

Training Method Improvements of a WiFi Navigation System Based on POMDP

Manuel Ocaña, Luis Miguel Bergasa, Miguel Ángel Sotelo, Ramón Flores, Elena López, Rafael Barea

Department of Electronics, University of Alcalá

Polytechnic School, Alcalá de Henares, Madrid (Spain)

Email: {mocana,bergasa,sotelo,rflores,elena,barea}@depeca.uah.es

Abstract—The framework of this paper is the robotics navigation inside buildings using WiFi signal strength measure. This navigation is achieved using a Partially Observable Markov Decision Process (POMDP). In the localization phase we used WiFi signal strength and Ultrasound measures as observations. The localization system works in two stages: map construction and localization stage. The map construction stage usually requires a great effort, therefore in this paper we address the problem of minimizing this calibration effort using an automatic training method. We describe the method based on Simultaneous Localization And Mapping (SLAM) techniques and in a robust local navigation task. This automatic method is compared with a manual method to obtain a deterministic map. Also we demonstrate that using this one in a on-line training stage the system is able to adapt the WiFi map to the variations of the WiFi signal measure. Additionally, we analyze the optimal parameters for this automatic training system. The system has been tested in a real environment using two commercial robotic platforms. Some experimental results and the conclusions are presented.

I. INTRODUCTION

The recent interest in location sensing for wireless network applications and the rising demand on the deployment of such systems has brought network researchers up against a mainly and well-known problem in the field of the robotics, the localization.

To find the pose (position and orientation) of a robot from physical sensors is not a trivial problem and is often referred to "the most important problem to provide a mobile robot with autonomous capabilities" [1]. Several systems for localization have been proposed and successfully deployed for an indoor environment. These systems are based on: infrared sensors [2], computer vision [3], ultrasonic sensors [4], laser [5] or radio frequency (RF) [6]. Within the last group we can find localization systems that use WiFi signal strength measure.

WiFi localizations systems take advantage of the boom in wireless networks over the last few years. The WiFi 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). Therefore these localization systems can determine the device location without any extra hardware in the environment. It makes these systems attractive for indoor environments where traditional techniques, such as Global Positioning System (GPS) [7], fail.

In order to estimate the robot location, these systems measure the signal strength of received packets in wireless Ethernet interface. The signal strength depends on the distance and obstacles between wireless nodes and the robot. Moreover, the system needs more than one base stations or Access Points (APs) to measure the distance from them to the device. Using these measures they can apply a triangulation algorithm to infer the estimated position [8].

Unfortunately, in indoor environments, the wireless channel is very noisy and the RF signal can suffer from reflection, diffraction and multipath effect, which makes the signal strength a complex function of distance [6]. To solve this problem, several WiFi location determination systems use a priori radio map (WiFi signal strength map), which captures the signature of each AP at certain points in the area of interest [9] [10] [11]. These systems work in two phases: training and estimation. During the training phase, the radio map is built in a previous setup normally achieved in manual way. 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. The problem is that this method involves an enormous calibration effort because the observations are manually obtained.

For surveillance robots navigation over huge indoor environments design, in which the objective is to guidance the robot to a goal room and if some low level behaviours are used to perform local navigation, a topological discretization is appropriate to facilitate the planning and learning tasks. A POMDP model provides solutions to localization, planning and learning in this robotic context. These models use probabilistic reasoning process to deal with uncertainties, very important in the case of WiFi localization sensors, and a topological representation of the environment to reduce memory planing easybility and process time of the algorithms

In this paper we present an improved WiFi navigation system based on POMDP. The localization stage of our system uses a priori radio map. In this work we compare the map automatically obtained, using a robot moving in the environment in autonomous way and SLAM techniques, with a deterministic map obtained in manual mode. We demonstrate that the proposed automatic calibration method improves the manual one and then manages the adaptability of the map, very important in a WiFi system.

The rest of the paper is organized in the following sections:

Section 2 provides a description of the POMDP navigation system. Section 3 explains the implementation of the WiFi Simultaneous Localization And Mapping (WSLAM) algorithm. Section 4 provides the description of the reducing calibration effort method. Section 5 shows the implementation and some experimental results, as well as a description of the used test bed. Finally, the conclusions and future work are described in Section 6.

