INTERNATIONAL JOURNAL OF INTELLIGENT CONTROL AND SYSTEMS
VOL. 10, NO. 2, JUNE 2005, 131-142

A New Rapid Sensor Deployment Approach for First Responders

Hui LIU, Houshang DARABI, and Pat BANERJEE

Abstract- **This paper introduces a new approach for selecting the location of Radio Frequency (RF) sensors used for detecting objects in a building. It is assumed that a building, its map and its main building materials are given. A number of moving or static objects, carrying RF tags are to be detected in this building. RF sensors are used to detect the tags. The time interval between the moment that the building information (map, etc.) become available and the moment that the detection must start is very short, perhaps less than 15 minutes. In this short duration of time, two problems are of interest and must be solved. First, the number and the deployment locations of sensors must be computed. Second, the detection procedure that uses the data from the deployed sensors must be developed. We present two algorithms to solve the stated problems. Our results from real world tests are satisfactory and show that the developed methods have the potential to be used by first responders such as police and firemen in detecting their indoor crews in the event of fire and other threats.**

Index Terms—**RF Sensor Network, Sensor Deployment, Indoor Object Detection, Zone Detection, Sensor Location Selection**

1. INTRODUCTION

Communication systems could play an essential role in emergency situations such as fires, building collapses or extreme weather phenomena. Unfortunately, existing systems often provide minimal communication infrastructure for supplying information about the nature or the extent of a disaster *in situ*. As a result, first responders typically enter emergency situations with little real-time information about the site, and, should they become trapped, only a haphazard means of rescue are available to them. One promising method for providing real time feedback from disaster sites involves the use of *sensor networks* and/or *wireless networks.*

Recent advances in sensor technologies make it possible to install and interconnect tiny devices within existing infrastructure, such as smoke detectors or overhead lighting, for networked use in case of an emergency. These networks could provide emergency control centers with 3D building visualization, real-time monitoring of hot spots or structure failures, and tracking of victims and personnel.

Central to such features is the ability to perform indoor *location detection* in the face of unpredictable reflections (from furniture, people, walls), occlusions (due to smoke, fire), and changing building topology (from falling walls, collapsed ceilings). Indeed, many essential tasks of an advanced first responder system require the following capabilities:

• To help crew members to identify their own and others' locations.

• To locate potential hazards, victims, or sources of the emergency.

• To identify and rescue trapped personnel.

• To identify the location of non human moving objects such as police dogs that are used to find the location of threat (e.g. a bomb) or unconscious victims in the building.

Existing approaches either take a long time to calibrate or learn [8, 17, 19, 20], or have a resolution of 10 meters or more [14] thereby making the results too unrefined for use by first responders.

User position estimation methods based on the strength of the radio signals received from multiple wireless access points have been recently proposed and implemented by several independent research groups [1, 2, 6, 8, 9, 10, 11, 14, 15, 17, 19, and 20]. For instance, in [1] and [18], the RADAR location system by Microsoft Research calculates the position of a Wi-Fi device either by similarity with previous measurements, or by modeling signal propagation. In [13], a comparative study of radio frequency-based indoor location sensing systems is presented. In this paper a new approach to wireless access point placement techniques is described. The method integrates coverage requirements with the reduced of the error of the user position estimate.

The literature seems to be rather scarce on sensor placement in the context of indoor localization using RF techniques. One issue in the design and implementation of a wireless local area network is the selection of access point (AP) locations. Proper AP placement is necessary to provide adequate signal coverage and also to minimize co-channel coverage overlap. The impact of incorrect placement of APs is significant. Placing APs too far apart can lead to gaps in coverage. On the other hand, placing the units too close together leads to excessive co-channel coverage overlap, degrading system performance. A method to deploy a 3-dimensional large-scale indoor IEEE 802.11b WLAN or other micro-cellular networks is proposed in [4], where placement is constrained so that there are no coverage gaps, and no overlaps between APs operating on the same channel are allowed. In [5], the same author of [4] provides a procedure for estimating the coverage areas of relocated APs.

A wireless AP placement technique for optimal signal coverage inside buildings is analyzed in [7]. The signal propagation model used is the empirical Motley-Keenan indoor wave propagation model where the path loss

Manuscript received March 12, 2005; revised May 20, 2005. This work was partially funded by a NIST ATP grant and by the Illinois Law Enforcement Alarm System under a Department of Homeland Security grant.

Hui Liu is with Dept. of Electrical and Computer Engineering, University of Illinois at Chicago, 1004 South Oakley, Chicago, Illinois, 60612, USA (e-mail: hliu13@uic.edu); Houshang Darabi and Pat Banerjee are with Dept. of Mechanical and Industrial Engineering, University of Illinois at Chicago, 842 W. Taylor, 2039 ERF, M/C 251, Chicago, Illinois, 60607 USA (e-mail: hdarabi@uic.edu and banerjee@uic.edu).

depends on the type of walls, ceiling and floors. The cost function is the ratio of the points on the floor with higher path loss than a specific threshold; and the function is minimized by using genetic algorithms.

The issue of location detection through a novel framework based on the theory of identifying codes has been addressed in [6]. The key idea of this approach is to allow sensor coverage areas to overlap in such a way that each resolvable position is covered by a unique set of sensors. In this setting, determining a sensor-placement with a minimum number of sensors is equivalent to constructing an optimal identifying code. It has a moderate computational complexity.

Some algorithms for the efficient placement of sensors in a sensor field are described in [3, 7]. They are aimed at optimizing the number of sensors and determining their placement to support distributed sensor networks. They represent the sensor field as a grid (two- or three-dimensional) of points. A target in the sensor field is therefore a *logical object*, which is represented by a set of sensors that *see* it. An irregular sensor field is modeled as a collection of grids. The optimization framework is inherently probabilistic due to the uncertainty associated with sensor detections. These algorithms are targeted at average coverage as well as at maximizing the coverage of the most vulnerable grid points. These algorithms are suitable for outdoor field, unsuitable for indoor environment.

To our knowledge, the only two papers that relate signal strength detection errors and location errors, hence suggesting localization-aware AP placement, are [11] and [12]. In [12], the linear and multiple regression methods are used to estimate the signal strength model of an indoor wireless AP by experimental data. Some results are obtained by analyzing the relationship between signal strength error and localization error. In [11], a method integrates coverage requirements with the reduction of the error of the user position estimates. They propose a mathematical model of user localization error for algorithms based on the variability of signal strength measurements. This model has been designed to be independent from the actual localization techniques, and it is only based on generic assumption on the behavior of the localization algorithm employed. This model is implemented in reactive *tabu* search, local search and simulated annealing techniques in order to optimize wireless access point placement with respect to user localization error. This method is suitable for planar environment, maybe not suitable for 3D environment.

