Research Article

INEMO: Distributed RF-Based Indoor Location Determination with Confidence Indicator

Hongbin Li,[1] Xingfa Shen,[2] Jun Zhao,[1] Zhi Wang,[1] and Youxian Sun[1]

[1] *State Key Laboratory of Industrial Control Technology, Zhejiang University, Hangzhou 310027, China*
[2] *Institute of Computer Application Technology, Hangzhou Dianzi University, Hangzhou 310018, China*

Correspondence should be addressed to Zhi Wang, wangzhi@iipc.zju.edu.cn

Using radio signal strength (RSS) in sensor networks localization is an attractive method since it is a cost-efficient method to provide range indication. In this paper, we present a two-tier distributed approach for RF-based indoor location determination. Our approach, namely, INEMO, provides positioning accuracy of room granularity and office cube granularity. A target can first give a room granularity request and the background anchor nodes cooperate to accomplish the positioning process. Anchors in the same room can give cube granularity if the target requires further accuracy. Fixed anchor nodes keep monitoring status of nearby anchors and local reference matching is used to support room separation. Furthermore, we utilize the RSS difference to infer the positioning confidence. The simulation results demonstrate the efficiency of the proposed RF-based indoor location determination.

1. INTRODUCTION

With the recent development of micro-electro-mechanical system (MEMS), inexpensive networked sensor systems which work autonomously are available for context-aware computing. A context-aware system can sense time, location, temperature, and related resources to handle the current situation. Moreover, this kind of system can utilize the variance of context to adapt its behaviors, such as communication and sensing patterns, without user intervention.

Determining user's location is one of the most important issues in context-aware computing. Sensing result without location information may be inapplicable. For example, if the office resource system is able to manage the locations of assets, users can always check out the assets location online without bothering other staff. A cell phone chooses to ring or reject a business call based on the situation whether the user is in his/her office or not. In a scenario of museum navigation, an electronic narrator speaks to the visitors based on their current locations. All in all, knowing the location can help a system do the right thing at the right place.

Previously, we have proposed NemoTrack [1], an RF-based outdoor tracking prototype system. In our latest exper-iment with 20 Mica2 nodes [2] placed on a 5×4 grid with 1 meter displacement in between, the result shows an over-all tracking accuracy of around 30 cm. The main feature of NemoTrack is the dynamic tracking group management [3], which enables sensor nodes waking-up and quitting based on whether the target of interest is approaching or leaving the specific region. The autonomously elected group leader manages the sensor result at each sensing circle and hands off the leadership to the prospective node when the target is leaving the current group. In an indoor environment, however, sensor nodes cannot be placed regularly in grid form due to complex and unfavorable building layout. Moreover, the characteristic of RF propagation is severely affected by multipath interference phenomenon. As a result, it is very difficult to import an outdoor localization system directly into an indoor environment.

In this paper, we propose a novel approach for RF-based indoor location determination Indoor NEMO track, or IN-EMO for short. INEMO provides two levels for positioning accuracy: room separation and cube determination. Room separation computes which room or corridor that the target is in and cube determination computes which office cube the target is placed in. The key idea of INEMO is that all sensor

nodes maintain small sets of latest neighboring RSS data and utilize the data sets as reference in target positioning. Our method does not require nodes to keep global information and it is free from site-survey and signal precollection. However, we assume that all background sensor nodes know their room/corridor ID and relative coordinates, which is easy to satisfy during a setup stage. Additionally, a positioning confidence indicator (PCI), derived from RSS differences between pairs of nodes, is provided for every estimate to capture the environmental complexity. The simulation results demonstrate the efficiency of the proposed RF-based indoor location determination.

This paper is organized as follows. Section 2 presents a brief survey of related work. Then, Section 3 introduces the wireless environment and reports the characteristics of RSS difference between a pair of Mica2 nodes. Section 4 describes our approach while Section 5 presents simulation results of room separation. Section 6 validates our idea through system implementation and analysis. Section 7 concludes this paper and states our future work.

2. RELATED WORK

Many efforts have been made to provide reliable indoor location service. The active badge location system [4] is an early user-tracking system. The building is populated with a wired network of sensors, which receive a unique code emitted in infrared by users. Infrared is chosen because of its inability to penetrate partition walls in office buildings.

The cricket location-support system [5] uses RF and ultrasound together to achieve accurate ranging. The beacons, which are mounted on chosen locations, emit RF and ultrasound signals simultaneously. The moving targets, namely, listeners, infer distance from a beacon by estimating time difference between reception of RF and ultrasound. Thus, listeners can easily estimate their position by triangulation.

The above two techniques require line of sight (LOS) for receivers and transmitters and they suffer from limited range. The RF technique is a promising option since it has longer communication range, non-LOS transmission ability, and is becoming more pervasive with the development of Wi-Fi and wireless sensor networks.

The RADAR [6] uses RF to estimate locations. Two methods are proposed. The first one is called empirical method, in which a site-survey is needed to create a signal database. At runtime, the system tries to match the signal measured to the database and give location estimations. The second method skips the site-survey and uses a radio propagation model to infer signal patterns in certain positions. However, it suffers from the inaccuracy of the radio propagation model due to the multipath phenomenon.

Many other Wi-Fi-based localization systems have been proposed, which can be further categorized according to their signal processing methods. Model-based approaches collect RSS measurements to infer distances between target and reference points, and then apply triangulation method to derive the target location [7–9]. Other approaches establish RSS-maps through site-survey and signal precollection

and compute the targets position using different algorithms [10–12].

Wi-Fi-based localization systems rely on electrical and network infrastructures, thus cannot be easily deployed in noninfrastructured environments such as a warehouse and a greenhouse. Even in infrastructured environments such as office buildings, APs deployment are still constrained by electrical and network profile. Conversely, the wireless sensor network paradigm, which hardly relies on infrastructure, provides an alternative. In the following paragraph, we summarize some of the most recent advances in sensor network localization.

MoteTrack [13] collects signal strength signatures from numerous beacon nodes and stores the signature database on beacon nodes. At runtime, the target matches the received signature to the database and gives the positioning result. The main feature of MoteTrack is that it can tolerate the failure of up to 60% of the beacon nodes without severely degrading the accuracy. However, MoteTrack suffers from complex signal map construction. Dozens of signatures have to be collected for every reference point. MERIT [14] tracks users to a room granularity by comparing average values of RSS in different rooms and it introduces RF reflectors for better spatial diversity. It achieved an accuracy of 98.9% in best cases for room granularity. Ecolocation [15] determines the location of unknown nodes by examining the ordered sequence of RSS measurements taken at multiple reference nodes. Reference [16] uses techniques like frequency diversity and averaging multiple measured data to overcome multipath propagation and enhances the accuracy of weighted centroid localization by simple optimizations. In this paper, we follow the signature database approach, similar to MoteTrack but considering room granularity instead and removing the signature collection, which allows us to improve the localization accuracy.

