This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2017.2737426, IEEE Transactions on Mobile Computing

Low-Overhead WiFi Fingerprinting

Junghyun Jun, *Member, IEEE,* Liang He, *Senior Member, IEEE,* Yu Gu, *Senior Member, IEEE,* Wenchao Jiang, *Member, IEEE,* Gaurav Kushwaha, *Student Member, IEEE,* Vipin A, *Student Member, IEEE,* Long Cheng, *Member, IEEE,* Cong Liu, *Member, IEEE,* and Ting Zhu, *Senior Member, IEEE,*

Abstract—WiFi-fingerprint localization is recognized as promising indoor localization technique. However, it suffers from high implementation overhead such as heavy initial training and fingerprint map maintenance over time. In this paper, we present the design, implementation, and evaluation of *AP-Sequence*. It is a fingerprint-based localization system that achieves extremely low overhead in fingerprint map construction and maintenance. AP-Sequence achieves this by treating a scan from any reference locations as an input to adjust a large portion of the fingerprint map. The power of AP-Sequence comes from dynamic region partitioning mechanism generating a fingerprint based on relative RSS values. AP-Sequence offers several advantages over existing methods with respect to robustness against environment noises, ability to handle dynamic power control, and mobile device heterogeneity. We have implemented AP-Sequence on Android platform. Experiment results with over one month of evaluation demonstrate that our design achieves an average localization accuracy of $4 - 7.6$ m over an extended time period with low-overhead in fingerprint map construction and maintenance.

Keywords—Indoor Localization, WiFi Fingerprint, Indoor Positioning, Relative RSSI, Linear Boundary

1 Introduction

With the ubiquitous availability of WLAN infrastructures in most indoor environments [1], [2], [3], [4], [5], [6], [7], WiFi-fingerprint-based localization becomes one of the most viable techniques. Other range-based indoor localizations treat radio signal distortion due to environment as a major huddle. Unlike them, WiFi-fingerprint-based localization utilizes this radio signal distortion as a unique fingerprint of a location.

WiFi-fingerprint localization typically consists of two phases: a training phase and a fingerprint matching phase. During the training phase, the fingerprint map (i.e., the radio map) is constructed by the site survey, where RSS values from multiple WiFi APs at different reference points are collected. In the fingerprint matching phase, a user is localized to one of the nearest reference points by matching the user measured fingerprints to the recorded fingerprints in the fingerprint map.

These two phases became a standard approach for all state-of-art WiFi-fingerprint localization. Unfortunately, these two phases have led to an unresolved dilemma between achieving better accuracy and lowering implementation overheads for existing WiFi-fingerprint-based localization techniques. This is due to the heavy initial training phase to construct the accurate fingerprint map [5] and the frequent fingerprint map updating [6]. The heavy training involves a large number of reference points to better represent an area of interest. Also, maintaining a reliable fingerprint map requires the site surveyor to frequently collect RSS scans at every reference points. In case of heterogeneous devices, it is impractical to collect RSS scans for all types of devices. To tackle these technical challenges, several approaches were proposed [5], [6], [8], [9], [10].

While all of them address one of these challenges very well, none of them was able to provide a solution that reduces heavy training while maintaining the reliable fingerprint map. This is because all of them applied a rule: *one geographical location must be represented by the set of reference points.* Under this rule, it is difficult to provide a unified solution that could handle all the challenges simultaneously.

In this work, we present a new direction to WiFi-fingerprint localization. It does not apply the rule of reference points for geographical locations anymore. Instead, it divides the area of interest into small regions and directly assign a unique sequence to each region as a fingerprint. The reference points are used as additional inputs to adjust the boundaries of the regions. This paradigm shift allowed us to design a unified WiFi-fingerprint-based localization system which provides the reliable localization accuracy and demands low system overhead.

