INTERNATIONAL JOURNAL OF INTELLIGENT CONTROL AND SYSTEMS
VOL. 12, NO.3, SEPTEMBER 2007, 272-281

An Improved Wireless Sensor Location Selection Algorithm for First Responders

MengChu ZHOU, Wei SU, Zaihan JIANG and Wenqi (Wendy) GUO

***Abstract*- This paper introduces an improved approach for selecting the location of Radio Frequency (RF) sensors used for detecting objects in a building. Given the building map and materials, a number of moving or static objects carrying RF tags are to be detected in a building in an emergency situation. Liu *et al.* (2005) presented two algorithms that can be used to compute the number and deployment locations of sensors and detect objects using the data from the deployed sensors. However, their first algorithm only accounts for absolute deployment cost. It can be greatly enhanced by using relative deployment costs. By comparing the results of the simulation of the existing and improved algorithms using MatLab^{TM}/C++, this work concludes that when determining the locations to deploy sensors, one must take into account the locations relative to the locations of sensors already deployed. The results can be used by first responders such as police and firemen in detecting their indoor crews in the event of fire and other emergency situations.**

***Index Terms*—Wireless Sensor Network, Sensor Deployment, Sensor Location Selection, Intelligent Buildings**

1. INTRODUCTION

Wireless sensor networks can be integrated in critical emergency and alarm systems. The existing infrastructure provides only minimal communication capability for supplying information about the nature or extent of a disaster *in situ*. As a result, first responders typically enter emergency situations with little real-time information about the site. Should they become trapped, only a haphazard means of rescue are available to them. The recent progress in sensing technologies and wireless networks makes it possible to provide real time feedback from disaster sites.

The response time for a first responder is crucial given an emergency or disastrous event. Sensors in a network have the ability to provide an emergency control system with real-time monitoring, 3D building visualization, hot spots or structure failures, and tracking victims or personnel in danger. As indicated by Liu *et al*. (2005), an advanced first responder system must have the following capabilities:

a. To help crew members to identify their own and others' locations;
b. To locate potential hazards, victims, or sources of the emergency;
c. To identify and rescue trapped personnel; and
d. To identify the location of certain objects such as police dogs that are used to find the location of threat, e.g., a bomb.

Many existing approaches either take a long time to calibrate or learn (Hightower *et al*., 2000; Smailagic and Kogan, 2002; Ladd, *et al*., 2002; Youssef and Agrawala, 2004), or have a low resolution (MeshNetworks, 2006), thereby benefiting first responders insignificantly.

Many researchers have proposed user position estimation methods based on the strength of the radio signals received from multiple wireless access points (Bahl and Padmanabhan, 2000; Ray, *et al*., 2003; Gwon *et al*., 2004; Prasithsangaree, *et al*, 2002; Krishnan, *et al.*, 2004; Smailagic and Kogan, 2002; Manapure, *et al.*, 2004). Sensor placement problem is related to the access point placement in constructing a wireless local area network in a building. Nagy and Farkas (2000) use a genetic algorithm to optimize the placement of access points in terms of signal coverage inside a building. Hills (2001) proposes a method to deploy a 3-dimensional large-scale indoor IEEE 802.11b WLAN such that a) no coverage gap exists and b) no overlaps between access points are allowed to operate on the same channel. Hills, *et al*. (2004) present an evaluation algorithm for estimating the signal strength and coverage areas of relocated access points. Chen and Kobayashi (2002) present a linear and multiple regression method to estimate the signal strength model of an indoor wireless access points by experimental data. They have analyzed the relationship between signal strength error and localization error.

Ray *et al*. (2003) deal with a location detection problem via a novel framework based on the theory of identifying codes. They allow sensor coverage areas to overlap in such a way that each resolvable position is covered by a unique set of sensors. Hence, determining a sensor-placement with a minimum number of sensors is equivalent to constructing an optimal identifying code. Guo *et al*. (2006) proposes an adaptive sensor placement method based on the real-time estimates of object distribution via a recursive distributed Expectation-Maximization algorithm. These optimization frameworks are inherently probabilistic due to the uncertainty associated with sensor detections.

It is well recognized that in crisis situations emergency responders have to acquire timely all information critical to their task performance. In practice, slow accessing and sharing the relevant information among collaborating individuals or organizations leads to unnecessary,

Manuscript received July 9, 2006; revised May 8, 2007. This work was partially funded by DoD via Booz Allen Hamilton under ETOSS #W15P7T-06-D-A401 and SubContract 86190NBS21.

M. C. Zhou is with New Jersey Institute of Technology, ECE Dept., Newark, NJ 07102, USA and School of Electro-Mechanical Engineering, Xidian University, Xi'An, Shaanxi 710071, China (e-mail: zhou@njit.edu). W. Su is with U.S. Army RDECOM, Ft Monmouth, NJ 07703, USA (e-mail: Wei.Su@us.army.mil). Z. Jiang is now with Free-Scale Semiconductor Inc., Lake Zurich, IL 60047 while the work was done at New Jersey Institute of Technology (zaihanjiang@yahoo.com). W. Guo is with New Jersey Institute of Technology, ECE, Newark, NJ 07102, USA (wenqiguo@gmail.com).

preventable errors and delays, thereby causing more damage to the situations. Motivated by it, Netten *et al*. (2006) propose a Task-Adaptive Information Distribution method. It consists of a system for adaptive information distribution that distributes relevant information to collaborating emergency responders and of an adaptive workflow system used to obtain knowledge of tasks and work processes. Their method is shown effective by using data from a real incident.

Effective knowledge management is important in emergency information systems. Murphy and Jennex (2006) describe and analyze two knowledge management systems that were utilized during Hurricane Katrina response. The systems were developed by both federal agencies and grass root efforts without the support or mandate of government programs. These programs were able to share data and interact in life saving capacities, transcending traditional geo-political boundaries. In other words, emergency information systems are enhanced by incorporating knowledge management tools and concepts.

Ramaswamy *et al*. (2006) presents a state-of-the-art review of existing emergency response systems. They investigate the principles, theories, and practices from four diverse, yet related, fields of knowledge with respect to information representation and decision support capability requirements for emergency planning and response systems are investigated. Their integration enables the cooperation between constituent agencies (e.g., fire, police and medical) and surrounding municipalities that operate using assorted decision support protocols, system architectures, networking strategies and along different levels of data security needs. A service architectural framework is proposed to provide an integrated platform of knowledge and support affordable integration for municipalities of all sizes. A prototype web service based implementation is presented.