3. INDOOR WIRELESS ENVIRONMENT

In this section, we characterize the wireless medium used in our system through a series of experiments. We begin with a description of our experimental setup and then we discuss the RF signal propagation and the noisy wireless channel characteristics that make location estimation a challenging task and constitute the motivation for our approach. Later we report on the characteristics of RSS differences between pairs of nodes.

3.1. Overview of the environment

Our experimental testbed is set up in the fifth floor of our department. Figure 1 shows the testbed layout. The building is equipped with Berkeley Mica2 nodes. They communicate in 900 MHz frequency bandwidth with the low power RF transceiver CC1000. We place the nodes AC1 and AC2 in room 517. AC2 is connected to an MIB510 programmer board via 51 pin interface and the programmer forwards the message to an IBM laptop via RS232 serial port. Using the program we developed in TinyOS [17], AC2 sends *Hello* messages to AC1 periodically and AC1 sends back the replies.
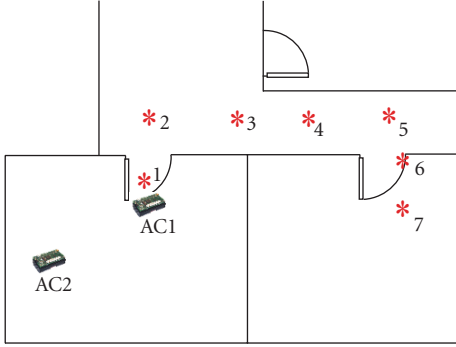
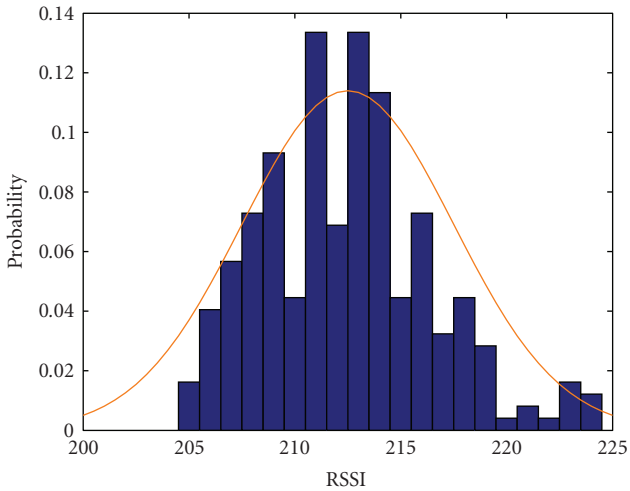FIGURE 1: Test layout for wireless environment.

FIGURE 2: The RSSI histogram for AC1 to AC2 at fixed location.

The signal strengths of these messages are stored in a laptop for analysis.

3.2. Characteristics of signal propagation and measurement

We collected 250 sequential RSS readings from AC2 and depict them in Figure 2 in the form of a normalized histogram. The RSS is quantified by the received signal strength indicator (RSSI) which is provided by the CC1000 [18] component of Mica2. We will explore more about RSSI in Section 4.

As we can see the wireless channel is very noisy. Due to reflection, diffraction, refraction, and absorption by obstacles and moving objects (e.g., human), signal propagation suffers from severe multipath effects in an indoor environment [19]. That is, RF signal can reach the destination through different paths, with different amplitude and phase. The multipath power at receiver is determined as the sum of all individual powers regardless of the phase of each path.

Also, changes in the environmental conditions, for example, temperature, humidity, or light, affect the propagation to a certain extent. We also observe that hardware diversity has large impact on RSSI measurements. For one transmitter, different receivers measure different RSSI readings, while one receiver can measure different RSSI readings from transmitters working on the same output power. The orientation and height of the omnidirectional antenna also affect the measurement in a certain degree. For the sake of simplicity, we define the synthetical impact of hardware diversity as *RSSI offset*. For a pair of nodes, the measured RSSI is the sum of ideal RSSI and the corresponding offset. For the detail of the hardware diversity and RSSI behavior of Mica2, the readers shall refer to [20].

3.3. Characteristics of RSS difference

In indoor positioning applications, anchors and targets exchange messages regularly, which helps the distributed anchors identify the attributes and needs of the target. Here we conduct experiments for RSS difference characteristics in both temporal and spatial senses. To characterize the RSS difference variation with time, we placed a pair of nodes AC1 and AC2 in room 517 as shown in Figure 1. AC1 sends a message to AC2 every 6 seconds, and AC2 sends replies to AC1 with sensed RSSI after reply intervals 0.5, 1, 2, 3, 4, 5 seconds respectively. We show the RSSI readings for 0.5- and 5-second intervals in Figure 3. Generally, in cases of small reply interval, RSSI readings of AC1 and AC2 match well, and the curve of RSSI difference is stable. With larger reply interval, RSSI readings of AC1 and AC2 behave more differently. As a result, the RSSI difference varies severely. With respect to the variations with space, we fix AC1 and place AC2 in the locations indicated by * as shown in Figure 1. AC1 sends one message every 3 seconds and AC2 replies 1 second later. In Figure 4, we show the RSSI readings and their difference with AC1 in places 1 and 7. Obviously, RSSI curves of AC1 and AC2 and their difference are stable when AC2 is placed near AC1. When they are separated with longer distance, the curves show larger fluctuation.

The fluctuation of the RSSI difference is able to reflect the temporal and spatial characteristics of the environment. The degree of fluctuation indicates the environmental complexity, that is, we can use the RSSI difference between a pair of nodes to infer the environmental complexity and the trustworthiness of the localization determined in the positioning phase.