Our proposed solution utilizes a metric called *AP-Sequence* to handle the temporal fluctuation of RSS and the device heterogeneity and a *dynamic* region-partitioning mechanism is designed to reduce number of required reference locations. AP-Sequence uses *relative* RSS differences among various APs. The RSS difference between two APs is known to be robust against ambient noises and interferences, indicating a good tolerance of short-term RSS variation [5], [11]. Also, it has been shown that device heterogeneity has little impact on the RSS difference [12]. AP-Sequence remains relatively stable over long period of time even when AP transmission power and environment change. The

• *J. Jun, G. Kushwahais, and Vipin are with the Department of Computer Science and Engineering, Indian Institute of Technology, Ropar, India (E-mail: peterjun@iitrpr.ac.in); L. He is with University of Colorado Denver, CO, USA; Y. Gu is with Visa Inc., USA; W. Jiang is with University of Minnesota, Twin Cities, USA; L. Cheng is with Virginia Tech, USA; C. Liu is with University of Texas, Dallas, USA; T. Zhu is with University of Maryland, Baltimore County, USA.*

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2017.2737426, IEEE Transactions on Mobile Computing

rational is that the environment often changes slowly and only a few APs change their power significantly at the same moment. Since our dynamic region-partitioning mechanism reduces required reference points, the fingerprint map gets updated quickly. Consequently, this reduces the negative impact of long-term RSS variation. The region-partitioning mechanism divides the area into small sub-areas by *dynamic* linear boundaries between all pairs of APs and each sub-area is associated with a unique AP-Sequence. As demonstrated in our empirical experiments, a single or double scanning of APs at few reference points (covering as little as 10% of the area of interest) is sufficient to estimate all the linear boundaries and generate/update the reliable fingerprint map. This paper makes following contributions by taking these advantages of AP-sequence as a fingerprint:

- We show a line is a good approximation of a real boundary between a pair of APs. The boundary is where relative RSS order between two APs change.
- In our design, we simplify the radio map construction by partitioning an area into a set of regions and each region is identified by the unique AP-sequence. The partitioning is done by the line arrangement. We further simplify the implementation of radio map construction by applying affine transformation of geometric space into dual space.
- We have implemented a AP-sequence-fingerprint localization system on Android platform. It is a complete WiFi-fingerprint localization system which constructs radio map based on the AP-sequence with no or few site surveying and localizes users by AP-sequence matching.
- We extensively test accuracy and practicality of AP-sequence-fingerprint localization system with real traces that we have collected for over one month.

2 Related Work

WiFi-fingerprint-based localization has attracted intensive attention because of the ubiquitous deployment of WiFi access points [13], [14], [15], [16], [17]. However, heavy initial training, handling of temporal fluctuation of RSS, and device heterogeneity still hinder its wide acceptance as a practical solution to the indoor localization.

The heavy initial training is the most significant bottleneck of fingerprint-based localization. ZEE [2] and LiFS [1] present a system that introduces the crowdsourcing (participatory) based training data collection and thus makes the calibration effortless. However, this still requires RSS scans from all the reference points to guarantee high localization accuracy. Also, this approach introduces new challenges — e.g., the device diversity issue. Attempts were made to reduce the number of needed reference points. NearMe [18] achieves this by estimating the distance between a user and reference points based on their distance model. In [19], a compartmental attenuation model for various signal modalities such as radio, acoustic and visible light is developed to provide high accuracy for node localization and tracking in sensor networks. Although the model based approach reduces the required reference points, the heavy training is still required to construct the environment-dependent distance models. EZ [20] is another RF modelling approach that does not require any calibration. Instead of manual site survey, it relies on occasional GPS information inside the building. However, GPS information can be completely absent in some buildings or inaccurate due to poor GPS signal reception. To reduce the initial heavy training, model-based approach as been proposed [21]. Zhang et al. proposed a model-based fingerprint-free localizing which uses a RSSI distribution model to dynamically divide field into subareas with unique signature [22]. Although this work eliminates the initial RSSI sampling to create fingerprints, it still requires manual deployment of anchors with known hardware configuration which is not feasible when localization is required on a large scale.

Another technical challenge is the temporal fluctuation of RSS [23], [24], [25], due to channel noise, change of environment, and dynamic power controlling of WLAN. A large number of RSS scans from every reference point was used to address this issue in [23], [24], [26]. After realizing RSSI difference between a pair of APs is more reliable than absolute RSS values, Rightspot [11] and FreeLoc [5] proposed more reliable fingerprint localization method based on this RSS difference.

