Measurement-based AP Deployment Mechanism for Fingerprint-based Indoor Location Systems

Dong Li[1], Yan Yan[1], Baoxian Zhang[1], and Cheng Li [2], Peng Xu[3]
[1] Research Center of Ubiquitous Sensor Networks, University of Chinese Academy of Sciences
Beijing 100049, China
[e-mail: {yany, bxzhang}@ucas.ac.cn]
[2] Faculty of Engineering and Applied Science, Memorial University, St. John's, NL A1B 3X5, Canada
[e-mail: licheng@mun.ca]
[3] Research Institute of information technology, Shandong Police Training College, Jinan, Shandong Prov., China
[e-mail: xu-peng@189.cn]
*Corresponding author: Peng Xu

Abstract

Recently, deploying WiFi access points (APs) for facilitating indoor localization has attracted increasing attention. However, most existing mechanisms in this aspect are typically simulation based and further they did not consider how to jointly utilize pre-existing APs in target environment and newly deployed APs for achieving high localization performance. In this paper, we propose a measurement-based AP deployment mechanism (MAPD) for placing APs in target indoor environment for assisting fingerprint based indoor localization. In the mechanism design, MAPD takes full consideration of pre-existing APs to assist the selection of good candidate positions for deploying new APs. For this purpose, we first choose a number of candidate positions with low location accuracy on a radio map calibrated using the pre-existing APs and then use over-deployment and on-site measurement to determine the actual positions for AP deployment. MAPD uses minimal mean location error and progressive greedy search for actual AP position selection. Experimental results demonstrate that MAPD can largely reduce the localization error as compared with existing work.

1. Introduction

Indoor localization has obtained considerable interests because position information is essential in many applications and systems such as Internet of Things [1]. In general, indoor localization technologies can be divided into two categories based on whether range information is used. One type of such technologies is range based. They measure the distance (range) by using radio signal strength (RSS) [2], time of arrival (TOA) [3], time difference of arrival (TDOA) [4], time of flight (TOF) [5], or arrival of angle (AOA) [6] and then estimate the position of a mobile target using triangulation, trilateration, multilateration algorithms, or etc. The performance of these technologies highly depends on the characteristics of wireless signals, which in general change with distance and time. Another type of technologies is range free. Such technologies estimate the position of a mobile target according to the relative positions between the mobile target and beacons or pre-calibrated points such as binary proximity [7], approximate points in triangle (APIT) [8], or pre-calibrated fingerprints [9-12]. In reality, the performances of most localization technologies are highly relevant to the number and also the positions of beacons.

Recently, WiFi fingerprint based indoor localization has attracted a lot of attention and much work has been carried out by using WiFi APs as beacons due to the increasing popularity of existing WLAN infrastructure. However, how to properly deploy WiFi APs to achieve high localization performance has not received sufficient attention. Traditional WLAN facilities were mainly deployed for Internet access and existing work in this aspect mainly focused on achieving maximal signal coverage [13-14] or enhancing the network throughput [15]. Some existing work (e.g., [16]) indicated that the accuracy and precision of a positioning system is affected by the number of APs. Moreover, they suggested to carefully place APs in a way such that every location can receive the minimum required number of signals to achieve desired positioning performance goal. Some other existing mechanisms for AP deployment (e.g., [17-24]) used simulated based methods for fingerprint generation in order to determine the positions for AP deployment. They typically use a simplified radio propagation model to characterize the characteristics of wireless signals in target environment and also an abstraction for the target environment. However, the propagation properties of wireless signals in indoor environment are usually very complex to be characterized. Moreover, a solution identified by using such simulated fingerprints may be far away from the solution for the real environment. In addition, it is difficult, if not impossible at all, to introduce measured signals from pre-existing APs for radio map generation into a simulator, due to the difficulty in simultaneously feeding the simulator with simulated signals (from supposed newly deployed APs) and measured signals (from actual preexisting APs in the target environment).

To address the above problem, in this paper, we propose a measurement-based AP deployment mechanism (MAPD), which works to deploy APs in target environment to improve the performance of a WLAN fingerprint based indoor localization system. The key idea behind MAPD is to take full advantage of pre-existing APs for identifying candidate AP positions with poor accuracy and also improved localization performance. In MAPD, two rounds of measurements are required. In the first round measurement, we calibrate a radio map by using those pre-existing APs only and then calculate the location errors of certain pre-selected points in the target environment. We then choose *n* candidate positions with the lowest location accuracy among the pre-selected positions. The second round measurement is conducted by all the pre-existing APs and overly deployed *n* APs, one AP at each of the above

selected candidate positions. We progressively select m from the n over-deployed APs, one for each time. During the progressive selection process, minimal mean location error and greedy search are used to find a near-optimal AP deployment layout. It should be noted that during the application phase, only the m APs are kept and those other $n\text{-}m$ over-deployed APs will be removed. This is economically acceptable due to the quick decrease in WiFi AP cost. Finally, we conduct experiments in 320m^2 office environment. The experimental results demonstrate that MAPD can largely improve the performance of indoor localization system in term of mean location error and maximum location error as compared with existing work.

The rest of this paper is organized as follows. Section 2 gives a brief overview of related work for AP deployment for indoor localization. Section 3 presents the detailed design of our proposed mechanism MAPD. Section 4 presents some implementation details of MAPD regarding certain design choices and parameter tuning via experiments. Section 5 compares the performance of MAPD with other related work. Finally, Section 6 concludes this paper.

2. Related Work

Recently, deployment of WiFi APs for indoor localization has attracted much attention and a lot of work has been carried out in this aspect. Most existing mechanisms in this aspect (e.g., [17-22]) typically involve the following aspects: 1) choose a proper objective function (or called a key metric sometimes) to judge the quality of an AP deployment layout, 2) choose a search algorithm for choosing m positions out of n candidate positions ($m \leq n$) for the AP deployment; 3) generate fingerprints (via measurement or simulation) for evaluating the performance of a specific deployment layout. Next, we shall introduce existing mechanism based on how they address each of these three aspects.

