HAS: Hidden Anti-Theft System Based on Wireless Sensor Networks

Longjiang Guo[1,§], Jinsheng Duan[1], Jinbao Li[1], Lei Yu[2], and Haiying Shen[2]
[1]School of Computer Science and Technology, Heilongjiang University, Harbin, China 150080
[2]Department of Electrical and Computer Engineering, Clemson University, Clemson, SC, USA 29634
Email: {longjiangguo@gmail.com, jinshengduan@hotmail.com, leiy@clemson.edu, shenh@clemson.edu}

***Abstract*—Wireless sensor networks(WSNs) are being widely deployed for many monitoring applications, of which a popular one is anti-theft. However, current WSN technologies for anti-theft are either very susceptible to the environment interference or easily compromised by the thieves. Hence, they fail to achieve the desired effectiveness in many anti-theft scenarios. To address these limitations, this paper proposes a novel Hidden Anti-theft System (HAS) to monitor theft intrusion, which is based on the influences of theft intrusion on signal strength according to the shadowing effect in wireless communication. The theft intrusion is detected by finding abnormal RSSI samples of a wireless link compared with the stable range of signal strength in the link's normal state. Through the proposed monitoring approach, HAS can effectively detect intrusion while keeping invisible. In HAS system, to achieve load balance for detection task, we propose an efficient algorithm to determine the set of links that each node monitors. To reduce the response time, a dual-layer scanning solution with dual-radio nodes is proposed to scan the monitoring area more intensively. The experiment results show that the efficiency of HAS. HAS achieves very low false positive and false negative rates, and the response time of dual-layer scanning is 54.2% less than single layer scanning.**

***Index Terms*—Motion detection, Anti-theft technology, Shadowing effect.**

I. Introduction

A wireless sensor network (WSN) consists of a number of sensor nodes with limited capacity of storage, computation and communication. The sensor nodes measure the light [1], vibration, pressure and other physical quantities of the environment. The users or applications make critical decision with the information collected from the WSN. Recent advances in WSN technologies have lead to increasing deployment of WSNs in our daily life for various applications such as health monitoring, traffic control and anti-theft.

As an important application, WSNs for anti-theft significantly improves the effectiveness of traditional anti-theft ways such as using cameras, sensor door locks and infrared devices that have very limited detecting capability. A WSN detects possible intrusions by measuring distance changes [2] and sensing light [1] as well as vibration. However, the intrusion detection capability of WSNs is very susceptible to environment interference, which makes WSN only be applied in few scenarios such as indoor anti-theft. Besides, the sensor nodes can be easily found and destroyed by the thieves if they are deployed in visible places, and then the whole anti-theft system could be rendered ineffective.

In this paper, we propose a novel detecting approach which exploits the transmitting characteristics of wireless signals to make sensor nodes invisible while still effectively detecting intrusion. On one hand, received signal strength (RSS) will attenuate significantly when people or other objects move into the transmitting area. The degree of RSS change depends on the type of physical medium and transmitting distance, etc. Such fact indicates that the intrusion can be detected by monitoring RSS changes. On the other hand, wireless signal can penetrate indoor obstacle easily because of its high frequency, which indicates that the sensor nodes can be deployed in hidden places such as inside walls or furniture so that the entire system is invisible. In this way, a hidden anti-theft system, referred to as **HAS**, is proposed, in which sensor nodes are concealed in the places such as walls or furniture. HAS can protect the sensor nodes from being easily discovered and attacked by adversaries.

In HAS, sensor nodes detect possible intrusions of thieves by monitoring the RSS changes of wireless links which cross the areas needed to be monitored. Given these links, each sensor node is assigned to monitor a part of the links. To achieve load balance, an optimal algorithm named Poker-Dealing is proposed to determine the set of links that each node monitors such that the number of links monitored by every node is uniformly distributed. We also exploit dual-radio capability of sensor nodes and propose a dual-layer scanning scheme to reduce the response time of the system to intrusion activities (declines 54.2%) . The experiments on real test bed show that HAS achieves very low false positive (0.0083%) and false negative rates (0% so far), which indicates that it can be applied in the environment with high secure demand such as museum or open indoor area.

The rest of this paper is organized as follow. Related works are described in Section II. Section III describes preliminary knowledge and anti-theft models. Section IV describes HAS system in detail. SectionV introduces a novel scanning way with dual-radio sensor networks. SectionVI presents our experimental results. SectionVII discusses the application of HAS works on a large scale. SectionVIII concludes this paper.

[1]§To whom correspondences should be addressed. Email: longjiangguo@gmail.com

II. Related Work

Video camera is a commonly used anti-theft device, which can monitor fan-shaped spaces. Users observe the real-time situations of monitoring areas. Yoshiura et al. [3] created a community security system by organizing personal camera watching systems to a watch network system. Onoe et al. [4] proposed a system to monitor the area with omnidirectional video cameras to reduce the movement of cameras in a wider coverage area. Debaque et al. [5] investigated the coverage optimization problem of multiple cameras.

Besides video camera, other types of anti-theft devices have been deployed for theft intrusion detection and access control. Mao et al. [1] proposed to use light sensors to detect and track motions. Hwang et al. [6] proposed a novel wireless access monitoring and control system based on the digital door lock. The door will not be open until a correct password is entered or a digital key is inserted. Li [7] extracted human iris image to identify whether the image matches the user's in database. The iris recognition enhances security because of the uniqueness of a person's iris in the world. The above traditional anti-theft approaches have limited application scenarios.

WSNs can be used for intrusion detection and object tracking, and thus can be applied for anti-theft in wider areas. Cao et al. [2] developed a vehicle anti-theft system according to the relationship between communication distance and Received Signal Strength Indicator (RSSI) of the links. A base station is setup in a park and sensor nodes are deployed in cars. Cars login when they enter in park and logout when they leave. Any illegal driving-away action without logout can be detected by the base station. The park will notifies user and track the illegal driving with sensors outside. Woyach et al. [8] investigated sensing capability due to shadowing effect with MICA2 and MICAz motes. They found that RSSI changes if a device-free person enters into a transmitting area (usually Line-Of-Sight path). Human will be detected as role outside system. Lee et al. [9] analyzed RSSI with and without moving objects. Zhang et al. [10] proposed a scheme to track the object according to the influences of objects on the Non-LOS paths of wireless signals. Wilson and Palwari [11] developed a through-wall tracking system based on Woyach et al.'s findings [8] on the characteristics of 2.4GHz microwave.

III. Preliminary

This section briefly introduces RSSI and shadowing effect in wireless communication.