II. INTRODUCTION TO A POMDP NAVIGATION SYSTEM

In this section we provide a resume of our POMDP system which was explained by the authors in [12].

When a robot moves across an environment executing several actions (a_t), in execution step t , and it has free of uncertainty in the environment observation, we can modelize this system as a Markov Decision Process (MDP). The MDP is a mathematic model that permit characterize robotics systems without noise in the environment observation. The MDP considers that only the effect of the actions has uncertainty.

When a MDP achieves some execution steps and it goes along a different states ($s_0, s_1 \dots s_n$) executing some actions ($a_0, a_1 \dots a_n$), the probability of being in a s_{t+1} state in the $t + 1$ execution step is obtained using equation 1.

$$p(s_{t+1}|s_0, a_0, s_1, a_1, \dots, s_t, a_t) = p(s_{t+1}|s_t, a_t) \quad (1)$$

The actions uncertainty model represents the real errors or failures in the execution of the actions. The transition function T incorporates this information to the MDP. In the discrete case, T is a matrix that represents the probability of reaching the state s_{t+1} when the robot is in the state s_t and it has executed the action a_t .

There is a recompense function R for each state s and action a . The robot reaches the maximum value of the recompense function when it reaches the target state travelling through the ideal trajectory and executing the ideal actions.

Although MDP considers that the environment observation is free of uncertainty, in the real robotic systems, there are some uncertainties associated to their sensors observations. These are more significant when the observations are provided by the noisy WiFi sensor [13].

The POMDPs are mathematic models that permit to characterize these noisy systems. A POMDP is defined by the same elements than in a MDP: S (states set), A (actions set), T (transition function), R (recompense function); and then it adds the following elements: O (observations set ($o \in O$)) and ν (observation function).

A POMDP doesn't know its real state because the uncertainty of the observation. 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 a probability that indicates the possibility of being in the real state. This is the main reason to divide the control stage of a POMDP in two stages, as can be seen in Figure 1:

- 1) State estimator: the inputs of this block are the current observations and its output is the Bel . This block calculates 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).

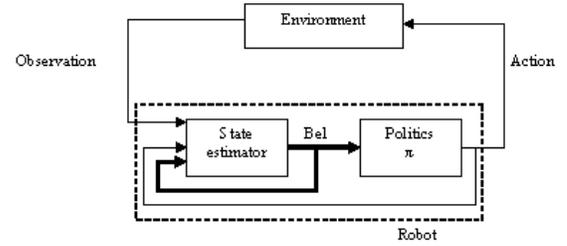


Fig. 1. Partially Observable Markov Decision Process (POMDP)

The state estimator block is known as localization system. This updates the Belief Distribution when a new action or observation is carried out. In the robotics context, these conditions usually are simultaneous. When an action a is executed and a new observation o is taken, the new probabilities become as it is shown in equation 2.

$$Bel_t(s') = \eta \cdot p(o|s') \cdot \sum_{s \in S} p(s'|s, a) \cdot Bel_{t-1}(s), \forall s' \in S \quad (2)$$

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 navigations behaviours that the robot executes to move from a state to another, such as move forward (a_F), turn around (a_T), turn to the left (a_L) and turn to the right (a_R). The observations are perceptions of the environment that the robot can extract from its sensors that in our case are obtained from the WiFi ($obs_{WiFi_{APx}}$) and Ultrasound (obs_{US}) sensors. In this case, the Markov model is partially observable because the robot never may exactly know the state where the robot is. To solve the POMDP model we have used the WiFi Simultaneous Localization And Mapping (WSLAM) in order to obtain the WiFi observation function and, an extension of the EM algorithm to obtain the Ultrasound observation function.

Observations from the WiFi and the Ultrasound sensors are complementary. The first one obtains an estimation of the global localization and the second one obtains a good 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 observations models ($p(\vec{o}|s)$) by mean of Bayes rule. Assuming that the observations are independent, the observation model can be simplified as in the following way:

$$p(\vec{o}|s) = p(obs_{WiFi1}, \dots, obs_{WiFix}, obs_{US}|s) = p(obs_{WiFi1}|s) \cdot \dots \cdot (obs_{WiFix}|s) \cdot p(obs_{US}|s) \quad (3)$$

In the next sections we provide a description of the techniques that we have used to obtain a reduction in the calibration stage effort and how it is possible to adapt the radio

map provided with this technique to the WiFi signal strength measure variations.