Regarding choosing a proper objective function, most mechanisms take similarity distance between fingerprints or its variants as the objective function. In [17], Battiti et al. integrated location error minimization and signal coverage maximization as the objective function. In [18], Zirazi et al. presented an algorithm that minimizes geometric dilution of precision to improve the positioning accuracy. In [19], Sharma et al. proposed a method for minimizing the total number of similar fingerprints. In [20], Zhao et al. suggested to maximize the signal distance between each pair of fingerprints. In [21], He et al. aimed to make the resulting location error in Euclidean distance to be greater than a given threshold, which was chosen to be 2 meters. Fang et al. [22] aimed at choosing a set of APs' locations such that the signal distance is maximized and the noise is minimized simultaneously. The advantages of using similarity distance based objective functions are easy to obtain and independent of specific localization algorithm. However, the correlation between similarity distance and position accuracy is not quite high. It means increasing the similarity distance between fingerprints may not necessarily contribute to the performance of a localization system.

Regarding selection of proper search algorithm, most existing mechanisms use heuristic algorithms to search for a good solution. In this aspect, Refs. [17,19] use simulated annealing (SA), Refs. [18,21] use genetic algorithm, and Ref. [20] adopts differential evolution algorithm. Use of such heuristic algorithms are suitable to search near-optimal solution when no apriori knowledge for the combinatorial optimization problem is available. The performance of the solutions returned by these algorithms highly depend on the number of iterations allows and obtaining near-optimal solutions using such algorithms are usually quite time consuming.

To generate the fingerprints for judging the quality of a specific AP deployment layout, most existing mechanisms use simulations to generate such fingerprints due to newly

deployed APs by using wireless signal propagation models. In this aspect, Refs. [17,18,20,21] use logarithmic loss model and Ref. [19] uses Ray tracing propagation model. However, indoor wireless environment is usually complex to be well characterized by a generic propagation model. That is, a solution identified by using such simulated fingerprints may largely deviate away from the ideal case in reality. Different from the above simulation based mechanisms, in [22], Fang et al. collected the fingerprints in a real office environment and used weighted k nearest neighbor algorithm to calculate the location error. However, they did not provide details regarding how to choose initial candidate positions for AP deployment and also how to choose actual AP positions from them.

Refs. [23,24] give some general suggestions for AP deployment for assisting indoor localization. In [23], the experiment results demonstrated that localization performance by using symmetric deployment is better than that by using asymmetric deployments. In contrast, Ref. [24] gave a suggestion: Deploying APs at the superset of interior and perimeter access points in a staggered fashion could achieve better performance.

One key problem in all the above mechanisms is that they did not consider how to jointly use newly deployed APs and pre-existing APs for achieving high localization performance, a key issue that will be studied in this paper.

3. Proposed Mechanism

In this section, we propose our MAPD mechanism. In the design, we assume there have already existed some APs (but at unknown positions) in the target environment, which is quite common nowadays.
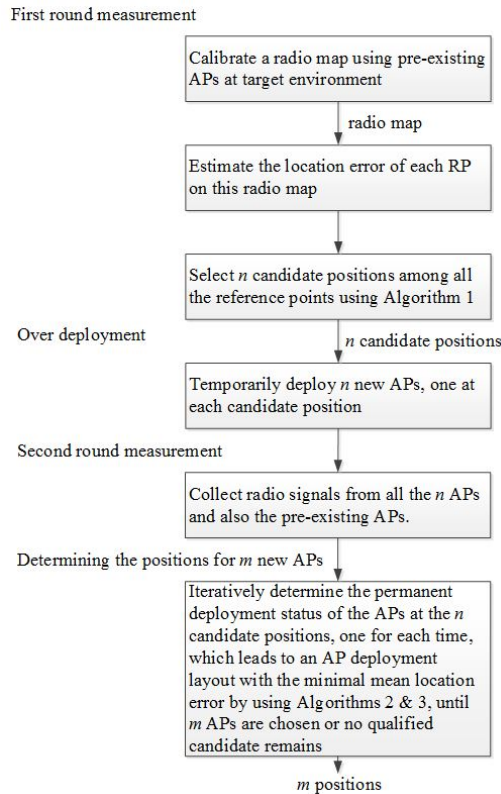

Fig. 1. Flow chart describing how our measurement-based AP deployment mechanism works.

The key idea behind MAPD is to make full use of the pre-existing APs via two rounds of on-site measurements for assisting the new AP deployment and also for achieving high localization performance. The first round is to calibrate the target environment using those pre-existing APs only and then determine some candidate positions with low position accuracy (whose total number is denoted by n) for the AP deployment. The second round is to chooses m ($m<n$) positions for actual AP deployment, one AP at each of the m positions, among the n candidate positions identified in the first round for achieving improved localization performance. **Fig. 1** gives a flowchart showing how our mechanism works.

Design details of MAPD are as follows.

3.1 Determine *n* Candidate Positions via One Round of Measurement

This step is to determine n candidate positions in the target environment for assisting the AP deployment. For this purpose, we first calibrate a radio map for the target environment by only using those pre-existing APs and then choose a number of n candidate positions among the reference points (RPs). Regarding this, we select those RPs with low accuracies as the candidate positions. To ease the presentation, let P represent the number of pre-existing APs.

1)   Calibrate a radio map using those pre-existing APs only and estimate the location error of each RP on this radio map

For this purpose, we first divide the target environment into a grid structure and define the intersections of the grid as reference points (RPs).[1] We then use a mobile phone with WiFi interface to scan the WiFi channels at each of the reference points and sample radio signal strength (RSS) T times from each detected AP. At each RP, a vector of mean RSS values (known as a fingerprint) associated with the RP is recorded in a two dimensional matrix (known as a radio map). In the matrix, each row represents a fingerprint (i.e., a RP). Let RP_i ($i \in [1, N]$) represent the i-th RP and AP_m ($m \in [1, M]$) represent the m-th AP, where N and M represent the total number of RPs and that of APs, respectively. After obtaining the radio map, a mobile device at an unknown position reports its sampled vector of RSS values (denoted by V) sampled from various APs to a remote server. The remote server calculates the similarity distance between the reported vector and each fingerprint in the radio map and selects k fingerprints (RPs) with the smallest similarity distances as the k nearest neighbors to the mobile device. The positions of the k neighbors are then weighted averaged as the estimated position of the mobile device.

In this paper, the similarity distance is calculated as follows.

$$d_i = \sqrt{\sum_{m=1}^{M}(\bar{o}_{i,m} - v_m)^2}, \tag{1}$$

where d_i is the similarity distance between an instantly reported RSS vector V and the fingerprint associated with RP_i, v_m represents the mean RSS value received from the m-th AP in vector V at RP_i, and $\bar{o}_{i,m}$ represents the mean RSS value received from the m-th AP at the i-th RP. $\bar{o}_{i,m}$ is calculated by (2).