A. Wireless Communication and RSSI

In the process of wireless communication, the transmitter emits radio waves to space through antenna. The wireless signal expands in the space and forms a spatial sphere [12]. The receiver can sense the intensity of the electromagnetic signal as it hits the antenna. The value of the intensity is called Received Signal Strength Indicator (RSSI). RSSI is measured by the difference between transmitting power and fading energy in unit of dBm, given as follows [11]:

$$\text{RSS(dB)} = P - L - S - F - V \tag{1}$$

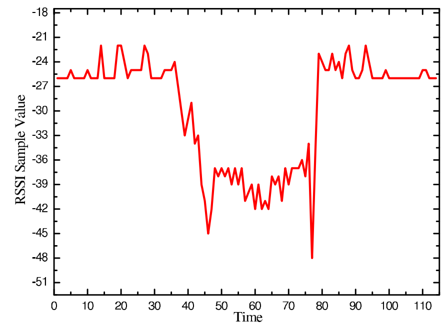


Fig. 1. Shadowing Effect. Wireless signal is absorbed by human body for 4 seconds through the wave trough.

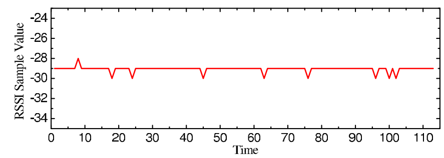


Fig. 2. RSSI sample value when a series of motions occur in the edge of the transmitting area.

where P denotes transmitting power, L is the static losses due to distance and device inconsistencies etc, S is the shadowing loss due to objects that attenuate the signal, F is the fading loss in multipath environments, and V is the measurement of noise. According to Equation (1), RSSI is the remaining energy of the signal after a series of attenuation.

B. Shadowing Effect

The shadowing effect appears when a physical medium enters the transmitting area of wireless signals. The obstacles between transmitter and receiver can attenuate signal power by absorbing, reflecting and scattering signals, and even can break the transmitting links in the worst case [13]. Figure 1 shows an example of shadowing effect given by our experiment on our real test-bed (Crossbow's TelosB TPR2400). In our experiments, a sensor node with ID #07 (sender) sends beacon packets to another node with ID #08 (receiver). The RSSI changes of node #08 over time is shown in Figure 1. A significant fluctuation in the figure is caused by a person staying on the Line of Sight (LoS) path between the sender and receiver for 4 seconds. In contrast, if obstacles only appear at the edge regions of the transmitting area, they have little influence to the RSSI of wireless signals. Figure 2 shows the measurement of RSSI at the receiver while only some motions occur at the edge regions of the transmitting area. HAS is based on the shadowing effect to find theft intrusions.

IV. Hidden Anti-Theft System

This section describes the details of HAS design.

A. System Overview

In HAS, the user decides the interested areas to be monitored and finds wireless links of the sensor network that cross these areas through a GUI (Graphical User Interface) program installed in the base station. The base station is a desktop PC. After the user specifies the links that should be monitored, the base station determines the set of links that each node monitors and sends the assignment information to all nodes. Then, each node starts to record the RSSI samples of each monitored link.

Definition 1: **Agent for link (A, B)**. An agent for link (A, B) is a node which is either A or B. If A is the agent for link (A, B), then the agent A receives packets from the node B, records and monitors the RSSI samples of the link, reports to base station when abnormity is detected. For a given link (A, B), there must be only one agent to work for link (A, B). One agent can monitor multiple links.

In HAS, every node is an agent for the links assigned to it. We define one scanning cycle as a period with length T_{cycle} of n time slot (t_{slot}), i.e., $T_{cycle} = n \cdot t_{slot}$. Each node broadcasts a beacon packet once in a scanning cycle, and receives the packet from another node on the link it monitors.

For example, in Figure 4, D could be the agent for link (D, E) and (D, F). D will analyze RSSI of the link, say (D, E), by receiving the packets from E. For the wireless link (D, A), if it is not chosen by the user, i.e., the area it crosses is not necessary to be monitored, D will drop the packet from A.

Every agent consists of three components: adapting process, time scheduling and abnormity handling. Adapting process is used to determine stable RSSI value in normal state of wireless links. Time scheduling ensures that agents communicate to each other with no collision. Abnormity handling decides whether intrusion detection exists according to abnormal RSSI samples. The framework of HAS is shown in Figure 3.

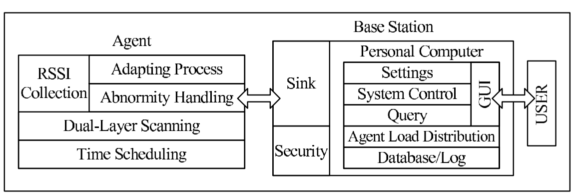


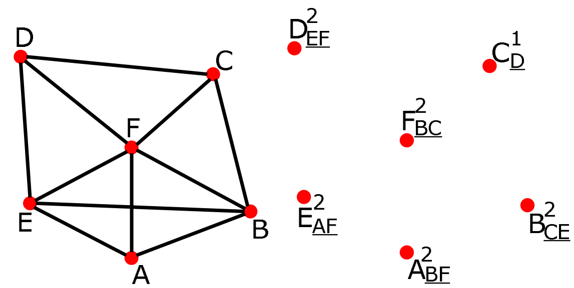
Fig. 3. The framework of HAS.

Once receiving settings from the base station, each agent starts adapting process. Sensor nodes send packets in the order of time scheduling. Adapting process lasts until every agent has collected M RSSI samples from each link which the agent monitors. Agents continue receiving beacon packets, and analyze each RSSI.

If A detects an abnormal link (A, B) where A is the agent, A sends alarm packet to the base station. The alarm packet contains information of link (A, B). Once the base station receives the alarm packet, the edge (A, B) will be colored by `RED` in the layout on screen. At the same time, alarm rings to notify users that there is a theft intrusion.

B. Agent Load Distribution

HAS requires that each link specified by users should be monitored. This section addresses the problem of how to assign links to agents while balancing the number of links



(a) (b)

Fig. 4. An Illustration of B-EPVP

that each agent monitors. We formulate this problem as the following Balanced Edges Partitioned by Vertices Problem (B-EPVP for short):

Definition 2: **B-EPVP**. Given an undirected graph $G = (V, E)$, where E is the edge set (links) specified by users and V is the vertex set, find a partition of edge set $\varepsilon = (\varepsilon_1, \cdots, \varepsilon_n)$ in G, where ε_i is an edge set which node v_i monitors, and

- $\sum_{i=1}^{|V|} (|\varepsilon_i| - \varphi)^2$ is minimized;
- $\bigcup_{i=1}^{n} \varepsilon_i = E$. For $\forall i \neq j$, $\varepsilon_i \cap \varepsilon_j = \varnothing$, $|\varepsilon_i| \leqslant |\varepsilon_j|$ if $d_i \leqslant d_j$;

where $\varphi = |E|/|V|$ that is the average number of edges that a vertex sets, and d_i is the degree of vertex i.