Chen and Dahanayake (2007) indicate that crisis response is an information intensive process, producing and consuming large quantities of information from and for different relief/response organizations. The traditional centralized system design principle used to address inter-organizational information retrieval over boundaries proves insufficient. It lacks flexibility and adaptability to deal with dynamically changing information needs caused by the unpredictable nature of a crisis. Realizing such limitation, they propose a new service architecture for information seeking and retrieval, which allows one to seek and retrieve role-based, situation-aware information in the context of a crisis situation.

There are two main design concerns when wireless sensors are deployed in a sensor network for buildings. The first one is the sensor location selection problem and the second one is the detection problem. Liu *et al*. (2005) pioneered in the study of both problems by proposing effective algorithms to solve them. They can find the best locations for sensor deployment, and establish the effective rules for target detection. Several real world experiments prove that they can achieve a high level of detection accuracy.

However, their first sensor location selection algorithm only accounts for absolute deployment cost. One can greatly enhance it by using relative deployment costs. Section 2 presents problem statement and necessary assumptions and summarizes the algorithm proposed by Liu *et al*. (2005). Section 3 presents the improved algorithm. Section 4 reports the performance of the existing and proposed algorithms for a large example. Section 5 presents the conclusion.

2. PROBLEM STATEMENT

This work focuses on the following sensor location problem in a building. The following assumptions are made (Liu *et al*., 2005):

a) A computer layout of the building map is given.
b) Normal materials, e.g., cement, glass, brick, and metal, are used in the building;
c) A zone is defined as an area in a building. It is a finite and continuous space. Zones in a building represent non-overlapping spaces. In a detection zone, each point has the same detection value. That is, distinguishing between different points of a given zone is not important for the system users. Examples of typical zones could be rooms, offices, exit areas and elevators. Each neighboring zone is separated by a wall or partition of some sort. We assume that zone dimensions are greater than 10’ (length) by 10’ (width). The dimension of zone height is not important as the tag objects move or remain on the zone floor.
d) Tags are RF transmitters assigned to a set of objects somewhere in a building. Each one is independent and unique.
e) Sensor deployment effort weight per location is given. The deployment effort weight can be measured by different metrics such as the time to reach the deployment location and the degree of location accessibility. It is much faster and easier to deploy a sensor in a location outside in front of the entrance door than a location on the ninth floor of the building.

Given a set of sensors, we need to assert a set of rules that assign a zone to every combination of signal strength from the sensors. One needs to find only which zone the tag is in, and not the specific coordinates of the tag.

In dealing with the location problem, we need to have the coverage areas overlap so that each position is covered by a unique set of sensors. Meanwhile, we need to minimize the deployment cost and number of sensors. Two problems are of interest in designing this system.

Problem 1: Location Problem

The goal of this problem is to find the sensor placement locations that optimize several objectives. The objectives include high accuracy of detection, minimum number of sensors, minimum deployment time, and covering all the zones.

Problem 2: Detection Problem

Given a set of sensors inside and outside the building, establish a set of rules that assigns a zone to every possible combination of received signal strength from the sensors. In other words, one needs to determine which zone a tag is located in and not the exact coordinates of the tag.

The location and detection algorithms address these problems. We declare a list of symbols to use in the algorithms.

C : Signal score binary threshold matrix of dimension $m \times n$ where m is the number of locations and n is the number of zones;

D : Tag zone-sensor location signal score matrix of dimension $m \times n$;

E : Signal strength range impact amplification matrix of dimension $m \times p$ where p is the number of signal strength levels;

$E(SS_{ij})$: Expected value of signal strength for the sensor located in location l_i when the tag is in zone z_j

G : Conflict resolution matrix of dimension $n \times n$;

$f_j(x)$: Conditional probability distribution of receiving sensor network state vector x given that the tag is present in zone z_j

$g_x(z_j)$: Conditional probability distribution of the tag presence in zone z_j given that the received sensor network state is x

$h(z_j)$: Probability distribution of the tag presence in zone z_j

$L = \{l_1, l_2, \cdots, l_m\}$: Set of candidate locations

n_{oj} : Number of observations that their signal strengths received from sensor in location o fall in the $s_{oj}th$ interval

$O = \{o_1, o_2, \cdots o_k\}$: Set of selected locations

s_r State of the sensor deployed at location $o_r \in O$

w_i : Cost (time) of reaching location l_i from outside the building

$x = \{s_1, s_2, \cdots, s_k\}$: Sensor network state

X : Set of possible states of the sensory network

$y_1, y_2, \cdots, y_a$: SS range break point values

$Z = \{z_1, z_2, \cdots, z_n\}$: Set of zones

α_{ij} : Lower bound of the estimation interval

β_{ij} : Upper bound of the estimation interval

δ_{ij} : Total reduction for zone z_j and location l_i

The following objectives must be met by the location algorithm.

a. We must cover the zones entirely. In other words, for a given sensor location, the sensor must be able to get a signal strength greater than zero from a tag in the zone.

b. We reach the maximum conflict resolution. In selecting locations for sensors, minimizing conflicts maximizes detection resolution.

c. We minimize the deployment time of the sensors. It is easy to cover the first two objectives by placing a sensor at every possible location, but it is expensive and time consuming. We then have to take into account cost or time (w_i) of a location (l_i).

We will refer to W as either deployment cost or weight throughout the paper. There are no standard units of measure used to describe W for each location. We had rough estimate or approximation of these weights. The actual numeric value of the weights do not factor into what we use to determine the locations to deploy, it is only used in comparison with other values.

Location Algorithm (Liu *et al.,* 2005)

1. Generate *tag zone-sensor location signal score matrix* $D = [d_{ij}]_{mxn}$ where d_{ij} is the signal score when the tag is in zone z_j and sensor is in location l_i. We assign d_{ij} value according to the following rules:

$$d_{ij} = \begin{cases} 0 & \alpha_{ij} = \beta_{ij} = 0 \\ 1 & \alpha_{ij} = 0, \beta_{ij} = 10 \\ t+1 & \alpha_{ij} = y_t, \beta_{ij} = y_{t+1},\ 1 \le t \le a-1 \\ 8 & \alpha_{ij} = 70, \beta_{ij} = 80 \end{cases}.$$

Define *signal score binary threshold matrix* $C = [c_{ij}]_{mxn}$ where $c_{ij} = \begin{cases} 1 & d_{ij} \ge 0 \\ 0 & d_{ij} = 0 \end{cases}$.