III. WiFi SIMULTANEOUS LOCALIZATION AND MAPPING (WSLAM)

This section presents the implemented WSLAM algorithm to automatically provide a WiFi radio map that represents the observation function O . The environment is divided in certain discrete positions or nodes $s \in S$, where S is the total number of nodes. Nodes are coarse-grained regions of variable size in accordance with the environment topology and its centres are separated by more than 80 cm. This simplification obtains a reduction in the computation time and it's appropriate to minimize the WiFi small scale effect [13]. This effect happens due to the chaotic variations of the received WiFi signal when the robot moves over a small distance (order of wavelength).

The method consists of two different phases:

- 1) In the first one the training data are obtained. These can be obtained in a manual or automatic mode.
- 2) In the second phase the WSLAM algorithm is executed over the loaded data.

Both steps should be executed only one time for each environment as long as the environment doesn't change. Once the WiFi radio map is obtained can be used in the localization stage of the POMDP.

During the training stage the robot travels through the environment executing local autonomous motion actions (a_t), and halting in the centre of the nodes (s_t), where t represents the execution step. At each node the robot stores the observations of the WiFi signal strength measures ($obs_{WiFiAPx_t}$), received from all x environment APs.

In order to minimize the WiFi measure error, each AP observation is achieved using the mean value of 60 consecutive samples. We have empirically demonstrated that this value is the optimum for obtaining few errors in the localization phase with a minimum number of samples. The observed values mean are rounded from 0 to 99 because they correspond with the 0 to -99dBm measure range in the WiFi interface. Equation 4 shows the observation and action training set during t execution steps.

$$\begin{aligned} obs_{WiFi}^t &= \{obs_{WiFi1}, \dots, obs_{WiFi_t}\} \\ a^t &= \{a_1, \dots, a_t\} \\ training_set &= \{obs_{WiFi}^t, a^t\} \end{aligned} \quad (4)$$

Where the observation obs_{WiFi_z} is the tupla shown in equation 5, and z is a value between 1 and t execution steps.

$$obs_{WiFi_z} = \{(obs_{WiFiAP1}, \dots, obs_{WiFiAPx})_z\} \quad (5)$$

The stored observations at each node and the executed actions through the different trials represent the training data set and they constitute the inputs for the WSLAM algorithm that will be executed in the second stage.

The WSLAM is a particularization of the Baum-Welch algorithm, also known as EM (Expectation-Maximization), using WiFi observations and a novel setup process. The EM

algorithm is a hill-climbing process that iteratively alternates two steps:

- The E-step (expectation step) calculates the state evolution, it estimates the robot localizations based on the currently available map parameters.
- The M-step (maximization step) uses the estimated trajectory, computed in the E-step, to recalculate the map in order to obtain the maximum likelihood parameters. When the map improves it is easier to estimate the evolution of the states.

The E-step re-estimates, at each iteration, the robot trajectory, using the Belief distribution over all states, the training data set and the available map. The Bel is obtained through two distributions: $\alpha_z(s)$ and $\beta_z(s)$, as we show in equation 6.

$$\begin{aligned} Bel_z(s_z | training_set, O) &= \eta \cdot \alpha_z(s_z) \cdot \beta_z(s_z), \forall s_z \in S_t \\ \alpha_z(s_z) &= p(s_z | obs_{WiFi1}, a_1, \dots, obs_{WiFi_z}, O) \\ \beta_z(s_z) &= p(s_z | a_z, \dots, o_t, a_t, O) \end{aligned} \quad (6)$$

Where η is a normalization factor. The α distribution is forwarded computed. It is the probability of reaching a state when the robot has executed several actions and has moved through some several states. The β distribution is backward computed. It represents the probability of travelling through some several states when the robot has started at certain state and it has executed several actions.

