Studying of WiFi range-only sensor and its application to localization and mapping systems

F. Herranz, M. Ocaña, L. M. Bergasa, M. A. Sotelo, D. F. Llorca,
N. Hernández, A. Llamazares and C. Fernández
Department of Electronics, University of Alcalá, Madrid (Spain)
Email: {fherranz,mocana,bergasa,sotelo,llorca,nhernandez,allamazares,cfernandez}@depeca.uah.es

***Abstract*— The goal of this paper is to study a noisy WiFi range-only sensor and its application in the development of localization and mapping systems. Moreover, the paper shows several localization and mapping techniques to be compared. These techniques have been applied successfully with other technologies, like ultra-wide band (UWB), but we demonstrate that even using a much more noisier sensor these systems can be applied correctly. We use two trilateration techniques and a particle filter to develop the localization and mapping systems based on the range-only sensor. Some experimental results and conclusions are presented.**

I. INTRODUCTION

For most outdoor applications, i.e. surveillance tasks or vehicle navigation systems, Global Positioning System (GPS) [1] provide enough accuracy. On the contrary, when GPS receiver is in urban environments with high buildings or trees, the signal can suffer multipath fading or even Line-Of-Sight (LOS) blockage. In addition, it is important to remark that GPS signal is not strong enough to penetrate inside buildings, then this problem discards this technique to use it like an indoor localization system.

Vehicle navigation systems use a combination of a previous map with localization information to guide the vehicle through a mesh of connected ways. Maps are usually obtained in a semi-autonomous way process known as mapping [2]. Mapping is based on sensor observations which extract main features of the environment and allow to represent them into a topolocical or metric map.

Autonomous localization and mapping are two problems with similar features. It is not possible to built a map if the localization process does not work well, and it is impossible to locate a device with high precision without an accurate map. The SLAM(Simultaneous Localization And Mapping) techniques [3] [4] are used to solve these problems simultaneously, because the uncertainty of both processes can be reduced by doing localization and mapping at the same time .

Several systems for localization and mapping have been proposed and successfully deployed for indoor environments. These systems are based on: infrared sensors [5], computer vision [6], ultrasonic sensors [7], laser [8] or radio frequency (RF) [9] [10] [11] [4]. Within the last group we can find localization systems that use WiFi and Ultra Wide Band (UWB) signal level (SL). In order to estimate the vehicle or map feature location, these systems measure the signal strength and then apply a deterministic (i.e. trilateration) or probabilistic (i.e. particle filter) algorithm to infer the estimated position. In addition, these techniques can be used in the same way in outdoor environments.

While the UWB systems achieve a high accuracy in both systems (localization and mapping), by mean of adding UWB reference beacons in the environment, WiFi technology uses 802.11b/g network infrastructure to estimate a device position without using additional hardware. Unfortunately, signal propagation is affected by reflection, refraction and diffraction in indoor environments. This effect, known as multipath effect, turns the received SL into a complex function of the distance. To solve this problem, several localization systems use a previous map and then, in the estimation phase, the received signal measure from each Access Point (AP) is compared with the map to obtain the estimated position [12] [13] [14]. This last technique is not recommended when the environment is dynamic or when its size increases.

In this work, we use the combination of the WiFi signal measure and a propagation model to obtain a range-only sensor that can be used both indoor and outdoor. We compare two deterministic and one probabilistic techniques to obtain the accuracy of all of them. These techniques are used in the same way for localization and mapping with slightly modifications. This work represents a previous step before obtaining a WiFi range-only SLAM system.

The rest of the paper is organized as follows: section 2 shows propagation models and WiFi signal variations; section 3 shows the localization with propagation model techniques; section 4 shows the mapping process; section 5 describes the results obtained by WiFi localization and mapping systems; and finally, section 6 shows some conclusions and future works.

II. WIFI RANGE-ONLY SENSOR

This section provides an introduction about the WiFi signal measure and its application as a range-only sensor. It is important to highlight that WiFi technology works at 2.4Ghz, a closer frequency to water resonant one, then it can be affected by several variations.