In this paper we present two algorithms that are used to find the best locations for sensor deployment, and the detection rules. Through several real world experiments we show that the developed system can be deployed quickly and offers a high level of detection accuracy. These features make our system unique when compared with similar systems reported by other researchers. In Section 2 of the paper we define the research problem and its specific assumptions. Two different algorithms are used to solve the problem. These algorithms are presented in Sections 3 and 4. In Section 5 we report the performance of the developed algorithms for three large examples. Section 6 concludes the paper and suggests the future research problems that can be followed from this research.

2. PROBLEM DEFINITION

We consider the following sensor location and object tag detection problem. A building is determined and the following information regarding the selected building is provided:

- Computerized version of the building map (ideally in AutoCAD or an equivalent format)
- General types of the main construction materials used in the building exterior and interior (e.g. cement, glass, metal).
- Building detection zones: a discrete division (partition) of the building to non-overlapping spaces; the definition of zone is based on the detection objectives. A zone is a finite and continuous space. All the points in a zone must have 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. Two neighbor zones must be separated by a partition such as wall, glass, etc. We assume that zone dimensions are greater than 10’ (length) by 10’ (width). The dimension of zone height is not important as we assume that the tag objects move or remain on the zone floor.
- Feasible sensor deployment locations, inside and outside of the building, are known.
- 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. For example it may be much faster and easier to deploy a sensor in a location outside in front of the building entrance door than a location on the ninth floor of the building.
- A set of objects are equipped with unique RF transmitters (tags) that are present in the building. The tags are independent from each other.

Without loss of generality, we assume that only one tag is given. Two problems are of interest in this system.

Problem 1: Detection Problem

Given a set of sensors inside and outside the building, what is the set of rules that assigns a zone to every possible combination of received signal strength from the sensors? Henceforth we are interested in determining which zone a tag is located in and not the exact coordinates of the tag.

Problem 2: Location Problem

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

In the following we present two algorithms to solve the above stated problems. Figure 1 shows the relationships of these two problems. As shown in Figure 1, we first solve the location problem. The obtained set of sensor locations is then used to detect the tag zone. However, in the following, we first present the detection algorithm. The insight to this algorithm will help us derive the location algorithm.

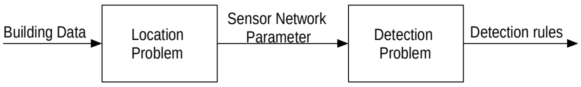


Fig. 1. Location and detection problems and their relationship

In order to address our approach in the following sections clearly, we make a list of symbols here.

C : Signal score binary threshold matrix
D : Tag zone-sensor location signal score matrix
E : Signal strength range impact amplification matrix
$E(SS_{ij})$: 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
$f_j(x)$: 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_{oi} : 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_i and location l_i

3. DECTION ALGORITHM

Let $Z = \{z_1, z_2, \cdots, z_n\}$ and $L = \{l_1, l_2, \cdots, l_m\}$ be the set of zones and the set of feasible deployment locations respectively. Denote the set of selected locations for sensor deployment by $O = \{o_1, o_2, \cdots o_k\}$ where $O \subseteq L$. Let s_r be the state of the sensor deployed at location $o_r \in O$. In our experimental setup, the value of s_r is either zero or an integer number between 1 and 8. This value is generated by a function of the signal strength reported by the sensor deployed in location $o_r \in O$. Define $x = \{s_1, s_2, \cdots, s_k\}$ as the sensor network state vector and X as the set of possible states of the sensory network. We let $f_j(x)$ be conditional probability distribution of receiving sensor network state vector x given that the tag is present in zone z_j. Also let $g_x(z_j)$ be the conditional probability distribution of the tag presence in zone z_j given that the received sensor network state is x. Finally, define $h(z_j)$ as the probability distribution of the tag presence in zone z_j.

Suppose that we have solved the location problem. That is, we have found the members of O. After deploying the sensors, the state of the sensor network is continuously read and used to estimate the tag zone. The process of estimation starts by finding the tag presence probabilities in different zones given the received sensor network state. After calculating these probabilities, it is reasonable to assume that the tag is present in the zone with the highest calculated probability. We call this detection policy the *maximum probability rule*. In order for this policy to be effective, i.e., to establish a good detection precision, the maximum probability value must be significantly larger than other probabilities. For example, consider a problem with three zones. Suppose that based on a given sensor network state we conclude that the probabilities related to the presence of tag in those zones are 0.3, 0.32, and 0.33, for zones 1, 2 and 3 respectively. Here based on maximum probability rule zone 3 is selected. However, this selection has a very low precision as with probability of 0.3+0.32=0.62 the tag is not in zone 3.

We notice that zone probabilities are a function of sensor locations. Therefore in order to use the maximum probability rules effectively, we need to select the sensor locations in a way that the maximum probability for every possible state of the sensory network is significantly different for other probability values. For example in the previous scenario, if the probabilities were 0.02, 0.03 and 0.95 then our conclusion of the presence of tag in zone 3 would be very accurate. In the location algorithm we will use this concept (increasing the maximum probability) as the basis of our algorithm. Here we present the detection algorithm first and use the insight into this algorithm to discuss the location algorithm. As shown in Figure 1, the location algorithm determines the sensor network information that is used by the detection algorithm to construct the detection rules. Formally, the zone estimation procedure includes the following steps.

Detection Algorithm

1- Read x
2- Calculate $g_x(z_j / x)$ for all $z_j \in Z$

3- Let $M(x) = Maximum\{g_x(z_j / x), z_j \in Z\}$

4- Select $z_j \in Z$ such that $g_x(z_j / x) = M(x)$. Solve the ties arbitrarily.

End of Detection Algorithm.

Based on the above procedure the detection error of sensor network state x is $1 - M(x)$. We recall that $\sum_j g_x(z_j / x) = 1$ because Z establishes a zone partitioning of the building, that is, $\bigcup_j z_i$ covers all possible zones for the presence of the tag, and $z_i \cap z_j = \phi$ for $z_i, z_j \in Z$ where $i \neq j$.