---

[1] Other ways of choosing references points are also acceptable. In this paper, we assume the reference points are distributed on a grid structure.

$$\bar{o}_{i,m} = \frac{1}{T_{i,m}}\sum_{x=1}^{T_{i,m}} o_{i,m}(x), \tag{2}$$

where $o_{i,m}(x)$ is the x-th RSS value received from the m-th AP at the i-th RP, $T_{i,m}$ is the total sampling time from the m-th AP at the i-th RP. In our design, the number of nearest neighbors k in wkNN algorithm is empirically set to 3. We had also tested other choices ranging from 1 to 10 and found that wkNN reaches its best performance when $k = 3$. The estimated position (x', y') is as follows.

$$x' = \sum_{i}^{k}(x_i \times w_i), y' = \sum_{i}^{k}(y_i \times w_i) \tag{3}$$

where (x_i, y_i) is the position of the i-th nearest neighbor, and w_i is the weight associated with the i-th nearest neighbor, which is computed as follows.

$$w_i = \frac{1/d_i}{\sum_{j=1}^{k}(1/d_j)}, (1 \leq i \leq k) \tag{4}$$

Eq. (4) ensures that the nearer a neighbor is, the greater its associated weight will be.

To determine the location error of RP_i (denoted by e_i) on the radio map (calibrated using pre-existing APs), we use the Leave-one-out cross-validation (LOOCV) method [25]. Specifically, we let each individual fingerprint (RP) as a testing set and all other remaining fingerprints as the training set. Then we substitute such two sets into (1)-(4) and apply the result into (5) in order to determine the location error associated with RP_i (denoted by e_i).

$$e_i = \sqrt{(x_i - x_i')^2 + (y_i - y_i')^2} \tag{5}$$

where (x_i', y_i') is the estimated position of RP_i. In this way, we get the location error of each RP. Next, we introduce how to choose the candidate AP positions from the RPs.

2)   Select n Candidate Positions

The selection is based on the following two observations. First, reference positions with poor accuracy should be chosen with high priority. Second, newly deployed APs should keep certain distance. The latter choice is because a newly deployed AP is expected to be helpful for assisting the localization in its vicinity. Detailed algorithm design for candidate position selection is shown in **Algorithm 1**.

Algorithm 1. Selection of candidate AP positions.

Input:	S_1: An array storing the location errors of all the RPs.
	RP_S_1: An accompanying array recording the ID of the RP for each corresponding item in S_1.
	D: Minimal required distance between newly deployed APs.
Output:	S_2: The set of candidate positions.

---

float d_{xy}: The distance between two positions x and y.
$N(x)$: The set of all RP neighbors of RP x within distance D.
1. $S_2 \leftarrow \varnothing$

---

2. Sort all the elements in S_1 in the descending order of estimated location errors.
3. **while** $S_1 \neq \varnothing$
4. $x \leftarrow$ RP_S_1[0] //get the ID of the RP leading to the largest location error.
5.   **for** each $y \in$ RP_S_1-$\{x\}$
6.   **do if** $d_{xy} \leq D$, **then** insert y into $N(x)$, **end if**
7.   **end for**
8.   insert x into S_2
9.   remove x and elements in $N(x)$ from RP_S_1 and also remove their corresponding items from S_1
10. **end while**
11. **return** S_2

By properly adjusting the value of D, which represents the minimal required distance between newly deployed APs, Algorithm 1 can return an expected number (denoted by n) of candidate positions for deploying new APs. How to tune the parameter D will be discussed later.

3.2 Select *m* Positions from the *n* Candidate Positions via a Second Round Measurement

After completing the first round measurement, we have n candidate AP positions. Now, our goal is to select m out of the n candidate positions for the actual AP deployment while minimizing the mean location error. However, finding the optimal solution is known to be a combinatorial optimization problem and is NP-Complete. To find an efficient AP deployment layout, in MAPD, a greedy search algorithm is used. The selection process is as follows.

- AP over-deployment is conducted to prepare radio map due to any interested AP combination of the n candidate positions (plus those pre-existing APs) and then judge the quality of the combination (measured by mean location error in this paper). The AP over-deployment deploys one AP at each candidate position and it is supposed to be economically acceptable because WiFi APs are now cheaper and cheaper.
- Calculation of mean location error, which is for calculating the resulting mean location error due to an interested AP deployment layout, and
- Progressive selection of m AP positions for actual deployment, one for each time, which leads to the minimal mean location error if the AP was deployed.

Detailed designs are as follows.

1)   Radio signal collection via AP over-deployment and on-site measurement

To judge the quality of a particular AP deployment layout, we need the radio map calibrated by using APs deployed at the positions suggested by the layout. However, the total number of such layouts can be huge. In this paper, we handle this problem by over-deployment and on-site measurement. Specifically, we deploy n APs, one at each candidate position and then conduct another round of on-site measurement to collect the radio signals from all these n APs and also the P pre-existing APs at each RP. In this way, we can easily get the radio map due to any combination of a subset of the n APs (plus all the pre-existing APs of course). It should be noted that during the application phase, only m APs need to be kept and other $n-m$ over-deployed APs will be removed since they are deployed only for evaluation purpose. The effectiveness of such over-deployment is based on the conclusion in [16]: Signals from different APs within the range of reception can be considered as uncorrelated or independent because the correlation coefficient between any pair of signals is often small or trivial.

2) Calculation of mean location error due to a particular AP deployment layout

For a possible AP deployment layout, we use mean location error as the key metric to evaluate its performance. The procedure for calculating the mean location error (MLE) due to a specific AP deployment layout is given in **Algorithm 2**.

Algorithm 2. Calculate the MLE of a specific AP deployment layout.

Input: A radio map calibrated based on a certain combination of (new) APs and also those pre-existing APs

Output: MLE due to the above new AP deployment

---

1. **For** each fingerprint (RP) in the radio map, **do**
2.   Select it as the testing set and all other remaining fingerprints as the training set;
3.   Calculate the estimated position of this fingerprint by using (3);
4.   Calculate the location error of this fingerprint based on (5);
5. **end for**
6. Average the location errors of all fingerprints to get the MLE value and return it.

To evaluate the effectiveness of using MLE as the key metric for selecting an AP deployment layout, we had also conducted experiments to compare it with other metrics such as minimal total number of similar fingerprints (MNSF) [19], maximal signal distance between each pair of fingerprints (MSD) [20], and maximal signal noise rate (MSNR) [22], all were conducted by using the same data set and in the same testing environment. The results show that MLE performs the best. Details will be presented in Section 3.2.