4. INEMO OVERVIEW

The design of INEMO aims at satisfying two goals: providing coarse-grained (room) and fine-grained (office cube) positioning information with one system. In our approach, an office building is populated with tiny Mica2 nodes. Typically, we place 4 nodes per room, one in each corner, so that most regions can be covered. In corridors we place nodes according to the building profile. These static nodes act as anchors, with unique ID and user designated coordinates. In the current version, all anchor IDs and coordinates are injected in the setup phase. Then, they run in a totally distributed way, spontaneously maintaining neighbors status and cooperating for positioning with no central supervision. The functions of anchors are the following.
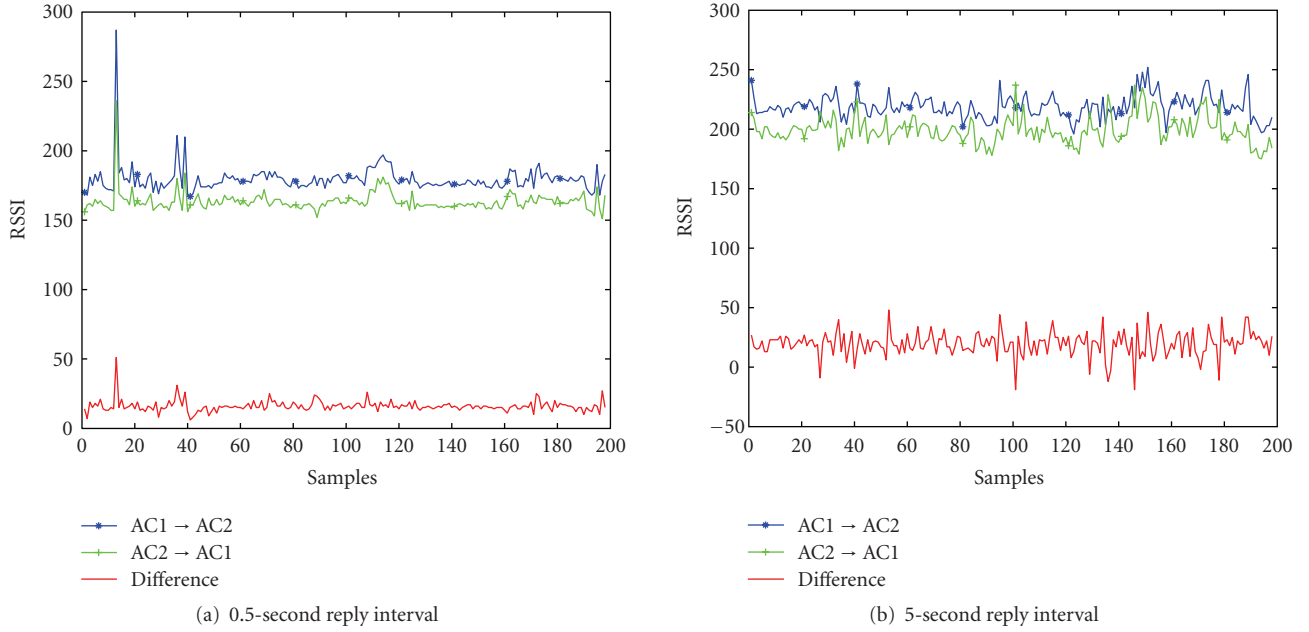
(a) 0.5-second reply interval

(b) 5-second reply interval

FIGURE 3: Temporal characteristics of RSSI difference.
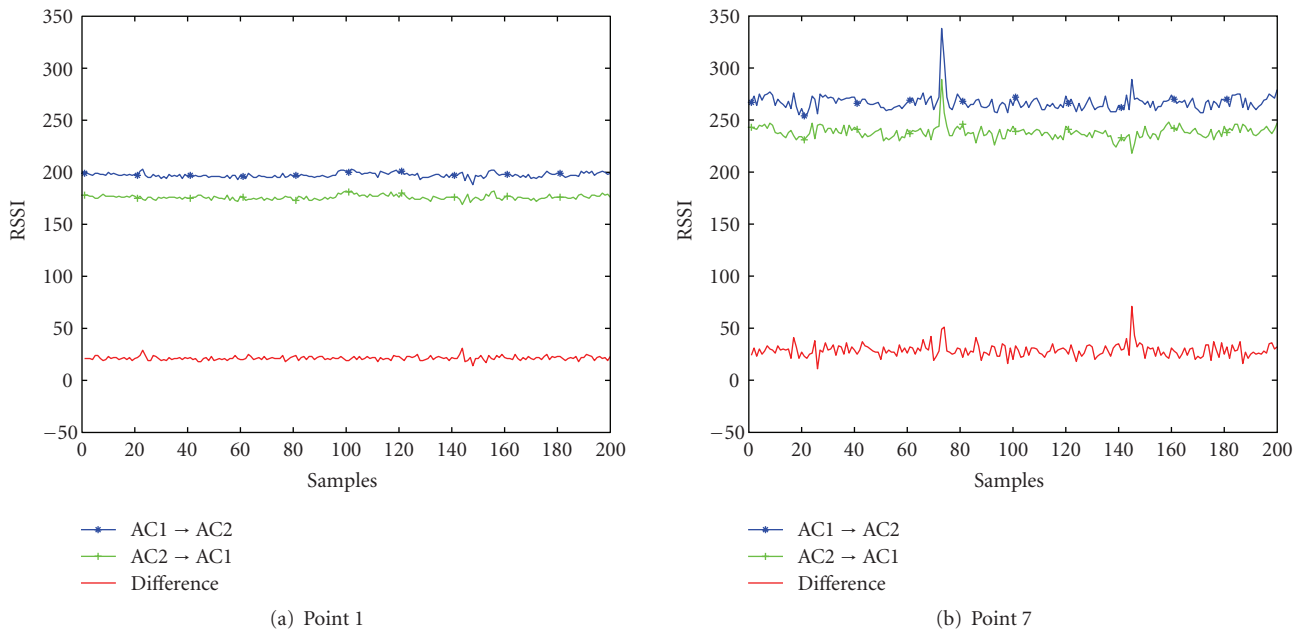

(a) Point 1

(b) Point 7

FIGURE 4: Spatial characteristics of RSSI difference.

(1) *Periodic "Hello" message broadcasts:* each anchor periodically sends *"Hello"* messages with a fixed transmission power.

(2) *Monitoring the nearby anchors:* each anchor receives *"Hello"* messages and maintains a statistical list of RSS values sensed from other anchors.

(3) *Reply to target positioning requests:* on hearing targets requests, the relevant anchors reply with the concerned information while the others stay silent.

The first two functions enable the monitoring of environment dynamics and tracking of anchors removal and joining. By periodically broadcasting and updating, anchors keep an up-to-date status of nearby ones, that is, both anchor existence and RSS behavior. The third function helps the target to acquire its position. If the target requests room granularity, all nearby anchors reply with the full statistical RSS list. If cube granularity is requested, only anchors in a specific room reply with a statistical RSS list
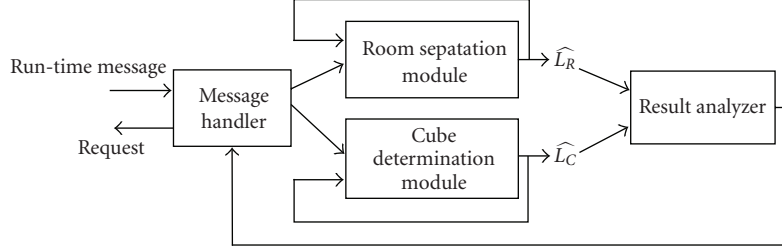
FIGURE 5: The target centric INEMO positioning approach.

which contains RSS information of the anchors in the same room.

The fixed anchors work in the background, which means they only exchange "*Hello*" messages among themselves and passively answer requests from targets. The positioning procedure is target centric, which is depicted in Figure 5. The *message handler* keeps the current granularity requirement, according to which the handler sends out the corresponding request and also dispatches received replies to *room separation module* or *cube determination module*. After determining the location, the two computation modules keep the localization results $\widehat{L_R}$ or $\widehat{L_C}$ via closed loops as required by Bayesian methods, and also send them to the *result analyzer*. This last component decides whether the current localization result is correct and satisfactory according to the requirement.

The positioning procedure works as follows: when a moving target (typically a person carrying a device) enters a room and wants to know its location, it first broadcasts a room granularity request to nearby anchors. Then the *message handler* forwards the upcoming replies to the *room separation module*. After several rounds of estimation, the *result analyzer* deems that the user is in a certain room and changes the requirement to cube granularity. The *message handler* begins to send requests of cube granularity and forwards the replies to *cube determination module*. When the target leaves the room, the *result analyzer* senses that the results are no longer correct, causing the requirement to switch back to room granularity.