In a previous work [15] authors have throughly studied the main variations that affect to WiFi signal. We identified five main variations that can appear when working with robots. Among this five ones, there are three main variations to take into account when we want to develop WiFi range-only sensor localization and mapping systems:

- **Temporal variations**: when the robot is standing at a fixed position, the signal strength measure can vary over time. SL variations can be up to 2 dBm. These variations are usually due to changes in the physical environment such as people in movement.
- **Small-scale variations**: these variations occur when the robot moves in a small distance, under the wavelength λ. As a result, there are significant changes in the average received SL. For the 802.11b networks working at the 2.4 GHz range, λ is 12.5 cm. This kind of variations are generated by multipath effect. Small-scale variations introduce a high uncertainty in the system. These variations make difficult to estimate the device position because they can be up to 10 dBm for positions around the same location.
- **Large-scale variations**: signal strength varies over a long distance due to attenuation of the RF signal [16]. Large-scale variations can be used to estimate the distance between the robot and reference positions (APs locations).

A propagation model [17] is an empirical mathematical formulation for the characterization of radio wave propagation as a function of frequency, distance and other conditions. A single model is usually developed to predict the behavior of propagation for all similar links under similar constraints. Created with the goal of formalizing the way in which the radio waves are propagated from one place to another, such models typically predict the path loss trough link or the effective coverage area of a transmitter.

In our system, we use a propagation model to estimate the distance between the APs and the robot through received SL. Our work is based on Hata-Okumura propagation model, which is studied in [18]. The equation (1) describes this model:

$$d = 10^{\frac{P_{TX} - P_{RX} + G_{TX} + G_{RX} - X_{\alpha} + 20\log\lambda - 20\log 4\pi}{10n}} \tag{1}$$

Where:

- d: is the distance between transceiver and receiver.
- P_{TX} and P_{RX}: are the transceiver and the receiver power (dBm).
- G_{TX} and G_{RX}: are the transceiver and receiver antenna gain (dBi).
- X_{α}: represents the error. It is a normal random variable with standard deviation α.
- λ: is the wavelength (12.5 cm).
- n: denotes influence of walls and other obstacles. In outdoors environments with LOS it is defined in the range from 2 to 3. In [19], the authors determine that the variable n must be approximately 2 in outdoor environments.

.

III. Localization with range-only sensors

Localization is the technique that estimates the position of a mobile device using reference positions and the distance provided by the range-only sensor. In this section we describe the three techniques that we have compared.

A. *Spherical trilateration*

This technique estimates the robot position using the APs positions and the distances between the mobile and the APs. The algorithm is based on the next constraints:

- The n APs positions are known, and are placed in the coordinates $(x_1, y_1, z_1), (x_2, y_2, z_2), \dots (x_n, y_n, z_n)$.
- The robot position is defined as (x_r, y_r, z_r), and it is the position to estimate by the algorithm.
- The distances between the robot and each AP are known $r_1, r_2, \dots r_n$.

The trilateration elements are showed in Figure 1.

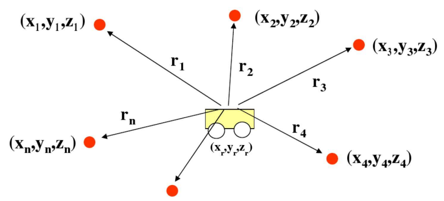


Fig. 1. Trilateration elements

The algorithm is based on these constraints to estimate the robot position using the equations (2), (3), (4) and (5).

$$\begin{gathered} r_i^2 = (x_r - x_i)^2 + (y_r - y_i)^2 + (z_r - z_i)^2 \Rightarrow \\ x_r^2 + y_r^2 + z_r^2 + x_i^2 + y_i^2 + z_i^2 - 2x_r x_i - 2y_r y_i - 2z_r z_i - r_i^2 = 0 \\ \forall i = 1, 2, \dots n \end{gathered} \tag{2}$$

Where t and S_i^2 are obtained as shown in (3):

$$\begin{aligned} t &= x_r^2 + y_r^2 + z_r^2 \\ S_i^2 &= x_i^2 + y_i^2 + z_i^2 \end{aligned} \tag{3}$$

