World Academy of Science, Engineering and Technology
International Journal of Electrical and Computer Engineering
Vol:6, No:4, 2012

A Novel Approach for Tracking of a Mobile Node Based on Particle Filter and Trilateration

Muhammad Haroon Siddiqui, Muhammad Rehan Khalid

Abstract**—**This paper evaluates the performance of a novel algorithm for tracking of a mobile node, interms of execution time and root mean square error (RMSE). Particle Filter algorithm is used to track the mobile node, however a new technique in particle filter algorithm is also proposed to reduce the execution time. The stationary points were calculated through trilateration and finally by averaging the number of points collected for a specific time, whereas tracking is done through trilateration as well as particle filter algorithm. Wi-Fi signal is used to get initial guess of the position of mobile node in x-y coordinates system. Commercially available software "Wireless Mon" was used to read the WiFi signal strength from the WiFi card. Visual C++ version 6 was used to interact with this software to read only the required data from the log-file generated by "Wireless Mon" software. Results are evaluated through mathematical modeling and MATLAB simulation.

Keywords**—**Particle Filter, Tracking, Wireless Local Area Network, WiFi, Trilateration

Open Science Index, Electrical and Computer Engineering Vol:6, No:4, 2012 waset.org/Publication/12519

I. INTRODUCTION

EVERY mobile node has wireless sensors which have very limited resources, such as energy and communication bandwidth. They also have limited sensing and communication ranges. Therefore, the issues that are related to these limitations need to be investigated before using these sensors in a specific application.

In this paper, we focus on mobile node tracking using Wi-Fi signals from three Access Points (APs) that are deployed in an area of interest. Laptop is used as a mobile node and wireless Local Area Network (WLAN) card is used as a sensor which processes its raw observations. The non-linear, fluctuating natures of radio signals, and their sensitivity to interference, dead spots and reflection, are well known [1]-[3]. In a structured environment such as our test area, multi-path fading is also an issue [4]. Thus radio propagation is complex enough to render analytic models impractical. The observation model must be built empirically, by sampling from the environment.

Probabilistic techniques such as the Bayes filter [5] are an attractive option to cope with the uncertainty inherent in sensing and action, and allow us to perform global localization with no initial estimate of mobile node pose.

In the present work, we choose to represent the belief function associated with the mobile node's pose by samples (or particles). A particle filter [6, 7] was chosen because of its ease of implementation [8] and fast real-time properties. The mathematical basis for our implementation of the particle filter is standard [8]. To reduce the hardware and computational complexity conventional particle filter algorithm is modified which will be discussed briefly in the next section.

Muhammad Haroon Siddiqui is with Electrical Engineering Department, National University of Sciences and Technology, NUST, Pakistan (e-mail: muhammad.haroon@ mcs.edu.pk).

Muhammad Rehan Khalid is with Electrical Engineering Department, National University of Sciences and Technology, NUST, Pakistan (e-mail: mrehan@ mcs.edu.pk).

The problem of mobile node localization has been studied in great depth. Research focusing on single-node localization with prior map knowledge includes [5, 9]. The problem of wireless signal-based localization indoors has been recently studied [10, 11] and is becoming particularly relevant given the recent proliferation of WiFi technology (particularly 802.11b). It is interesting to note that such technology could also be used to localize people using WiFi connectivity in a building. In fact, the robot localization problem is easier than the people-localization problem since it is possible to easily build a motion model for robots, and to equip them with motion sensors, which can use such a model. Previous attempts in WiFi-based localization have used data readings to model signal strength observations [10] as well as more general models that use the geometric properties of the indoor environment [11] together with knowledge about the wireless card and AP hardware characteristics. Our approach is similar to the approach in [10] in the use of a Bayes filter; however instead of using prior knowledge of signal-strength maps for all APs to construct the observation model, we have collected the data empirically and made the propagation model of the area. We incorporate a motion model into the localization estimate. We also explore the behavior of WiFi signal in the presence of different reflecting materials and objects. In one of the empirical methods studied in [11] for tracking wireless device users in a building, no Bayes filter was used and localization was performed by triangulation only. As more research effort is devoted to Wi-Fi location, it becomes increasingly important to compare algorithms fairly on common data sets taken under known conditions. The only previous effort in this regard of which we are aware is [12], which compared three different Wi-Fi location algorithms in a single-floor office building. The results revealed the behavior of the Wi-Fi signal, channel model and the performance of the filter. Our results show that accurate localization ($\approx$ 3m) is achieved in most test cases and reasonable time is reduced for execution of the algorithm.