4.1. Room separation

In the *room separation module,* we use the RSSI distance and Manhattan distance to evaluate the node-to-node closeness. It is expected that RF signals sent from neighboring rooms would encounter reasonable attenuation and the receiver would get lower RSS readings (or no readings at all) than those sent from the current room.

The RSSI is determined by an analogue-to-digital converter which measures the voltage over a 27 kΩ resistor, and the voltage is in the range of 0 to 1.2 V. The relationship between RSSI and distance is extensively studied in [16, 20, 21] whose conclusion is that RSSI is a reasonable distance metric. The RSSI value can also be easily converted into power (in dbm) from CC1000 datasheet [18]:

$$P_{\text{dbm}} = -50 \times \text{Voltage}_{\text{battery}} \times \frac{\text{RSSI}_{\text{raw}}}{1024 - 45.5}. \qquad (1)$$

Here, we present another distance metric that utilizes the RF indoor attenuation characteristics. We employ the Manhattan distance metric, which is inspired on MoteTrack [13]:

$$M(i, t) = \sum_{j=1, j \neq i}^{n} |\text{RSSI}_{jt} - \widetilde{\text{RSSI}_{ji}}| + \text{RSSI}_{it}, \qquad (2)$$

$$n = \text{number of } in - \text{range anchors of } i,$$

$$\begin{aligned}
\widetilde{\text{RSSI}_{ji}} &= \text{RSSI statistic of anchor } j \text{ to } i, \\
\text{RSSI}_{jt} &= \text{RSSI from anchor } j \text{ to target } t, \\
\text{RSSI}_{it} &= \text{RSSI from anchor } i \text{ to target } t.
\end{aligned} \qquad (3)$$

We use the Manhattan distance aiming at neutralizing the RSSI fluctuation. The Manhattan distance, derived from the difference of RSSI database of node pairs, infers how well these two nodes match. The RSSI readings of neighboring nodes are utilized in such a way that the Manhattan distance is referred to multiple nodes instead of measuring a single RSSI. Here we propose several methods for room separation using RSSI distance or Manhattan distance.

(1) Minimum distance (MD): the target is considered to be in the same place as the anchor with minimum distance.

(2) Minimum averaged distance (MAD): anchors are clustered by their room ID and distances are averaged. We deem a target to be in the room with the minimum average distance to the respective cluster.

(3) N-weighted sum distance (NWSD): we sort the distances in ascending order and assign weighting factors accordingly. For example, if we have M distances we then assign weight M to the smallest number and M-1 to the second smallest, and so forth. Finally, we pick the N smallest distances from each room and sum up their weighting factors. We deem a target to be in the room with the largest N-weighted sum. Note that method (1) is a special in which $N = 1$.

Furthermore, we utilize previous estimates to recursively compute the probability of each room with a Bayesian technique (shown by the closed loop in Figure 5). This filtering procedure can effectively filter outliers, for example, 100 consecutive estimates to room A can avoid a sudden estimate to room B. Note that the Bayesian technique is not included in the evaluation sections because we are more interested in knowing the performance from immediate results.

4.2. Cube determination

In the *cube determination module*, only anchors in a specific room are asked to reply, thus alleviating wireless channel collisions. As we deploy anchors in room corners, we use weighted centroid localization (WCL), which is easy to implement in energy-constrained sensor nodes. After gathering the results of related anchor nodes, the unknown target t estimates its approximate position by a weighted expression:

$$(x_{\text{est}\_t}, y_{\text{est}\_t}) = \left(\frac{\sum_{i=1}^{n} w_{it} x_{\text{anch}\_i}}{\sum_{i=1}^{n} w_{it}}, \frac{\sum_{i=1}^{n} w_{it} y_{\text{anch}\_i}}{\sum_{i=1}^{n} w_{it}} \right), \quad (4)$$

where

$$\begin{aligned} w_{it} &= \text{ weight of anchor } i \text{ and target } t \\ n &= \text{ number of anchors.} \end{aligned} \quad (5)$$

Each anchor node contributes with the sensing result as a distance metric to the computing process. The relation between sensing result and weighting factor is dependent on the sensor model. In ultrasound ranging technique, for instance, the time of flight is proportional to the distance between nodes. While in the case of RF ranging, the widely accepted relationship between distance and received power is defined by the log-normal shadowing model:

$$P_r(d)|_{\text{dBm}} = P_r(d_0)|_{\text{dBm}} - 10\alpha \, \log\left(\frac{d}{d_0}\right) + X_{\text{dB}}. \quad (6)$$

$P_r(d)$ and $P_r(d_0)$ denote the received power at an arbitrary distance d and a reference distance d_0 from a transmitter. α is the path loss exponent and it is environment dependent. For instance, line-of-sight of indoor environment shows an α value around 1.6 to 1.8, and around 4 to 6 in the presence of obstacles [21]. The last part of the model denotes the variation of the received power $X_{\text{dB}} \sim N(0, \sigma_{\text{dB}}^2)$. From this model, the weight w_{it} can be replaced by $\sqrt[\alpha]{P_{it}(mW)}$.

In our study, 4 anchors, one each corner, are placed at a typical office room. Consequently, these 4 anchors can cover most regions of the office making WCL possible and reasonable. We calculate weight factors $w_{1t}, w_{2t}, \ldots, w_{nt}$ by (1) and then calculate the estimated target location by (4).

4.3. Positioning confidence indicator

In our experience of developing positioning techniques, we found that merely giving location estimate without a confidence indication is not enough. In some regions, for example, wherever line-of-sight is available for all anchors with good connectivity, location estimates are very stable. When the target is in the vicinity of obstacles, for example, walls or doors, location estimates highly deviate from the true location. Thus, these estimates may not be acceptable. Inspired

by the RSS difference of pairs of nodes, we propose the positioning confidence indicator, which is defined as follows:

$$\begin{aligned} \text{PCI} &= \sum_{1 \le i \le n} \sqrt{\frac{1}{m-1} \sum_{j=1}^{m} \left(\left| \text{RSSI}_{tij} - \text{RSSI}_{itj} \right| - \overline{\lambda_i} \right)^2} \\ \overline{\lambda_i} &= \frac{1}{m} \sum_{j=1}^{m} \left| \text{RSSI}_{tij} - \text{RSSI}_{itj} \right|, \end{aligned}$$

n = number of anchors involved in positioning,

m = size of sliding window,

RSSI_{tij} = RSSI readings from target t to anchor i at period j.

$$(7)$$

Since our current sensor node platform does not support floating point calculation, we modify the PCI definition slightly to enhance computation efficiency:

$$\text{PCI}' = \sum_{i=1}^{n} \left(\frac{1}{m} \sum_{j=1}^{m} \left| \left| \text{RSSI}_{itj} - \text{RSSI}_{tij} \right| - \overline{\lambda_i} \right| \right). \quad (8)$$

This indicator works as follows: target t periodically broadcasts positioning requests to nearby anchors and in-range anchors send back the RSSI reading to the target. Then, the target gains pairs of RSSI readings in every period, namely, RSSI_{itj} and RSSI_{tij} for anchor i at time sequence j. We calculate the deviation of samples $\text{RSSI}_{itj} - \text{RSSI}_{tij}$ of the latest m periods, and sum up the deviations of in-range anchors to be the PCI.