Next, we present a greedy search algorithm (referred to as GSA) to find an AP deployment layout leading to low MLE. The design details are shown in Algorithm 3. An AP deployment layout means a number of APs chosen from the n APs and only APs in the resulting deployment layout will be kept during the application phase.

In **Algorithm 3**, the outer "for" loop between lines 2-15 is for maximally choosing m APs from the n candidates and the inner "for" loop between lines 4-11 is to select one new AP from S_2, which leads to the minimal MLE on the radio map calibrated by using all preexisting APs, already determined APs for deployment in S_1, and one new AP from S_2. The outer loop continues until m AP positions are selected or no qualified candidate remains (see line 12). In this way, we hope to greatly reduce the computational overhead with little penalty in location accuracy. To validate the effectiveness of GSA, we had compared GSA with simulated annealing (SA) and exhaustive search by using the same data set and the same objective function for error estimation. The details will be presented in section 3.3.

Algorithm 3. Procedure for GSA for finding an efficient combination of APs.

Input: m: Expected number of APs to be deployed for use during the application phase

         PREV: The set of all pre-existing APs

        S_1:The set containing the new APs for actual deployment during application phase, whose initial value is NIL.

        S_2: A set containing the remaining over-deployed APs, which is initially all the n over-deployed APs.

Output: S_1

---

1. $S_1 \leftarrow \varnothing$
2. **for** $i = 1$ to m, **do**
3.   *error* $\leftarrow$ INFINITY
4.   **for** each element $j \in S_2$, **do**
5.   MAP $\leftarrow$ A radio map calibrated by using APs = PREV + S_1 + $\{j\}$
6.   *tmp* $\leftarrow$ The MLE of the above radio map returned by Algorithm 2 (called MAP)
7.   **if** (*tmp* < *error*) **then**
8.   *error* $\leftarrow$ *tmp*
9.   *next* $\leftarrow$ *j*
10.   **end if**
11.   **end for**
12.   **if** the resulting MLE *error* is lower than a given threshold, **then return, end if**
13. $S_1 \leftarrow S_1 + \{next\}$
14. $S_2 \leftarrow S_2 - \{next\}$
15.   **end for**
16.   return S_1

4. Implementation Details

In this section, we explore the impact of various design choices and parameters in MAPD on its performance via extensive experiments.

4.1 Experiment environment and setup

The experiments were conducted on an office environment, on the third floor of the Science-and-Research Building on the campus of the University of Chinese Academy of Sciences (see **Fig. 2**). The target environment consists of three rooms (306, 308, and 310) and a corridor. The dimensions of this environment are 32 m by 10 m, an area of 320m^2. 85 reference points were selected with 1 meter space. RP1-RP32 were chosen in the corridor. RP33-RP49 were chosen in Room 310. RP50-RP66 were in Room 308. RP67-RP85 were chosen in Room 306.
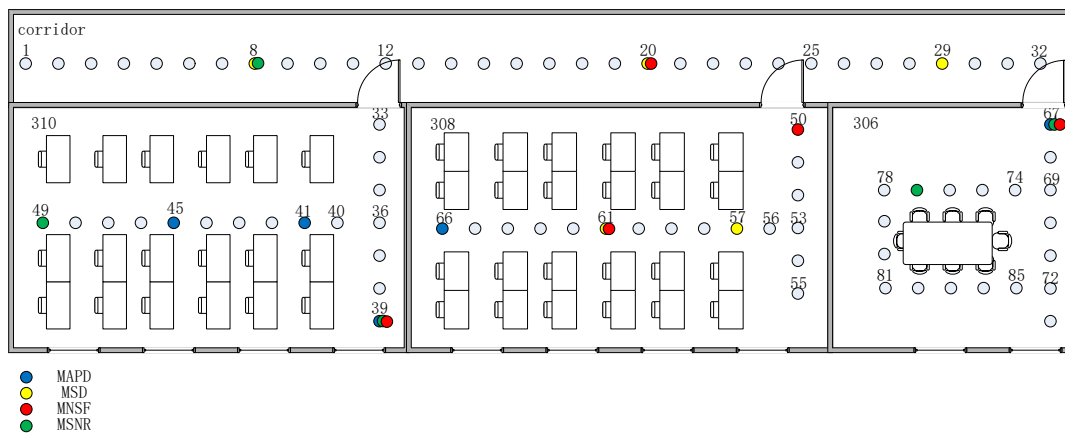

Fig. 2. Experiment environment and distribution of the reference points.

In our experiments, the testing personnel walked in the target environment and sampled RSS values 20 times at each reference point by using a Samsung 9100 mobile phone. Throughout the measurement, the direction of the mobile phone was always held consistent. Among the 22 APs detected at all the 85 RPs, 10 of them were detected at all the RPs. All these APs are pre-existing, whose hardware types and positions were unknown.

Fig. 3. TL-MR10 router.

In the experiments, we used TL-MR10U (see **Fig. 3**) as the new AP for deployment and radio map calibration. TL-MR10U is an 802.11 b/g/n router with a 2600 mAh battery, which can exempt the requirement of availability of power jack at each of the candidate positions.

4.2 Impact of minimal new AP distance *D*

In this subsection, we conduct experiment for tuning the parameter *D*, which is the minimal required distance between newly deployed neighbor APs in Algorithm 1. Recall that Algorithm 1 is to choose the *n* candidate positions for preparing the actual AP deployment.

We first use the radio map calibrated by using those pre-existing APs to get a view on the location error at each RP. **Fig. 4** shows the distribution of these location errors, where X and Y axes correspond to distances in meter, and Z axis represents the location error at each RP in meter. From **Fig. 4**, we can see that the location errors vary from 0.33 to 5.5 meter. Moreoever, it is seen that some of the errors are significantly worse than those around them.
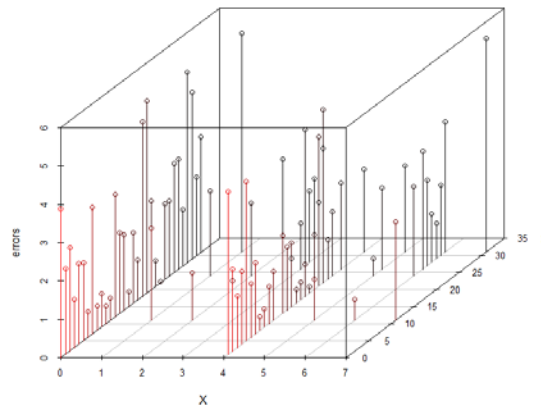

Fig. 4. Distribution of the location errors at different reference points.