It is possible to show the equations using the matrix way:

$$AX = B \tag{4}$$

$$\begin{aligned} X &= \begin{pmatrix} x_r \\ y_r \\ z_r \\ t \end{pmatrix} \\ A &= \begin{pmatrix} 2x_1 & 2y_1 & 2z_1 & -1 \\ 2x_2 & 2y_2 & 2z_2 & -1 \\ \dots & \dots & \dots & \dots \\ 2x_n & 2y_n & 2z_n & -1 \end{pmatrix} \\ B &= \begin{pmatrix} S_1^2 - r_1^2 \\ S_2^2 - r_2^2 \\ \dots \\ S_2^2 - r_2^2 \end{pmatrix} \end{aligned} \tag{5}$$

B. Spherical trilateration. Gauss-Newton algorithm

This method is based on the same elements than the previous trilateration (Figure 1). Moreover, a random position is used as initial one to estimate the robot position $emp = (\hat{x_r}, \hat{y_r}, \hat{z_r})$.

The distance between the robot and the AP_i is defined according to equation (6).

$$r_i = \sqrt{(x_i - x_r)^2 + (y_i - y_r)^2 + (z_i - z_r)^2} \qquad i = 1 \dots n \tag{6}$$

Now, we can define the distance between the AP_i and the emp (7).

$$\hat{r}_i = \sqrt{(x_i - \hat{x}_r)^2 + (y_i - \hat{y}_r)^2 + (z_i - \hat{z}_r)^2} \qquad i = 1 \dots n \tag{7}$$

This method is based on equations (6) and (7) and Gauss-Newton algorithm. This method is used to solve non-linear least squares problems like this. It makes possible to minimize a sum of squared function values through an iterative way (equation (8)).

$$F(\hat{x}_r, \hat{y}_r, \hat{z}_r) = \sum_{i=1}^{n} (\hat{r}_i - r_i)^2 = \sum_{i=1}^{n} [f_i(\hat{x}_r, \hat{y}_r, \hat{z}_r)]^2 \tag{8}$$

Where f_i is obtained as shown (9):

$$f_i = \sqrt{(x_i - \hat{x}_r)^2 + (y_i - \hat{y}_r)^2 + (z_i - \hat{z}_r)^2} - r_i \tag{9}$$

Deriving the equation (9) respect to $(\hat{x}_r, \hat{y}_r, \hat{z}_r)$ it is possible to obtain the equation (10).

$$\begin{aligned} \frac{\partial F}{\partial \hat{x}_r} &= 2 \sum_{i=1}^{n} f_i \frac{\partial f_i}{\partial \hat{x}_r}; \\ \frac{\partial F}{\partial \hat{y}_r} &= 2 \sum_{i=1}^{n} f_i \frac{\partial f_i}{\partial \hat{y}_r}; \\ \frac{\partial F}{\partial \hat{z}_r} &= 2 \sum_{i=1}^{n} f_i \frac{\partial f_i}{\partial \hat{z}_r}; \end{aligned} \tag{10}$$

Equation (10) can be showed using the matrix way, $A \cdot \Delta X = B$ (equation (11)).

$$\begin{aligned} \Delta X &= \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta z \end{pmatrix} \\ A &= \begin{pmatrix} \frac{(\hat{x}_r - x_1)}{\hat{r}_1} & \frac{(\hat{y}_r - y_1)}{\hat{r}_1} & \frac{(\hat{z}_r - z_1)}{\hat{r}_1} \\ \frac{(\hat{x}_r - x_2)}{\hat{r}_2} & \frac{(\hat{y}_r - y_2)}{\hat{r}_2} & \frac{(\hat{z}_r - z_2)}{\hat{r}_2} \\ \vdots & \vdots & \vdots \\ \frac{(\hat{x}_r - x_n)}{\hat{r}_n} & \frac{(\hat{y}_r - y_n)}{\hat{r}_n} & \frac{(\hat{z}_r - z_n)}{\hat{r}_n} \end{pmatrix} \\ B &= \begin{pmatrix} (\hat{r}_1 - r_1) \\ (\hat{r}_2 - r_2) \\ \vdots \\ (\hat{r}_n - r_n) \end{pmatrix} \end{aligned} \tag{11}$$

The system can be solved by least squares and it is possible to obtain the algorithm increases according to equation (12).

$$\Delta X = (A^T A)^{-1} A^T B \tag{12}$$

Finally, the estimated robot position is updated using the previous emp and the new increase (equation (13)).

$$emp_{k+1} = emp_k - \Delta X_k \tag{13}$$

The process continues running until the increases ΔX become acceptable by the system.