II. PROBLEM FORMULATION

We examine the problem of single mobile node localization using wireless signal strength as the only sensor available on the mobile node. Prior signal strength data are gathered on which the observation model is based. The mobile node starts with no idea about its location and moves about the environment guided by a human controller. The localization computation is done offline in the experiments reported here, but is fast enough for the technique to be adapted for online use. The stationary points were calculated through trilateration and finally by averaging the number of points collected for a specific time, whereas tracking is done through trilateration as well as particle filter algorithm.

A. Assumptions

We make a number of assumptions in our approach. The environment is assumed to be static with sparse activity if any.

Wireless AP signals in any given area of the environment are assumed to be time-invariant in the absence of interference and signal absorption by humans. We assume there are no large variations in signal strength over small distances. The robot is assumed to move in a smooth, continuous manner and is constrained to be somewhere in the area of interest. Observations are assumed to be independent. In the absence of a robot, the effect of human proximity was eliminated before accepting the data for modeling by rejecting the initial and final 10% samples at a point. We also neglected few outlaws present in the area. In the light of these first-order assumptions, the results from experiments are quite encouraging.

B. The Particle Filter

Particle filter [7] is a practical algorithm to solve Bayesian probability problem, also named conditional density propagation condensation algorithm. The so-called particle, which is described as small-scale filter, can be considered a representative a point of the target's state. The filter in this work, created in accordance to the tutorial about particle filter for mobile robots of Ioannis Rekleitis [13]. The presented algorithm basically tries many random poses to find the most suitable one. A particle is a pose associated with a weighing function. The weight of each particle is the measure of the quality of its pose estimate and such weights (the "quality" of each pose) are calculated using Bayesian reasoning.

The main objective of particle filtering is to "track" a variable of interest as it evolves over time, typically with a non-Gaussian and potentially multi model PDF. The basis of the method is to construct a sample-based representation of the entire PDF. A series of actions are taken, each one modifying the state of the variable of interest according to some model. Multiple copies (particles) of the variable of interest are used, each one associated with a weight that signifies the quality of that specific particle. An estimate of the variable of interest is obtained by the weighted sum of all particles. The particle filter algorithm is recursive in nature and operates in two phases: Prediction and Update. After each action, each particle is modified according to the existing model (prediction stage). Including the addition of random noise in order to simulate the effect of noise on the variable of interest. Then each particle's weight is re-evaluated based on the latest sensory information available (update stage). At times the particles with (infinitesimally) small weights are eliminated, a process called resampling.

More formally, the variable of interest $(X^k = x^k, y^k, \theta)^k$ at time t=k is represented as a set of M samples ($S_j^k = [X_j^k, W_j^k]$: j=1,2,3….M) where the index "j" denotes the particle and not the mobile node, each particle consisting of a copy of the variable of interest and a weight (W_j^k) that defines the contribution of this particle to the overall estimate of the variable.

Given a particle distribution, actions need to be taken based on the mobile node's pose. Three different methods of evaluation can be used in order to obtain an estimate of the pose. First the weighted mean can be used; second, the best particle (the P_j such that $w_j = \max(w_k) : k = 1...M$) and, third, the weighted mean in a small window around the best particle (also called robust mean) can be used. Each method has its advantages and disadvantages. In this work, second method i.e. weighted mean has been used.