Obviously, $|\varepsilon_i| \leqslant d_i$, $\sum_{i=1}^{n} |\varepsilon_i| = m$. v_i is an endpoint of any edge in ε_i. An instance of B-EPVP is shown in Figure 4. Figure 4(a) is the input, an undirected graph G, and Figure 4(b) is the output. Take vertex $F^2_{B,C}$ for an example. The superscript of F is 2, it denotes that the number of edges monitored by F. The subscript is BC. It shows that the edges (F, C) and (F, B) are monitored by F. In order to solve B-EPVP, a Poker-Dealing (PD for short) algorithm is proposed below. The input of PD algorithm is $G = (V, E)$, and the output is $\varepsilon = (\varepsilon_1, \cdots, \varepsilon_n)$.

Data Structure. In PD algorithm, an important data structure of adjacent list T is established for the graph with the links decided by the users. Figure 5(a) shows an instance of adjacent list T for the graph in Figure 4(a). T has 3 columns, which are v_i, D_i and ε_i. Here, v_i denotes the vertices in $G = (V, E)$. D_i is the neighbor vertices of v_i, ε_i has already been defined in Definition 2. In ε_i, there are several subcolumns, denoted as $\varepsilon_i[1]$, $\varepsilon_i[2]$, and so on. $\varepsilon_i[j]$ denotes the j^{th} element in ε_i. For example, in Figure 5(a), vertex F ($v_i = F$) stays in the first row. The row corresponding to F is denoted by R_F, namely, $R_F = \{F|D_F|\varepsilon_F\}$, $D_F = \{A^3B^1C^4D^5E^2\}$, $\varepsilon_F = \varnothing$. D_F is a neighbor set of vertex F. $A \in D_F$, it denotes that $(A, F) \in E$. $|D_F|$ is the number of neighbors of F, i.e., the current degree of F. The superscript of A^3 is the distance between R_F and R_A by sorting T according to $|D_i|$ in descending order. NR_F is denoted as the number of the row, where R_F stays in. For example, in Figure 5(a), R_F stays in the first row, $NR_F = 1$, R_A stays in the fourth row, $NR_A = 4$, $dist(R_F, R_A)$ denotes the distance between R_F and R_A, $dist(R_F, R_A) = NR_A - NR_F = 3$. Obviously, we have $dist(R_A, R_F) = -dist(R_F, R_A)$, and $dist(R_A, R_A) = 0$.

Initialization. For the input $G = (V, E)$, all vertices are put into the first column. Each D_i is initialized with v_i's neighbors in G, where $v_i \in V$. $\varepsilon_i = \varnothing$. T is sorted according to $|D_i|$ in descending order.

PD Algorithm. The process of PD algorithm can be regarded as that $|V|$ people paly $|E|$ cards in a poker game. $|E|$ cards will be distributed to $|V|$ people in a certain rule. The process of PD algorithm simulates the poker distribution. There are two stages in PD algorithm. **IF** $|E| \geqslant |V|$, **THEN** PD algorithm goes into stage 1 and stage 2 orderly; **ELSE** it goes into stage 2.

Stage 1: There are some rounds. In each round, there are $|V|$ iterations. For j^{th} round, in each iteration, PD algorithm sorts T according to $|D_i|$ in descending order, and finds v_i such that $\varepsilon_i[j] = null$ and $|D_i|$ is maximum. If $\exists M \in D_i$ such that $dist(R_{v_i}, R_M) > 0$, then PD selects vertex M which has smallest positive distance; If $\forall M \in D_i$ such that $dist(R_{v_i}, R_M) \leqslant 0$, then PD selects vertex M which has greatest negative distance. Then, PD puts (v_i, M) into $\varepsilon_i[j]$, deletes M and v_i from D_i and D_M respectively, i.e., removes the edge (v_i, M) from E. Above process is repeated until $|E| < |V|$.

Example for stage 1. Figure 5(b)~(g) is stage 1. There are only one round ($j = 1$), six iterations in this round. Figure 5(b) is iteration 1. $|D_F| = 5$ is maximum and $\varepsilon_F[1] = null$, distance in $D_F = \{A^3B^1C^4D^5E^2\}$ are all positive, PD selects B from D_F, since $dist(R_F, R_B) = 1$ is minimum. PD puts (F, B) into $\varepsilon_F[1]$. B and F will be deleted from D_F and D_B respectively, i.e., the edge (F, B) is removed from E. Then, PD sorts T according to $|D_i|$ in descending order.

Figure 5(c) is iteration 2. $|D_E| = 4$ is maximum and $\varepsilon_E[1] = null$ (since $\varepsilon_F[1] = (F, B)$), distance in $D_E = \{A^2B^3D^5F^1\}$ are all positive, PD selects F from D_E, since $dist(R_E, R_F) = 1$ is minimum. PD puts (E, F) into $\varepsilon_E[1]$. F and E will be deleted from D_E and D_F respectively, i.e., the edge (E, F) is removed from E. This process repeats until $|E| < |V|$. In Figure 5(g), $|E| = 5$ and $|V| = 6$, stage 1 ends.

Stage 2: Without loss of generality, suppose that stage 1 finishes at j^{th} round. Once stage 1 finishes, PD enters into stage 2. There are also some rounds in this stage. PD enters a new round, $j \leftarrow j + 1$.

First, PD sorts T according to initialized $|D_i|$ in descending order. T will be divided into several parts. Each part is a subset of adjacent rows in T. In the same part, for $\forall v_i$, the initialized $|D_i|$ are all the same. For example, in Figure 5(h), $\{R_F\}$, $\{R_E, R_B\}$, $\{R_A, R_C, R_D\}$ are three different parts. In the first part, there is only one row R_F, the initialized $|D_F|$ is the greatest one in T. In the second part, there are two rows R_E and R_B, the initialized $|D_E|$ and $|D_B|$ are same.

Then, PD sorts each part according to current $|D_i|$ in descending order. After that, PD finds the first v_i such that $\varepsilon_i[j] = null$ in T, and select a vertex M from D_i according to the following two conditions:

- If $|E| = 1$, select M from D_i where current $|D_M|$ is maximum in T.
- If $|E| > 1$, select M in D_{v_i} randomly where R_M is in the last part.