All of the above mentioned fingerprint-based localization techniques require heavy training of fingerprints for all reference locations with extensive WiFi scans, because they all represent the geographical locations with a set of reference points. Unlike them, AP-Sequence represents a geographical location with a small region and the reference points are used for shifting the boundaries of this region. This allows AP-Sequence to automatically create and update the fingerprint map based on sparse reference points. With such a low overhead, we were able to design a unified WiFi-fingerprint localization that is robust against the temporal RSS variances, AP's dynamic power control, and device heterogeneity. The idea of using Ap-sequence as location fingerprint has also been explored in [27], where the area is divided into regions by cutting the lines connecting APs in the middle (i.e., bisectors), assuming a universal radio propagation model. This is similar to the ideal case we present in Sec. 5. Similarly, IncVoronoi [28] is designed to provide zero-calibration accurate RF-based indoor localization. It utilizes the relative relation between RSS values from two APs and incrementally reduces the user ambiguity region based on refining the Voronoi tessellation of the area of interest. However, in practice, partitioning with bisectors could be unreliable and inaccurate due to different AP transmission powers and environmental interferences, as shown in Fig. 4 based on our experiment results. Also, the maintenance of the fingerprint map (e.g., due to building renovation and (un)installing furnitures) is not explored in [27], [28].

In the domain of wireless sensor node localization, it is not a new idea to represent the geographical location

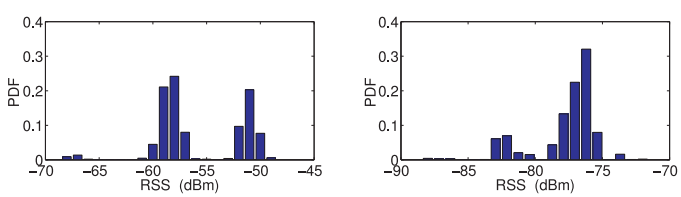
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2017.2737426, IEEE Transactions on Mobile Computing

with a small region (subareas) and mapping unknown sensors to one of the regions. This is known as a range-free localization [29], [30]. The range-free localizations assume homogeneous sensor nodes with same transmission powers. Therefore, the positions of any bisectors are fixed at a mid-point between two corresponding anchors. Unlike wireless sensor nodes, APs are heterogeneous in WLAN and WiFi and often their transmission power settings are unknown to the users. Therefore, the boundary partitioning two APs is no longer a bisector. In AP-Sequence, we first obtain new RSS scans from a surveyor/volunteer at some reference points and, based on this, we shift the position of bisector automatically to represent any environment changes such as dynamically changing transmission power or ambient noises.

3 Motivation

WiFi-fingerprint-based localization needs a fine-grained fingerprint map, leading to heavy training overhead. Here we explain the training overhead, as well as the stabilities and reliabilities of existing fingerprinting metrics based on absolute RSS values.

• **Heavy Training Overhead.** The accuracy of WiFi-fingerprint-based localization highly depends on the number of reference points in the space of interest. For example, Horus [23] uses 172 reference locations in a $68.2m \times 25.9m$ area and RADAR [31] uses 70 reference locations in a $22.5m \times 43.5m$ area. Typically, every reference points needs to be characterized by WiFi RSS values from multiple APs. According to our empirical experiments, completing the single scanning of all 11 WiFi channels costs about 1–1.5 seconds on Android platform (e.g. Samsung Galaxy 3, Sony Xperia, and HTC Rhyme). This means Horus would take at least 172 seconds for constructing its complete fingerprint map. This is significant initial overhead if we also consider the travelling time between the reference points as well. Worse, it is known that a single scanning is not sufficient to construct the fingerprint map due to short-term RSS fluctuation and human body blockage [7].



(a) AP_1 (b) AP_2

Fig. 1: Signal strength histograms from two APs.

• **Access Point Dynamic Power Control.** Dynamic power control has been increasingly adopted in advanced WiFi networks [32], affecting the absolute RSS values. We have collected RSS values from two APs for one day, and the histograms of the two RSS traces are shown in Fig. 1. The multiple peaks are the evidence of dynamic power control — dynamic power control changes AP transmission power several times a day. In practice, however, it is challenging to predict when the AP power will change because it depends on both the instantaneous user density and the traffic loads. As a result, the accuracy of WiFi-fingerprint system can not be maintained unless the entire fingerprint map is updated frequently.