5. SIMULATION AND PERFORMANCE EVALUATION

In this section, we present a performance evaluation of room separation using simulations.

5.1. Simulation model and parameters

We use the log-normal shadowing model (6) and RSSI conversion (1) to generate RSSI samples as a function of distance. Then we scale the RSSI range from 0 to 300, for easier comparison without affecting the performance. We set our simulation environment and place our virtual anchors and testpoints exactly as our real world implementation (Figure 6), which will be shown in the next section. The path loss exponent α is set to be 3 for line-of-sight nodes, and 0.75 is added for each wall obstruction. For instance, $\alpha = 3$ for nodes 1, 12, 13, and 16; $\alpha = 3.75$ for nodes 1, 13, 13, and 17; $\alpha = 4.5$ for nodes 1 and 17.

The following RF channel characteristics are considered in our simulation:

(i) *RSSI variance* measures the degree of RSSI fluctuation due to the multipath phenomenon;

(ii) *RSSI offset* represents the hardware and environmental effects on the RSSI measurement;

(iii) *packet loss rate* represents the wireless channel traffic due to collisions.
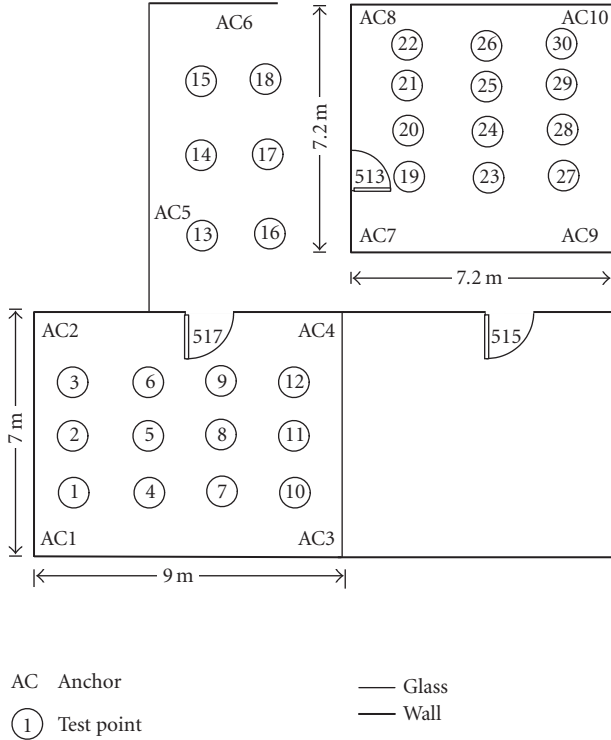
Figure 6: Test layout for INEMO.

5.2. Simulation results

The performance of the proposed algorithms are evaluated by room separation accuracy. For each set of parameters, 200 computations are made in each testpoint.

To study the effect of RSSI variation we assume ideal values for the remaining parameters which means RSSI readings have zero offsets and packet loss rate is 0%. As shown in Figure 7(a), positioning accuracies of all the six algorithms decrease as the RSSI variation increases. MD-Manhattan and MAD-Manhattan show the best performance, and 2WSD-RSSI is the poorest. When RSSI readings fluctuate severely, even sliding window filters fail to estimate the true value. 2WSD-RSSI is sensitive to RSSI variation because the RSSI values become more unpredictable as the variation increases. On the other hand, 2WSD-Manhattan shows median performance among all because Manhattan distance is a synthetic result of reference matching and RSSI variations are more or less neutralized.

Packets loss is common in sensor network applications. To analyze the sensitivity to this parameter we assume zero RSSI offsets and RSSI variation of 20. As shown in Figure 7(b), Manhattan-based algorithms degrade very fast as the packet loss rate increases. When packets from nearby anchors are lost, large penalties are added to the Manhattan distance. The incomplete reference information makes the Manhattan distance corrupted. RSSI-based algorithms are less sensitive to packet loss rate because the sliding window filter presents the stored readings when the current reading is missing. 2WSD has the poorest performance among RSSI-

based algorithms mainly because of the RSSI variance rather than the packet loss rate.

In the last evaluation, we add an offset to the RSSI measurements sensed from packets of AC5 and AC6. A positive offset means the receiver measures a lower received power while negative offset means the received power is higher. Figure 7(c) shows a result generated under the assumption that packet loss rate is zero and RSSI variance is 20. Manhattan-based algorithms show excellent performances with accuracy either near or equal to 100%. Manhattan distance is very robust to RSSI offsets since it is based on reference matching and does not concern the offsets of RSSI readings. RSSI-based algorithms show performance degradations when the offset is far from zero. The degradation is more severe when the offset is negative. This is expected because we place more test points in rooms than corridor. When offset is negative, more test points in rooms estimate they are near AC5 and AC6. Figure 7(d) shows a result under the assumption that packet loss rate is 20% and RSSI variance is 20, which is more realistic. We observe similar performance, that is, Manhattan-based algorithms are robust to RSSI offsets while this makes RSSI-based algorithms degrade.

In practice, the parameters are more complex. Each node has different variance and for each pair of nodes they have different RSSI offset and packet loss rate. Human activity also greatly affects the above parameters. However, the simulations give us a basic understanding of the expected performance. All six algorithms degrade with larger RSSI variance, which is very hard to overcome. With higher packet loss rate, Manhattan-based algorithms degrade faster than RSSI-based ones. However, we can improve the performance of Manhattan-based algorithm by controlling packets collisions. RSSI offsets have great impact on RSSI-based algorithms and nearly no impact on Manhattan-based ones. Using Manhattan-based algorithms for room separation, we can benefit from this property by assigning different output power to anchors. Then larger areas can be covered with fewer anchors without hardware calibration. MAD-Manhattan performs better than MD-Manhattan and 2WSD-Manhattan mainly because it takes all anchors into account and further neutralizes RSSI variance. In situations where the packet loss rate is uncontrollably high, RSSI-based algorithms can be used. MAD-RSSI shows the best performance among them and it is also the choice in [14].