The M-step re-adjusts the map parameters according to the previous map and the states evolution estimated in the E-step. This adjust is carried out using a frequency count as we show in equation 7. Where the $\gamma_z(s)$ distribution is computed as the product of $\alpha_z(s)$ and $\beta_z(s)$ distributions. It represents the best state estimation.

$$p'(o|s) = \frac{\sum_{z=1 \dots t | o_t=o} \gamma_z(s)}{\sum_{z=1 \dots t} \gamma_z(s)} \forall s \in S \quad (7)$$

In order to adjust the initial distribution we use equation 8 and the transition matrix, as it is shown in equation 9. The transition matrix defines the probability of reaching a state s' when the prior state is s and the robot executes an action a .

$$p(s_0 = s) = \gamma_0(s) \quad (8)$$

$$p(s'|s, a) = \frac{\sum_{z=1 \dots t-1 | a_t=a} \gamma_z(s, s')}{\sum_{z=1 \dots t-1 | a_t=a} \gamma_z(s)} \forall s, s' \in S \text{ and } \forall a \in A \quad (9)$$

One of the main problems in SLAM algorithms is the parameters initialization. The WSLAM introduces an important improvement for this purpose, because the initialization of the WiFi observations map is carried out using a coarse radio propagation model. The WiFi radio signal propagates through the air following a radio propagation model. This model is very difficult to obtain for indoor environments due to the multipath suffering and the temporal variation of the

WiFi signal. Although an exact and general model doesn't exist, an approximated model can be used to initialize the WiFi observation map. After that, a reduced searching range according to the model deviation respect the real propagation can be employed. In our case we use a generic log distance model as the shown in equation 10.

$$R_{SL} = T_{SL} + G_{TX} + G_{RX} + 20\log(4\pi) - 10n_W\log d - X_a \quad (10)$$

Where the R_{SL} is the received signal level, T_{SL} is the transmitted signal level, G_{TX} and G_{RX} are the transmitter and receiver antennas gain respectively, λ is the wavelength (12.5cm for the 2.4GHz of the WiFi signal), n_W is a factor that depends on the walls effect, X_a is a random variable and d is the distance between the emitter and the receiver

IV. REDUCTION OF THE CALIBRATION EFFORT

The methods to obtain the WiFi training data set involve an enormous calibration effort because at the moment observations in each node are manually done or using the robot in teleoperated mode. To solve this problem, we propose an automatic WiFi and Ultrasound training data set extraction method based on a robust and autonomous local navigation task that carries out the robot along the centre of the corridors and it's able to stop in the centre of the found nodes in its route using odometry and ultrasound sensors [14].

Using this local navigation task and a modification of the proposed POMDP, we can obtain the WiFi training data set with a low effort. The modification of the POMDP that we propose is to use only Ultrasound observations instead of WiFi+Ultrasound ones, because Ultrasound observations are very robust in the local context.

In our case using the robustness of our low level controller and Ultrasound observations in the POMDP we can build an autonomous learning system. The robot navigates in autonomous mode with the initial state known (we set to one the first known state in the Bel). The robot stores the actions executed at each transition (a_t) and the WiFi and Ultrasound observations (obs_{WiFi} , obs_{US}) in the training data set. Then the WSLAM algorithm is executed to obtain the WiFi radio map with a low effort.

This method represents an off-line training stage previous to the robot navigation through the environment. In this work we have enhanced the system with a new on-line training stage, in this case when the robot navigates in the environment with different missions the executed actions and the taken observations are stored. All this data in a trial form a frame. When the number of frames is enough to obtain the radio map, the WSLAM algorithm is executed to update it. This update can be ran while the robot is navigating in a parallel process without needed to stop the robot. We use an impact factor, lower than in the off-line stage, to soften the map update. With this on-line stage we obtain a flexible and adaptable system.

One of the main parameter to determine in this calibration method is the optimal number of frames that it is necessary to obtain a correct radio map. To determine this parameter

we have used the entropy and the divergence factor proposed by the authors in [15]. The entropy determines what is the uncertainty degree or scatter of the different states within the Bel, using the expression shown in equation 11.

$$H(Bel) = - \sum_{Bel(s) \neq 0} Bel(s) \cdot \log(Bel(s)) \quad (11)$$

The divergence factor determines what is the uncertainty degree of the Bel using the equation 12, where d_{max} is the difference between the first and the second maximum value in the Bel, p_{max} is the maximum value in the Bel and m is the number of states.

$$\tilde{D} = 1 - \frac{(d_{max} + p_{max}) \cdot m - 1}{2 \cdot m - 1} \quad (12)$$

The divergence factor takes one value when there is no uncertainty in the Bel, that means that this state has the maximum probability, and zero when there is the maximum uncertainty over the states, in this case the probability of each state is $1/m$.