To implement the detection procedure, one needs to calculate $g_x(z_j / x)$. We use Bayesian method for this calculation. Based on this method, we can write:

$$g_x(z_j / x) = \frac{f_j(x)h(z_j)}{\sum_{z_i \in Z} f_i(x)h(z_i)} \quad (1).$$

The information about $h(.)$ can be obtained from the movement pattern of the tag. This can be given by the system user. If no prior information related to $h(.)$ is available, one can simply assume that $h(z_j) = \frac{1}{n}$ for all $z_j \in Z$. Here we use this assumption to present our method. Formula (1) can be used for other forms of $h(.)$. Applying the $h(.)$ uniformity assumption to (1), we obtain:

$$g_x(z_j) = \frac{f_j(x)}{\sum_{z_i \in Z} f_i(x)} \quad (2).$$

The behavior of RF sensors is not known due to many factors such as signal propagation, noise, and building materials. Therefore we try to identify $f_j(x)$ by learning. We use the data frequency method to estimate $f_j(x)$ from a set of data collected during the learning. This method is described below.

Suppose we collect a random sample of N sensor network observations. We recall that the tag transmits a radio frequency wave to each sensor. Depending on the distance between the tag and a sensor, and other building and environment factors, the value of the signal strength (SS) reported by the sensor can change. In our experimental set-up, the state of the sensor network is known by values of the SS reported by sensors at any time. To define the sensor network state in a discrete manner, assume that the values $\{ y_1, y_2, \cdots, y_a \}$ break the total SS range to $a+1$ non-overlapping intervals as follows: $(0, y_1), [y_1, y_2), \cdots, [y_{a-1}, y_a), [y_a, \infty)$.

The state of the sensor deployed in location l_i when that tag is in zone z_j can be written as:

$$s_{ij} = \begin{cases} 0 & ss_{ij} = 0 \\ 1 & ss_{ij} \in (0, y_1) \\ t+1 & ss_{ij} \in [y_t, y_{t+1}), 1 \le t \le a-1 \\ a+1 & ss_{ij} \in [y_a, \infty) \end{cases} \quad (3)$$

where ss_{ij} is the SS reported by this sensor for the tag.

We estimat $f_j(x) = \prod_o f_{oj}(s_{oj})$ where $f_{oj}(s_{oj}) = \frac{n_{oj}}{N}$ for $0 \le s_{oj} \le a+1$. Here n_{oj} is the number of observations that their signal strengths received from the sensor in location o fall in the $s_{oj}th$ interval when the tag is in zone z_j. From the above definition of intervals, we have $\sum_{s_{oj}=0}^{a+1} f_{oj}(s_{oj}) = 1$. We also notice that $s_{oj}=0$ shows the case that the sensor is not receiving any signal from the tag.

We recall that the indoor signal propagation results in the generation of continuous values of SS. Therefore if the values $\{ y_1, y_2, \cdots, y_a \}$ are selected in a way that the lengths of intervals $(0, y_1), [y_1, y_2), \cdots, [y_{a-1}, y_a), [y_a, \infty)$ be large then we can assume that the probability function $f_{oj}(s_{oj})$ satisfies the *continuous SS spectrum* property. This means that for every zone z_j and a deployed sensor in location o, we can write that $\forall 2 \le s_{oj} \le a-1$ if ($f_{oj}(s_{oj}-1) \neq 0$ and $f_{oj}(s_{oj}+1) \neq 0$) then $f_{oj}(s_{oj}) \neq 0$.

4. LOCATION ALGORITHM

As stated in the detection algorithm, the detection procedure is based on the state of the sensor network, which depends on the deployed sensors. For a given building, there could be many locations inside or outside the building that can be used to place the sensors. Therefore the locations of the sensors have to be decided. Such decision is made by the location algorithm. Due to specific nature of our problem, we first discuss the design objectives that must be considered in developing the location algorithm.

Objective O1: Zone coverage objective

This objective states that every zone must be covered by at least one sensor. That is, for every zone there must be at least one sensor that reports an SS greater than zero when a tag is present in that zone. Suppose that zone z_j is not covered by any sensor, then from the frequency method we conclude that $f_j(0,0,\cdots,0) = \prod_o f_{oj}(0) = 1$ and $f_j(0,0,\cdots,0) = \prod_o f_{oj}(0) = 1$ for $x \neq (0,0,\cdots,0)$. Using (1) and (2), we have $g_x(z_j /(0,0,\cdots,0)) = 1$ and $g_x(z_j / x) = 0$ for $x \neq (0,0,\cdots,0)$. Therefore the maximum probability rule cannot be applied to this case because we cannot distinguish between the presence and absence of the tag in zone z_i. We assume that every zone is covered by at least one sensor. If such an assumption cannot be realized by any possible sensor deployment, then all uncovered zones will be left out of consideration. Our experience from the tested buildings (refer to section V) shows that almost in every case this assumption holds as we can usually find a sensor location that covers every given zone.

Objective O2: Maximum conflict resolution objective

As mentioned when discussing the detection algorithm, the detection error is a function of the maximum probability rule performance. Based on this objective the ideal sensor network should guarantee that $M(x)=1$ for every possible state x of the sensor network. Such a network has a 100% conflict resolution. Therefore one objective of the location problem is to select the location for sensor deployment in a way that it maximizes the detection resolution by minimizing conflicts.

Objective O3: Minimum deployment time objective

To achieve the best possible values for objectives O1 and O2, one trivial solution is a sensor network that is built based on $O = L$. This means that every possible location for sensor deployment is used. However, the deployment of this network is usually very time-consuming and therefore expensive. Based on our experiments, in a regular building, there could be hundreds of possible sensor locations. Many of these locations are inside the building and the deployment time for them could be large. For example, in a building with 10 floors, if the elevators are down, then the time to reach a location on the 10^{th} floor could be more than 2 minutes. We recall that in our specific problem time between identifying the target building and using the network for detection is very short and therefore it is important to minimize the deployment time. To address this objective we define w_i as the cost (time) of reaching location l_i from outside the building. Therefore the costs of locations that are located outside the building on the ground are almost zero as these locations can be immediately reached for deployment. For the locations inside the building, depending on their accessibility and time to reach (from outside), their costs can be defined. The location cost can be automatically derived using the AutoCAD map of the building by estimating the travel time from outside through the available pathways to the targeted location. The focus of the current paper is not to show how to use the building map to generate such cost weights. Here we only use these weights in the execution of our location algorithm.

All three objectives are simultaneously handled in our location algorithm. While we make sure that every zone is covered by at least one sensor objective (O1), we try to find the optimal deployment strategy by using a combined ratio that takes into consideration objectives O2 and O3. This combined ratio is the cost or time spent (objective O3) for resolving one unit of conflict (objective O2).