Table 1. Results returned by Algorithm 1 for different distance constraint *D* for new AP deployment.

Minimal new AP distance *D* (meter)	Number of candidate positions *n*	Candidate positions
2	25	67 73 20 29 57 49 45 1 77 8 13 81 26 39 70 5 66 51 17 34 40 84 61 54 23
3	17	67 73 20 29 57 49 45 1 77 8 13 81 39 50 66 61 41
4	13	67 20 57 49 1 77 8 13 39 25 64 44

Table 1 lists the candidate positions with changing new AP distance constraint *D*. In the first round measurement, we tested three different values (i.e., *D* = 2, 3, 4) for finding an appropriate value of *D*. We found that the sets of candidate AP positions based on different *D* values contain the same set of RPs (i.e., 67, 73, 20, 57, 49, 45, 1, 8, 13, 39), which are non-related to *D*. Note that when *D*<2 the resulting RP list still contain the above RP sequence. In this paper, we choose *D* = 3 since three meters is a reasonable distance within which RSS and distance exhibits a sharp relationship. Note that in this case, *n*=17.

4.3 Impact of MLE

This subsection compares the performance of using minimal mean location error MLE as the key metric for evaluating the performance of a specific AP deployment layout and three other metrics including MSD [20], MSNR [22], and MNSF [19]. MSD aims to maximize the Euclidean distance of received signal strength array among all the sampling points. MSNR aims to maximize SNR (signal-to-noise ratio). MNSF aims to minimize the total number of similar fingerprints.

In this experiment, we compare the performance of the four metrics in terms of mean location errors resulted by different AP deployment layouts. The mean location error by using each of the four metrics was calculated by using the wkNN algorithm. During the experiments, the same data set (i.e., *PreExistingAPs* + *NewAPs*) and the same AP searching algorithm (i.e., simulated annealing) were used. *PreExistingAPs* and *NewAPs* represent the set of pre-existing APs and that of newly deployed APs, respectively.
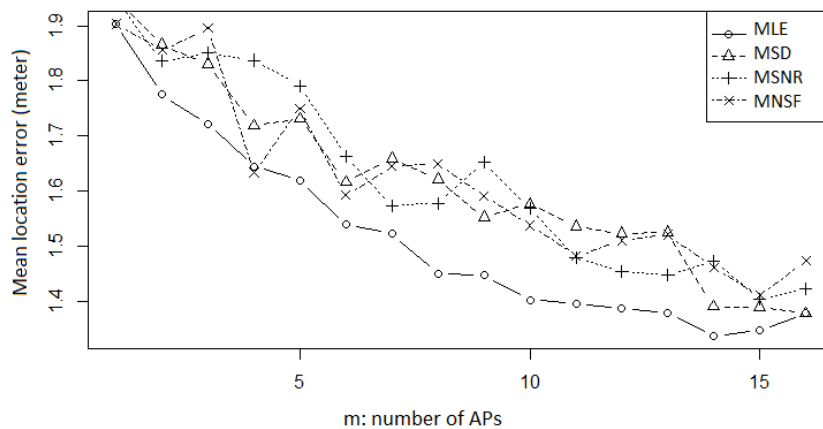

Fig. 5. Comparison of mean location error performance by different metrics versus number of newly deployed APs.

Fig. 5 shows that the location errors by all the four metrics decrease as the number of candidate APs increasing. Moreover, MLE is superior to all the other three. This is because MLE is directly related to the location accuracy of a localization system, whereas all the other three are not. Furthermore, the curve by MLE is smoother than those by the other three.

4.4 Effectiveness of greedy search algorithm (GSA)

This subsection compares the performance of GSA with two other searching algorithms including simulated annealing (SA) [26] and exhaustive search for finding efficient AP deployment layout. SA is a probabilistic search heuristic for solving the global optimization problem in a large search space. It uses a variable as the temperature for the cooling schedule, which is very high at the beginning and will become lower and lower during iterative computations. In our experiments, we start from a random initial combination, set the maximum iteration number to 300, and the starting temperature as 100. Exhaustive search enumerates all possible combinations and returns the optimal solution. However, its computational time is typically exponential to the number of candidate AP positions.

Under the same data set (i.e., *PreExistingAPs + NewAPs*) and the same objective function (i.e., minimal MLE), we compare the performance of the above three search algorithms. The performance is reflected by the mean location error calculated by using wkNN and the computation time. All the three search algorithms were implemented using R language on a PC with Intel Core i5-2400 CPU and 4 GB RAM.
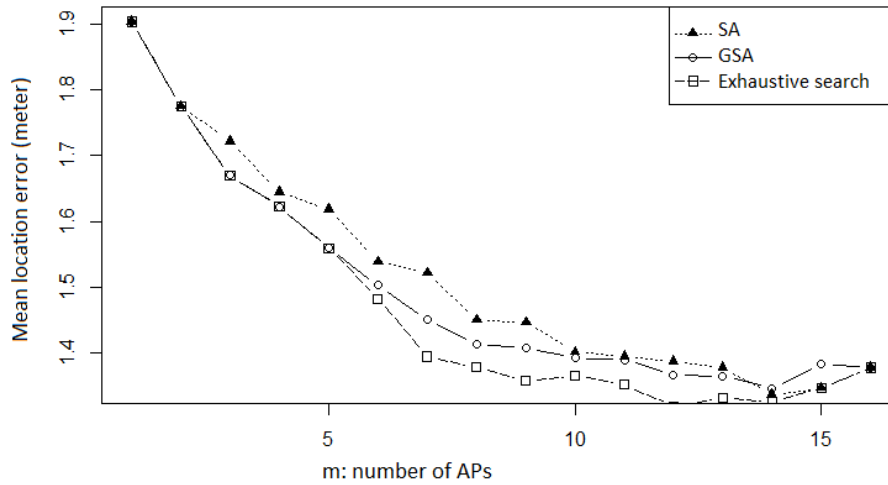

Fig. 6. Comparison of various search algorithms versus number of deployed APs in terms of mean location error.