At last, PD puts (v_i, M) into $\varepsilon_i[j]$, deletes M and v_i from D_i and D_M respectively, i.e., removes the edge (v_i, M) from E. Above process is repeated until $|E| = 0$.

Example for stage 2. In the end of stage 1, $|E| = 5$ and $n = 6$. PD enters stage 2. Figure5(h)~(m) is stage 2. PD starts from the second round, i.e., $j = 2$. After sorting T, T is divided into 3 parts, in Figure5(h), $\{R_F\}$, $\{R_E, R_B\}$, $\{R_A, R_C, R_D\}$ are three different parts. In (h), R_F is the first row in T, and $\varepsilon_F[2] = null$. Since $|E| = 5 > 1$, and D_F=$\{A, C\}$, A and C are all in the last part, randomly selects C in D_F where R_C is in the last part. PD puts (F, C) into $\varepsilon_F[2]$, delete C and F from D_F and D_C respectively, i.e., remove the edge (F, C) from E. Sort each part in descending order according to current $|D_i|$.

In Figure5(i), R_E is the first row such that $\varepsilon_E[2] = null$. PD randomly chooses A from D_E(A and D in D_E are all in last part). PD repeats above process until $|E| = 0$. In the end (Figure5(m)), PD stops when $|E| = 0$.

Theorem 1: The solution of PD algorithm satisfies:

- $\sum_{i=1}^{|V|} (|\varepsilon_i| - \varphi)^2$ is minimized;
- For $\forall i \neq j$, $|\varepsilon_i| \leqslant |\varepsilon_j|$ if $d_i \leqslant d_j$

Proof: We will prove that the output ε_i satisfies what the B-EPVP requires.

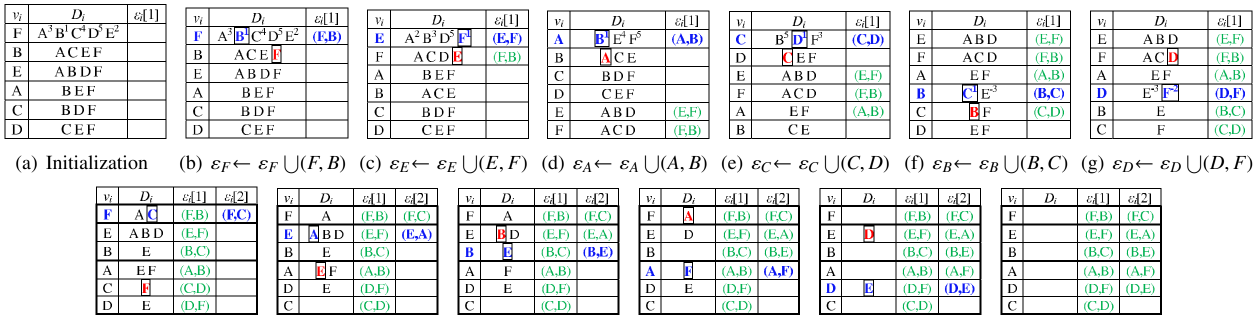
- When (v_i, M) is put into $\varepsilon_i[j]$, one affiliated relationship (v_i, M) forms (we call that assignment), and 1 edges are deleted from G. There are $|E|$ affiliated relationships during the algorithm.
- $|\varepsilon_i| \leqslant d_i$ is obvious. When $|D_i| = 0$, ε_i can't get any vertex.
- Each ε_i can get one edge as possible in every scan. If a vertex v_i whose D_i is very small, ε_i may get no edge in later scan. In Eq.2, we can unwrap $\sum_{i=1}^{n} (|\varepsilon_i| - \varphi)^2$. To minimize $\sum_{i=1}^{n} (|\varepsilon_i| - \varphi)^2$ is to minimize $\sum_{i=1}^{n} |\varepsilon_i|^2$. We know, when $x = y$, $(x+2)^2 + (y+0)^2 > (x+1)^2 + (y+1)^2$. $\sum_{i=1}^{n} |\varepsilon_i|^2$ is like that. If we put edges into ε_i in Poker-Dealing way, the increasement of $\sum_{i=1}^{n} |\varepsilon_i|^2$ is slower than any other ways (such as $(x + 2)^2 + (y + 0)^2$). So we can get $\min \sum_{i=1}^{n} (|\varepsilon_i| - \varphi)^2$.

$$\sum_{i=1}^{n} (|\varepsilon_i| - \varphi)^2 = |\varepsilon_1|^2 + \cdots + |\varepsilon_n|^2 - 2m\varphi + n\varphi^2 \quad (2)$$

- If a vertex has very small degree, it can get vertices as many as possible via Poker-Dealing to approximate φ. In the last iteration of the 2nd stage, the v_i on the top have more priority to get vertex to ensure the second requirement in definition of B-EPVP. Because in the last iteration, the v_i on the top have greater original degree. Therefore, for $\forall i \neq j$,$|\varepsilon_i| \leqslant |\varepsilon_j|$.
- In stage 1, the selecting of M can be the minimum influence to the lower v_i. The lower v_i is more possible to get vertex in i-th iteration, rather than gets nothing. □

C. Time Scheduling

Since HAS are designed for the indoor environment and the communication radius between nodes can be up to 100m or more, the nodes communicate with each other in one hop (MultiHop will be discussed in Section.VII), and all nodes in system are considered to be in the same collision area. Thus, in HAS, time scheduling is introduced to ensure that there is at most one node sending beacon packet in a time slot.



(a) Initialization (b) $\varepsilon_F \leftarrow \varepsilon_F \bigcup (F,B)$ (c) $\varepsilon_E \leftarrow \varepsilon_E \bigcup (E,F)$ (d) $\varepsilon_A \leftarrow \varepsilon_A \bigcup (A,B)$ (e) $\varepsilon_C \leftarrow \varepsilon_C \bigcup (C,D)$ (f) $\varepsilon_B \leftarrow \varepsilon_B \bigcup (B,C)$ (g) $\varepsilon_D \leftarrow \varepsilon_D \bigcup (D,F)$

(h) $\varepsilon_F \leftarrow \varepsilon_F \bigcup (F,C)$ (i) $\varepsilon_E \leftarrow \varepsilon_E \bigcup (E,A)$ (j) $\varepsilon_B \leftarrow \varepsilon_B \bigcup (B,E)$ (k) $\varepsilon_A \leftarrow \varepsilon_A \bigcup (A,F)$ (l) $\varepsilon_D \leftarrow \varepsilon_D \bigcup (D,E)$ (m) $T = \varnothing$

Fig. 5. An Example of Poker-Dealing Algorithm. Stage 1 is from (b) to (g), and iteration 1~6 respectively. (b)~(g) are in round 1 of PD. Stage 2 is from (h) to (l), and iteration 1~5 respectively. (h)~(l) are in round 2 of PD.