Before showing the algorithm steps, we need to define some notations. Due to our specific sensor specifications that will be discussed in section V, we use the following SS interval settings: $a = 7$, and $y_i = 10 * i$ for $1 \le i \le 7$. Since the maximum SS in our setup could be 80 dB, therefore we have defined 8 intervals each with a length of 10 dB. We notice that our selection of SS settings has been mainly based on our experiment history with different buildings and our sensor network hardware. The location algorithm can be modified easily for other settings as required by different types of detection hardware. In the following algorithm, let $p = a + 1 = 8$. Also we assume $O = \{\}$ before the algorithm starts.

The mean value of the SS for the sensor located in location l_i when the tag is in zone z_j is shown by $E(SS_{ij})$. Since when the sensor is not actually deployed in l_i, calculating this mean is not possible, we use an interval estimation of $E(SS_{ij})$ to implement the location algorithm. To obtain such estimation, one can use only the information available before the learning. Based on our experiments with different buildings and our specific detection hardware, we develop Table 1 that is used in the estimation process. We used three inputs in generating the data of this table: the Euclidean distance between the center of z_j and l_i, the number of barriers blocking the Euclidean line connection z_j and l_i, and the type of these barriers. All the barriers considered here are assumed to have a thickness of less than 3 inches. During our experiments we have many situations that the thickness of a barrier was more than 3 inches. This is especially true for building perimeter wall and floor separators. We simply assume that such barriers can be considered as multiple barriers of 3 inches or less. Therefore, for example, a 5-inch cement barrier is considered as two cement barriers, or a cement barrier with the thickness of 12 inches is considered as 4 such barriers.

We assume a linear barrier weight with the weight values directly proportional to the severity of the barrier, thereby linearly proportional to the SS reduction. We calculate the *total reduction* for z_i and l_i, shown by δ_{ij}, by adding the numbers obtained from Table 1. For example if the distance between z_i and l_i is 8 feet and there exist one wooden wall barrier with the thickness of 4 inches and one glass barrier with the thickness of 2 inches, on the Euclidean line between z_i and l_i, then δ_{ij} =2*1+2=4. We then use Table 2 to find corresponding lower (α_{ij} =30) and upper (β_{ij} =40) bounds for the estimation interval.

To test our estimation procedure, we randomly select a sample of 100 location-zone pairs in four different buildings and measured the average SS for every pair. The actual average falls in the estimated interval by the procedure in 92% of the time. We believe such a confidence level is satisfactory.

Table1: Barrier reductions 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 2: Interval estimations

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

Location Algorithm

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:

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

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

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 $1\leq k\leq p-1$;

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

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

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

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:

$$R(l)=\frac{1}{w_l}\left(\sum_{(i<j)\wedge(|g_{ij}|\geq 1)}\frac{Y_{ij}(l)}{|g_{ij}|}\right),$$ 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 still several *l's* are candidate then select one arbitrarily. Let l_0 be the selected location.

4.3.1. Remove l_0 from *H*.

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

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

5.3.3. 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 $0\leq s_{oj}\leq a+1$. 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 as defined by (3).

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.

Here we discuss how the location algorithm satisfies the three design objectives O1, O2 and O3. In the first iteration of the algorithm steps, we start with an empty set for the selected sensors. This is given by the assumption that $O = \{\}$ before the algorithm starts. In the first iteration of steps 1 through 5 an initial set of sensor locations are selected. In the first iteration of step 6 the selected locations are filled with sensors, the learning process is executed and based on the learning data it is decided if more locations have to be filled with sensors. If this is required then steps 2 to 5 are used to select new locations and step 6 is again used for learning about the new deployed sensors and checking if more locations are needed. Iterations of steps 2 to 6 are repeated until either all possible locations are selected ($O=L$) or the selected locations satisfy the maximum probability rule conditions and cover all the zones. In either case, the algorithm proceeds to its last step, where the probability distributions needed for detection algorithm are calculated. This step is executed once. Figure 2 shows the summary of the algorithm steps.

We recall that before the algorithm starts, the set of deployed sensors is empty and therefore no learning data is available. In order to select the first set of locations for sensor deployment, we start with tag zone sensor location signal score matrix D in step 1, which is constructed using the interval estimators of average SS for every pair of location and zone. These interval estimators (α_{ij} and β_{ij}) were discussed before the location algorithm. The main idea behind the creation of this matrix is to obtain a mechanism to find the number of conflicts in applying the maximum probability rules for different combinations of sensor locations. We have taken a conservative approach in finding the number of conflicts. This means, in some cases, we might attempt to remove a conflict that does not really exist. Such an attempt could result in selecting more sensors than what we really need to apply the maximum probability rule with the required precision. In such cases some sensors are redundant and do not affect the detection precision. However, we will know what sensors are redundant (or do not have any effect on the precision of the maximum probability rule) only after we finish the learning process and compute the signal probability distributions.

Steps 2 and 3 prepare the required data needed to start the location selection. We recall that the sensor location selection is based on interval estimators of average SS. Such estimators might originate a wider range of SS (when SS range impact amplification matrix E in step 2 is created) for every pair of location and zone. Again we have to use this inaccuracy due to the lack of learning data and our limited time in finding the probability rules required for actual detection.

The tag zone-sensor location signal score matrix D is used in step 2 to construct SS range impact amplification matrix. Matrix E implicitly extends the 10 dB interval estimators used in step 1 to 30 (20) dB estimators for intermediate (extreme) score numbers. This means that using matrix E we estimate that for every pair of location and zone the SS values fall in a 30 dB interval (or 20 dB for extreme score points). In practice based on our specific hardware and considering the effect of uncontrollable building factors such as distance, barriers, noise, magnetic fields, moving objects and many more, in most cases the range of SS for a specific pair of sensor and zone falls in less than a 20 dB interval. Our 30 dB approach, although conservative, compensates for most errors that could have resulted from our original estimations (α_{ij} and β_{ij}).

In step 3, using matrix E we generate the conflict resolution matrix G. Every defined element of matrix G corresponds to two different zones. The content of this element is all the sensor locations that can be used safely (with no conflict) to distinguish between the corresponding zones. We notice that if a defined element of matrix G is empty then using the available sensor locations we may not be able to effectively distinguish between these two zones. In other words, there may exist a state of the sensor network for which the maximum probability rule cannot accurately select one of the corresponding zones over the other.