2. Generate *SS range impact amplification matrix* $E = [e_{ik}]_{mxp}$ where

$e_{i1} = \{j : (d_{ij} = 1) \text{ or } (d_{ij} = 2)\}$;

$e_{ik} = \{j : (d_{ij} = k-1) \text{ or } (d_{ij} = k) \text{or } (d_{ij} = k+1)\}$

when $1 \le k \le p-1$;

and $e_{ip} = \{j : (d_{ij} = p-1) \text{ or } (d_{ij} = p)\}$.

3. Generate *conflict resolution matrix* $G = [g_{ij}]_{nxn}$ where

$$g_{ij} = \begin{cases} \{1, 2, \cdots, m\} - \{k : \exists q \mid \{i, j\} \subseteq e_{kq}\} & i < j \\ g_{ji} & i > j \\ \text{Undefined} & i = j \end{cases}$$

4. Let $H = L$. Use the following iterative procedure to select the sensor locations.

4.1. Stop and go to step 5 if all elements of G are either empty or undefined.

4.2. For every $l \in H$ calculate:

$$R(l) = \frac{1}{w_l}\left(\sum_{(i<j)\wedge(|g_{ij}|\ge 1)} \frac{Y_{ij}(l)}{|g_{ij}|}\right)$$, where $Y_{ij}(l) = 1$ if $l \in g_{ij}$, and $Y_{ij}(l) = 0$ otherwise.

4.3. Select $l \in H$ with maximum *R(l)*. If several *l's* satisfy the maximum criterion for *R(l)* then select the one with maximum value of $\sum_{j=1}^{n} c_{lj}$. If several *l's* are still candidates, then select one arbitrarily. Let l_0 be the selected location.

4.3.1. Remove l_0 from *H*.

4.3.2. For $1 \le j \le n$ if $c_{l_0 j} = 1$, set $c_{ij} = 0$ for $1 \le i \le m$.

4.3.3. Let $g_{ij} = \{\}$ if $Y_{ij}(l) = 1$.

4.3.4. Go to step 4.1.

5. Perform the following procedure.

5.1. Stop and go to step 6 if *H* is empty or $\sum_{i=1}^{m}\sum_{j=1}^{n} c_{lj} = 0$.

5.2. For every $l \in H$ calculate $R(l) = \frac{1}{w_l}(\sum_{j=1}^{n} c_{lj})$.

5.3. Select $l \in H$ with maximum *R(l).* If several *l's* satisfy the maximum criterion for *R(l)*, open ties arbitrarily. Let l_0 be the selected location.

5.3.1. Remove l_0 from *H*.

5.3.2. For $1 \le j \le n$ if $c_{l_0 j} = 1$, set $c_{ij} = 0$ for $1 \le i \le m$.

5.3.4. Go to step 5.1.

6. If any of the selected sensor locations in the latest iteration of steps 4 and 5 do not exist in O , add them to this set.

6.1. If no sensor location is added to set *O* then go to step 7, otherwise deploy all the added sensors and go to the next step.

6.2. For every zone we perform the learning by putting a tag in that zone and recording the resulting signal strengths from the sensors. Suppose that the tag is in zone *j* and we have collected *N* observations (of signal strength) from the sensor placed in location *o* (location *o* is selected from the newly added locations to *O* and no calculation is done for the old locations). We estimate $f_{oj}(s_{oj}) = \frac{n_{oj}}{N}$ for $0 \le s_{oj} \le a+1$. Here n_{oj} is the number of observations that their signal strengths received from a sensor in location o fall in the $s_{oj}th$ interval.

6.3. For all the selected locations and zones, calibrate their corresponding scores in matrix *D* based on the learning data.

6.3.1. For the sensor in the newly added location *o* and the tag in zone *j*, calculate the average signal strength by $E(S_{oj}) = \sum_{s_{oj}=1}^{a+1} s_{oj}.f_{oj}(s_{oj})$. If $d_{ij} - 1 \le E(S_{oj}) \le d_{ij} + 1$, where *i* is the row of matrix *D* corresponding to the sensor in location *o*, then leave d_{ij} as is; otherwise set

$$d_{ij} = \begin{cases} INT(E(S_{oj})) & \text{if } E(S_{oj}) - INT(E(S_{oj})) \le 0.5 \\ INT(E(S_{oj})) + 1 & \text{if } E(S_{oj}) - INT(E(S_{oj})) > 0.5 \end{cases},$$

where *INT*(•) is the integer function.

6.3.2. If no element of *D* changes in step 6.3.1, then proceed to step 7. Otherwise go to step 6.4.

6.4. Based on the new matrix *D*, update $c_{ij} = \begin{cases} 1 & d_{ij} \ge 0 \\ 0 & d_{ij} = 0 \end{cases}$. If all the locations are selected (that is $O = L$) go to step 7; otherwise go to step 2.

7. Calculate $f_j(x) = \prod_{o} f_{oj}(s_{oj})$ for all zones and selected locations and proceed to the detection algorithm.

End of location algorithm.

Liu *et al.* (2005) have in detail discussed how their location algorithm satisfies the mentioned design objectives.

3. IMPROVED ALGORITHM BASED ON RELATIVE WEIGHT MATRIX

3.1 Relative Weights and Algorithm Modification

The idea of this work is to modify the sensor location algorithm to address the issue of deploying to locations with proximity to one another. We propose to have a relative weight matrix RW in addition to the weight vector *W*. RW_{ij} gives the distance in terms of deployment cost from location *i* to location *j*. Through the first iteration of step 4 or in some trivial cases, the first iteration of step 5, the selected set of locations to deploy sensors, *O*, is empty. When calculating the *R*(*l*) value, we use the regular *W* for each location. Then, a location would be selected for *O.* In each subsequent iteration of steps 4 and 5, we use the modified *R*(*l*) values, which use *RW* instead of *W.* The following are steps 4 and 5 of the location algorithm rewritten to the modified weight scheme.

4. Let $H = L$. Use the following iterative procedure to select the sensor locations.

4.1. Stop and go to step 5 if all elements of G are either empty or undefined.

4.2. If O is empty go to 4.2.1. Otherwise go to 4.2.2

4.2.1. For every $l \in H$ calculate:

$$R(l) = \frac{1}{w_l}\left(\sum_{(i<j)\wedge(|g_{ij}|\ge 1)} \frac{Y_{ij}(l)}{|g_{ij}|}\right) ,$$ where $Y_{ij}(l) = 1$ if $l \in g_{ij}$, and $Y_{ij}(l) = 0$ otherwise. Go to 4.3.