C. Particle filter

The particle filter is a sequential Monte Carlo algorithm, i.e. a sampling method to approximate a distribution that uses its temporal structure. A ”particle representation” of distributions is used. In particular, we will be concerned with the distribution $P(X_{rt}|z_{0:t})$ where $X_{rt} = (x_{rt}, y_{rt}, \theta_{rt})$ is the observed robot state at time t, and $z_{0:t} = (r_1, r_2, ..., r_n)$ is the sequence of observations from time 0 to time t. The transition and sensor models, $P(X_{rt}|z_{0:t})$ are represented using a collection of N weighted samples or particles, $\{X_{rt}^{(i)}, \pi_t^{(i)}\}_{i=1}^{N}$ where $\pi_t^{(i)}$ is the weight of particle $X_{rt}^{(i)}$ (equation (14)).

$$P(X_{rt}|z_{0:t}) \approx \sum_i \pi_{t-1} \delta(X_{rt} - X_{rt-1}^{(i)}) \tag{14}$$

The particles are propagated using the movement model $p(X_{rt}|X_{rt-1}, a_t)$ and the verisimilitude $P(z_t|X_{rt})$.

Firstly, the particles are uniformly distributed at the state space. Next, the particles are updated by the previous actions a_{t-1}, the actual observation z_t and the movement model. Finally, the updated particles are weighted, so the density probability function of the particles represents the estimated robot position.

IV. Mapping with range-only sensors

Mapping is the process that makes possible to estimate the APs positions using the distance between them and the robot. First of all, to map the positions of the reference is needed to know the trajectory of the mobile, and then estimate the APs positions using this knowledge. This problem is similar to the localization one but with a different point of view, we suppose that the robot position is known and static at different steps, and then it seems like the APs are moving around it.

A. Spherical trilateration and Gauss-Newton Spherical trilateration

These algorithms are used on mapping in a similar way than trilateration algorithms are used on the localization. The main difference between both is the previous knowledge. Localization algorithms know the APs location and estimate the robot position, however, mapping algorithms know the robot trajectory and they estimate the APs position. Figure 2 shows the elements used to estimate the position of one AP.

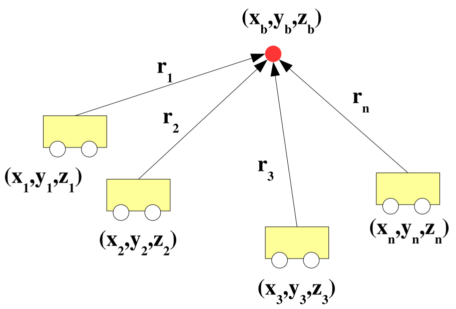


Fig. 2. Mapping elements

Finally, the algorithm is based on the equations (2), (3), (4) and (5), swapping reference (x_i, y_i, z_i) by beacon (x_b, y_b, z_b) positions.

Work [20] puts forward some situations that can make the system fails:

- When the reference positions are align.
- If the beacon position and the reference are on the same plane.
- If the reference position is over one reference.

According to these constraints, close positions of the robot trajectory are useless to map the APs position because they can be aligned. Moreover, it is recommended to design a ”zigzag” path for the robot to avoid the alignment of the reference positions.

B. Particle filter

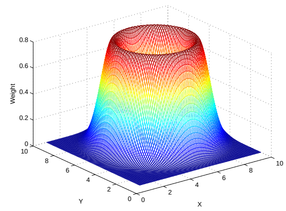
A particle filter like III-C is also used to map the APs. The main difference between both particle filters is the orientation. In this case, only a measurement of distance is obtained and then the AP can be everywhere within a circumference. To adapt the previous filter for mapping some modifications have been performed. These modifications are:

- Measurement vectors Z are the distances between the AP and the robot. The measures depend on the robot location.
- The verisimilitude $P(z_t|X_{rt})$ uses a vectorial space to represent the observations. Thus, we use a circumference equation based on a vectorial space. It is written in parametric form using trigonometric functions as is shown in equation (15). Then, 360 observations (one per angle ϕ) are generated. These observations form a circumference with radius equal to the distance between the AP and the robot.