In steps 4 and 5 the locations are selected. In step 4 we use the conflict resolution matrix and select the actual locations for sensor deployment. If this is the first iteration of this step then all the selected locations have to be filled by the sensors. For any other iteration of this step, only the selected locations that have not been previously selected should be filled with sensors. The objective of step 4 is to

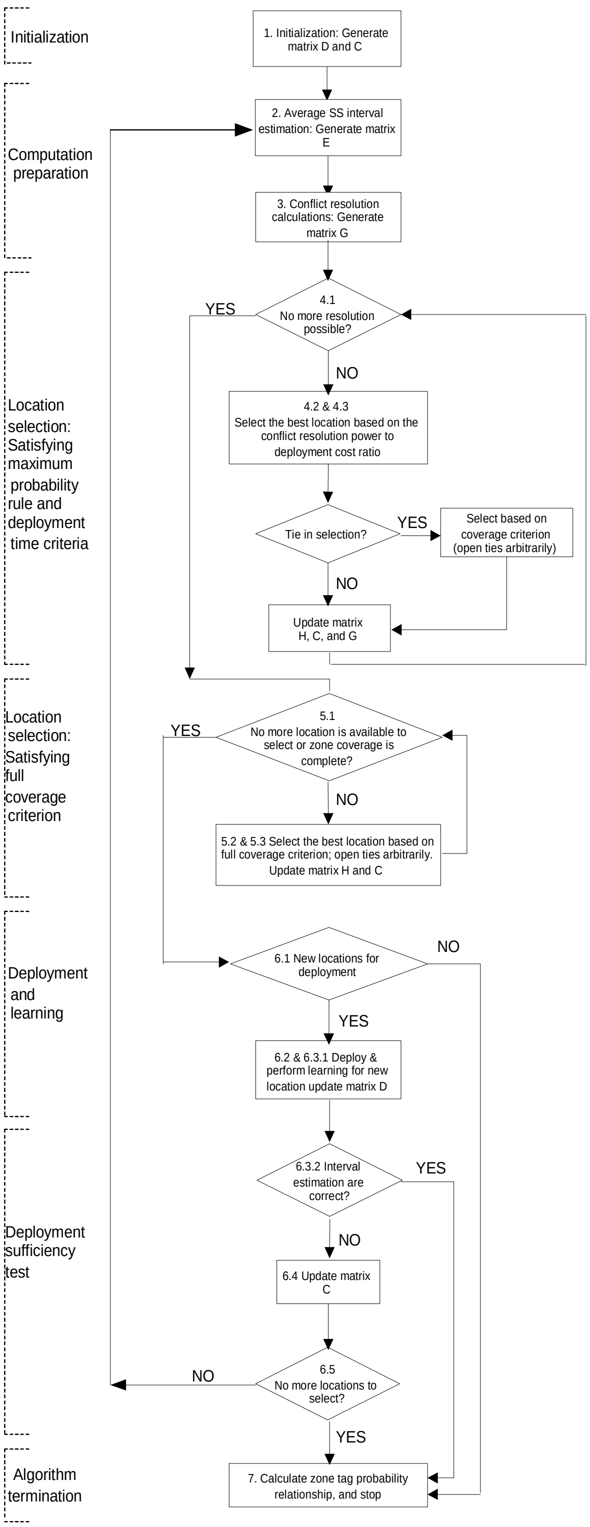


Fig. 2. Summary of location algorithm steps

resolve possible conflicts between every two zones by selecting the locations that can be used for this resolution. These are the locations listed by matrix G. We continue the selection process until no two zones can have conflict (the stopping rule for the selection process is given by step 4.1). This process guarantees the satisfaction of objective O2. During the selection process, we select the locations based on the value of their combined ratios that is calculated in step 4.2. As already mentioned, the combined ratio is the cost or time spent (based on objective O3) for resolving one unit of conflict (based on objective O2). Therefore, in the selection process, we give priority to the locations that resolve maximum number of conflicts and require minimum deployment time. In calculating the maximum number of conflicts we have distinguished between the effects of locations. The effect of a location is the summation of individual effect scores that the location has on resolving the conflicts between every two zones. For location l and pair of zones i and j, the effective score is zero if $l \notin g_{ij}$ and is $\frac{1}{|g_{ij}|}$ otherwise. That is the effect of location l would decrease if the number of locations that can resolve the conflict between zones i and j increases. If there is a tie based on the calculated combined ratios, then we use objective O1 to select the location. In this case the location that has more zone coverage is selected. This is done in step 4.3. Step 4.3 also performs different updates to continue the selection process. In step 5, we guarantee the complete satisfaction of objective O1.

Step 6 is where we deploy the sensors and perform the learning process. If based on the learning data we conclude that the elements of matrix D have to be updated, we update that matrix based on real data (step 6.3.1) and start the next iteration from step 2. We notice that the algorithm definitely terminates because the set of locations L is finite. In most cases we do not select all the locations and we stop after one of the stopping rules is met.

As the last statement in this section, we emphasize the point that our approach is not sensitive to exact probability numbers or the coordinates of the tag as addressed by other researchers. This is because we rely on conflict resolution between signal intervals. In other words we use interval estimators as the basis of our detection rather than point estimators. This has increased the speed of our deployment

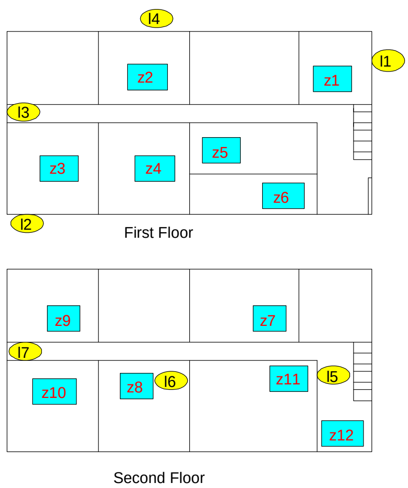


Fig. 3. The map of each floor of a small example

while keeping an acceptable detection error.

A small example

To illustrate the location algorithm, we consider the following example. This example is intentionally small to show the details of the algorithm steps. In the next section, we report our experience and the algorithm performance for larger size examples.

Our target building has two floors. The map of each floor is shown in Figure 3. The circles represent the possible sensor locations, and the rectangles show the zones that must be detected. Again it is assumed that the tag is always in one of the zones. The goal is to select a subset of $L = \{l_1, l_2, \cdots, l_7\}$ for sensor deployment, while satisfying our design objectives O1, O2 and O3. The following weights are considered as the location deployment costs: (1 1 3 1 6 8 9).