4.2.2 For every $l \in H$ calculate:

Find lowest RW_{ol} for every o in O. Calculate:

$$R(l) = \frac{1}{RW_{ol}}\left(\sum_{(i<j)\wedge(|g_{ij}|\ge 1)} \frac{Y_{ij}(l)}{|g_{ij}|}\right) ,$$ where $Y_{ij}(l) = 1$ if $l \in g_{ij}$, and $Y_{ij}(l) = 0$ otherwise.

4.3. Select $l \in H$ with maximum *R*(*l*). If several *l*'s satisfy the maximum criterion for *R*(*l*) then select the one with maximum value of $\sum_{j=1}^{n} c_{lj}$. If several *l*'s are stil candidates, then select one arbitrarily. Let l_0 be the selected location.

4.3.1. Remove l_0 from H.

4.3.2. For $1 \le j \le n$ if $c_{l_0 j} = 1$, set $c_{ij} = 0$ for $1 \le i \le m$.

4.3.3. Let $g_{ij} = \{\}$ if $Y_{ij}(l) = 1$.

4.3.4. Go to step 4.1.

5. Perform the following procedure.

5.1. Stop and go to step 6 if H is empty or $\sum_{i=1}^{m}\sum_{j=1}^{n} c_{lj} = 0$.

5.2. If O is empty, go to 5.2.1. Otherwise go to 5.2.2.

5.2.1. For every $l \in H$ calculate $R(l) = \frac{1}{w_l}(\sum_{j=1}^{n} c_{lj})$.

5.2.2. For every $l \in H$

Find lowest RW_{ol} for every o in O. Calculate:

$$R(l) = \frac{1}{RW_{ol}}(\sum_{j=1}^{n} c_{lj})$$

5.3. Select $l \in H$ with maximum $R(l)$. If several l's satisfy the maximum criterion for $R(l)$, open ties arbitrarily. Let l_0 be the selected location.

5.3.1. Remove l_0 from H.

5.3.2. For $1 \le j \le n$ if $c_{l_0 j} = 1$, set $c_{ij} = 0$ for $1 \le i \le m$.

5.3.3. Go to step 5.1.

3.2 Algorithm Implementation

The entire algorithm is coded using Visual C++ and MatLab. The program read four files:

d.txt - (The initial *D* matrix determined by the Barrier reduction and interval estimation tables.)

dm.txt - (The "measured" values for the *D* matrix from the signal simulations from MatLab.)

w.txt - (The weights of the locations.)

rw.txt (The relative weights of the locations.)

There were two output files for the algorithm:

r_scores.txt - (The *R*(*l*) scores from the algorithm in a comma delimited file. This file was exported to Excel for formatting purposes.)

cost.txt - (The shortest total cost of deployment of the locations selected from the algorithm.)

The program inputs *D*, *W*, and *RW* matrices and then asks whether to run the on the regular weight algorithm or to run on the relative weight algorithm. The algorithm runs until step 6.2. The learning tag processing part is done through simulation in MatLab, to be discussed next. The program outputs the locations to deploy sensors, and that data is entered into the MatLab code in a binary vector. MatLab processes all the data for step 6 and outputs the estimated interval "measurements" from signal simulation. This data is then copied into **dm.txt**. The prompt from the C++ program asks to press a key once the dm.txt file has been updated according to the MatLab simulation. After reading **dm.txt**, the program compares the data of D_m after rounding the interval numbers to that of the *D* matrix. If there is a change, the algorithm iterates once more from step 2, and repeats the same process. If new locations are deployed, we follow the same procedure.

Step 7 was not included in the C++ program, because the implication of step 7 is used mainly for the detection algorithm (Liu *et al.* 2005). In this case, we are only examining the speedup of the sensor deployment. This speedup was measured in total cost of deployment in comparing the two versions of the algorithm.

4. PERFORMANCE COMPARISON VIA A CASE STUDY

Because of hardware, time, and map constraints, we are unable to perform the physical tag learning in Step 6.2 of the location algorithm. In order to supplant this, we simulated the tag signals using MatLab. First we considered the mean signal value for a zone based on our interval estimations in the *D* table. For instance, if a signal score for a zone *x* and a location *y* was 5, that would give an estimated dB range of [40, 50]. The mean of that would be 45. We assume a 95% confidence interval using a normal distribution that any given signal for a tag in zone *x* for location *y* would fall into the range [25, 65]. For every signal with range [*a*, *b*], the confidence interval would be [a - 15, b + 15]. This gives us a standard deviation of 10 dB. To produce a learning tag signal matrix, we did the following:

Produce a $\boldsymbol{\mu}$ matrix corresponding to *D* for the estimated signal strengths. Table 1 gives us a translation of *D* matrix into $\boldsymbol{\mu}$ matrix.

Table 1. Translation from *D* matrix into $\boldsymbol{\mu}$ matrix

D score	μ	Zone/Location Distance
0	-10	> 75 feet
0	0	<= 75 feet
1	5	--
2	15	--
3	25	--
4	35	--
5	45	--
6	55	--
7	65	--
8	75	--

Use *SO* as a binary vector corresponding to the selected sensors from steps 4 and 5. For example, in a 5 location set, if *O* = {1 4 5} then *SO* = [1 0 0 1 1]. Diagonalize the vector *SO*to obtain a 5×5 matrix

$$\boldsymbol{SO}= \begin{pmatrix} 1&0&0&0&0 \\ 0&0&0&0&0 \\ 0&0&0&0&0 \\ 0&0&0&1&0 \\ 0&0&0&0&1 \end{pmatrix}$$

Produce an $m \times n$ random z-score matrix $\boldsymbol{Z} = R_z \bullet SO$ where R_z is an $m \times n$ matrix whose elements are random samples with mean 0 and variance 1. Using the model $Z = (\boldsymbol{\mu} - \boldsymbol{x}_i)/\sigma$, where $\boldsymbol{\mu}$ is a matrix of means, $\boldsymbol{x}_i$ is a matrix of "measured" signals, σ is the standard deviation, solve for $\boldsymbol{x}_i$ to obtain:

$$\boldsymbol{x}_i = \boldsymbol{\mu} - \sigma Z$$

We produced the $\boldsymbol{Z}$ matrix N times to give us N total signal readings of the learning tag. We add all the "measured" value matrices $\boldsymbol{x}_i$ and divide by N to obtain a matrix X of the average recorded signal strengths between each tag and every selected location.

When X is read into the program, the average signal readings translate into an interval estimation matrix D_m. We then compare D_m to D for step 6.3.2 in the location algorithm.