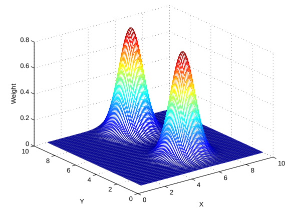
$$P = X_{r0} + r(\cos\phi, \sin\phi) \tag{15}$$

Where:

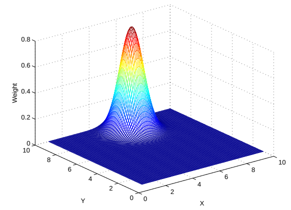
- X_{r0} is the actual robot position.
- r is the radius or the distance between the AP and the robot.
- ϕ is the angle.
- P are the observed AP coordinates (x_b, y_b).



(a) 360 hypothesis



(b) 2 hypothesis



(c) 1 hypothesis

Fig. 3. Mapping with particle filter

Figure 3 shows the particle filter process, at the beginning there are 360 possible positions (one per angle) where the AP can be. Then, the possible positions are less and usually there are only 2 possible positions: the real AP position and the ”mirror” one. Finally, only one hypothesis is followed and this position usually corresponds with the real AP position.

It is important to highlight that this algorithm does not need to collect a high number of samples to estimate the AP position. It is a online process and the accuracy is improving when the time is increasing.

V. Implementation and Results

This section describes some implementation features and the experimental results obtained with the designed tests.

A. Test-Bed Environment

The environment to test the localization and mapping systems is established outdoor and close to the Polytechnic School at the University of Alcalá (UAH).

The environment dimensions are approximately 20x20 metres. Moreover, three APs are used, these APs are located at coordinates (x, y, z) $(5.35, -2.36, 1.70)$, $(14, -2.36, 1.67)$, $(15.10, 7.4, 1.61)$. The WiFi antenna is placed at the mobile robot at 0.71 metres height. The robot trajectory is a pseudo-rectangle, this path is showed in Figure 4. The localization process in this work is calculated in 3D, and it has been necessary to convert the measurements from 3D to 2D to simplify the problem.

The tests have been performed with a laptop using an Orinoco Gold PCMCIA card, Linux Kubuntu 8.04, Wireless Tools v29 and Matlab 2008a. Signal level measure is obtained by the WiFi interface installed in the laptop. This interface scans the APs close to the device. Samples are got at 4 Hz, which is the highest frequency that the interface supports.

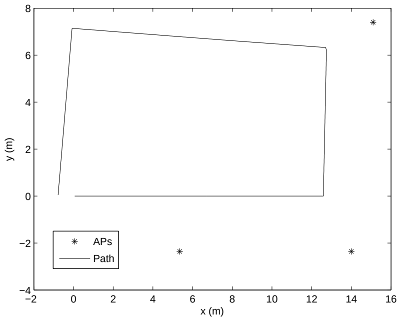


Fig. 4. Real test environment

TABLE I
WIFI RANGE ONLY SENSOR SAMPLES

d (m)	2	4	8	12	16	20
μ_{SL} (dBm)	-44.40	-53.25	-61.44	-66.00	-74.24	-78.04
σ_{SL}(dBm)	-3	-5	-5	-5	-7	-7

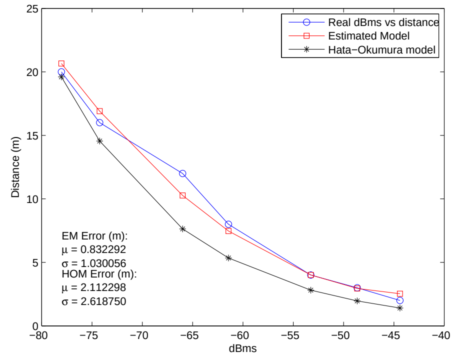


Fig. 5. Large-scale variations on propagation model

TABLE II
PROPAGATION MODELS ERROR

Distance (m)	2	3	4	8	12	16	20
EM error (m)	0.28	0.01	0.00	0.29	3.01	0.83	0.44
HOM error (m)	0.35	1.08	1.41	7.01	19.08	2.09	0.16