In step 1, the tag zone-sensor location signal score matrix *D* is calculated based on the estimation intervals. This matrix is shown in Table 3. As an example, location l_1 and the center of zone z_1 have a distance of 12 feet. One barrier exists on the Euclidean line connecting location l_1 to the center of zone z_1. The type of this barrier is glass and its thickness is less than 3 inches. Therefore, the reduction is 2 (refer to Table 1) and the estimated signal interval is [50,60] from Table 2. Using this estimation the signal score is 6. The signal score binary threshold matrix in Table 4 is calculated accordingly. In step 2 we construct the SS range amplification matrix. The result is shown in Table 5. As an example, consider the cell at the intersection of row l_1 and column [0,10]. This cell includes all the zone numbers that the presence of tag in them could cause the sensor located in l_1 to report a signal strength in the interval [0,10]. In this case zones z_2, z_5, z_7 and z_{12} belong to this cell. In the case of zone z_2, for example, from matrix *D*, we conclude that the interval estimator for the average SS for this zone and location l_1 is [10,20]. Accounting for the 30dB signal distribution range, the SS for this combination could be in anywhere in [0,30] including [0,10]. That is why this zone belongs to the related cell in matrix *E*.

Table 3: Tag zone-sensor location signal score matrix in the first iteration

D	z1	z2	z3	z4	z5	z6	z7	z8	z9	z10	z11	z12
l1	6	2	0	0	2	3	1	0	0	0	0	2
l2	0	2	5	4	2	2	0	1	0	2	0	1
l3	2	4	6	4	3	2	0	0	1	1	0	0
l4	4	6	1	2	2	1	1	0	1	0	0	0
l5	1	0	0	0	0	1	4	2	1	0	6	7
l6	0	0	0	1	0	0	3	8	3	4	4	2
l7	0	0	1	0	0	0	2	3	6	6	2	1

Table 4: Signal score binary threshold matrix in the first iteration

C	z1	z2	z3	z4	z5	z6	z7	z8	z9	z10	z11	z12
l1	1	1	0	0	1	1	1	0	0	0	0	1
l2	0	1	1	1	1	1	0	1	0	1	0	1
l3	1	1	1	1	1	1	0	0	1	1	0	0
l4	1	1	1	1	1	1	1	0	1	0	0	0
l5	1	0	0	0	0	1	1	1	1	0	1	1
l6	0	0	0	1	0	0	1	1	1	1	1	1
l7	0	0	1	0	0	0	1	1	1	1	1	1

Table 5: *SS* range impact amplification matrix in the first iteration

E	[0,10]	[10,20]	[20,30]	[30,40]	[40,50]	[50,60]	[60,70]	[70,80]
l1	2 5 7 12	2 5 6 7 12	2 5 6 12	6	1	1	1	0
l2	2 5 6 8 10 12	2 5 6 8 10 12	2 4 5 6 10	3 4	3 4	3	0	0
l3	1 6 9 10	1 5 6 9 10	1 2 4 5 6	2 4 5	2 3 4	3	3	0
l4	3 4 5 6 7 9	3 4 5 6 7 9	1 4 5	1	1 2	2	2	0
l5	1 6 8 9	1 6 8 9	7 8	7	7 11	11 12	11 12	12
l6	4 12	4 7 9 12	7 9 10 11 12	7 9 10 11	10 11	0	8	8
l7	3 7 11 12	3 7 8 11 12	7 8 11	8	9 10	9 10	9 10	0

In step 3 we construct the conflict resolution matrix *G*. This matrix is shown in Table 6. Every cell of this matrix is corresponding to a pair of zones (except the cells on the diameter). Sign "\" means "undefined" and "--" means "all sensor locations". The cells below the diagonal have not been filled for the sake of brevity. We recall that $g_{ij} = g_{ji}$ for $i > j$. The numbers inside every cell show alternative locations that could be used safely in distinguishing between the corresponding zones of that cell. For example, consider $g_{12} = \{1, 2, 5, 6, 7\}$. This means either of locations l_1, l_2, l_5, l_6 or l_7 can be used to distinguish between zones z_1 and z_2. To illustrate the idea, we explain this for location l_1 and zones z_1 and z_2. Based on the interval estimation for average signal strength reported by the sensor located in l_1 for the tag in z_1, the SS is a value between 40dB and 70dB. We recall that the interval estimation based on matrix *D* is [50,60] and using the 30dB SS range we find the range of [40,70] in this case. By a similar reason the SS range for the sensor in l_1 and tag in z_2 is [0,30]. This means that the sensor located in l_1 can distinguish between the tag in z_1 and the tag in z_2 as the values of SS reported in these cases do not have any overlap (two intervals [40,70] and [0,30] are disjoint). In this case for every reported SS value we can claim, with probability of 1, where the tag is, and this results in 100% satisfaction of the maximum probability rule.

Table 6: Initial conflict resolution matrix

G	z1	z2	z3	z4	z5	z6	z7	z8	z9	z10	z11	z12
z1	\	1 2 5 6 7	--	12 5 6 7	1 2 5 6 7	1 2 4 6 7	--	1 2 3 4 6 7	1 2 4 6 7	1 2 4 5 6 7	--	--
z2		\	1 2 4 5 6 7	1 4 5 6 7	4 5 6 7	4 5 6 7	2 3 4 5 6 7	1 3 4 5 6 7	--	1 3 4 5 6 7	--	3 4 5 6 7
z3			\	1 5 6 7	1 2 3 5 6 7	1 2 3 5 6 7	1 2 3 5 6	1 2 3 4 5 6	1 2 3 5 6 7	--	1 2 3 4 5 6	1 2 3 4 5 6
z4				\	1 5 6 7	1 5 6 7	1 2 3 5 7	--	1 2 3 5 7	1 3 4 5 6 7	--	1 2 3 4 5 7
z5					\	5 6 7	2 3 5 6 7	1 3 4 5 6 7	1 2 5 6 7	1 4 5 6 7	--	3 4 5 6 7
z6						\	2 3 5 6 7	1 3 4 6 7	1 2 6 7	1 4 5 6 7	--	3 4 5 6 7
z7							\	1 2 3 4 6	1 2 3 5 7	1 2 3 4 5 7	1 2 3 4	2 3 4 5
z8								\	1 2 3 4 6 7	1 3 4 5 6 7	1 2 3 4 5 6	1 3 4 5 6
z9									\	1 2 4 5	1 2 3 4 5 7	1 2 3 4 5 7
z10										\	1 2 3 4 5 7	1 3 4 5 7
z11											\	1 2 3 4
z12												\

In step 4 we try to select a subset of locations that could distinguish between all the zones. As mentioned, the conflict resolution matrix carries all possible choices of location. We start our selection from the location that has the maximum ratio of conflict resolution opportunity to its deployment time. In this example, our first choice is l_1. This is because l_1 has a deployment time of 1 and it has an opportunity of 10.2214 in removing the confusion. The number 10.2214 is calculated by adding the scores of l_1 for all the cells above the diagonal of matrix *G*. This score for the cells that do not contain l_1 is zero. For each cell containing l_1 the score is 1 divided by the number of elements in that cell. The less the number of elements in a cell the higher the calculated score will be. This reflects the importance of the locations attending in that cell as our choices would be more limited if the number of attending locations in a cell is less. After selecting every location, matrix *G* will be updated by setting the cells containing the selected location to empty set. If a tie happens in the selection process the rule of maximum zone coverage is used for the selection. In this example steps 4.1 to 4.3.4 are repeated 3 times in the first iteration of the algorithm resulting in the selection of locations l_1, l_4 l_2 and l_5. The location scores related to these four runs are shown in Table 7. The bold numbers show the maximum scores in each run that determine the selected locations. The four selected locations completely resolve all the conflict in detecting the zones.