1) Prediction: In order to predict the probability distribution of the pose of the moving node after a motion there is a need to have a model of the effect of noise on the resulting pose. Many different approaches have been used, most of which use n additive Gaussian noise model for the motion. Any arbitrary motion $[\Delta x, \Delta y]^{\mathrm{T}}$ can be performed as a rotation followed by a translation. Mobile node's initial pose is given by $[\Delta x, \Delta y, \hat{\theta}]^{\mathrm{T}}$. First mobile node rotates by $\delta\hat{\theta} = \hat{\theta}_k - \hat{\theta}$ where $\hat{\theta}_k$= arctan $\Delta y / \Delta x$ to face the destination position, and then it translates forward by distance $\rho = \sqrt{\Delta x^2 + \Delta y^2}$. If the starting pose is $[\Delta x, \Delta y, \hat{\theta}]^{\mathrm{T}}$, the resulting pose $[x', y', \hat{\theta}_k]^{\mathrm{T}}$ is given in (1). consequently, the noise model is applied separately to each of the two types of motion because they are assumed independent.

$$\begin{bmatrix} x' \\ y' \\ \hat{\theta}' \end{bmatrix} = \begin{bmatrix} x + \rho\cos\hat{\theta}_k \\ y + \rho\sin\hat{\theta}_k \\ \hat{\theta}_k \end{bmatrix} \qquad (1)$$

2) Rotation: When mobile node performs a relative rotation by $\delta\hat{\theta}$ the noise from the odometry error is modelled as a Gaussian with mean (M_{rot}) experimentally established and sigma proportional to $\delta\hat{\theta}$. More formally, if at time $t = k$ mobile node has an orientation $\hat{\theta}_k$ then after the rotation (time $t = k + 1$) the orientation of the mobile node is given by (2). Therefore to model the rotation of $\delta\hat{\theta}$, the orientation $\hat{\theta}_j$ of each particle j is updated by adding $\delta\hat{\theta}$ plus a random number drawn from a normal distribution with mean M_{rot} and standard deviation $\sigma_{rot}\ \delta\hat{\theta}\ \delta\hat{\theta}$ $(N[M_{rot}, \sigma_{rot}\ \delta\hat{\theta}])$, where σ_{rot} is in degrees per 360°).

$$\hat{\theta}_{k+1} = \hat{\theta}_k + \delta\hat{\theta} + N(M_{rot}, \sigma_{rot}\ \delta\hat{\theta}) \qquad (2)$$

3) Translation: Modeling the forward translation is more complicated. There are two different sources of error, the first related to the actual distance travelled and the second related to changes in orientation during forward translation. During the translation the orientation of the mobile node changes constantly resulting in a deviation from the desired direction of the translation. Such effect is called drift and we model it by adding a small amount of noise to the orientation of the mobile node before and after each step as well, if the intended distance is ρ. The actual distance travelled is given by ρ plus some noise following a Gaussian distribution. Experimental results provide the expected value and the standard deviation for the drift and pure translation. Because it is very difficult to analytically model the continuous process, a simulation is used that discretizes the motion the K steps, where K is chosen to be low enough for computational efficiency but high enough in order to describe the effect of noise in forward translation.

If [$\sigma_{translation}, \sigma_{drift}$] are experimentally obtained values per distance travelled then at each step of the simulation the standard deviation used is given in (3) and (4).

$$\sigma_{trs} = \sigma_{translation}\sqrt{k} \tag{3}$$

$$\sigma_{drift} = \sigma_{drift}\sqrt{k/2} \tag{4}$$

4) Weight Update: In conventional Particle Filter every particle is assigned a particular weight as per the Euclidean distance from the point sensed by the sensor. Gaussian function is used for the purpose as given in (5).

$$W = \frac{1}{\sqrt{2\pi\sigma^2}}\, e^{\frac{-(x_s-x_p)^2}{2\sigma^2}} \tag{5}$$

5) Resampling: It is depletion and replication phase and in this phase we replicates the particle with greater weights and dropped the particles with lower weights, however overall the total number of the particles remain same. Finally, we compute the mean of the PDF to get the actual point or position of the mobile node at a particular instance as shown in Fig. 1.