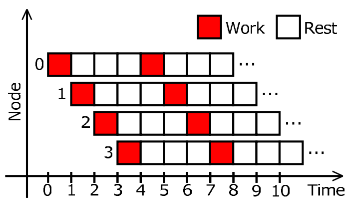


Fig. 6. Scheduling of Nodes. The dark block is time slot node sends packets, and the blank one is listening time slot. As system starts in order of 0, 1, 2, 3, no two nodes broadcast packets in one time slot.

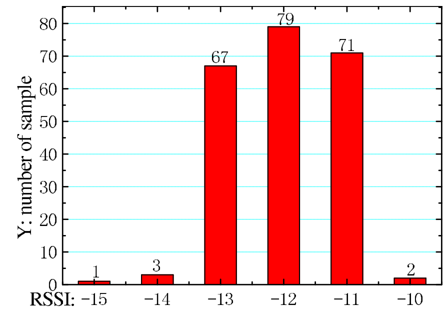


Fig. 7. Histogram of adapting process. This example shows 6 kinds of smaples and M = 223 for a particular link. Here α=-15, β=-10.

Given a list of n nodes in the system, and $n \times t_{slot}$ in one scanning cycle, here t_{slot} is the duration of a time slot. The i-th node broadcasts packet at time slot $i+\theta\times n$, where θ is the cycle amount since the system runs, and it starts from 0. In other time slots, an agent listens on the channel and receives packets from the link it monitors. Figure.6 shows an example of nodes scheduling.

D. Abnormity Detection

HAS detects the intrusion of thieves by discovering the changes of link RSSI due to shadowing effect. When a node received a packet from its monitored link, it measures RSSI of the link and checks whether there is any anomaly, i.e., significant changes. Here, we define two link states according to the value of RSSI.

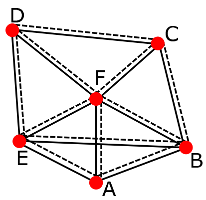
Definition 3: **Link State**. Link state is classified into *normal state* and *abnormal state*. We describe the normal state of link as a triple (α, β, δ), which means that RSSI of a link in normal state should satisfy $\alpha - \delta \leqslant$RSSI$\leqslant \beta + \delta$. If RSSI is beyond the range, the link is abnormal. Here α and β denote lower and upper bound of RSSI of normal state respectively, and δ denotes the variance tolerance.

To recognize the range of RSSI of a link in the normal state, the system performs an adapting process at the beginning in a static environment where nobody in monitoring area and nodes are settled at a fixed place. In the adapting process, every node collects RSSI samples and establishes a histogram for each link it monitors. In the histogram (in Figure 7), the x-axis represents sample values, and the y-axis is the amount of the samples. When a new RSSI is obtained, it is put into the corresponding bin where its value is. The adapting process for a link ends when the amount of samples reaches M. Then, α is set to the minimum value and β is set to the maximum value among all M samples. The mean amplitude of RSSI caused by human motion we measured is about 15~20dBm. Therefore, $(\beta + \delta) - (\alpha - \delta)$ should be less than 15 otherwise some human motions might not be detected. Therefore, we choose δ by $\delta < (\alpha - \beta + 15)/2$.

E. Abnormity Processing

After we obtain (α, β, δ) for a normal link, any RSSI sample not in that range would be considered as abnormal sample. When an agent need to alarm, it will broadcast the beacon packet including alarm information at its time slot. If an agent finds one abnormal sample, it sends alarm packet immediately, the system is too sensitive, and results performs in a high false positive. If the agent alarms according two consecutive abnormal samples, sensitivity decreases greatly, and the response time increases. Therefore, there is a trade-off between sensitivity and response time.

A beacon packet broadcasted by an agent contains the number of abnormal links. Every agent accumulates the number of abnormal links informed in beacon packets from other nodes in a scanning cycle. The total number is denoted as om. For a monitored link i, its agent holds a flag pre_i which denotes link i is normal ($pre_i = 0$) or abnormal ($pre_i = 1$) in previous cycle. When current agent receives a new RSSI sample: (1)if



(a) Dual-radio Network

Time Slot	Ch 1	Ch 2
1	D	B
2	C	A
3	E	C
4	A	D
5	F	E
6	B	F

(b) Schedule

Fig. 8. Dual-Radio Scanning. In (a) real line means channel1 and dash line denotes channel2. Real line and dash line can exits in one time slot.

$pre_i = 0$ and RSSI is normal, DO NOTHING; (2)if $pre_i = 0$ and RSSI is abnormal, $pre_i = 1$, DO NOTHING; (3)if $pre_i = 1$ and RSSI is abnormal, ALARM; (4)if $pre_i = 1$ and RSSI is normal, ALARM with probability $p = \frac{om}{m}$.

V. Dual-Layer Scanning

A. Problem Description

Currently, multi-radio sensor networks have attracted a great deal of attention, in which the sensor nodes contain two or more radios. With multi-radio hardware, two different sensors can broadcast at different channel in the same time slot. Once two non-adjacent nodes in G broadcast in two different channels simultaneously, more links can be monitored. For example, in Figure 8(a), if a single-radio sensor network is adopted, according to agent load distribution illustrated in Figure 4(b), if node D broadcasts a packet, node C can receive the packet, and only one link (C, D) is monitored by node C in that time slot. If a dual-radio sensor network is adopted, D and B can broadcast packets in two different channels simultaneously, links (C, D), (B, F) and (B, A) are monitored at the same time by nodes C, F and A respectively.

By adopting dual-radio sensor networks, it can achieve following benefits: (1) more links can be monitored in one time slot, i.e. scanning monitoring area more intensively; (2) response time can be reduced; (3) a possible decrease of false negative. In this section, dual-scanning by dual-radio sensors in HAS is proposed.

A challenge facing such dual-layer scanning is how to find a schedule of n time slots, and decide which two nodes broadcast in the i-th time slot simultaneously. For balanced scanning, we expect to see that each node broadcasts in both channels only once within a scanning cycle, and no node sends on both channels in one time slot. To find a schedule in dual-layer scanning, we introduce Unrelated Two Vertices Sequence Problem (U2VSP for short) as follows.