Table 7: Location scores in the first three runs

	l1	l2	l3	l4	l5	l6	l7
First run	**10.2214**	8.2548	2.7516	8.638	1.7647	1.2298	1.1339
Second run	--	1.2167	0.4722	**1.5167**	0.8750	0.5500	0.3222
Third run	--	**0.8400**	0.2333	--	0.5542	0.0917	0.0815
Fourth run	--	--	0.0333	--	**0.2421**	0.0057	0.0050

After finishing step 4, we obtain *H*= {3 6 7} and *O*= {1 4 2 5}. By this information we start the next step. In step 5 we satisfy the full coverage criterion. We start step 5 by knowing the following values for the coverage matrix: $c_{6,11} = c_{7,11} = 1$ and $c_{ij} = 0$ for all other i and j. This means that the condition $\sum_{i=1}^{m}\sum_{j=1}^{n} c_{lj} = 0$ is not satisfied. In fact zone z_{11} is not covered by any of the selected locations. In step 5.2 we calculate the values of *R(l)* for all the unselected locations, that is l_3, l_6, and l_7. We have $R(l_3) = 0$, $R(l_6) = \frac{1}{8}$, and $R(l_7) = \frac{1}{9}$. Therefore, in step 5.3 we select location l_6. After completing the steps 5.3.1 to 5.3.4 we go to step 5.1. Here the condition $\sum_{i=1}^{m}\sum_{j=1}^{n} c_{lj} = 0$ is satisfied, i.e., the selected set of locations covers all the zones. We go to step 6.

In step 6, we start with $O = \{l_1, l_2, l_4, l_5, l_6\}$. Since these locations do not have any sensor and have been added to set *O* in the last iteration of steps 4 and 5, we deploy them (by step 6.1). We then perform the learning in all the zones and collect the SS from all the deployed sensors. For this example the learning time is only 10 seconds. This time only includes the data collection time and not the time for positioning the tag in every zone for learning purposes. In this paper we have not considered an objective of reducing the time to position the learning tags in zones. We consider this problem as a future research issue and we refer to that in section VI. The deployment time is 90 seconds based on the deployment time of the selected locations. Every unit of deployment time is 10 seconds. Due to lack of space, these data are not shown here. After calculating the average SS based on actual data, we revise the score matrix (based on the rules in step 6.3.1). The revised matrix is shown in Table 8. Bold numbers show the revised cells.

Table 8: Revised tag zone-sensor location signal score matrix after the first iteration

D	z1	z2	z3	z4	z5	z6	z7	z8	z9	z10	z11	z12
l1	6	2	0	**1**	2	3	1	0	0	0	0	**1**
l2	0	2	5	4	2	2	0	1	0	**1**	0	1
l3	2	4	6	4	3	2	0	0	1	1	0	0
l4	**3**	6	1	2	2	**0**	1	0	1	0	0	0
l5	1	0	0	0	0	1	4	2	**0**	0	6	7
l6	0	0	0	1	0	0	3	8	3	**3**	4	2
l7	0	0	1	0	0	0	2	3	6	6	2	1

We repeat steps 2-5. The set *O* does not change and therefore we stop in step 7. Due to lack of space the actual

Table 9: Experiment results

Building Name	# of floors	# of zones	# of locations	#of selected locations	# of location algorithm iterations	# of zones covered	Learning time	Deployment time	Detection Precision
SEL	3	80	46	24	6	80	36 Sec.	320 Sec.	95%
ERF	2	113	75	29	8	112	51 Sec.	530 Sec.	96.4%
BH	2	15	8	3	2	15	16 Sec.	55 Sec.	100%

data are not shown here.

To calculate the precision of the detection algorithm, we collected another set of data by putting the tag in different zones and recording the SS reported from sensors. We observed 1200 data points, 100 observations per zone. We then calculated the network precision by dividing the number of correct location detection decisions based on the probability rules divided by 1200. Out of 1200 observed sensor network states, in 1176 cases the zone detected by the maximum probability rule is the actual zone. Therefore the precision is determined as $1176/1200 = 0.98$. We receive 24 observation points (network states) that are not among the available network states with a detection rule. This is due to the existence of some reported SS, during the detection phase, that are not available in the learning data. Next we report a summary of our experimental results and the performance of the algorithm for larger size cases.

We selected three different buildings in the University of Illinois at Chicago. Table 9 shows the outcome of our test results. The APs (sensors) used in our experiment were Linksys WAP11 Wireless-B Network Access Points, whose transmit power is 100mW (20dBm). The tag was HP iPAQ h2210 Mobile Media Companion PDA handheld, which is integrated with Wi-Fi 802.11b technology. HP iPAQ running application software MiniStumbler version 4.00 records signal strength values from each AP in a log file and sorted them from the strongest one to the weakest. Measured signal strength values are in the range of -92 dBm and -12 dBm, but values displayed in the software are signal strength values plus a 92 offset in order to obtain positive numbers. That yielded a SS from 0 to 80. Although the hardware used during these experiments was only experimental and not appropriate to be used for real applications (due to its size and number of tags) but we believe that the development of a hardware system that can satisfy the requirements of real world applications is not a research issue and is a matter of production. We use Matlab and C++ to implement the location and detection algorithms. As evident from Table 9, the algorithm performance for regular size buildings is very satisfactory. The detection precision is very good. The learning and deployment times are acceptable compared to the desirable system preparation time (less than 15 minutes). For ERF building only one zone could not be covered due to the limitation of the candidate locations for sensor deployment. In all other cases, all the zones are covered.

5. Conclusion and Future Research

In this paper we have presented a set of algorithms that are used for the identification and utilization of a sensory network. This network is used to detect the object zone in an indoor building. We divide the target problem into two parts. In the first part we select indoor/outdoor locations to deploy the sensors. In the second part we construct detection rules to detect the presence of the tag. By real world experiments we have showed the satisfactory performance of our method.