C. Proposed Method

It is a simple method in which exponential and higher powers are avoided as in case of Gaussian weighting function. In this method we place a triangle with the center at origin as shown in Fig. 2. The sensed particle is placed at the middle and predicted particles are assigned weights according to the Euclidean distance from the sensed particle. Particle which is on the exact centre in other words at the exact point where the sensed particle is placed, will get the highest weight and the particle placed at the maximum distance i.e. at 'X' as shown in Fig. 2, will get minimum weight. All those particles which are out of range i.e. 'X', will be dropped or neglected as their contribution in formation of overall PDF is negligible. Mathematically triangulation weighting function is given in (6).

$$W = 1 - \frac{|x_s - x_p|}{X} \tag{6}$$

'W' is the weight, 'x_s' is the sensed particle,' x_p' is the predicted particle and 'X' is the range of weighting window used to normalize the weights.

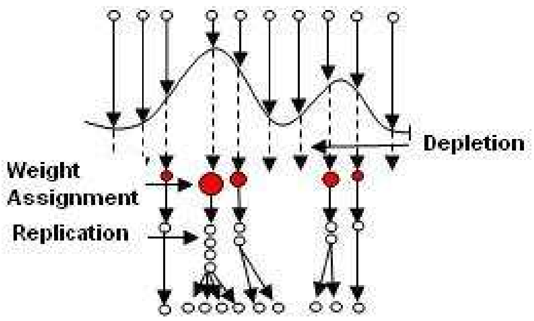

Fig. 1 Resampling process: Starts with ten particles assigning them weights and after depletion and replication again ten particles are received

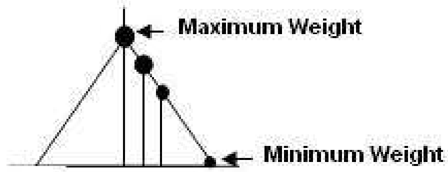

Fig. 1 Triangulation Weighting Function

III. System Architecture

The accurate positioning of a particular device is a major challenge. State-of-the-art wireless tracking technologies, such as wireless sensors and proprietary WLAN based sensors; typically require a costly dedicated network infrastructure. However, a 802.11 based tracking system, can be deployed without significant additional cost. Since many buildings are now rapidly deploying facility-wide 802.11 WLAN infrastructures, a WLAN based tracking approach would provide a cost-effective solution that takes advantages of their wireless infrastructure for tracking.

A. Access Points APs

A real-time system based on an existing 802.11 infrastructure is deployed in order to track WLAN tags. WiFi routers are used as APs. The first step of the deployment procedure is the data collection phase. Once the APs were established on known positions, the Received Signal Strength (RSS) values were collected from the APs as a function of the mobile node's location and orientation. From the measurements, we noted that the RSS values at a given location vary significantly depending on the mobile Node's position. Thus, RSS values were collected at a number of selected physical locations on the ground. After the data collection phase, the RSS values were imported into the positioning engine and processed to enhance the accuracy of location estimation. After constructing a database of RSS measurements, along with their known 2-dimensional locations and orientations, called Look-up table, the system can estimate mobile node's position by comparing the measured RSS data to the known values in the Look up table or database. To reduce the computation cost, the search is performed only on some portion of the RSS measurements in the database. If mobile node's previous location lies at a point *P*, then the search space is limited to its neighboring points within the distance *d* from *P*. These neighboring points are grouped into clusters based on their physical closeness. For each cluster, the most probable location of the mobile node is calculated based on the Euclidean distance of RSS measurements.

B. WiFi Received Signal Strength Indicator (RSSI)

Laptop is used as a mobile node to move around in the area illuminated by the APs. Commercially available software "wireless Mon" is used to read the WiFi signal strength from the WiFi card. It generates a log file in a particular file folder as per the path given to it with AP's name, date, time and signal strength etc. A code, written in Visual C++ version 6, to interact with this software and read only required data from the log file so that the signal strength of each AP is acquired and is converted into distance with the help of Look up table.