V. IMPLEMENTATION AND RESULTS

In this section we describe the test-bed used for our WiFi navigation system, some characteristics of the system implementation and then some experimental results to validate the proposed improvements of the navigation system with the real robots.

A. Test-Bed

The Test-Bed environment was established on the 3rd floor of the Polytechnic School building, concretely in the corridor number 4 of the Electronic Department at the University of Alcalá. The layout of this zone is shown in Figure 2. It has a surface of 60m x 60m, with about 50 different rooms, including offices, labs, bathrooms, storerooms and meeting rooms.

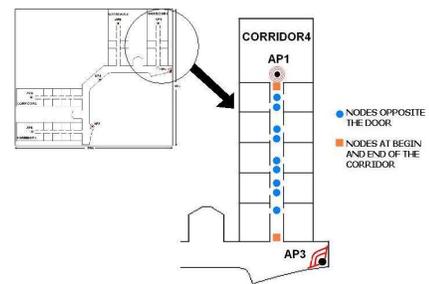


Fig. 2. Test-bed. Department of Electronics (Corridor 4)

Seven Buffalo Access Points (APs) (WBRE-54G) were installed at the all environment. Five APs were connected to omnidirectional antennas and two APs (AP3 and AP7) were connected to antennas of 120 degrees of horizontal beam-width. The APs act as wireless signal transmitters or base stations.

For simplicity, the tests were achieved in the corridor 4. This was discretized into 11 nodes placed at the positions indicated

in Figure 2. For each node some radio measures from all the APs in the two main orientations of the corridor were taken to obtain the WiFi radio map, and to extract the results of the WiFi POMDP navigation system.

B. Implementation

In order to obtain experimental results of this improved WiFi navigation system we have used two robots as shown in Figure 3. These robots are called BART (Basic Agent for Robotics Tasks) and SIRA (Spanish acronym for Assistant Robotic System). The two robots are based on the 2AT platform of Activmedia Robotics. For SIRA a metallic structure was added in order to carry a laptop and to increase the height of the camera. They have the following configuration: Orinoco PCMCIA Gold wireless card, Linux Red Hat 9.0 operating system, wireless tools, a 16 ultrasound sensor ring and a SONY pan-tilt-zoom camera.



Fig. 3. Real prototypes used in the results extraction

The architecture of the global WiFi POMDP navigation system is shown in Figure 4. The main blocks are:

- The robot sensors provide two observations (WiFi and Ultrasound) and they are the inputs of the training block and localization system. The actions commanded by the planner are executed through the local navigation system by the actuators of the robot (the four engines connected two by two).
- The localization system uses the observations provided by the robot and the priori map to obtain the estimated position over the all states (Bel).
- The planning system has two inputs, the Bel over the all states and the commanded state that it is introduced by the final user. The commanded action is used as input of the training system and the local navigation system.
- The training system is used to obtain the priori radio map and the POMDP parameters, such as the transition and observations matrices, from the training data system.
- The Man-Machine Interface (MMI) based on a friendly graphical interface built in Java, in a typical client-server application. This interface works under Linux and Windows through a TCP-IP connection to the robot server (in our case Saphira).
- The local navigation system is in charged of executing the actions commanded by the planning system.

C. Results

First we have compared the automatic training method with the manual one. We trained it in a manual mode by mean

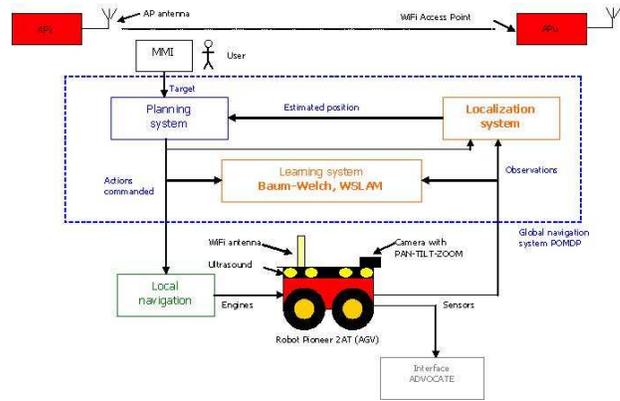


Fig. 4. WiFi POMDP navigation system

of positioning the robot along the several states and then the robot took the WiFi signal samples to calculate the mean value at each state. This needed 3 hours of an intensive man-work. With the automatic training system, the user only needed to launch the local navigation application with a slight supervision during about 1 hour to ensure that the task was carried out correctly by the robot. We repeated this automatic method for 8 times more and the robot never failed. This demonstrates that the calibration effort has been enormously reduced.