6. TESTBED AND PERFORMANCE EVALUATION

We implemented a simplified system prototype of INEMO in our department building, as shown in Figure 6. In the deployment phase, room 513 and room 517 are equipped with four anchors each and two anchors are placed in the corridor, all at 2 meters height. Each anchor knows its place (room/corridor) ID and relative coordinates. We select 3×4, 2×3, and 4×3 points in room 517, corridor, and room 513, respectively, for performance testing. The reason room 515 is not selected is that rooms 517 and 515 are divided by a big piece of glass, thus it is not a typical environment in positioning application. Due to limited number of Mica2 nodes at hand, we can only support an evaluation of three

(a) The effect of RSSI fluctuation

(b) The effect of packet loss rate

(c) The effect of RSSI offsets on 0% packet loss rate

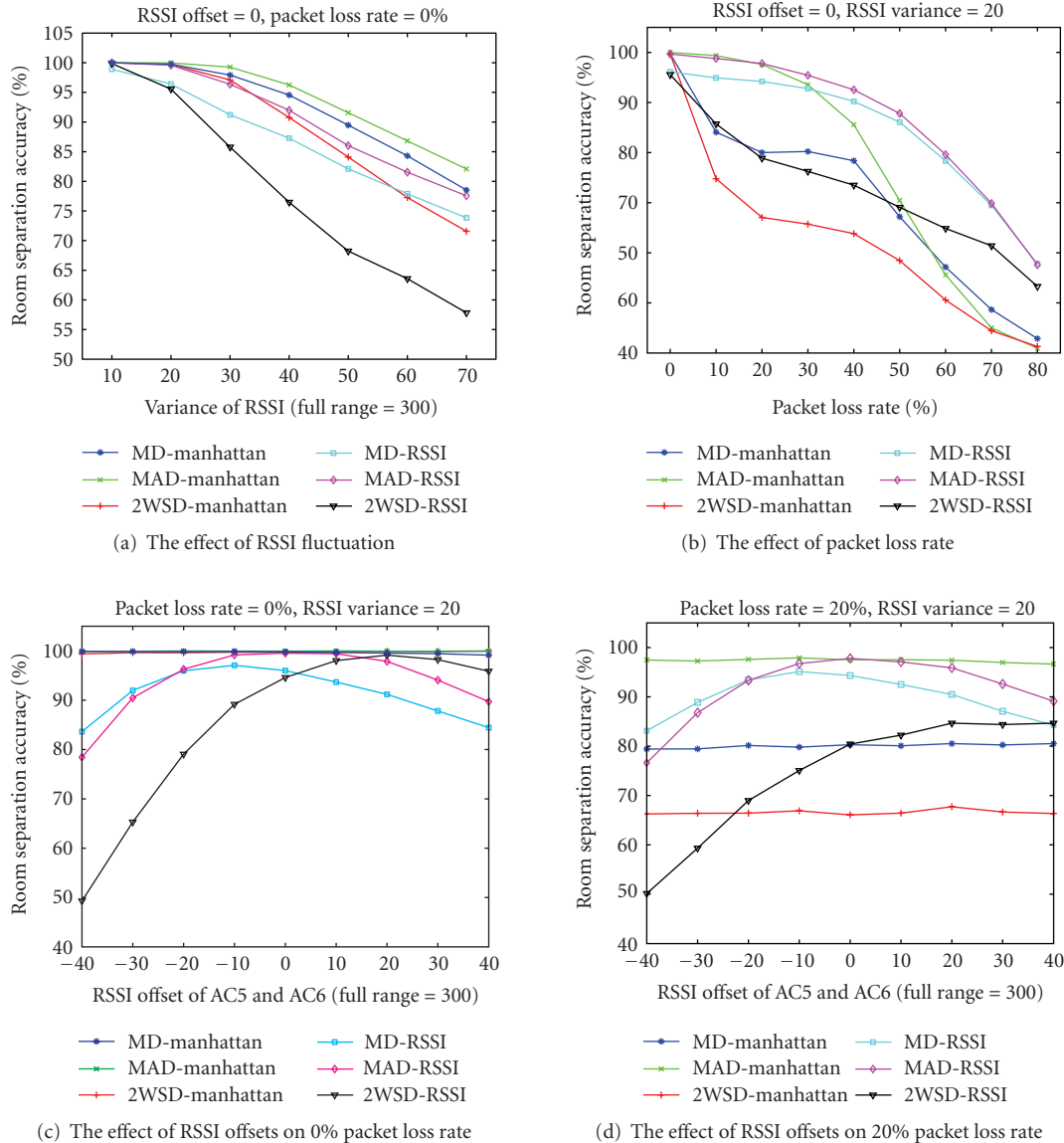(d) The effect of RSSI offsets on 20% packet loss rate

FIGURE 7: Simulation results for room separation.

places (two rooms and one corridor). But consider the limited range of wireless communication, a typical output power of 0 dBm has a communication range of 6 to 12 meters for CC 1000. No matter how large the future deployment scale will be, a positioning request activates only anchors in nearby places (two to three rooms, typically) by tuning to appropriate output power. Our experiment provides a representative case study for room separation.

In nearby anchors monitoring, we set a sliding window filter to keep the latest five periodic instances. Anchors average the valid RSSI readings, for each neighbor (valid means the anchor received a message successfully in that period). If no valid reading exists in five consecutive periods, the anchor assigns a Max_RSSI to the neighbor status. In each test point, the target broadcasts a room separation request every six seconds and about 300 requests are sent. All messages gathered by the target are forwarded to a laptop for offline analysis.

Our cube determination experiment was conducted in room 517, with test points 1 to 12. In each test point, the target broadcasts a cube determination request every six seconds and about 300 requests are sent in total. Note that anchors in neighboring places (e.g., rooms) do not reply to these requests.

6.1. Overall accuracy of room separation

Figure 8 shows the distribution of overall accuracy of room separation in 30 test points. MD-RSSI and 2WSD-RSSI outperform others as they obtain an accuracy of above 90% in 25 test points. MD-Manhattan, MAD-RSSI, and 2WSD-Manhattan achieve an accuracy of above 80% in more than a half of the test points. MAD-Manhattan shows the worst performance, with only one third of test points achieving 80% accuracy.
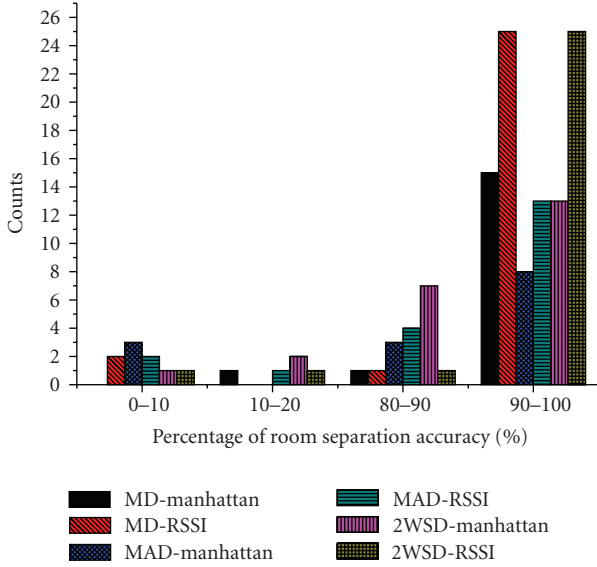
FIGURE 8: Accuracy of different algorithms in room separation.

