a very interesting point in a localization and navigation system for autonomous robots.

Other tests that we have carried out have demonstrated the presence of small orientation variations in the WiFi signal measurement. These small orientation variations represent a new constraint of the low level controller. The tests were carried out in several positions of our environment and we conclude that the maximum small orientation variations appear for orientations higher than 6°. Then, this is the maximum orientation error that the controller can assume.

Various practical trials were conducted to test the validity of the controller in the global navigation system for different initial conditions in real circumstances. During the tests, the reference robot velocity is assumed to be kept constant. The robot linear velocity is set to 20 cm/s and the initial orientation was fixed to 3°. Fig. 12 depicts the lateral error, the orientation error, and the commanded angular velocity during online operation, in which the robot was located at the beginning of a corridor. From that point the robot navigated until reaching the end of the corridor. Upon acquisition of ultrasound measures, the robot gets the adequate orientation in order to get parallel to the corridor walls. As observed in Fig. 12, both d_e and θ_e tend to zero as the robot moves forward.

In the next trial the reference robot velocity was also assumed to be kept constant. The robot linear velocity is set to 20 cm/s and the initial orientation was fixed to 3°. Fig. 13 depicts the lateral error, the orientation error, and the commanded angular velocity during this new operation in which the robot was located at the end of corridor 4 (see Fig. 1). From that point the robot navigated until reaching the end of corridor 3, carrying out the corresponding changes of corridor employing a basic spline. As can be observed in the next figure, both d_e and θ_e tend again to zero as the robot moves forward. It is important to note that the lateral error was not higher than 6 cm and the orientation error was not higher than 4° even when the robot carried out the changes of corridor (for times 80 s and 140 s).

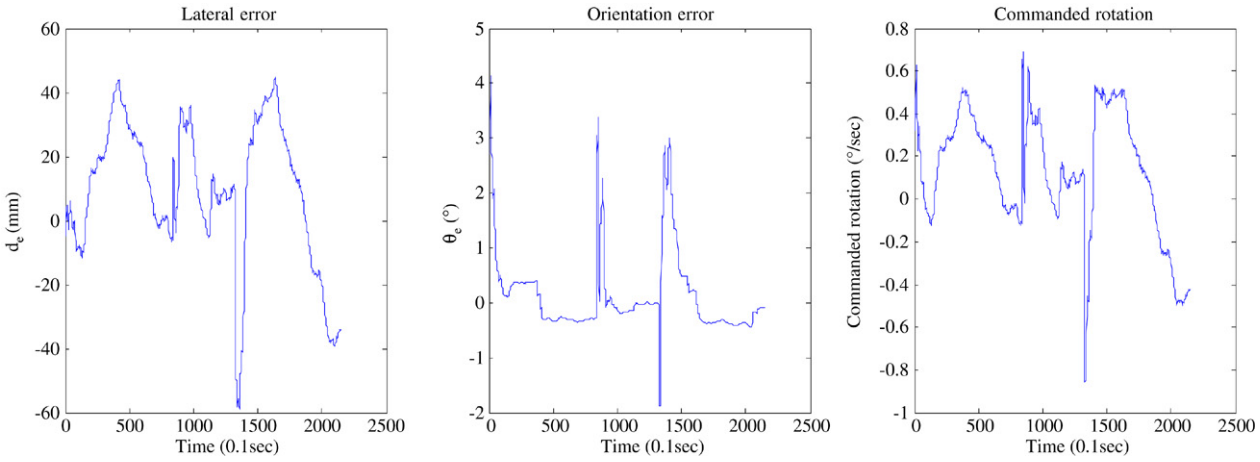

Fig. 13. Real results for a complete route in three corridors with $v = 20$ cm/s and initial orientation $= 3°$.