As an example, we generated a two floor building map with 10 possible sensor locations and 16 zones. The map is shown in Figure 1. The D matrix (Table 2) was found using the barrier reduction (Table 3) and interval estimation table (Table 4) corresponding to the distances (Table 5) from each sensor location to the center of each zone. The W (Table 6) and RW (Table 7) matrices were also measured from the maps. These inputs were plugged into the program using both relative and absolute weights. The MatLab simulations were conducted according to the output of the program, and the simulations were input into the program. The MatLab simulations for interval estimations are given below in Tables 8-9. The results for both cases are given in Table 10.

In this example, the relative weights adjusted algorithm requires one to deploy to 5 locations only, while the regular weight-based algorithm requires 6 locations. Summing up the regular weights from outside of the building, the proposed algorithm has a weighted cost of 13, while the absolute algorithm has a weighted cost of 12. Summing up the relative weights from location to location, starting from outside of the building to the closest location, the proposed algorithm has a cost of 19, while the existing algorithm has a cost of 22.

In other words, if one is to deploy to each location individually with respect to their position from outside of the building, the existing algorithm has a slightly lower deployment cost for this example. However, deploying from location to location has a much lower cost using the proposed algorithm than using the existing one. Considering that the proposed one requires one to deploy to only 5 locations, it is clear that the modified algorithm brings significant advantages over the existing one.

Given a larger building, the proposed relative weight-based method can be expected to be much more favored. From our calculations with the data we have, we clearly see that the proposed algorithm has a more time-efficient way of deploying sensors in one area. The existing one in this case has a slightly lower cost when dealing with deploying to each location, one person per location. In real time applications, it is certainly more plausible to believe that few people are dispatched to deploy sensors for a limited number of units instead one person per location. It is more sensible to think that to be most efficient, especially in larger buildings, one wants to deploy sensors in areas based on their proximity to other important sensors. For instance, with our example, if the building is one area of a much larger campus, one needs many people to deploy all sensors in it if the existing algorithm is used. If the total number of sensors for the campus is 60, 60 people are needed to deploy sensors. This is not cost-efficient for certainty. In real life, it is much more likely that only a couple of people per building would be expected to deploy sensors as the consequence of using the improved algorithm.

5. Conclusion and Future Research

Motivated by the work by Liu *et al*. (2005), this paper presents a modified algorithm based on relative weighting matrix for a sensor deployment problem in a building. Although this work uses a two-story building example, the modified algorithm is shown to be more efficient than the existing one (Liu *et al*., 2005). Further testing with larger amounts of data is expected to determine that the proposed algorithm should outperform significantly the existing one in terms of several optimization criteria.

As indicated in (Liu *et al*., 2005), there are several future research directions. In particular, one direction is to reduce the tag learning time. One idea is to advance the simulation of the learning to accurately approximate the actual learning time. This is a difficult task, but to simulate the learning can cut out a large part of time in the algorithm and provide for a much faster response time. Some issues to consider with that include the approximation of the signal strength from the tag in the farthest and closest areas of the zone. This gives the worst case and best case scenarios for signal strength for a tag-sensor relationship. The application of wireless sensing technologies, networking solutions, and developed algorithms to large-scale buildings and infrastructure requires much more work from researchers and engineers.

Acknowledgement

We would acknowledge the significant help from a number of senior students including M. Thanasides, R. Rana, M. Szydlowski, and M. Faloyin, ECE Dept., New Jersey Institute of Technology, Newark, NJ, USA, in designing and implementing the related algorithms and their experimental validation. We would also thank Prof. H. Darabi and Dr. H. Liu of University of Illinois at Chicago for their kind assistance.

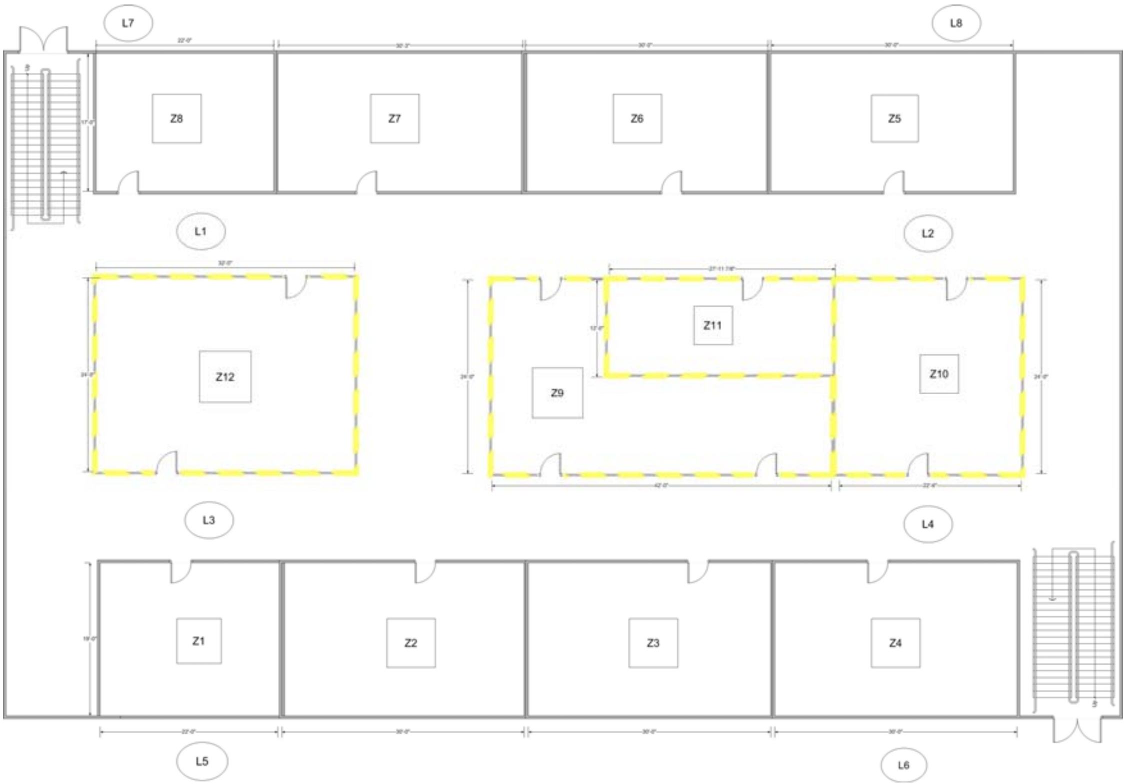


Fig. 1. Building Maps Level 1 & Level 2

Table 2. D-Matrix:

ZONE/LOC	Z1	Z2	Z3	Z4	Z5	Z6	Z7	Z8	Z9	Z10	Z11	Z12	Z13	Z14	Z15	Z16
L1	0	0	0	0	0	1	4	5	3	0	2	6	2	0	3	3
L2	0	0	0	0	5	2	1	0	4	6	6	0	0	4	2	2
L3	5	4	2	0	0	0	0	0	2	1	2	6	4	0	1	1
L4	0	1	2	5	0	0	0	0	3	6	4	0	0	4	1	1
L5	5	3	2	0	0	0	0	0	0	0	0	0	4	0	2	0
L6	0	1	2	4	0	0	0	0	0	0	0	0	0	2	0	0
L7	0	0	0	0	0	1	2	5	0	0	0	0	0	0	2	2
L8	0	0	0	0	5	2	1	0	0	0	0	0	0	2	0	2
L9	4	4	3	0	0	0	1	1	2	0	2	4	4	1	6	0
L10	0	0	1	2	4	4	2	0	2	4	4	0	0	6	2	3

Table 3. Barrier Reduction Classification Table

Barrier Weights / Distance	No Barrier	Cement					Wood					Glass					Metal				
	0	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
0~4.9 feet	1	2	\	\	\	\	1	1	1	\	\	1	2	2	\	\	2	2	\	\	\
5~14.9 feet	1	3	4	5	\	\	1	2	2	3	\	2	2	3	3	\	2	3	4	\	\
15~29.9 feet	2	4	5	5	6	\	2	3	3	4	4	2	3	3	4	4	3	4	4	5	\
30~44.9 feet	3	5	6	6	7	7	3	4	4	5	5	3	3	4	4	5	4	4	5	5	6
45~59.9 feet	5	6	6	7	8	8	5	5	6	7	7	5	5	6	6	7	5	6	6	7	7
60~79.9 feet	6	7	7	8	8	8	6	6	7	7	8	6	6	7	7	8	7	7	8	8	8
80~99.9 feet	7	7	8	8	8	8	7	7	8	8	8	7	8	8	8	8	8	8	8	8	8
>100 feet	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8

Table 4. Interval Estimation Table

δ_{ij}	≥ 8	7	6	5	4	3	2	1	0
α_{ij}	0	0	10	20	30	40	50	60	70
β_{ij}	0	10	20	30	40	50	60	70	80

Table 5. Distances from locations to each zone

Z/L	Z1	Z2	Z3	Z4	Z5	Z6	Z7	Z8	Z9	Z10	Z11	Z12	Z13	Z14	Z15	Z16
L1	50.1	57.2	77.2	101.8	87.1	57.8	29.6	14.1	50.1	93.8	67.1	18.1	51	94.5	42.7	43.6
L2	103	82.4	61.5	51	14.4	38.8	67.6	93.1	48.2	18	28.2	87.9	97.7	20.8	65.6	58.2
L3	12	27.6	56.3	87.2	100.9	75.6	56.6	52.1	46	92.5	67.6	20.1	17.5	92.6	33	66
L4	92	64.6	36.5	13.7	52.8	64.4	83.9	107	49	20	38	89.8	84	21.8	60.4	73.9
L5	13.5	31.6	59.5	88.9	116	95	83.6	79.6	64.4	103.3	83.2	47.7	20	101	56	90
L6	88	60.8	32.6	15.1	80	85.7	101.7	119	61.5	48.2	57.6	95.8	82	49.4	79.9	93
L7	75	85.1	100.9	122	96	65.9	35.6	14.1	71.3	110.4	82	44.4	77.9	104	55	51
L8	128.8	100	83.2	75	11.4	41.1	70.7	97.5	65.6	41.2	46	99.11	112	45.6	79	57
L9	22.8	21.6	43.6	72.9	86.8	63	51	52.3	33	78.9	54	25	15.6	75.5	22.3	59.4
L10	90	71	55	52	21.6	28.6	54	80.8	38.8	25	20.8	75.9	85	20	51.9	34.1

Table 6. Regular Weights

L1	L2	L3	L4	L5	L6	L7	L8	L9	L10
4	6	6	4	1	1	1	1	10	10

Table 7. Relative Weights

L-L	L1	L2	L3	L4	L5	L6	L7	L8	L9	L10
L1	1	4	2	5	8	9	4	8	8	9
L2	4	1	5	2	10	6	8	10	11	10
L3	2	5	1	4	10	8	6	10	10	11
L4	5	2	4	1	8	4	9	8	9	8
L5	8	10	10	8	1	4	6	10	15	14
L6	9	6	8	4	4	1	10	6	11	10
L7	4	8	6	9	6	10	1	4	10	11
L8	8	10	10	8	10	6	4	1	14	15
L9	8	11	10	9	15	11	10	14	1	5
L10	9	10	11	8	14	10	11	15	5	1

Table 8. Signal Simulations to compare with D matrix (Normal Weights)

-0.19	0.02	-0.62	-0.23	-0.74	0.64	3.69	5.00	3.01	-0.95	1.85	5.89	1.55	-0.83	3.12	3.07
-0.50	-0.50	0.00	0.00	5.00	2.00	1.00	-0.50	4.00	6.00	6.00	-0.50	-0.50	4.00	2.00	2.00
5.00	4.00	2.00	-0.50	-0.50	-0.50	0.00	0.00	2.00	1.00	2.00	6.00	4.00	-0.50	1.00	1.00
-0.31	0.66	1.78	5.29	-0.13	-0.24	-0.47	-0.28	3.04	6.46	4.01	-0.81	-0.19	4.14	0.72	1.06
4.91	3.00	1.98	-0.56	-0.75	-0.27	-0.55	-0.25	-0.39	-0.48	-0.75	0.18	4.27	-0.46	1.93	-0.20
-0.57	0.74	2.32	3.91	-0.60	-0.47	-0.62	-0.41	-0.17	0.03	-0.13	-0.46	-0.54	1.67	-0.42	0.02
0.07	-0.96	-0.47	-0.29	-0.45	0.95	1.75	5.09	0.13	-0.88	-0.86	0.04	-0.42	-0.90	2.25	1.83
0.16	-0.54	-0.45	-0.16	4.66	2.18	1.12	-0.48	0.11	-0.11	-0.21	-0.54	0.08	2.27	-0.34	2.14
4.00	4.00	3.00	0.00	-0.50	0.00	1.00	1.00	2.00	-0.50	2.00	4.00	4.00	1.00	6.00	0.00
-0.50	0.00	1.00	2.00	4.00	4.00	2.00	-0.50	2.00	4.00	4.00	-0.50	-0.50	6.00	2.00	3.00