Next we concentrate on the test points with extremely low accuracy. MD-Manhattan has one point with 17.7%, MAD-Manhattan has 3 point with about 3.0%, and 2WSD-Manhttan has 3 points with about 10%. Correspondingly, MD-RSSI has 2 points with 0%, MAD-RSSI has 2 points with 0% and 1 point with 8%, and 2WSD-RSSI has one 0% and one 14%. Our explanation of the zero accuracy cases is that taking only RSSI as the distance metric may encounter extremely bad performance. In test points 16 and 17, the minimum RSSI readings are measured from AC4, so MD-RSSI would make a decision that target is in room 517. In the same test points, MD-Manhattan gives accuracy of 41.7% and 27.8%. That is, Manhattan distance can effectively neutralize abnormal RSSI readings. In test point 4, we notice that anchor 2 can reply only a few messages to other nodes due to unknown reason, making the average RSSI deteriorate. So MAD-RSSI gives an accuracy of 0%, while MAD-Manhattan gives 64.7% and 2WSD-Manhattan gives 95%. In test point 17, RSSIs from AC7 and AC8 are smaller than AC5 and AC6. 2WSD-RSSI gives an accuracy of 0% while 2WSD-Manhattan gives 18.5%. These results show that Manhattan distance is robust to RSSI offset. If anchors behave abnormally, nearby anchors can sense and adapt to the offset. Abnormal anchor(s) would not affect the Manhattan distances in a strong sense, since target and anchors can counteract offsets in reference matching.

Despite Manhattan distance doing better in extreme cases, our experiment did not show an encouraging result in overall accuracy. We analyzed the message lists in anchors and target and found that packets are lost occasionally. Among test points in room 513, the packet loss rate is 64.3% on average, which means anchors cannot receive target requests or target fails to receive replies. In room 517 and corridor, the packet loss rate is 55.2% and 48.6%, respectively. In other words, anchors and target cannot send and receive messages in a reliable way. Therefore, anchors fail to estimate

the correct neighboring status unexpectedly, which results in corrupted Manhattan distance estimates. These results are compatible with the expected performance derived from simulation. As the packet loss rate increases, Manhattan-based algorithms degrade faster than RSSI-based algorithms. In our room separation process, all anchors, which received the request, contend to send reply in the same wireless channel. The CSMA-based MAC used in TinyOS 1.1.7 cannot handle this situation successfully. We plan to implement advanced MAC protocols, for example, ZMAC [22], S-MAC [23] or even some cross-layer protocols, to enhance communication reliability.

6.2. Overall accuracy of cube determination

This experiment assesses the accuracy of cube determination of INEMO. We collected about 300 sets of RSSI readings of target-anchor and anchor-target in each test point. These readings are used to calculate position estimates and errors offline.

A path loss exponent α of 3.2 is selected empirically for our office environment. Our results show that in the 9 m × 7 m room 517, the mean error of all test points is 127 cm. As depicted in Figure 9(a), we achieve a 50th percentile and 80th percentile positioning accuracy of 1.1 and 2.2 m, respectively. The largest error is below 3 m, about 1/3 of the room length. However, our result of cube determination is derived from raw RSSI readings, with no scale adjustment [16], and calibration. Better accuracy is expected by using hardware calibration and optimizing anchor placement.

6.3. Positioning confidence indicator

For each positioning result, a PCI is also given to infer the confidence of recent estimates. In our experiment, the PCI window size is 6. Figures 9(b)–9(d) illustrate how the positioning error of cube determination changes with time. The proposed PCI can efficiently denote the variation amplitude of the recent positioning error. The sharper the positioning error curve fluctuates, the bigger the corresponding PCI value is, which means that the recent positioning results are not stable. Note that from a users perspective, only the PCI is available and users cannot know the error curve. In our current implementation, the PCI only gives an indication of whether the error is stable or whether the environment is stable enough for positioning. We believe that by using calibration or learning techniques, more precise positioning results can be derived from stable environment. This is part of our future work.

In room separation phase, due to high packet loss rate, we cannot collect enough RSSI information for PCI computation. We take room 517, for example. When the target broadcasts cube determination requests, only 12.3% packets are lost from anchor 1 to 4. When room separation requests are broadcast, 45.0% packets are lost from anchor 1 to 4, due to channel content from anchors of neighboring rooms. These two different packet loss rates prove that our two-tier positioning method can alleviate wireless channel effectively.