Fig. 6 shows that the mean location errors by all the three search algorithms decrease as the number of deployed APs increasing. Exhaustive search always yields the optimal solution. GSA is superior to SA in most cases. Specifically, the performance of SA is only relevant to the number of combinations and the number of iterations. In our experiments, the maximum iteration number of SA was fixed to 300, which is greater than the maximum number of possible combinations of AP positions when $m \leq 3$ or $m \geq 14$ (note that $n=17$). With the number of combinations increasing, the difference between SA and exhaustive search increases. Also, GSA has the same trend with SA. In practice, we can choose the value of n to

be much larger than m in order to achieve high localization performance improvement. In this case, GSA usually outperforms SA.
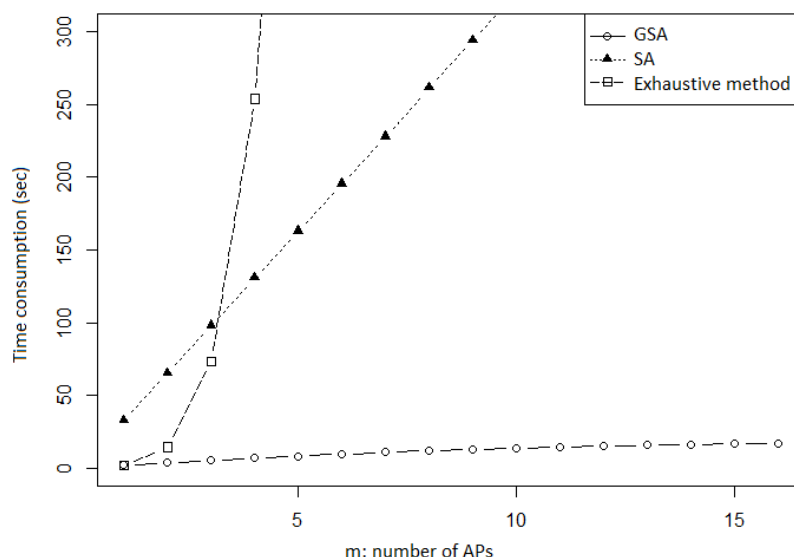

Fig. 7. Comparison of searching times by different search algorithms versus number of deployed APs.

Fig. 7 shows that all the three search algorithms' computation times increase with the number of deployed APs. Among the three algorithms, GSA consumes the least time, SA is the second, and exhaustive search needs the largest amount of time.

5. Performance Evaluation

In this section, we evaluate the performance of MAPD via extensive experiments by comparing it with related work.

5.1 Experiments setup

In this subsection, we introduce in details the experiment environment, performance criteria used, and mechanisms performed for comparison.

1)   Experiments environment

The experiments were again conducted in the $320m^2$ office environment in **Fig. 2**. We implemented the MAPD mechanism via a client program on an Android mobile phone and backend services on a PC. At the client side, we sampled RSS values 20 times at each reference point. Throughout the measurements, the mobile phone holding direction is always consistent. At the server side, all localization mechanisms were implemented using R language on a PC with Intel Core i5-2400 CPU and 4 GB RAM (as backend server).

In the initial experiment, the number of deployed APs m was fixed to be 5. The minimal required distance D was set to 3, accordingly the number of candidate AP positions $n = 17$ (see **Table 1**). During the measurements, in total 22 preexisting APs were detected at the 85 RPs and 10 of them were detected at all the RPs. In the second round of measurement, in total 39 APs were detected at the 85 RPs and 27 of them were detected at all these RPs. Note that, at each RP, all the 17 new APs can be detected. Accordingly, new radio map can be built by using

the 10 preexisting APs (denoted by Preexisting_APs) and the 17 candidate APs (denoted by nCandidateAPs).

2)   *Performance criteria*

The performance of an AP deployment mechanism is measured by the gain of localization accuracy. In this paper, we chose to use the following three criteria.
- Mean location error: it is reported as positioning accuracy.
- Distribution of the location error: it represents positioning precision.
- Maximum location error: it reflects the worst case of a localization system.

3)   *Mechanisms for comparison*

We compared MAPD with the case without new APs (called NoNewAP), MSD [20], MNSF [19] and MSNR [22]. The main implementation details for various mechanisms are shown in **Table 2**.

Table 2. Parameters and design choices for various mechanisms.

Mechanisms	Radio map used for identifying candidate AP positions	Radio map used for online localization	Objective function	Search algorithm
NoNewAP	N/A	PreexistingAPs	N/A	N/A
MSD[20]	PrePreexistingAPs + nCandidateAPs	PreexistingAPs + 5 new APs	maximal signal distance of each pair fingerprints	Simulated annealing
MAPD	PreexistingAPs + nCandidateAPs	PreexistingAPs + 5 new APs	minimal mean localization error	greedy search
MNSF[19]	PreexistingAPs + nCandidateAPs	PreexistingAPs + 5 new APs	minimal total number of similar fingerprints	simulated annealing
MSNR[22]	PreexistingAPs + nCandidateAPs	PreexistingAPs + 5 new APs	maximal signal distance and minimal noise	simulated annealing

According to the parameters and design choices in **Table 2**, the positions for the 5 new APs returned by different mechanisms are as follows (represented by the IDs of the corresponding RPs, see **Fig. 2** regarding how the deployment positions are chosen by different mechanisms):
- MAPD: 41, 39, 67, 66, 45;
- MSD: 61, 57, 29, 8, 20;
- MNSF: 67, 61, 39, 20, 50;
- MSNR: 78, 8, 39, 67, 49.

5.2 Location accuracy

To determine the location error performance, we used wkNN onto the radio maps calibrated using pre-existing APs and also the 5 new APs suggested above, by using different mechanisms. The results are shown in **Table 3** and **Fig. 8**.

Table 3. Comparison of performance of various mechanisms in terms of location errors.

Mechanisms	Mean location error (meter)	Maximum location error (meter)
NoNewAP	2.12	5.55
MSD	1.73	4.53
MAPD	1.56	4.02
MNSF	1.69	5.20
MSNR	1.74	4.92

Table 3 shows that the mean location error by MAPD is 1.56 meter which is 0.56 meter (26%) 0.17 meter (10%), 0.13 meter (7%), and 0.18 meter (10%) smaller than their counterparts by NoNewAP, MSD, MNSF, and MSNR, respectively. Also, the maximum location error by MAPD is reduced by 1.53meter (27%), 0.51meter (11%), 1.18 meter (24%), and 0.9 meter (22.4%), as compared with the above mechanisms, respectively.
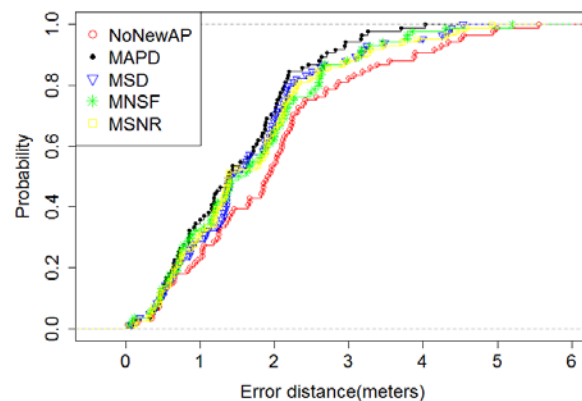

Fig. 8. CDF of location error distances by different mechanisms.