C. Signal Propagation model of the Area

Collecting the RSS database empirically is performed by war-walking or war-driving. The process entails systematically measuring RSS data from the detectable AP's over the area where localization is to be performed. Each RSS measurement is tagged with a location reference and stored as tuples of the form {RSS, Location}. The process of measuring the RSS's is time consuming, however, it is essential for the calculation of the distance. There are two methods to calculate the distance, firstly, either by making a lookup table mentioning the distance against the RSS basing on the empirical data collection and secondly, a model by comparing and observing the relationship between the theoretical and empirical data i.e. the theoretical value of the distance calculated by the mathematical equations with the change of RSS by increasing the distance. By using these distances and already known APs location with the help of trilateration method, the location of mobile node is calculated.

D. Trilateration

In this paper trilateration approach is proposed to determine the position (two dimensional locations: X and Y Coordinates) of a mobile node using received signal strength indicator (RSSI), received from a fixed set of access points with known locations. The proposed algorithm employs the Particle Filter to generate a better estimation of RSSI and of the distance calculated from RSSI.

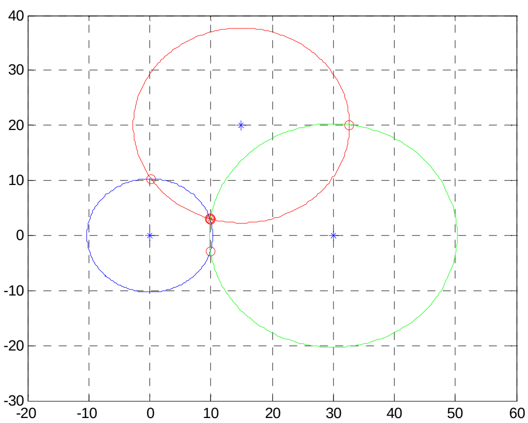

Fig. 3 Trilateration

In this phase the distance between mobile device and access point (AP) is estimated based on the signal strength received by the device. The power of the transmitted signal from the AP is different from that received at the mobile device, due to attenuation and other factors. The received signal strength indicator (RSSI) of an AP at a location is an indicator of how strong the signal from that access point is at that location.

A node N has determined the distances from itself to each of three (or more) other nodes $A, B, C,$ and so on Fig.3. The other nodes' geographic locations (coordinates) are known: $A = (a_x\, a_y), B = (b_x\, b_y),\ C = (c_x c_y)$, and so on. The distances from node N to each of the other nodes are a_r, b_r, c_r, and so on. The trilateration problem is to find the coordinates of node $N = (n_x n_y)$ from the given information. Calculations of coordinates X and Y is given by (7) and (8).

$$X_{AB} = \frac{x_A + x_B}{2} + \frac{(x_B - x_A)(r_A^2 - r_B^2)}{2d^2} \pm \frac{y_B - y_A}{2d^2}\sqrt{[(r_A + r_B)^2 - d^2)(d^2 - (r_B - r_A)^2]} \quad (7)$$

$$Y_{AB} = \frac{y_A + y_B}{2} + \frac{(y_B - y_A)(r_A^2 - r_B^2)}{2d^2} \pm \frac{x_B - x_A}{2d^2}\sqrt{[(r_A + r_B)^2 - d^2)(d^2 - (r_B - r_A)^2]} \quad (8)$$

IV. Results

The objective was to simulate the process that depicts a reasonable level of accuracy in order to comprehensively test the proposed methodology in real-time scenario. It was also necessary to maintain a certain degree of realism while ensuring the possibility of occurrence of worst case scenario without design. Accordingly, maximum randomness was incorporated in throughout the process, whereas randomness was within the bounds of the MATLAB software used.

The system used for achieving the results given is DELL Inspiron mini-10 laptop with genuine Windows XP SP2; having a 1 GB of RAM and an Intel Atom processor running at 1.6 GHz. MATLAB was used to carry out simulations and Microsoft Visual C++ version was 6.0.

A. Track Generation

A path of a mobile node is generated by using random data. In initial tests, 100 points are taken. The resultant Path is plotted in a curve form and is shown in Fig. 4.