Table 9. Signal Simulations to compare with D matrix (Relative Weights)

-0.08	-0.06	-0.45	-0.54	-0.49	1.01	4.32	5.26	3.24	-0.24	2.03	5.77	2.06	-0.37	2.82	3.19
-0.31	-0.29	-0.07	-0.22	4.87	2.28	1.07	-0.68	3.86	5.85	5.97	-0.30	-0.65	3.90	1.79	1.99
5.00	4.00	2.00	-0.50	-0.50	-0.50	0.00	0.00	2.00	1.00	2.00	6.00	4.00	-0.50	1.00	1.00
-0.50	1.00	2.00	5.00	0.00	0.00	-0.50	-0.50	3.00	6.00	4.00	-0.50	-0.50	4.00	1.00	1.00
4.95	2.81	2.18	-1.05	-0.63	-0.43	-0.65	-0.21	0.16	-0.42	-0.26	-0.24	3.97	-0.68	2.11	-0.47
-0.45	0.98	1.87	4.22	-0.78	-0.55	-0.76	-0.62	0.24	0.20	0.20	-0.42	-0.62	1.70	-0.45	-0.21
0.08	-0.60	-0.63	-0.46	0.00	1.16	1.86	5.26	0.21	-0.53	-0.74	0.24	-0.65	-0.41	1.70	1.93
-0.50	-0.50	-0.50	0.00	5.00	2.00	1.00	-0.50	0.00	0.00	0.00	-0.50	-0.50	2.00	-0.50	2.00
4.00	4.00	3.00	0.00	-0.50	0.00	1.00	1.00	2.00	-0.50	2.00	4.00	4.00	1.00	6.00	0.00
-0.50	0.00	1.00	2.00	4.00	4.00	2.00	-0.50	2.00	4.00	4.00	-0.50	-0.50	6.00	2.00	3.00

Table 10. R Scores Using Regular and Relative Weights

R Scores Using Regular Weights

	Loc#1	Loc#2	Loc#3	**Loc#4**	**Loc#5**	**Loc#6**	**Loc#7**	**Loc#8**	Loc#9	Loc#10
1	**2.78234**	1.91243	1.81382	**2.92014**	**13.8345**	**14.3544**	**13.9083**	**14.0988**	1.00528	0.858373
2	**0.175992**	0.117328	0.083995	**0.059028**	**0.503968**	~~	**0.703968**	**0.703968**	0.036111	0.057897
3	**1**	0.666667	1	**0.75**	**2**	~~	~~	**1**	0.5	0.4
4	**0.75**	0.666667	0.666667	**0.75**	~~	~~	~~	**1**	0.3	0.4
5	**0.75**	0.5	0.666667	**0.75**	~~	~~	~~	~~	0.3	0.3
6	~~	0.166667	0.166667	**0.25**	~~	~~	~~	~~	0	0.1

Get Learning Data

Interval Estimations are Correct

Total Deployment Cost Using Regular Weights: 12

Deploying in this order: 7 -> 6 -> 0 -> 3 -> 5 -> 4

With Regular Weights the Total Deployment Cost is 22

R Scores Using Relative Weights

	Loc#1	Loc#2	Loc#3	**Loc#4**	**Loc#5**	**Loc#6**	**Loc#7**	**Loc#8**	Loc#9	Loc#10
1	**2.78234**	**1.91243**	1.81382	2.92014	**13.8345**	**14.3544**	**13.9083**	14.0988	1.00528	0.858373
2	**0.078219**	**0.117328**	0.062996	0.059028	**0.125992**	~~	**0.070397**	0.117328	0.032828	0.057897
3	**0.025**	**0.033333**	0	0	~~	~~	**0.033333**	0.033333	0	0.02
4	**0.5**	~~	0.166667	0	~~	~~	**1**	0	0.2	0
5	**0.25**	~~	0.166667	0	~~	~~	~~	0	0.1	0

Get Learning Data

Interval Estimations are Correct

Total Deployment Cost Using Relative Weights: 13

Deploying in this order: 6 -> 0 -> 1 -> 5 -> 4

With Relative Weights the Total Deployment Cost is 19

REFERENCES

[1] P. Bahl and V. N. Padmanabhan, "RADAR: An In-Building RF-Based User Location and Tracking System," in *Proc. of the IEEE INFOCOM Conf.*, pp. 775-784, 2000.

[2] N. Chen and A. Dahanayake, "Role-based Situation-aware Information Seeking and Retrieval for Crisis Response," *Int. J. of Intelligent Control and Systems*, Vol. 12, No. 2, pp. 186- 197, June 2007.

[3] Y. Chen and H. Kobayashi, "Signal strength based indoor geolocation," in *Proc. of IEEE Int. Conf. on Communications (ICC)* , vol. 1, pp. 436-439, 2002.

[4] Guo, Z., M. C. Zhou and G. Jiang, "Recursive Online EM Algorithm for Adaptive Sensor Deployment and Boundary Estimation in Sensor Networks," in *Proc. of 2006 IEEE Int. Conf. on Networking, Sensing and Control*, pp. 862 - 867, Ft. Lauderdale, Florida, U.S.A. April 23-25, 2006.

[5] Y. Gwon, R. Jain, and T. Kawahara, "Robust Indoor Location of Stationary and Mobile Users," in *Proc. of the IEEE INFOCOM Conf.*, pp. 1032-1043, 2004.

[6] J. Hightower, G. Borriello, and Roy Want, "SpotON: An Indoor 3D Location Sensing Technology Based on RF Signal Strength," *The University of Washington, Technical Report: UW-CSE 2000-02-02*, February 2000.

[7] A. Hills, "Large-scale wireless LAN design," *IEEE Communication Magazine*, Vol.39, pp. 98-107, 2001.

[8] A. Hills, J. Schlegel, and Ben.Jenkins, "Estimating Signal Strengths in the Design of an Indoor Wireless Network," *IEEE Trans. on Wireless Communications*, vol.3, no.1, pp. 17-19, January 2004.

[9] P. Krishnan, A.S. Krishnakumar, W.-H. Ju, C. Mallows, S. N. Gamt, "A System for LEASE: Location Estimation Assisted by Stationary Emitters for Indoor RF Wireless Networks," in *Proc. of the IEEE INFOCOM Conf.*, pp. 1001-1011, March 2004.