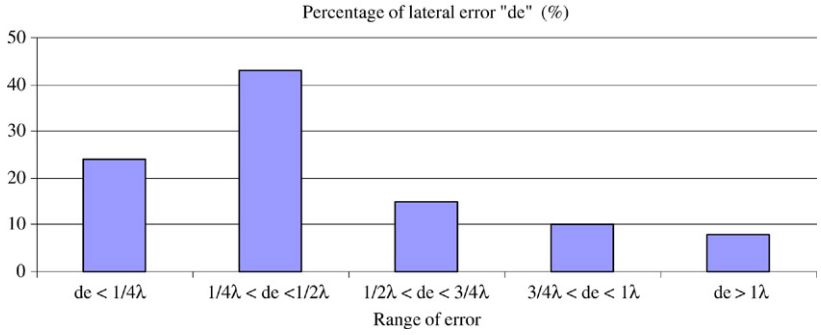

Fig. 14. Results from 40 experiments in lateral error (%).

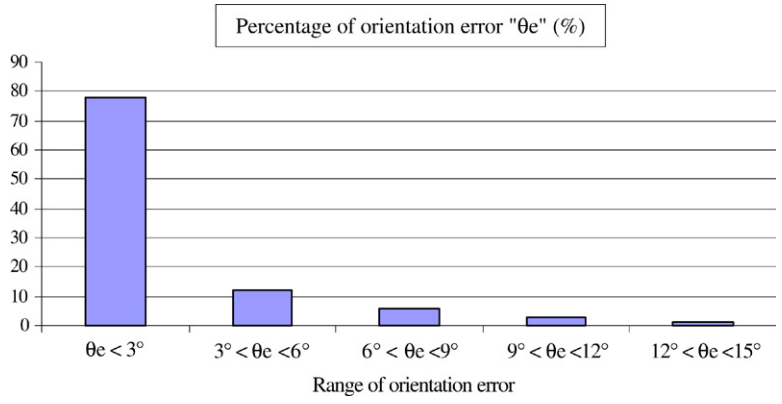

Fig. 15. Results from several experiments in orientation error (%).

We have programmed several targets for the POMDP and obtained the lateral and orientation error of the robot. Fig. 14 shows the statistics for the lateral error obtained from 40 different targets. The lateral error has been divided by ranges referenced to the wavelength. As can be seen in Fig. 14, the accumulated probability that the lateral error is less than $\frac{1}{2}\lambda$ is 70%.

With the same 40 targets in the POMDP we have obtained the orientation error. Fig. 15 shows a statistic of orientation error. The errors have been divided by ranges, but in this case they are referenced in degrees. The accumulated probability that the orientation error is less than 6° is 90%.

In both cases the accumulated percentage between the constraint margins allows us to obtain WiFi observations with low uncertainty. Therefore we can state that the low level controller developed in this work meets the constraints for being successfully deployed in a WiFi-based localization and navigation system.

To check the robustness of the global system as a function of time, we programmed 50 chained targets. The robot was

Table 2
Global results comparison with López 2004

		This work		López 2004	
		Number	Percentage (%)	Number	Percentage (%)
Successful	Direct path	45	90	35	70
	Indirect path	4	8	11	22
Failures	Incorrect target	1	2	2	4
	Loops	0	0	2	4

navigating for 3 hours and we obtained a few localization errors as can be seen in Table 2. These results are compared with the results obtained in the work of López (2004) [13] and Simmons (1997) [14], where the observations used were ultrasound and computer vision.

"Direct path" means that the target is reached following the ideal trajectory while "Indirect path" means that the target is reached after a recovering maneuver. As can be seen in the table, the total percentage of successful navigations was slightly higher than in López (2004), 98% versus 92%, but it is interesting to note that the direct path percentage was 90% which is much higher than in López (2004) with 70%. In Simmons (1997), the authors only obtained 90% of successful navigations in similar environments.

6. Conclusions

To conclude, the next key points should be remarked.

- First of all, the low level controller described in this work has been proved in two real Pioneer 2AT robots within a POMDP based on WiFi observations. It has been demonstrated that it is useful to obtain low uncertainty observations as long as they exhibit small scale and orientation errors.
- We have presented a navigation system for autonomous robots in indoor environments using a POMDP based on WiFi and ultrasound observations. According to the authors' knowledge this is the first work that uses this kind of observation in a POMDP.
- By combining a low level controller and a POMDP, a robust autonomous navigation system for indoor environments has been achieved. In fact, the robot is able to recover its position in spite of sensor uncertainty. It is specially recommended for localization and navigation in environments that are structured into corridors, where the H-shape model is easily obtained by means of ultrasound sensors.