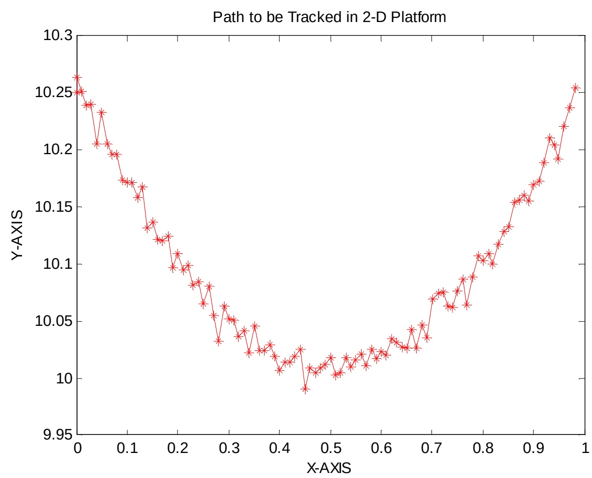

Fig. 4 Path Created Using Synthetic Data

B. Tracking using Gaussian weighting function

This is an existing method for giving weights to each predicted particle as per the Euclidean distance from the sensed particle, in conventional particle filter algorithm. In this method Gaussian expression is used (5) for the calculation of the weights of each predicted particle. The result with Gaussian weighting function is shown in Fig.5. Red '*' shows the original path of the mobile node generated using synthetic data, and blue 'o' shows the output of the particle filter algorithm with Gaussian weighting function.

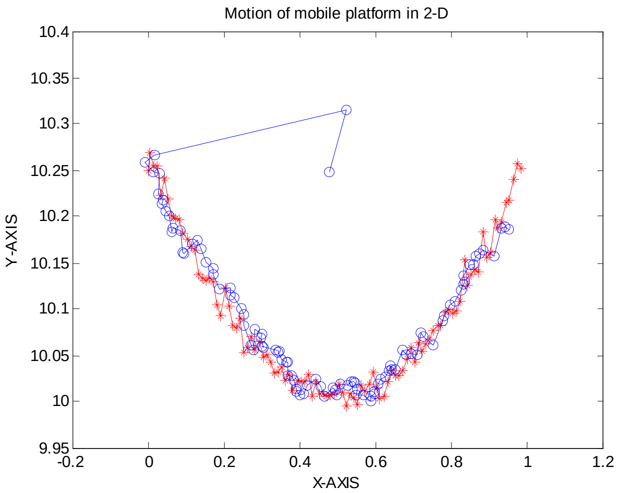

Fig. 5 Tracking Using Gaussian Weighting Function

C. Tracking using Triangular weighting function

The result with Triangular weighting function is shown in Fig.6. Where red '*' is the original path of the mobile node generated by using synthetic data and blue 'o' are the output of the particle filter algorithm with Triangular weighting function.

Mobile nodes are generally battery powered. The critical aspects to face concern how to reduce the energy consumption of nodes, so that the node's lifetime can be extended to reasonable times. Typically, a mobile node includes three basic components: a sensing subsystem for data acquisition from the physical surrounding environment, a processing subsystem for local data processing and storage, and a wireless communication subsystem for data transmission. In addition, a power source supplies the energy needed by the node to perform the programmed task.

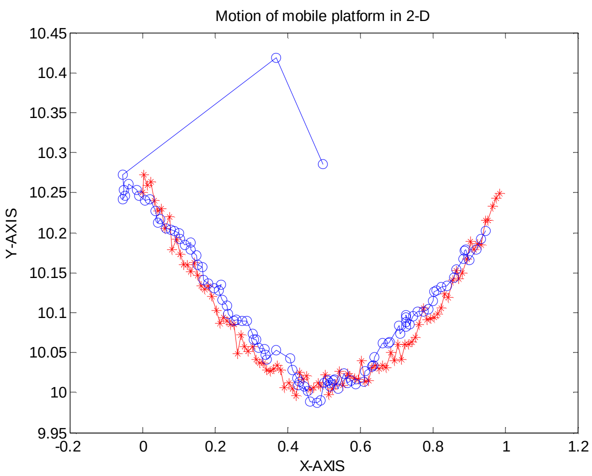

Fig. 6 Tracking Using Triangular Weighting Function

In case of Particle Filter (PF) we need to assign weights to each particle for the estimation of a single location. In our proposed system, there are about one thousand particles in each estimate. Gaussian window method is used for assigning weights, whereas triangular weighting function is proposed in this work to reduce the computational complexity. Triangular weighting function requires comparatively lesser execution time than Gaussian weighting function, at the cost of a slight increase in RMSE.