(a) Overall accuracy of cube determination

(b) Error and PCI for test point 3

(c) Error and PCI for test point 5

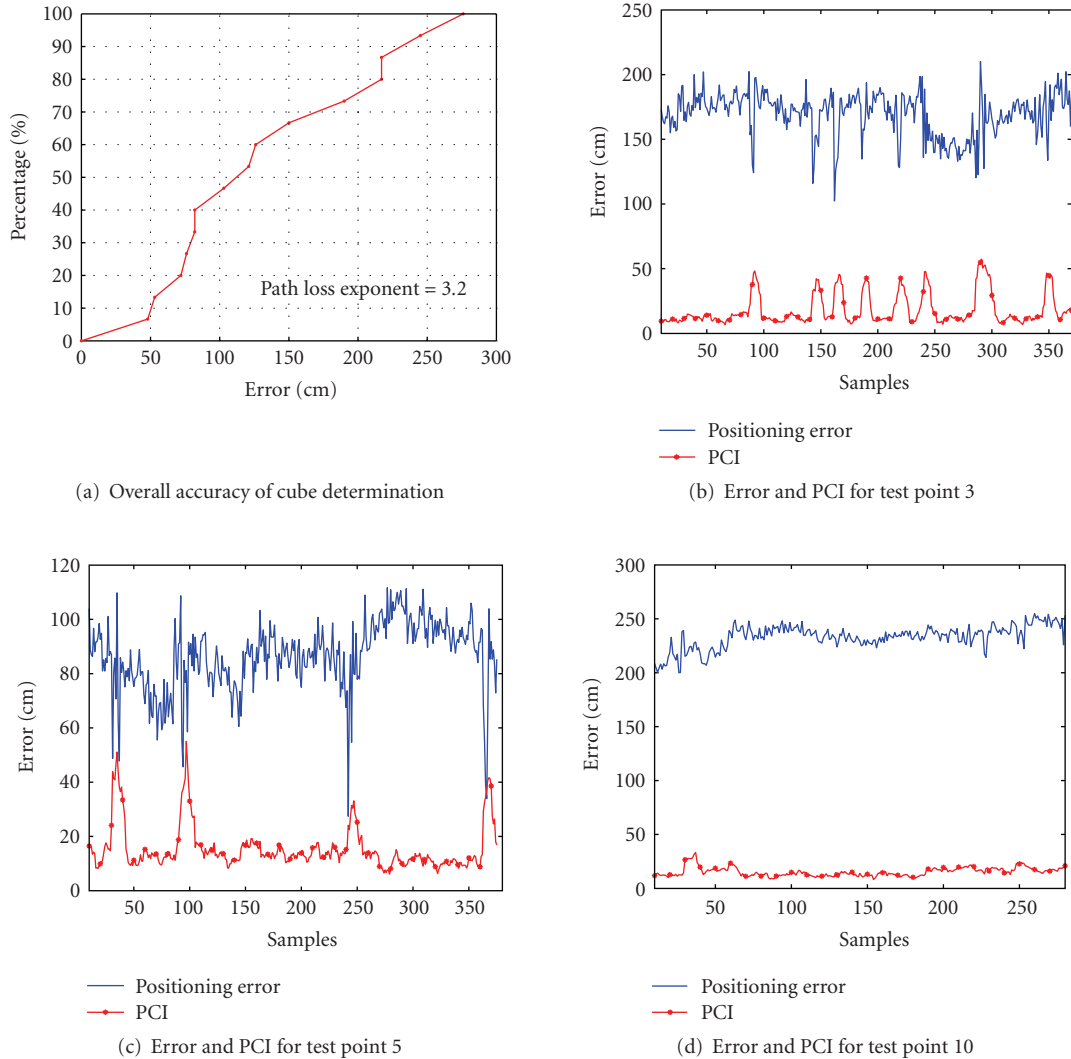(d) Error and PCI for test point 10

FIGURE 9: Accuracy of cube determination.

7. CONCLUSIONS

In this paper, a novel approach is proposed for indoor position determination using RF signal. We utilize the newly developed wireless sensor nodes to construct a distributed network for location service. The two-tier system, which obtains environment dynamics locally without site-survey and signal map precollection, provides services of room separation and cube determination. A reference matching method, with is robust to hardware diversity, is used to support room separation. Then weighted centroid localization is used in cube determination. We reach an accuracy of over 90% in room separation and 80 percentile accuracy of 2.2 m in cube determination, with reasonable confidence indicator inferring the certainty of positioning.

Future work involves testing the approach in other conditions (anchor density and anchor failure), using advanced MAC protocols to reduce packets loss, and getting more precise positioning result in stable environment.

REFERENCES

[1] X. Shen, H. Li, J. Zhao, J. Chen, Z. Wang, and Y. Sun, "Nemo-track: a RF-based robot tracking system in wireless sensor

networks (DEMO)," in *Adjunct Proceedings of the European Conference on Wireless Sensor Networks (EWSN '06)*, Zurich, Switzerland, February 2006.

[2] "Crossbow," http://www.xbow.com.

[3] X. Shen, H. Li, J. Zhao, Z. Wang, and Y. Sun, "Distributed group management scheme for target tracking in sensor networks," *Chinese Journal of Scientific Instrument*, vol. 28, no. 6, pp. 966–972, 2007.

[4] A. Hopper, V. Falcao, and G. Gibbons, "The active badge location system," *ACM Transactions on Information Systems*, vol. 10, no. 1, pp. 91–102, 1992.

[5] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan, "The cricket location supporting system," in *Proceedings of the 6th Annual ACM International Conference on Mobile Computing and Networking*, pp. 32–43, Boston, Mass, USA, August 2000.

[6] P. Bahl and V. N. Padmanabhan, "RADAR: An in-building RF-based user location and tracking system," in *Proceedings of the 19th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '00)*, vol. 2, pp. 775–784, Tel Aviv, Israel, March 2000.

[7] M. Youssef and A. Agrawala, "Small-scale compensation for WLAN location determination systems," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '03)*, vol. 3, pp. 1974–1978, March 2003.

[8] D. Madigan, E. Elnahrawy, R. P. Martin, W.-H. Ju, P. Krishnan, and A. S. Krishnakumar, "Bayesian indoor positioning systems," in *Proceedings of the IEEE INFOCOM '05*, vol. 2, pp. 1217–1227, Miami, Fla, USA, March 2005.

[9] H. Lim, L. C. Kung, J. C. Hou, and H. Luo, "Zero-configuration, robust indoor localization: theory and experimentation," in *Proceedings of the IEEE INFOCOM '06*, pp. 1–12, Barcelona, Spain, April 2006.

[10] A. M. Ladd, K. E. Bekris, A. Rudys, G. Marceau, L. E. Kavraki, and D. S. Wallach, "Robotics-based location sensing using wireless Ethernet," in *Proceedings of the Annual International Conference on Mobile Computing and Networking (MOBICOM '02)*, pp. 227–238, Atlanta, Ga, USA, September 2002.

[11] S. Ganu, A. S. Krishnakumar, and P. Krishnan, "Infrastructure-based location estimation in WLAN networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '04)*, vol. 1, pp. 465–470, Atlanta, Ga, USA, March 2004.

[12] Y. Ji, S. Biaz, S. Pandey, and P. Agrawal, "ARIADNE: A dynamic indoor signal map construction and localization system," in *Proceedings of the 4th International Conference on Mobile Systems, Applications and Services (MobiSys '06)*, vol. 2006, pp. 151–164, Uppsala, Sweden, June 2006.

[13] K. Lorincz and M. Welsh, "A robust, decentralized approach to RF-based location tracking," in *Proceedings of the 1st International Workshop on Location- and Context- (Awareness '05)*, vol. 3479, pp. 63–82, Oberpfaffenhofen, Germany, May 2005.

[14] Y. W. Lee, E. Stuntebeck, and S. C. Miller, "MERIT: mesh of RF sensors for indoor tracking," in *Proceedings of the IEEE SECON*, pp. 545–554, Reston, Va, USA, September 2006.

[15] K. Yedavalli, B. Krishnamachari, S. Ravulat, and B. Srinivasan, "Ecolocation: A sequence based technique for RF localization in wireless sensor networks," in *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks (IPSN '05)*, vol. 2005, pp. 285–292, Los Angeles, Calif, USA, April 2005.

[16] F. Reichenbach and D. Timmermann, "Indoor localization with low complexity in wireless sensor networks," in *Proceedings of the IEEE (INDIN '06)*, pp. 1018–1023, 2006.

[17] "TinyOs," http://www.tinyos.net.

[18] "ChipCon," http://www.chipcon.com.

[19] H. Hashemi, "The indoor rdaio propagation channel," *Proceedings of the IEEE*, vol. 81, pp. 943–968, 1993.

[20] J. Ma, Q. Chen, D. Zhang, and L. M. Ni, "An empirical study of signal strength in sensor networks using MICA2 nodes," Tech. Rep., Hong Kong University of Science and Technology, 2006.

[21] X. Shen, Z. Wang, P. Jiang, R. Lin, and Y. Sun, "Connectivity and RSSI based localization scheme for wireless sensor networks," in *Proceedings of the International Conference on Intelligent Computing (ICIC '05)*, vol. 3645, pt. 2, pp. 578–587, Hefei, China, August 2005.

[22] I. Rhee, A. Warrier, M. Aia, and J. Min, "Z-MAC: hybrid MAC for wireless sensor networks," in *Proceedings of the ACM SenSys '05*, pp. 90–101, 2005.

[23] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient MAC protocol for wireless sensor networks," in *Proceedings of the IEEE (INFOCOM '02)*, vol. 3, pp. 1567–1576, New York, NY, USA, June 2002.