• **Heterogeneous Mobile Devices.** Many existing WiFi fingerprinting techniques *implicitly* assume that homogeneous devices are used for both the offline fingerprint map construction and the online localization. In practice, this is the unlikely case. Different devices may have different RSS readings even when placed at the same location. For example, if a Samsung Galaxy 3 was used to construct the fingerprint map and a user to be localized is using Sony Xperia, there may be no matching between the constructed fingerprint map and the real-time user measurement. Shifting and scaling of the measured RSS values can not solve this problem since even devices of the same model may perform differently [33].

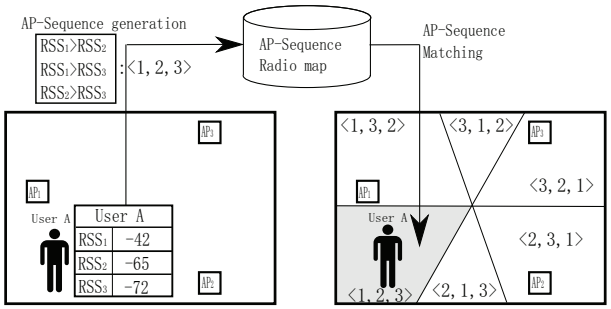


Fig. 2: System overview

4 System Overview

In order to simultaneously tackle all the technical challenges mentioned in our motivation, we propose AP-Sequence fingerprint localization. Figure 2 illustrates a high-level application scenario of AP-Sequence fingerprint localization. When a user A wants to know her location, she first scans the RSS values of APs in her proximity. For example, the user A scans wireless channels and observes AP_1, AP_2 and AP_3 with RSS values of -42 dBm, -65 dBm, and -72 dBm, respectively. Based on the relative difference between RSS values of the three APs, the AP-Sequence of $<1,\ 2,\ 3>$ is generated. Essentially, it is an ordered sequence of APs from high to low in terms of RSS strength. Then the user A uploads AP-Sequence $<1, 2, 3>$ to AP-Sequence fingerprint map server for localization. The AP-Sequence of $<1,\ 2,\ 3>$ is compared to AP-Sequence fingerprint map and she is ultimately localized to a region associated with the best matching AP-Sequence.

Two challenges need to be addressed for the AP-Sequence fingerprint localization: AP-Sequence fingerprint map construction and AP-Sequence matching.

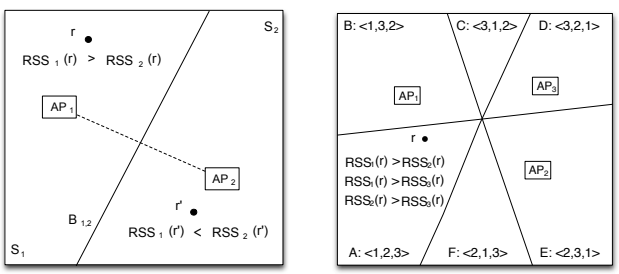
• **AP-Sequence fingerprint map** is constructed by partitioning the area of interest into a set of small regions. To partition the area, the locations of APs are required, which are readily available in most commercial and industrial buildings because their locations are

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2017.2737426, IEEE Transactions on Mobile Computing

often predetermined by WLAN infrastructure designers for maximizing AP coverage and link connectivity. AP manufactures such as ProCurve Networking by HP [34] and Cisco [32] normally provide guidance to assist in determining the optimal AP locations. Even in the case where the AP locations are unknown, there are many effective solutions to localize those APs in practice [35].

• For **AP-Sequence matching**, we adopt a novel sequence matching algorithm that takes the stability of measured RSS values into consideration [5]. This algorithm essentially quantifies the robustness of the measured AP-Sequence against noise and interference, which outperforms mechanisms such as the classic Longest Common Sequence [36].

Next, we provide the details of radio map construction and localization by AP-sequence.