We have obtained the optimum number of frames needed for the radio map construction. We had the robot navigating around the environment for two hours in an automatic way. Then, we obtained the mean entropy of the training data set using from 1 to 7 frames (figure 5(a)) and the localization error percentage during the training stage (figure 5(b)). We concluded that with a number of frames higher than 5, the mean entropy value is low ($0.12 \cdot 10^{-3}$) and the localization error percentage is lower than 0.5%, therefore 6 is the optimal frames number for the training data set.

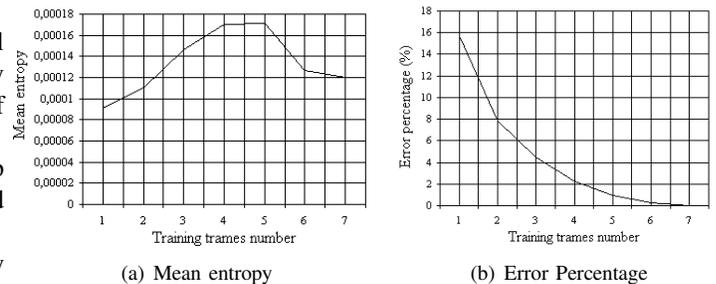


Fig. 5. Optimal frames number in the training data set

Once we know what is the optimal number of frames, we obtain the map through the two methods, first we obtain the probabilistic radio map with the WSLAM method and then we obtain a deterministic map using the manual method. Figure 6 shows the both maps, the probabilistic one (figure 6(a)) and the deterministic one (figure 6(b)). These maps represent, in gray scale, the probability of obtaining a WiFi signal level measure

TABLE I
LOCALIZATION ERROR

Method	Localizations	True	False	Error Percentage(%)
Manual	50	38	12	24
WSLAM	50	46	4	8

in a certain state of the environment (i.e. $p(o|s)$), where white color represents a 0% probability and black color represents a 100%.

We used both maps in the WiFi navigation system proposed in this work. The robot was navigating through 50 states and the statistics of localization error was obtained. As you can see in Table I, with the WSLAM method the localization error percentage during the navigation was lower than 10%, while with the manual one the error was up to 20%.

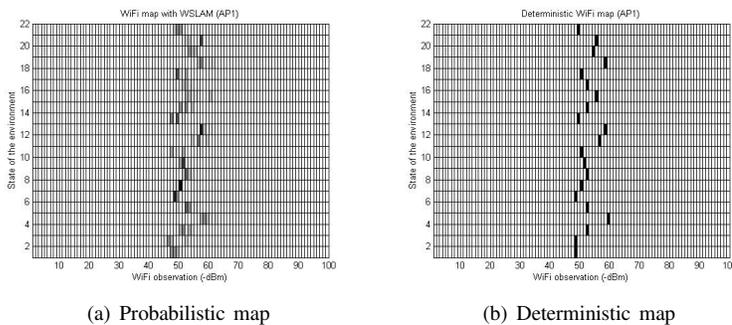


Fig. 6. Maps obtained with WSLAM and manual method

Finally we tested the on-line training system. The robot BART was training in the environment using 6 frames in order to obtain the WiFi radio map. We used this priori radio map in the robot SIRA to navigate, obtaining a major localization error number than with BART robot. This demonstrates that the map obtained with one robot is not useful for another robot with a different structure. To improved the results of this test, we used the on-line training method with SIRA, it was training during different missions and it was updating the radio map. Finally we improve the results up to 90% of true locations.

VI. CONCLUSIONS AND FUTURE WORKS

In this work we have presented an enhancement WiFi navigation system. This navigation system uses an automatic training method to obtain WiFi radio map useful in the localization process of a POMDP.

We have obtained the optimal number of training frames to be used with this automatic training system. This effectively diminishes one of the most daunting practical barriers to wider adoption of this type of localization measurement technique.