In future, we will try to speed up the algorithm. Then, WiFi observations will be taken only in the interesting states, such as at the end of corridors, thus obtaining a faster and more robust algorithm for real applications.

Acknowledgements

This work has been funded by grant IST-2001-34508 from the European Commission through ADVOCATE II Research Project, GR/SAL/0860/2004 (TEAPEM Project) from the Spanish CAM, and DPI2002-02193 (SIRAPEM Project) from the Spanish Ministry of Science and Technology (MCyT).

References

[1] M.E. López, L.M. Bergasa, R. Barea, M.S. Escudero, A navigation system for assistant robots using visually augmented POMDPs, Autonomous Robots 19 (1) (2005) 77–87.
[2] I. Cox, Blanche-an experiment in guidance and navigation of an autonomous robot vehicle, IEEE Transactions on Robotics and Automation 7 (2) (1991) 193–204.
[3] R. Want, A. Hopper, V. Falco, J. Gibbons, The active badge location system, ACM Transactions on Information Systems 10 (1992) 91–102.
[4] J. Krumm, S. Harris, B. Meyers, B. Brumitt, M. Hale, S. Shafer, Multi-camera multi-person tracking for easy living, in: Proceedings of 3rd IEEE International Workshop on Visual Surveillance, 2002, pp. 3–10.
[5] N.B. Priyantha, A. Chakraborthy, H. Balakrishnan, The cricket location support system, in: Proceedings of the 6th ACM MobiCom, 2002, pp. 155–164.
[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, in: Proceedings of 28th Annual Conference of the IEEE Industrial Electronics Society, 2002, pp. 2779–2784.
[7] P. Bahl, V.N. Padmanabhan, RADAR: A, In-building RF-based user location and tracking system, in: Proceedings of the IEEE Infocom 2000, vol. 2, 2000, pp. 775–784.
[8] R. Casas, A. Marco, J.J. Guerrero, J. Falcó, Robust estimator for non-line-of-sight error mitigation in indoor localization, in: EURASIP Journal on Applied Signal Processing, 2006, pp. 1–8.
[9] P. Enge, P. Misra, Special issue on GPS: The global positioning system, in: Proceedings of the IEEE, January 1999, pp. 3–172.
[10] M. Youssef, A. Agrawala, Small-scale compensation for WLAN location determination systems, in: Proceedings of the 2003 ACM Workshop on Wireless security, 2003, pp. 11–20.
[11] M. Ocaña, Sistema de localización global WiFi aplicado a la navegación de un robot semiautónomo, Ph.D. Thesis, Department of Electronics, University of Alcalá, Spain, 2005.
[12] E. Lopez, R. Barea, L.M. Bergasa, M.S. Escudero, A human-robot cooperative learning system for easy installation of assistant robots in new working environments, Journal of Intelligent and Robotic System 40 (2004) 233–265.
[13] M.E. López, Sistema de navegación global basado en procesos de decisión de Markov parcialmente observables, Aplicación a un robot de asistencia personal, Ph.D. Thesis, Department of Electronics, University of Alcalá, Spain, 2004.
[14] R. Simmons, R. Coodwin, K.Z. Haigh, S. Koenig, J. O'Sullivan, A layered architecture for office delivery robots, in: Proceedings of the First International Conference on Autonomous Agents, Agents'97, 1997, pp. 245–252.
[15] J. Luo, P. Tsiotras, Control design for systems in chained form with bounded inputs, in: Proceedings of the American Control Conference, Philadelphia, 1998.
[16] M.A. Sotelo, Lateral control strategy for autonomous steering of Ackerman-like vehicles, Robotics and Autonomous Systems 45 (2003) 223–233.