B. *WiFi range only sensor*

To study the WiFi range only sensor a real test has been performed, which consists of measuring the signal level at different distances. The collected samples are processed calculating the mean and the variance of them. Table I shows the mean and the variance values of the samples for each distance. The mean values shows how the SL decreases with the distance, however, the variance increases with the distance. A high variance in the samples, produced by the noise, makes difficult to estimate a distance from a SL value.

Based on Table I values, and paying attention to large-scale variations it is possible to estimate a propagation model. Figure 5 shows an estimated propagation model and a comparison with Hata-Okumura model (HOM) using a set of training data. The propagation model has been estimated obtaining the mean SL of each distance and fitting a polynomial function using a approximation by least squares. The estimated model (EM) obtains better results than HOM because it fits perfectly the training data, however, HOM is a generic model and it can be adapted to new environments.

Table II shows the error for each distance. In both models the error tends to increase with the distance. This error is specially important in the HOM at 12 metres, it is due to a high noise in the training samples. HOM does not fit well to the training samples when they contain a huge noise, however, the EM obtains better results in this case. It is important to remark that in other cases the samples can contain more or less noise and the EM will not obtain good results. Then, the EM is a better choice in an under control environment, however, the HOM is more general and adaptable to new environments.

C. *Localization results*

Localization results are showed in Table III. Trilateration algorithm gets a mean error of 9.04 metres, which is the highest error of the three algorithms. On the other hand, Gauss-Newton obtains better results, the mean error is 6.26 metres and the error is more constant than the trilateration one. Both algorithms estimate the robot position without previous information, which decreases the accuracy, but these can be used with a low computational cost. However, the particle filter uses the previous information to estimate the position to avoid high changes in the error. Then, it makes the particle filter the most accurate algorithm, it gets a mean error of 3.16 metres and a maximum error of 9.24 metres.

D. *Mapping results*

The mapping techniques obtain the following results: trilateration algorithm gets the smallest mean error, 9.7 metres, using 125 reference positions to estimate the AP position. With a lower number of 100 positions it is impossible to estimate the position.

Gauss-Newton algorithm uses 125 robot positions to obtain a mean error of 10.58 metres, this error is higher than the trilateration one. This method obtains better results in several situations but in some occasions it finds a local minimum and then it does not obtain the optimal solution. Moreover, Table IV shows that this method obtains good results using only

TABLE III
LOCALIZATION ERROR

Method	Trilateration	Gauss-Newton	Particle filter
Mean (m)	9.04	6.26	3.16
Max (m)	23.48	22.13	9.24
Min (m)	0.56	0.1	0.31

TABLE IV

GAUSS-NEWTON ERROR

Num_pos	5	10	30	50	100	150
Mean (m)	15.92	15.80	15.49	13.48	14.68	9.16
Max (m)	27.10	36.01	41.47	25.40	16.82	9.40
Min (m)	1.99	6.40	5.09	6.13	11.23	8.92

TABLE V

PARTICLE FILTER ERROR

Particles	100	1000	1500	2500	3500	4000
Mean (m)	5.50	4.41	4.10	3.95	3.62	3.51

50 robot positions to estimate the AP location. Then, it is necessary to spend only 12 seconds to localize the AP position.

Both methods are affected by the robot trajectory, this problem was previously commented and we have obtained better results using other robot paths in simulation mode.

Finally, the particle filter has been tested varying the number of particles in 100 experiments. Table V shows the mean error obtained in 100 experiments, it has been obtained from the moment that filter converges to the real beacon position. Sometimes, the filter converges to the mirror position due to the high noise, this noise is approximately 10 dBm and it can introduce an error of 10 metres in the observation. Then, in several executions the observation can be near to the mirror position. Results show an error that tents to decrease. The smallest error, 3.51 metres, was obtained using 4000 particles.