Fig. 8 shows the cumulative distribution function (CDF) of location errors by using different mechanisms. Compared with NoNewAP, the positioning precision by MAPD is much better. For example, in MAPD, 60 percentile of the location errors are within 2 meter, while only 50 percentile of location errors are within 2 meter for the case without new APs. Also, it is seen that MSD, MNSF and MSNR have similar performance. Note that MAPD outperforms all the other mechanisms. We had also implemented our mechanism (and also the other mechanisms) in other indoor environment including a home environment (an apartment in a building) and a 15m × 15m hall in the 8th floor in a mall. The results also show that our mechanism outperforms existing mechanisms. Due to limited space, we did not report the results here.

5.3 Impact of AP number *m*

Here, we explore the impact of the number of deployed APs *m* on the location accuracy performance. We took the pre-existing APs PreExistingAPs with the APs suggested by our MAPD mechanism and fed them into wkNN algorithm to calculate the resulting mean location error. **Fig. 9** shows that the mean location error by MAPD decreases gradually with m increasing. After the number of APs reaches 10, however, the change in mean location error is insignificant. In reality, the choice on the number of deployed APs should consider the required accuracy and deployment cost.
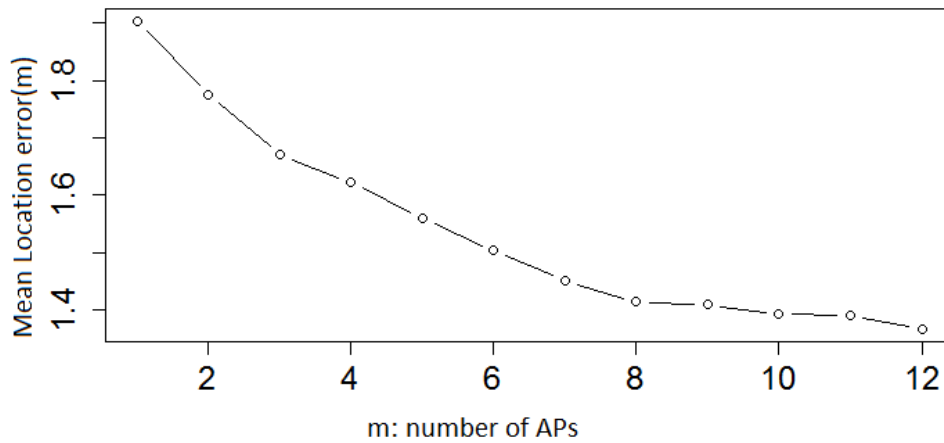

Fig. 9. Mean location error by MAPD versus number of deployed APs.

In summary, from the above experimental results, it is clearly seen our MAPD mechanism can significantly improve the location accuracy as compared with previous work.

6. Conclusion

In this paper, we proposed a measurement-based AP deployment mechanism MAPD, which supports the deployment of new APs in indoor environment to improve the performance of a WLAN fingerprinting based indoor localization system. MAPD takes advantage of the pre-existing APs in target environment for assisting the new AP deployment. In the design of MAPD, two rounds of on-site measurements, over-deployment of APs for determining candidate locations, minimal mean location error calculation, and progressive greedy search are used for finding a set of near-optimal AP positions. Experimental results show that MAPD can achieve high localization performance.

References

[1]   P. Bellavista, A. Kupper, and S. Helal, "Location-Based Services: Back to the Future," *IEEE Pervasive Computing*, vol. 7, no. 2, pp. 85-89, Apr.-June 2008. Article (CrossRef Link)

[2]   P. Bahl and V.N. Padmanabhan, "RADAR: an in-building RF-based user location and tracking system," in *Proc. of IEEE INFOCOM'00*, pp. 775-784, Mar. 2000. Article (CrossRef Link)

[3]   S. Golden and S. Bateman, "Sensor measurements for wi-fi location with emphasis on time-of-arrival ranging," *IEEE Transactions on Mobile Computing*, vol. 6, no. 10, pp. 1185–1198, Oct. 2007. Article (CrossRef Link)

[4]   AeroScout Company. [Online]. Available at: http://www.aeroscout.com/.

[5]   K. Muthukrishnan, G.T. Koprinkov, N. Meratnia, and M.E.M. Lijding, "Using time-of-flight for WLAN localization: feasibility study," *Technical Report*, University of Twente, Electrical Engineering, Mathematics and Computer Science, 2006. Article (CrossRef Link)

[6]   C.H. Lim, P.N. Boon, and D. Duan, "Robust methods for AOA geo-location in a real-time indoor WiFi system," *Journal of Location Based Services*, vol. 2, issue 2, pp. 112-121, June 2008. Article (CrossRef Link)

[7]   N. Shrivastava, R. Mudumbai U. Madhow, and S. Suri, "Target tracking with binary proximity sensors: fundamental limits, minimal descriptions, and algorithms," in *Proc. of ACM SenSys'06*, pp. 51-264, Oct. 2006. Article (CrossRef Link)

[8]   T. He, C.D. Huang, B.M. Blum, J. A. Stankovic, and T. Abdelzaher, "Range-free localization schemes for large scale sensor networks," in *Proc. of ACM MobiCom'03*, pp. 81-95, Sept. 2003. Article (CrossRef Link)

[9]   S. Sen, B. Radunovic, R.R. Choudhury, and T. Minka, "You are facing the Mona Lisa: Spot Localization Using PHY Layer Information," in *Proc. of ACM MobiSys'12*, pp. 183-196, June 2012. Article (CrossRef Link)

[10] Y. Chen, D. Lymberopoulos, J. Liu, and B. Priyantha, "FM-based indoor localization," in *Proc. of ACM MobiSys'12*, pp. 169-182, June 2012. Article (CrossRef Link)

[11] M. A. Youssef and A. Agrawala, "The Horus WLAN location determination system," in *Proc. of ACM MobiSys'05*, pp. 205-218, June 2005. Article (CrossRef Link)