(a) Partition with bisector (b) Partition into 6 regions

Fig. 3: Bisector-based area partition

5 AP-Sequence for the Ideal Case

The key idea behind AP-Sequence fingerprinting is to partition the WiFi-covered area into small regions according to the relative RSS ordering of APs. Ideally, the RSS values of an AP are expected to be inversely proportional to the Euclidean distance between a reference location and the AP [37]. In this ideal case, if the two APs have identical transmission powers, the measured RSS values from these two APs on the bisector of the line connecting these two APs are also identical. This implies that an area with two APs can be cut into two distinct regions by a bisector line and each region has a unique AP-Sequence. However, this bisector-based partitioning approach may not work well in practice because radio signals are subjected to multipath fading and environment noises. In addition, the transmission powers of different APs may also vary over time, and are dynamically adapted by the network controller to maximize bandwidth and coverage.

For presentation clarity, we first describe how to construct the AP-Sequence-based fingerprint map in an ideal scenario where all APs possess equal transmission powers and the received RSS at a receiver is strictly inversely proportional to Euclidean distances to the corresponding APs. In the next section, we present effective and efficient techniques to construct the fingerprint map with accurate AP-Sequences in practice, which handle practical issues such as dynamic power control, signal noises and heterogeneous devices well.

Let us consider an area covered by multiple APs. Any location in this area receives periodic beacon messages from a set of nearby APs. In an ideal scenario, the area can be partitioned into several distinct regions by bisector lines that connect different pairs of APs. For example, consider an area S that is covered by AP_1 and AP_2, as illustrated in Fig. 3(a). S can be partitioned into two distinct regions S_1 and S_2 by the bisector B_{12} of the line that connects AP_1 and AP_2. Let $RSS_i(r)$ denote the RSS value of AP_i measured at location r, then $RSS_1(r) > RSS_2(r)$ is expected to hold for any $r \in S_1$ because the distance from r to AP_1 is always shorter than the distance from r to AP_2. Similarly $\forall r' \in S_2$, we have $RSS_2(r') > RSS_1(r')$ in the ideal scenario.

Aided with the above-mentioned bisector-based partitioning in the ideal scenario, we are able to partition an area with n APs into at most $O(n^4)$ regions [38]. In our experiment with 15 production APs, we are able to partition a $22\ m \times 107\ m$ area into 947 distinct regions. For instance, Fig. 3(b) shows an area that is partitioned into six distinct regions by the bisector lines connecting each pair of three APs. To reflect the order of RSS values from different APs in each partitioned region, we define AP-Sequence in each region by an AP string. Each element in the string represents a specific AP, ordered from the strongest to the weakest RSS (from left to right). For example, the AP-Sequence $<1, 2, 3>$ in region A of Fig. 3(b) indicates that any location p residing in A will experience the received RSS in the order of $AP_1 > AP_2 > AP_3$. The list of AP-Sequences for all partitioned regions is defined as an *AP-Sequence fingerprint map*, which will be used in the localization phase.

6 AP-Sequence In Practice

In the previous section, we have discussed the basic idea behind partitioning the area using bisectors and assigning AP-Sequences to the partitioned regions. However, partitioning with bisectors in practice could be unreliable and inaccurate due to different AP transmission powers and environmental interferences. In this section, we present a novel design that efficiently identifies reliable AP-Sequence in practice with a few (organic) reference training locations.

The objective of area partitioning is to identify regions with which the strengths of RSS received from all APs in the area can be ordered in a reliable way. To precisely identify such regions in practice, RSS values of APs must be collected from every possible locations, which is clearly impractical due to the heavy training overhead. In our design, the identification of these regions is accomplished by identifying *(i)* the shape of boundary that defines any two adjacent regions, and *(ii)* the locations of such boundaries.