There are several future research problems that can be followed from here. One potential research direction is to include the learning tag deployment time as a design objective. The current version of algorithm does not consider this objective. When the number of zones increases the tag deployment time could increase significantly. Some zones can also include selected sensor locations. In this case by going to that zone, both the learning tag and the sensor can be deployed.

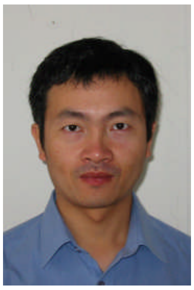
The other interesting problem is to consider the concept of class of locations in the location selection process. If a group of the selected locations are close and can be reached from each other, then the deployment time for that group is not necessarily the summation of the individual location deployment times. In fact the group deployment time could be the time to reach to that group from outside of the building and the time to cover all the members by moving from one to another.

The other research direction can be to identify the conditions for partial learning. By partial learning we mean not putting the learning tag in all zones. During our experiments we discover that the modification of matrix D after learning usually happens for extreme zones. By extreme zones we mean the zones that are not surrounded by other zones. Therefore, the problem here is to find the subset of zones such that by putting the learning tags in them, we can find the actual probability distributions for the zones without a learning tag.

References

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

[2] M. Vossiek, L. Wiebking, etc. "Wireless Local Positioning – Concepts, Solution, Application". *Microwave Magazine, IEEE*, Vol.4, No.4, Dec. 2003, pp. 77-86.
[3] S. Dhillon, "Sensor Placement for Effective Coverage and Surveillance in Distributed Sensor Networks". *IEEE Wireless Communications and Networking Conference (WCNC)*, 2003, pp. 1609-1614.
[4] A. Hills, "Large-scale wireless LAN design". *IEEE Communication Magazine*, Vol.39, 2001, pp. 98-107.
[5] 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, January 2004, pp. 17-19.
[6] S. Ray, R. Ungrangsi, etc., "Robust Location Detection in Emergency Sensor Networks". *Proceedings of the IEEE INFOCOM Conference*, 2003, pp. 1044-1053.
[7] L. Nagy and L. Farkas, "Indoor base station location optimization using genetic algorithm". *IEEE International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC)*, 2000, pp. 843-846.
[8] J. Hightower, Gaetano 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.
[9] Y. Gwon, R. Jain, and T. Kawahara, "Robust Indoor Location of Stationary and Mobile Users". *Proceedings of the IEEE INFOCOM Conference*, 2004, pp. 1032-1043.
[10] P. Prasithsangaree, etc, "On indoor position with wireless LANs". *IEEE International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC)*, 2002, pp. 720-724.
[11] R. Battiti, etc., Optimal wireless access point placement for location-dependent services". Technical Report DIT-03-052, Informatica e Telecomunicazioni, University of Trento. [Online]. Available: http://eprints.biblio.unitn.it/archive/00000489/
[12] Y. Chen and H. Kobayashi, "Signal strength based indoor geolocation". *IEEE International Conference on Communications (ICC)* , vol. 1, 2002, pp. 436-439.
[13] S. Manapure, H. Darabi, V. Patel and P. Banerjee, "A comparative study of radio frequency-based indoor location systems". *IEEE International Conference on Networking, Sensing and Control*, 2004, pp. 1265-1272.
[14] MeshNetworks, "Mesh Positioning Systems (MPS)". [Online], Available: http://www.meshnetworks.com/pages/technology.
[15] P. Krishnan, etc., "A System for LEASE: Location Estimation Assisted by Stationary Emitters for Indoor RF Wireless Networks". In *Proceedings of the IEEE INFOCOM Conference*, 2004, pp. 1001-1011.
[16] P. Castro, etc., "A Probabilistic Room Location Service for Wireless Networked Environments". *Proceedings of the 3rd international conference on Ubiquitous Computing*, 2001, pp. 18-34.
[17] A. Smailagic and S. Kogan, "Location Sensing and Privacy in a Context-Aware Computing Environment". *IEEE wireless Communications* [see also *IEEE Personal Communications*], vol. 9, no. 5, Oct. 2002, pp. 10-17.
[18] P. Bahl, etc., "Enhancements to the RADAR User Location and Tracking System". http://research.microsoft.com/~padmanab/papers/msr-tr-2000-12.pdf.
[19] A. Ladd, etc., "Robotics-Based Location Sensing using Wireless Ethernet". *ACM International Conference on Mobile Computing and Networking (MOBICOM)*, 2002, pp. 227-238.
[20] M. Youssef, A. Agrawala, "Handling Samples Correlation in the Horus System". In *Proceedings of the IEEE INFOCOM Conference*, 2004, pp. 1023-1031.



Hui Liu is a Ph.D. student at Department of Electrical and Computer Engineering, University of Illinois at Chicago. He received the B.S. degree from North China Electric Power University, Baoding, China, and his Master degree from Beijing University of Telecommunications and Posts, China. His research interests include communication, statistical signal processing, control and automation, and artificial intelligence.

Houshang Darabi is currently an Assistant Professor with the Department of Mechanical and Industrial Engineering, University of Illinois at Chicago. He received the Ph.D. degree in industrial and systems engineering from Rutgers University, New Brunswick, NJ, in 2000. His research interests include the application of discrete-event systems control theory in modeling and analysis of service and manufacturing systems, computer-integrated manufacturing, supply chain networks, and manufacturing information systems. His research has been supported by the National Science Foundation, the Department of Commerce, Motorola Inc., and several other agencies. He has published in different prestigious journals and has presented his research in national and international conferences. Dr. Darabi is a Senior Member of the Institute of Industrial Engineers (IIE), and a member of the Instrumentation, Systems and Automation Society (ISA) and the Institute for Operations Research and the Management Sciences (INFORMS).

Pat Banerjee is currently a Professor in the Department of Mechanical and Industrial Engineering and of Computer Science (by courtesy) at University of Illinois at Chicago (UIC). His current research interests include virtual reality applications; haptics applications; sensors, diagnostics and prognostics; immersive learning effectiveness and display interfaces; and linear and non-linear design optimization models. Professional education includes a MS (1987) and Ph.D. (1990) from Purdue University and a BTech (1984) from Indian Institute of Technology, Kanpur. In addition to over 100 publications in his area of research expertise, he has served as a department editor of IIE Transactions, as an associate editor of IEEE Transactions on Robotics and Automation. His textbook entitled Virtual Manufacturing was published by John Wiley in 2001. Banerjee is a Fellow of ASME and has served on the 2002 Research Visionary Board at Motorola Labs, and has received the ASME MED/MHED Best Technical Paper Award.