Definition 4: **Unrelated Value** $\mu(v_i, v_j)$**.** For a given undirected graph $G = (V, E)$, $v_i, v_j \in V$, $\mu(v_i, v_j) = 1$ iff $(v_i, v_j) \notin E$; $\mu(v_i, v_j) = 0$, iff $(v_i, v_j) \in E$.

Definition 5: **Unrelated 2 Vertices Sequence Problem.** For a given undirected graph $G = (V, E)$, find a sequence $S = \{ (v_1, v'_1), \cdots, (v_n, v'_n) \}$ with length $n = |V|$, for $\forall i$, $v_i, v'_i \in V$, where S satisfies:

- $\bigcup v_i = \bigcup v'_i = V$, for $\forall i \neq j$, $v_i \neq v_j, v'_i \neq v'_j, v_i \neq v'_i$.
- $\max \sum_{i=1}^{n} \mu(v_i, v'_i)$.

Every element (v_i, v'_i) in S expresses the two broadcasting sensors at two different channel in i-th time slot. The scanning edges are not overlapped when $\mu(v_i, v'_i) = 1$. Therefore, we can solve U2VSP to make $\sum_{i=1}^{n} \mu(v_i, v'_i)$ to be maximum. In Figure 8(b), $(v_2, v'_2) = (E, C)$ denotes that E and C broadcast at channel 1 and 2 respectively in the 2nd time slot. Here, $\sum_{i=1}^{n} \mu(v_i, v'_i) = 4$ gets maximum. To solve U2VSP, Sequence Generator algorithm(SG for short) is proposed.

B. SG Algorithm

Data Structure. In SG algorithm, an important data structure of adjacent list T is established for the graph $G = (V, E)$ which is similarly defined in SectionIV-B. Figure 5(a) shows an instance of adjacent list T for the graph in Figure 4(a). T has 3 columns, which are v_i, D^*_i and D_i. v_i and D_i are already defined in Section IV-B. Element in D^*_i is in the form of M^*. $M^* \in D^*_i$ iff $(v_i, M) \notin E$, i.e., $D^*_i = \{M^* | (v_i, M) \notin E\}$. For a particular M^*, N_{M^*} is the amount of M^* in T. For example, in Figure 9(a), $N_{C^*} = 2$. In Figure 9(a), vertex A ($v_i = A$) stays in the first row. The row corresponding to A is denoted by R_A, namely, $R_A = \{A | D^*_A | D_A\}$, $D^*_A = \{C^*, D^*\}$, where $(A, C) \notin E$ and $(A, D) \notin E$. $D_A = \{B, E, F\}$.

Initialization. For the input $G = (V, E)$, put all vertices into the first column. Fill each D_i with v_i's neighbors in G, fill each D^*_i with v_i's non-neighbors in G, where $v_i \in V$.

Algorithm. There are two stages in SG algorithm. **IF** $\exists v_i, |D^*_i| > 0$, **THEN** SG goes into stage 1 and stage 2 orderly; **ELSE** SG goes into stage 2.

Stage 1. There are some iterations in stage 1.

In each iteration, if $\exists v_i \in V$, such that $|D_i \bigcup D^*_i| = 1$, SG calls procedure **Making-Pair**(v_i, M), where M is exclusive element in $D_i \bigcup D^*_i$. **Making-Pair**(v_i,M) is a subfunction. In the procedure of **Making-Pair**(v_i,M), R_{v_i} will be deleted from T, for $\forall J \in V$, M and M^* will be removed from D_J and D^*_J respectively, if $M \in D_J$ or $M^* \in D^*_J$.

If $\forall v_i \in V$, such that $|D_i \bigcup D^*_i| > 1$, then

- Select the element M in T, where N_{M^*} is minimum. If there multiple elements are minimum, SG randomly chooses one.
- Construct a row set $Q = \{R_K | M^* \in D^*_K\}$, i.e. Q is the set of rows where M^* appears.
- Select R_K, where $|D^*_K|$ is the minimum in Q.
- SG calls **Making-Pair**(K,M).

Above process is repeated until all D^*_i are empty

Example for stage 1. Figure 9(a) is the initialization of the adjacent list of G in Figure 8(a). In Figure 9(a), $\forall v_i \in V$, such that $|D_i \bigcup D^*_i| > 1$. Since N_{E^*}=N_{B^*}=1, they are all minimum. Assume SG randomly chooses B^*. $B^* \in D^*_D$, Q=$\{R_D\}$. In Q, there is only one element R_D. SG selects R_D, and calls **Making-Pair**(D, B). In Figure 9(b), the element in the pane is to be removed. R_D, B and B^* are removed from T. In Figure 9(c), since N_{A^*}=N_{E^*}=1, they are all minimum, assume SG randomly chooses A^*. $A^* \in D^*_C$, Q=$\{R_C\}$. SG calls **Making-Pair**(C,A). R_C, A and A^* are removed from T. The process repeats until all D^*_i are empty, and the stage 1 ends as shown in Figure 9(e).

Stage 2. Once, all D_i^* are empty, i.e. $\forall v_i \in V$, $|D_i^*| = 0$, SG goes into stage 2. While $T \neq \varnothing$, SG executes as follows:

- Select K, where current $|D_K|$ is the minimum in T.
- Select element M in D_K randomly.
- Call **Making-Pair**(K,M).

The above process repeats until T is empty.

Example for stage 2. In Figure 9(f), all D_i^* are empty, SG goes into stage 2. Since $T \neq \varnothing$, First, SG selects F, where $|D_F|$=1, it is the minimum in T. Next, SG selects element E in D_F since there is only one element in D_F. Then, SG calls **Making-Pair**(F,E). R_F and E are removed from T. The process repeats until T is empty as shown in Figure 9(h).

The final schedule is S={(D,B),(C,A),(E,C),(A,D),(F,E),(B,F)} $\mu(D,B)$=1, $\mu(C,A)$=1, $\mu(E,C)$=1, $\mu(A,D)$=1, $\mu(F,E)$=0, $\mu(B,F)$=0. $\sum_{i=1}^{n} \mu(v_i, v_i')$=4.