V. Conclusion

In this research work, a filtering algorithm is presented for the localization of mobile node, without any constraint or assumption about the node mobility model. Our approach relies on an analytical evaluation of the perturbation of the received signal strength from IEEE 802.11 access points. Based on signal strength measurements, the algorithm aims to correct the estimated node position in real time. The proposed method has the advantage of being based on analytical calculations, which gives closed formulas for the position estimates. Therefore, the implementation of the algorithm is simple and suitable for nodes of limited computational power, while still sufficiently accurate for use in indoor or outdoor environments. The performance of the proposed algorithm is evaluated with numerical simulations. The results highlight that proposed algorithm results in lower execution time at the cost of a slight increase in RMSE.

Acknowledgment

The authors would like to thank Aamir Munir (FAST-NUCES, Islamabad, Pakistan) and Khubaib Ahmed (MAJU, Islamabad, Pakistan) for their influential support and cooperation. Aamir Munir helped in creating log files and Khubaib Ahmed helped in simulation setup.

References

[1] A.Chakraborty, "A distributed architecture for mobile, location-dependent applications," M.S. thesis, Massachusetts Institute of Technology, May 2000.
[2] T.Cutler, "Wireless ethernet and how to use it," in *The* Online Industrial Ethernet Book, Issue 5. 1999.
[3] A.Neskovic, N.Neskovic, and G.Paunovic, "Modern approaches in modeling of mobile radio systems propagation environment," IEEE Communications Surveys, 2000.
[4] H.Hashemi, "The indoor radio propagation channel," in *Proceedings of the IEEE*, July 1993, vol. 81, pp. 943–968
[5] D.Fox, W.Burgard, and S.Thrun, "Markov localization for mobile robots in dynamic environments," *Journal of Artificial Intelligence Research*, vol. 11, pp. 391–427, 1999.
[6] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 174–188, Feb. 2002.
[7] P. M. Djuric, M. Vemula, and M. F. Bugallo, "Target tracking by particle filtering in binary sensor networks," *IEEE Trans. Signal Process.*, vol. 56, no. 6, pp. 2229–2238, Jun. 2006.
[8] D.Fox, S.Thrun, F.Dellaert, andW.Burgard, "Particle filters for mobile robot localization," in *Sequential Monte Carlo Methods in Practice*. Springer-Verlag, New York, 2001.
[9] J.J.Leonard and Durrant-Whyte, "Mobile robot localization by tracking geometric beacons," *IEEE Transactions on Robotics and Automation*, vol. 2, pp. 1080–1087, 1991.

[10] A.M.Ladd, K.E.Bekris, G.Marceau, A.Rudys, D.S.Wallach, and L.E.Kavraki, "Robotics-based location sensing for wireless ethernet," in *Eighth ACM International Conference of Mobile Computing and Networking (MOBICOM 2002)*, Atlanta,GA, September 2002.
[11] P.Bahl and V.N.Padmanabhan, "Radar: An in-building rfbased user location and tracking system," in *Proceedings of IEEE Infocom 2000*, Tel- viv,Israel, March 2000, vol. 2, pp. 775–784.
[12] T. Roos, P. Myllymaki, H. Tirri, P. Misikangas, and J. Sievanan. A probabilistic approach to wlan user location estimation. International Journal of Wireless Information *Networks*, 9(3), July 2002.
[13] Ioannis M. Rekleitis, "A Particle Filter Tutorial for Mobile Robot Localization".Technical Report TR-CIM-04-02,Centre for Intelligent Machines, McGill University, Montreal, Quebec, Canada, 2004.
[14] A. Goldsmith, "*Wireless Communications"*, Cambridge University Press, 2005.