[12] M. Brunato and R. Battiti, "Statistical learning theory for location fingerprinting in wireless LANs," *Computer Networks*, vol. 47, no. 6, pp. 825-845, Apr. 2005. Article (CrossRef Link)

[13]   K. Maksuriwong, V. Varavithya, and N. Chaiyaratana, "Wireless LAN access point placement using a multi-objective genetic algorithm," in *Proc. of IEEE International Conference on Systems, Man and Cybernetics*, pp. 1944–1949, Oct. 2003. Article (CrossRef Link)

[14] M. Kamenetsky and M. Unbehaun, "Coverage planning for outdoor wireless LAN systems," in *Proc. of 2002 International Zurich Seminar on Broadband Communications, Access, Transmission, Networking*, pp. 49.1-49.6, 2002. Article (CrossRef Link)

[15] A. Eisenblatter, H.-F. Geerdes, and I. Siomina, "Integrated access point placement and channel assignment for wireless LANs in an indoor office environment," in *Proc. of World of Wireless, Mobile and Multimedia Networks*, pp. 1–10, June 2007. Article (CrossRef Link)

[16] K. Kaemarungsi and P. Krishnamurthy, "Analysis of WLAN's received signal strength indication for indoor location fingerprinting," *Pervasive and Mobile Computing*, vol. 8, no. 2, pp. 292-316, Apr. 2012. Article (CrossRef Link)

[17] R. Battiti, M. Brunato, and A. Delai, "Optimal Wireless Access Point Placement for Location-Dependent Services," *Technical report DIT-03-052*, Universit`a di Trento, Oct. 2003.

[18] S. Zirazi, P. Canalda, H. Mabed, and F. Spies, "Wi-Fi access point placement within stand-alone, hybrid and combined wireless positioning systems," in *Proc. of ICCE'12*, pp. 279-284, Aug. 2012. Article (CrossRef Link)

[19] C. Sharma, Y.F. Wong, W.S. Soh, and W.C. Wong, "Access point placement for fingerprint-based localization," in *Proc. of ICCS'10*, pp. 238-243, Nov. 2010. Article (CrossRef Link)

[20] Y.X. Zhao, H.B. Zhou, and M.F. Li, "Indoor Access Points Location Optimization Using Differential Evolution," in *Proc. of CSSE'08*, pp. 382-385, Dec. 2008. Article (CrossRef Link)

[21] Y. He, W.X. Meng, L. Ma, and Z.A. Deng, "Rapid deployment of APs in WLAN indoor positioning system," in *Proc. of ICST CHINACOM'11*, pp. 268-273, Aug. 2011. [Article (CrossRef Link)](#)

[22] S.H. Fang and T.N. Lin, "A Novel Access Point Placement Approach for WLAN-Based Location Systems," in *Proc. of IEEE WCNC'10*, pp.1-4, Apr. 2010. [Article (CrossRef Link)](#)

[23] O. Baala, Y. Zheng, and A. Caminada, "The Impact of AP Placement in WLAN-Based Indoor Positioning System," in *Proc. of ICN'09*, pp. 12-17, Mar. 2009. [Article (CrossRef Link)](#)

[24] Cisco Systems, "Wi-Fi Location-Based Services 4.1 Design Guide," Available at: http://www.cisco.com/c/en/us/td/ docs/solutions/Enterprise/Mobility/WiFiLBS-DG.pdf.

[25] J. Starkweather, "Cross Validation techniques in R: A brief overview of some methods, packages, and functions for assessing prediction models," Available at: http://www.unt.edu/rss class/Jon/Benchmarks/ CrossValidation1_JDS_May2011.pdf

[26] P. A. Devijver and J. Kittler, "*Pattern Recognition: A Statistical Approach*," Prentice-Hall, 1982

Dong Li received his Ph.D. degree in computer science from the University of Chinese Academy of Sciences, Beijing, China in January 2016. He is currently a research scientist with Siemens Beijing. His research areas include indoor localization, Internet of Things, and embedded software design.

Yan Yan obtained his Ph.D. degree in computer science from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China in 2010. Since January 2010, he has been a lecturer with the University of Chinese Academy of Sciences, Beijing, China. He has served on technical program committees for international conferences and symposia such as IEEE ICC and AdHocNets. He has published over 10 papers in archival journal and conference proceedings. His research interests cover protocols and algorithm design and network coding for wireless ad hoc and sensor networks.

Baoxian Zhang received the PhD degree from Northern Jiaotong University, China, in 2000. He is currently a full professor with the Research Center of Ubiquitous Sensor Networks at the University of Chinese Academy of Sciences (UCAS), Beijing, China. From 2002 to 2005, he was a Research Scientist with School of Information Technology and Engineering, University of Ottawa, Canada, from 2002 to 2005. He is an associated editor with IEEE Systems Journal. He is a co-chair of IEEE VTC'14 Fall - Mobile Networks, Applications and Services Symposium. He has served on technical program committees for many international conferences and symposia such as IEEE Globecom and ICC. He is a joint holder of best paper award for WCSP 2012. He has coauthored a book in Wireless Sensor Networks and published over 150 refereed technical papers in archival journals and conference proceedings. His research interests cover network protocol and algorithm design, wireless ad hoc and sensor networks. He is a senior member of the IEEE (2012).

Cheng Li received the B. Eng. and M. Eng. degrees from Harbin Institute of Technology, Harbin, P. R. China, in 1992 and 1995, respectively, and the Ph.D. degree in Electrical and Computer Engineering from Memorial University, St. John's, Canada, in 2004. He is currently a Full Professor at the Faculty of Engineering and Applied Science of Memorial University, St. John's, Canada. His research interests include mobile ad hoc and wireless sensor networks, wireless communications and mobile computing, switching and routing, and broadband communication networks. He is an editorial board member of Wiley Wireless Communications and Mobile Computing and Journal of Networks, and an associate editor of Wiley Security and Communication Networks. He has served as a co-chair for various technical symposia of many international conferences, including the IEEE GLOBECOM and ICC. He has served as the TPC member for many international conferences, including the IEEE ICC, GLOBECOM, and WCNC. Dr. Li is a registered Professional Engineer (P. Eng.) in Canada and is a Senior Member of the IEEE and a member of the IEEE Communication Society, Computer Society, Vehicular Technology Society, and Ocean Engineering Society.

Peng Xu received his M.S. degree from the School of Information Science and Engineering, Shandong University in 2008. He is currently a senior engineer with the national information systems analyst. He is the director of Research Institute of information technology, Shandong Police Training College. His research interests focus on networking technology, computer software engineering, and communication technology.