Theorem 2: For a given undirected graph $G = (V, E)$, SG algorithm finds a sequence $S = \{ (v_1, v_1'), \cdots, (v_n, v_n') \}$ with length $n = |V|$, for $\forall i$, $v_i, v_i' \in V$, where S satisfies:

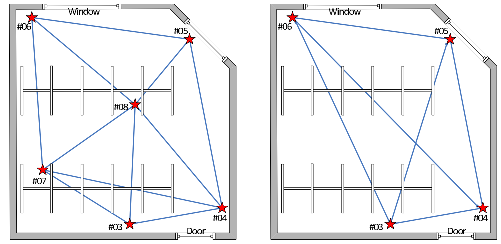
- $\bigcup v_i = \bigcup v_i' = V$, for $\forall i \neq j$, $v_i \neq v_j, v_i' \neq v_j', v_i \neq v_i'$.
- $\max \sum_{i=1}^{n} \mu(v_i, v_i')$.(Due to space, the proof is omitted.)

VI. Experimental Evaluation

We conducted experiments on real testbeds (Crossbow's TelosB TPR2400), shown in Figure 10. Our deployment uses TelosB motes (shown in red circle). In our testbed, every two sensors are bound together on a fixed stick with height of 0.9m. Program on motes is developed in Ubuntu10.10 + TinyOS2.x. Layouts are shown in Figure 11 such that the monitoring area can cover the whole room (the room area is $7.25 \times 7.45m^2$). The edges are selected by users. The base station distributes agents and assigns the edges to agents, and also computes dual-channel sequence with PD and SG algorithm. Then, the configuration information and parameters are sent to all agents.

Fig. 10. Testbed: Key Laboratory of Database and Parallel Computing, College of Heilongjiang Province, Room 329, in Heilongjiang University.

The theft enters in monitoring area at t_1, system alarms at t_2. Response time is $t_2 - t_1$. It is an important performance metric of any anti-theft system to reflect the system efficiency. False positive means the rate that system alarms in static environment. False negative means the rate theft enters the monitoring area without alarm. This two metrics are used to evaluate the detection accuracy of HAS.



(a) Layout1: 6 nodes (b) Layout2: 4 nodes

Fig. 11. System layout. Supposed that the layouts are optimized. The stars mean nodes and lines mean links.

Response Time. In single-layer scanning, Figure 12 shows the response time with object speed 1m/s and 2m/s in layout1 and layout2 respectively. Human moving speed is within a range of $0 \leqslant v \leqslant 10m/s$. Given a layout, response time is less for a higher object moving speed with fixed time slot duration, and is higher for a greater time slot with fixed object moving speed. False negative hasn't been found yet.

In dual-layer scanning, Figure 14 compare the response time with single-layer versus dual-layer scanning in layout1 with $t_{slot} = 100ms$, $M = 30$, where M is the number of samples in the adapting process. Dual-layer scanning method performs much better than single-layer system. The response time of dual-layer declines 54.2%. False negative rate is also zero.

False Positive and False Negative Rate. We enters monitoring area for 379 times with layout1. HAS scans about 36000 times in one hour, with $t_{slot} = 100ms$ and $M = 30$, by dual-layer scanning. The results show no false negative. False positive rate is 0.0083%. Adapting process parameter M and environment can influence false positive. In Figure 13, false positive declines extremely when M increases. Obviously, when M is greater, adapting process can get greater β and smaller α with a higher probability, which enlarges normal status range.

The B-EPVP Evaluation. Solution of PD algorithm is more balanced than any other way. This paper evaluates the result of TD (Total Deviation) and DR (Distribution Range). TD=$\sum_{i=1}^{n}(|\varepsilon_i| - \varphi)^2$, DR=$\max\frac{|\varepsilon_i|}{d_i} - \min\frac{|\varepsilon_i|}{d_i}$. In Figure 15(a), x-axis means topologies (7 random graphes, each graph contains 16 vertices), y-axis denotes TD. The result of PD is obviously better than Random Distribution. In every input, TD of PD is much lower than Random way. In (b) y-axis is distribution rate ($\frac{|\varepsilon_i|}{d_i}$). DRs of PD are all narrower than Random way. This evaluation shows that PD algorithm can obtain balanced result.

VII. Discussion on Multi-hop and Big-range Network

When HAS is applied to a whole floor of a building, the wireless signal becomes weak after goes through wall. Therefore, multi-hop is required. Figure 16 shows a possible solution. Figure 16(a) describes area (rooms) distribution by a graph model. Nodes in G denotes rooms with subnetworks or clusters in each room. Some agents and a sink (cluster-head) are deployed in one room. All the cluster-heads communicate

(a) Initialization

v_i	D_i^*	D_i
A	C*D*	BEF
B	D*	ACEF
C	A*E*	BDF
D	A*B*	CEF
E	C*	ABDF
F		ABCDE

(b) (D,B)

v_i	D_i^*	D_i
A	C*D*	$\boxed{B}$EF
B	D*	ACEF
C	A*E*	$\boxed{B}$DF
$\boxed{D}$	A*$\boxed{B*}$	CEF
E	C*	A$\boxed{B}$DF
F		A$\boxed{B}$CDE

(c) (C,A)

v_i	D_i^*	D_i
A	C*D*	EF
B	D*	$\boxed{A}$CEF
$\boxed{C}$	$\boxed{A*}$E*	DF
E	C*	$\boxed{A}$DF
F		$\boxed{A}$CDE

(d) (E,C)

v_i	D_i^*	D_i
A	$\boxed{C*}$D*	EF
B	D*	$\boxed{C}$EF
$\boxed{E}$	$\boxed{C*}$	DF
F		$\boxed{C}$DE

(e) (A,D)

v_i	D_i^*	D_i
$\boxed{A}$	$\boxed{D*}$	EF
B	$\boxed{D*}$	EF
F		$\boxed{D}$E

(f) (F,E)

v_i	D_i^*	D_i
B		$\boxed{E}$F
$\boxed{F}$		$\boxed{E}$

(g) (B,F)