VI. CONCLUSIONS AND FUTURE WORKS

In this work has been presented a WiFi range-only sensor and its application to localization and mapping system. In the first time, we have analyzed the main variations of this sensor and we have proposed to use a propagation model to obtain the distance between the robot and a certain reference positions (APs). Three different techniques have been compared to localization and mapping process. Each technique has been configured and performed to obtain the best possible accuracy. We have obtained an accuracy of 3.16 metres to localize the mobile and 3.51 metres to map the environment references. In the future, we have the intention of improving the localization and mapping systems using a WiFi range-only SLAM algorithm and using an Inertial Measurement Unit (IMU) to improve the movement model and then the accuracy of the system.

VII. ACKNOWLEDGEMENTS

This work has been funded by grant S2009/DPI-1559 (Robocity2030 II-CM Project) from the Science Department of Community of Madrid.

REFERENCES

[1] 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.

[2] S. Thrun, D. Fox, and W. Burgard, "A probabilistic approach to concurrent mapping and localization for mobile robots," in *Machine Learning*, 1998, pp. 253–271.

[3] K. E. Bekris, M. Click, and E. E. Kavraki, "Evaluation of algorithms for bearing-only slam," in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA'06)*, 2006, pp. 1937–1943.

[4] J. L. Blanco, J. A. Fernandez-Madrigal, and J. Gonzalez, "Efficient probabilistic range-only slam," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'08)*, 2008, pp. 1017–1022.

[5] 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.

[6] 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.

[7] N. Priyantha, A. Chakraborthy, and H. Balakrishnan, "The cricket location support system," in *Proc. of the 6th ACM MobiCom*, 2002, pp. 155–164.

[8] 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.

[9] V. Matellán, J. M. Cañas, and O. Serrano, "Wifi localization methods for autonomous robots," *Robotica*, vol. 24, no. 4, pp. 455–461, 2006.

[10] M. Ocaña, L. M. Bergasa, M. Á. Sotelo, R. Flores, E. López, and R. Barea, "Comparison of wifi map construction methods for wifi pomdp navigation systems," *Lecture Notes in Computer Science - Computer Aided Systems Theory*, vol. 4739, pp. 1216–1222, 2007.

[11] D. B. Jourdan, J. J. Deyst, M. Z. Win, and N. Roy, "Monte carlo localization in dense multipath environments using uwb ranging," in *Proceedings of IEEE International Conference on Ultra-Wideband*, 2005, pp. 314–319.

[12] 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.

[13] 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.

[14] M. Youssef, A. Agrawala, and A. Shankar, "Wlan location determination via clustering and probability distributions," in *Proc. of the IEEE PerCom 2003*, 2003.

[15] M. A. Sotelo, M. Ocaña, L. M. Bergasa, R. Flores, M. Marrón, and M. A. García, "Low level controller for a pomdp based on wifi observations," *Robot. Auton. Syst.*, vol. 55, no. 2, pp. 132–145, 2007.

[16] M. Youssef and A. Agrawala, "Small-scale compensation for wlan location determination systems," in *Proc. of the 2003 ACM workshop on wireless security*, 2003, pp. 11–20.

[17] L. Liechty, E. Reifsnider, and G. Durgin, "Developing the best 2.4 ghz propagation model from active network measurements," in *VTC Fall*, 2007, pp. 894–896.

[18] A. Kotanen, M. Hannikainen, H. Leppakoski, and T. Hamalainen, "Positioning with ieee 802.11b wireless lan," *Personal, Indoor and Mobile Radio Communications, 2003. PIMRC 2003. 14th IEEE Proceedings on*, vol. 3, pp. 2218–2222 vol.3, Sept. 2003.

[19] A. Bose and C. H. Foh, "A practical path loss model for indoor wifi positioning enhancement," *Information, Communications & Signal Processing, 2007 6th International Conference on*, pp. 1–5, Dec. 2007.

[20] J. O. Roa, A. R. Jiménez, F. Seco, J. C. Prieto, and J. Ealo, "Optimal placement of sensors for trilateration: Regular lattices vs meta-heuristic solutions," in *EUROCAST*, 2007, pp. 780–787.