[17] L. Cordesses, P. Martinet, B. Thuilot, M. Berducat, GPS-based control of a land vehicle, in: Proceedings of the IAARC/IFAC/IEEE International Symposium on Automation and Robotics in Construction, ISARC'99, Madrid, 1999.
[18] H. Sussmann, E. Sontag, Y. Yang, A general result on the stabilization of linear systems using bounded controls, IEEE Transactions on Automatic Control 39 (12) (1994) 2411–2425.
[19] Personal web page: http://www.hpl.hp.com/personal/Jean_Tourrilhes/.

M.A. Sotelo received the Ing. degree in Electrical Engineering in 1996 from the Technical University of Madrid, and the Ph.D. degree in Electrical Engineering in 2001 from the University of Alcalá, Alcalá de Henares, Madrid, Spain. From 1993 to 1994 he has been a Researcher at the Department of Electronics, University of Alcala, where he is currently an Associate Professor. His research interests include Real-time Computer Vision and Control Systems for Autonomous and Assisted Intelligent Road Vehicles. He has been a recipient of the Best Research Award in the domain of Automotive and Vehicle Applications in Spain, in 2002, the 3M Foundation Awards in the category of eSafety in 2003 and 2004, and the Best Young Researcher Award conceded by the University of Alcalá in 2004. He has been serving as Auditor and Expert at FITSA Foundation for R+D Projects in the domain of automotive applications since September 2004. He is the author of more than 100 refereed publications in international journals, book chapters, and conference proceedings. He is an IEEE member, as well as a member of the IEEE ITS Society and member of the ITS-Spain Committee.

M. Ocaña received his Ing. Degree in Electrical Engineering in 2002 from the University of Alcalá, and his Ph.D. degree in Electrical Engineering in 2005 from the University of Alcalá, Alcalá de Henares, Madrid, Spain. From 2002 to 2005 he has been researcher at the Department of Electronics, University of Alcalá, where he is currently an Associate Professor. He has been a recipient of the Best Research Award for the 3M Foundation Awards in the category of eSafety in 2003 and 2004. His research interests include robotics localization and navigation, assistant robotics and computer vision and control systems for autonomous and assisted intelligent vehicles. He is the author of more than 20 refereed publications in international journals, book chapters, and conference proceedings.

L.M. Bergasa received his Ing. Degree in Electrical Engineering in 1995 from the Technical University of Madrid, and his Ph.D. degree in Electrical Engineering in 1999 from the University of Alcalá, Alcalá de Henares, Madrid, Spain. From 1993 to 1994 he has been a researcher at the Department of Electronics, University of Alcalá, where he is currently an Associate Professor. He has been a recipient of the Best Research Award for the 3M Foundation Awards in the category of Industrial in 2004. His research interests include mobile robotics, real-time computer vision and control systems for autonomous and assisted intelligent vehicles. He is the author of more than 50 refereed publications in international journals, book chapters, and conference proceedings.

R. Flores received his Ing. Degree in Electrical Engineering in 2001 from the University of Alcalá, Alcalá de Henares, Madrid, Spain. From 1998 to 2004 he has been a researcher at the Department of Electronics, University of Alcalá, where he is currently an Assistant Professor. He has been a recipient of the Best Research Award for the 3M Foundation Awards in the category of eSafety in 2003. His research interests include mobile robotics and computer vision. He is the author of more than 20 refereed publications in international journals, book chapters, and conference proceedings.

M. Marrón received her Ing. Degree in Electrical Engineering in 1996 from the University of Alcalá, Alcalá de Henares, Madrid, Spain. She is currently a Lecturer in the Department of Electronics, University of Alcalá. Her research interests include mobile robotics and control systems. She is the author of more than 40 refereed publications in international journals, book chapters, and conference proceedings.

M.A. García received his Ing. Degree in Electrical Engineering in 2001 from the University of Alcalá, Alcalá de Henares, Madrid, Spain. From 1999 to 2000 he has been a researcher at the Department of Electronics, University of Alcalá, where he is currently a Lecturer. His research interests include computer vision and control systems for autonomous and assisted intelligent vehicles. He is the author of more than 30 refereed publications in international journals, book chapters, and conference proceedings.