v_i	D_i^*	D_i
$\boxed{B}$		$\boxed{F}$

(h) $T = \varnothing$

v_i	D_i^*	D_i

Fig. 9. An example of SG Algorithm

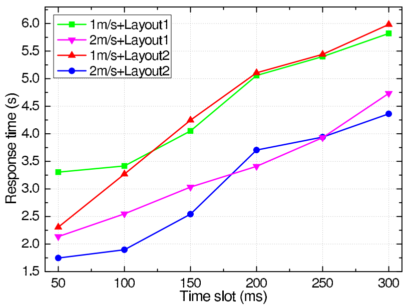


Fig. 12. Time Slot v.s. Response Time

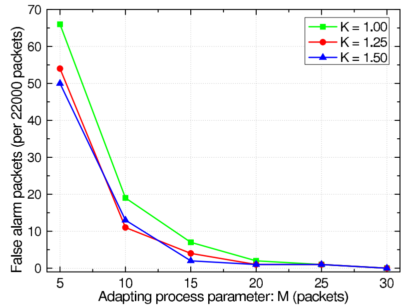


Fig. 13. Parameter M v.s. False Positive

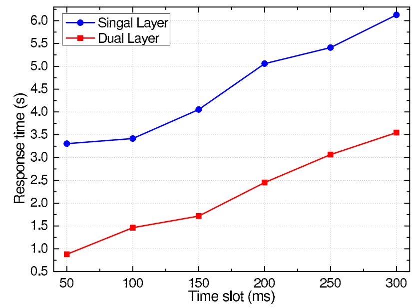
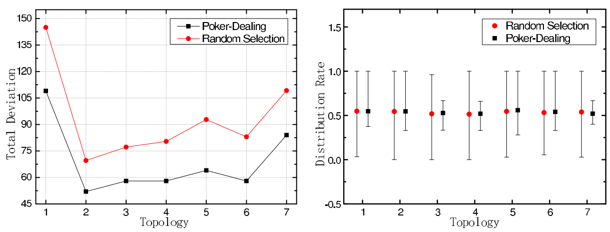
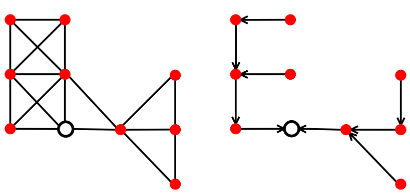


Fig. 14. Single-Layer v.s. Dual-Layer



(a) Total Deviation (b) Distribution Range

Fig. 15. Evaluation of PD Algorithm.



(a) Room Distribution (b) Spanning Tree

Fig. 16. A Multi-hop Network and Its Routing Scheme.

in the same channel. The channel all the agents communicate in room A should be different from the adjacent rooms.

A PC is installed in control center (the blank node). Figure 16(b) is the routing scheme determined by a spanning tree of G. The root of the spanning tree is the blank node. If an agent broadcasts alarm, it changes the channel which its cluster-head use. All the cluster-heads works on a same channel (including base station). The cluster-head sends the alarm packets to the control center by some routing scheme.

VIII. Conclusion

Traditional anti-theft technologies exist many limitations. HAS proposed in this paper can monitor more effectively. The system can be deployed in hidden place to avoid exposing to the adversaries. Monitoring task can be balanced distributed to agents by PD algorithm. Response time of Dual-Layer scanning is 54.2% less than single-layer scanning in average. No false negative is found. False positive is inversely proportional to M. when $M = 30$, false positive is nearly zero (0.0083%).

Acknowledgment

This work is supported by Program for New Century Excellent Talents in University under grant No.NCET-11-0955; Programs Foundation of Heilongjiang Educational Committee for New Century Excellent Talents in University under grant No.1252-NCET-011; Program for Group of Science and Technology Innovation of Heilongjiang Educational Committee under grant No.2011PYTD002; The National Natural Science Foundation of China under grant No.61033015, 60803015, 61070193; The Science and Technology Research of Heilongjiang Educational Committee under grant No.12511395; The Science and Technology Innovation Research Project of Harbin for Young Scholar under grant No.2008RFQXG107; Heilongjiang Province Founds for Distinguished Young Scientists under Grant No.JC201104; Heilongjiang Province Science and Technique Foundation under Grant No.GC09A109; This research was supported in part by U.S. NSF grants CNS-1249603, CNS-1049947, CNS-1156875, CNS-0917056 and CNS-1057530, CNS-1025652, CNS-0938189, CSR-2008826, CSR-2008827, Microsoft Research Faculty Fellowship 8300751, and U.S. Department of Energy's Oak Ridge National Laboratory including the Extreme Scale Systems Center located at ORNL and DoD 4000111689.

References

[1] X. Mao, S. Tang, X. Xu et al. iLight: Indoor device-free passive tracking using wireless sensor networks, *IEEE INFOCOM*, 2011: 281-285.

[2] H. Song, S. Zhu, G. Cao. SVATS: A Sensor-Network-Based Vehicle Anti-Theft System, *IEEE INFOCOM*, 2008: 2128-2136.

[3] N. Yoshiura, Y. Fujii and N. Ohta. Using the Security Camera System Based on Individually Maintained Computers for Homeland Security: The e-JIKEI Network Project, *IEEE IMTC*, 2005: 101-105.

[4] Y. Onoe, N. Yokoya, K. Yamazawa, and H. Takemura. Visual surveillance and monitoring system using an omnidirectional video camera, International Conference on Pattern Recognition, *IEEE ICPR*, 1998: 588-592.

[5] B. Debaque, R. Jedidi and D. Prevost. Optimal video camera network deployment to support security monitoring. Information Fusion, 2009: 1730-1736.

[6] I. Hwang, J. Baek. Wireless Access Monitoring and Control System based on Digital Door Lock, *IEEE Trans. Consumer Electronics*, 2007.

[7] P. Li, X. Liu. An incremental method for accurate iris segmentation, International Conference on Pattern Recognition, *IEEE ICPR*, 2008: 1-4.

[8] K. Woyach, D. Puccinelli and M. Haenggi. Sensorless Sensing in Wireless Networks: Implementation and Measurements, *IEEE Modeling and Optimization in mobile, Ad hoc and Wireless Network*, 4th International Symposium, 2006.

[9] P.W.Q. Lee, W.K.G. Seah, H. Tan and Zexi Yao. Wireless sensing without sensors - An experimental approach. 20th International Symposium on Personal, Indoor and Mobile Radio Communications, *IEEE PIMRC*. 2009: 62-66.

[10] D. Zhang , J. Ma , Q. Chen and L. Ni. An RF-based system for tracking transceiver-free objects, *IEEE PerCom*, 2007: 135-144.

[11] J. Wilson, N. Palwari. See-Through Walls: Motion Tracking Using Variance-Based Radio Tomography Networks, *IEEE Trans. Mobile Computing*, 10(5): 612-621, 2011.

[12] Carl J. Weisman. The Essential Guide to RF and Wireless, 2nd Edition, *Prentice Hall PTR Press*, 2002.

[13] A. Goldsmith. Wireless Communications, *Cambridge University Press*, 2005.

[14] G.D. Durgin, T.S. Rappaport, and D.A. de Wolf. New Analytical Model and Probability Density Functions for Fading in Wireless Communications, *IEEE Trans. Communications*, 50(6): 1005-1015, 2002.

[15] G. Zanca, F. Zorzi, A. Zanella and M. Zorzi. Experimental comparison of RSSI-based localization algorithms for indoor wireless sensor networks, *IEEE REALWSN*, 2008.