[10] A. M. Ladd, K. E. Bekris, A. Rudys, G. Marceau, L. E. Kavraki, D. S. Wallach, "Robotics-Based Location Sensing using Wireless Ethernet," *Proceedings of the 8th annual Int. Conf. on Mobile Computing and Networking (MOBICOM)*, pp. 227-238, Sept. 2002.

[11] H. Liu, H. Darabi, and P. Banerjee, "A New Rapid Sensor Deployment Approach for First Responders," *Int. J. of Intelligent Control and Systems*, Vol. 10, No. 2, pp. 131- 142, June 2005.

[12] S. Manapure, H. Darabi, V. Patel and P. Banerjee, "A comparative study of radio frequency-based indoor location systems," *IEEE Int. Conf. on Networking, Sensing and Control*, pp. 1265-1272, Taipei, Taiwan, March 2004.

[13] MeshNetworks, "Mesh Positioning Systems (MPS)," [Online], Available: http://www.meshnetworks.com/pages/technology, 2006.

[14] T. Murphy and M. E. Jennex, "Knowledge Management, Emergency Response, and Hurricane Katrina," *Int. J. of Intelligent Control and Systems*, Vol. 11, No. 4, pp. 199- 208, Dec. 2006.

[15] L. Nagy and L. Farkas, "Indoor base station location optimization using genetic algorithm," in *Proc. of IEEE Int. Symp. on Personal Indoor and Mobile Radio Communications*, pp. 843-846, 2000.

[16] N. Netten, G. Bruinsma, M. van Someren and R. de Hoog, "Task-Adaptive Information Distribution for Dynamic Collaborative Emergency Response," *Int. J. of Intelligent Control and Systems*, Vol. 11, No. 4, pp. 238- 247, Dec. 2006.

[17] P. Prasithsangaree, P. Krishnamurthy, and P. Chrysanthis, "On indoor position with wireless LANs," *Proc. of the 13th IEEE Int. Symp. on Personal Indoor and Mobile Radio Communications*, pp. 720-724, Sept. 2002.

[18] S. Ramaswamy, M. Rogers, A. D. Crockett, D. Feaker, and M. Carter, "WHISPER - Service Integrated Incident Management System," *Int. J. of Intelligent Control and Systems*, Vol. 11, No. 2, pp. 114- 123, June 2006.

[19] S. Ray, R. Ungrangsi, *et al.*, "Robust Location Detection in Emergency Sensor Networks," in *Proc. of the IEEE INFOCOM Conf.*, pp. 1044-1053, 2003.

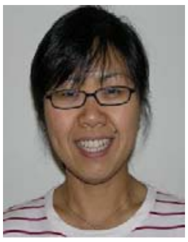
[20] A. Smailagic and S. Kogan, "Location Sensing and Privacy in a Context-Aware Computing Environment," *IEEE Wireless Communications,* vol. 9, no. 5, Oct. 2002, pp. 10-17.

[21] M. Youssef, and A. Agrawala, "Handling Samples Correlation in the Horus System," in *Proc. of the IEEE INFOCOM Conf.*, 2004, pp. 1023-1031.

MengChu Zhou received his B.S. degree from Nanjing University of Science and Technology, Nanjing, China in 1983, M.S. degree from Beijing Institute of Technology, Beijing, China in 1986, and Ph. D. degree in Computer and Systems Engineering from Rensselaer Polytechnic Institute, Troy, NY in 1990. He joined New Jersey Institute of Technology (NJIT), Newark, NJ in 1990, and is currently Professor of Electrical and Computer Engineering. His research interests are in computer-integrated systems, Petri nets, wireless ad hoc and sensor networks, system security, semiconductor manufacturing, and embedded control. He has over 250 publications including 6 books, 110 journal papers, and 16 book-chapters. His recent book includes *Modeling and Control of Discrete Event Dynamic Systems*, Springer, 2007, co-authored with B. Hruz. He served as Associate Editor of IEEE Transactions on Robotics and Automation from 1997 to 2000, and IEEE Transactions on Automation Science and Engineering from 2004-2007, and currently Managing Editor of IEEE Transactions on Systems, Man and Cybernetics: Part C, and Associate Editor of IEEE Transactions on Industrial Informatics. He was the recipient of NSF's Research Initiation Award, CIM University-LEAD Award by Society of Manufacturing Engineers, Perlis Research Award by NJIT, Humboldt Research Award for US Senior Scientists, Leadership Award. He is a life member of Chinese Association for Science and Technology-USA and served as its President in 1999. He is a Fellow of IEEE.

Wei Su received the B.S degree in electrical engineering and M.S degree in systems engineering from Shanghai Jiao Tong University, China, in 1983 and 1987, respectively. He received his Ph.D. degree in electrical engineering from The City University of New York, New York, in 1992. He is a senior research engineer in U.S. Army CERDEC, at Fort Monmouth, NJ since 1998. From 1991 to 1997, he was with US Army Research Laboratory. His research interests include wireless communication, signal and image processing, and adaptive control. He is the recipient of Superior Civilian Service Award and Medals, 2005 Army Research and Development Achievement Award, Army Material Command Top 10 Employee Nomination, 2004 and 2007 AOC International Research and Development Awards, and 2002 Thomas Alva Edison Patent Award. He is also recognized by Army CERDEC for Inventor's Wall of Honor and Electronic Warfare & Information Operations Association for Electronic Warfare Technology Hall of Fame. He is a senior member of IEEE.

Zaihan Jiang received his B.E from Xi'an Jiaotong University, Xi'an, China, M.S. from Chinese Academy of Science, Beijing, China, another M.S. from New Jersey Institute of Technology, Newark, NJ, US in 1999, and Ph.D. from New Jersey Institute of Technology in 2006, all in electrical engineering. He has more than eight year's industry experience, with Motorola Inc., US and Freescale Semiconductor. His research interest includes cross layer design, security and multi-media applications for cellular networks, wireless LAN, ad hoc, wireless sensor, and mesh networks. He has ten publications in conferences and journals. He is a member of IEEE.



Wenqi (Wendy) Guo is currently a Ph.D candidate at Department of Electrical and Computer Engineering, New Jersey Institute of Technology. She received the B.S degree from Beijing University of Civil Engineering and Architecture, China in 2001, and the Master degree from New Jersey Institute of Technology in 2006. Her research interests include wireless sensor network, mobile sensor network, control and automation, and intelligent building.