The WiFi map got through this technique is a probabilistic representation of the WiFi signal evolution in some certain discrete positions of the environment along the time and we demonstrate that it is more useful than a deterministic map.

The on-line training stage has demonstrated that the system is more flexible to the WiFi signal strength measure variations.

On the other hand we have demonstrated the performance of the system using two real robots at the Corridor 4 of the Electronics Department. Additionally, we have tested it over all the environment in simulation. In the near future, we have the intention to apply our system in all the Electronics Department environment in real mode.

VII. ACKNOWLEDGMENT

This work has been funded by grant S-0505/DPI/000176 (Robocity2030 Project) from the Science Department of Community of Madrid, TRA2005-08529-C02-01 (MOVICOM Project) from the Spanish Ministry of Science and Technology (MCyT) and SIMCA (CM-UAH: 2005/018) from the University of Alcalá.

REFERENCES

- [1] I. Cox, "Blanche-an experiment in guidance and navigation of an autonomous robot vehicle," *IEEE Trans. Robot. Automat.*, vol. 7, no. 2, pp. 193–204, 1991.
- [2] R. Want, A. Hopper, V. Falco, and J. Gibbons, "The active badge location system," *ACM Transactions on Information Systems*, vol. 10, pp. 91–102, Jan. 1992.
- [3] J. Krumm, S. Harris, B. Meyers, B. Brumitt, M. Hale, and S. Shafer, "Multi-camera multi-person tracking for easy living," in *Proc. of 3rd IEEE International Workshop on Visual Surveillance*, 2002, pp. 3–10.
- [4] N. Priyantha, A. Chakraborty, and H. Balakrishnan, "The cricket location support system," in *Proc. of the 6th ACM MobiCom*, 2002, pp. 155–164.
- [5] R. Barber, M. Mata, M. Boada, J. Armingol, and M. Salichs, "A perception system based on laser information for mobile robot topologic navigation," in *Proc. of 28th Annual Conference of the IEEE Industrial Electronics Society*, 2002, pp. 2779–2784.
- [6] P. Bahl and V. Padmanabhan, "Radar: A, in-building rf-based user location and tracking system," in *Proc. of the IEEE Infocom*, 2000, pp. 775–784.
- [7] P. Enge and P. Misra, "Special issue on gps: The global positioning system," in *Proc. of the IEEE*, vol. 87, no. 1, 1999, pp. 3–172.
- [8] O. Serrano, J. Cañas, V. Matellán, and L. Rodero, "Robot localization using wifi signal without intensity map," in *Proc. of the V Workshop Agentes Físicos (WAF'04)*, 2004, pp. 79–88.
- [9] A. Howard, S. Siddiqi, and G. Sukhatme, "An experimental study of localization using wireless ethernet," in *Proc. of the International Conference on Field and Service Robotics*, July 2003.
- [10] A. Ladd, K. Bekris, A. Rudys, G. Marceu, L. Kavraki, and D. Wallach, "Robotics-based location sensing using wireless ethernet," in *Proc. of the MOBICOM'02*, 2002.
- [11] M. Youssef, A. Agrawala, and A. Shankar, "Wlan location determination via clustering and probability distributions," in *Proc. of the IEEE PerCom 2003*, 2003.
- [12] M. Ocaña, L. Bergasa, M. Sotelo, and R. Flores, "Indoor robot navigation using a pomdp based on wifi and ultrasound observations," in *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS2005)*, Aug. 2005, pp. 503–509.
- [13] M. Ocaña, L. Bergasa, and M. Sotelo, "Robust navigation indoor using wifi localization," in *Proc. of the 10th IEEE International Conference on Methods and Models in Automation and Robotics (MMAR2004)*, Aug. 2004, pp. 851–856.
- [14] M. Ocaña, M. Sotelo, L. Bergasa, and R. Flores, "Low level navigation system for a pomdp based on wifi and ultrasound observations," in *Proc. of the 6th IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA2005)*, June 2005.
- [15] E. Lopez, R. Barea, L. Bergasa, and M. Escudero, "A human-robot cooperative learning system for easy installation of assistant robots in new working environments," *Journal of Intelligent and Robotic System*, vol. 40, no. 3, pp. 233–265, July 2004.