6.1 Partitioning with Line Boundaries

Unlike the ideal case, when AP_1's transmission power P_1 is different from AP_2's transmission power P_2, the boundaries of regions may no longer be bisectors. To

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2017.2737426, IEEE Transactions on Mobile Computing

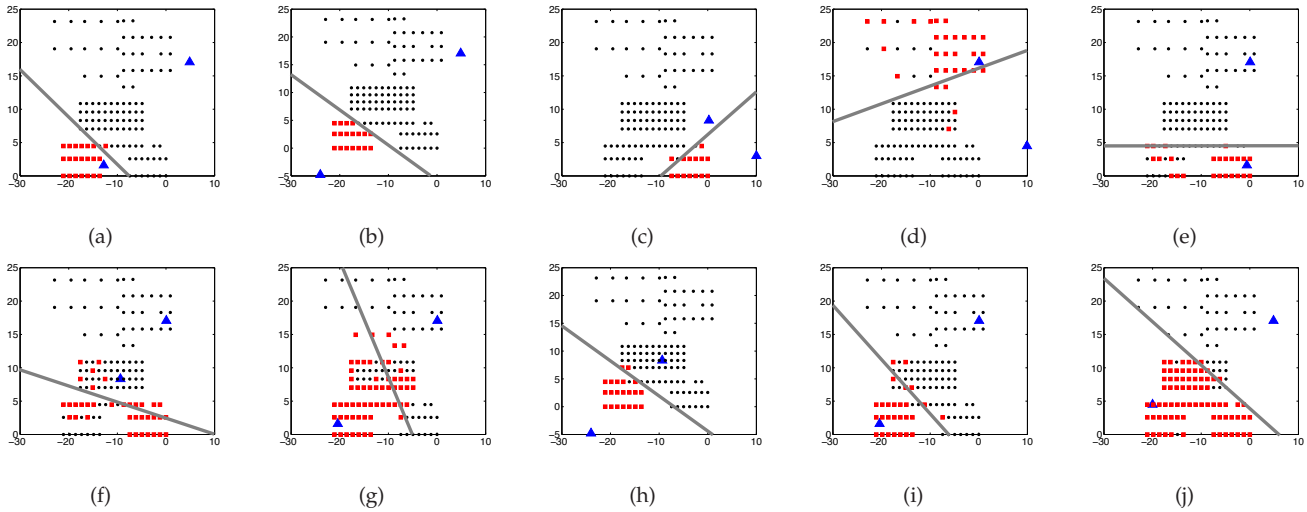


Fig. 4: Experimental results illustrating an area partitioned by a linear region boundary. The two triangles represent AP locations and the grey line denotes the boundary line. Squares and diamonds are reference locations presenting relative RSS differences among two APs.

determine the shape of boundaries in realistic indoor environments in a reliable and accurate manner, we have conducted extensive measurements in a $25\ m \times 25\ m$ area with a complex floor plan and diverse setup of office furniture. Within the area, there are 122 testing locations and 20 APs, with an average distance between adjacent testing locations of 1 m. At each testing location, we collect RSS values from all APs. For a pair of APs (AP_i and AP_j), the testing locations can be categorized into two sets based on their received RSS values. For any reference location r, if $RSS_i(r) > RSS_j(r)$, they are represented as the squares. Similarly, any location r with $RSS_j(r) > RSS_i(r)$ is represented as diamond. Figure 4 shows such categorization for two pairs of APs in our measurements. To validate whether lines are appropriate approximation of region boundaries, we use support vector machine to obtain a line that best classifies two sets of reference locations. We can see that the boundaries of regions for any AP pair can be reasonably approximated by lines in general. In our measurements, the classification of the 122 reference locations is performed with a total number of 190 APs pairs, and the distribution of the misclassified locations obtained throughout these measurements is shown in Fig. 5. We can see that only about 10 reference locations are located on the wrong side of the linear boundary on average.

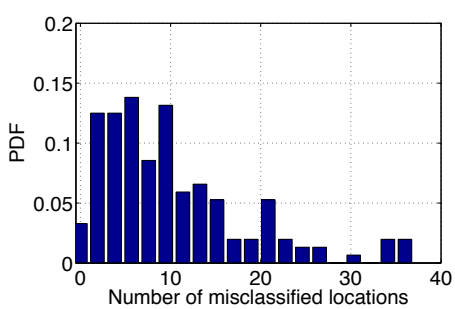


Fig. 5: Number of misclassified reference locations

These measurements show that even with different AP transmission powers and factors such as obstacles and multipath effect, we can still use lines to approximate the boundaries of the partitioned regions. However, as can be observed, these lines are no longer the bisectors of the AP pairs anymore.

6.2 The Location of Boundary Lines

Our design in determining the location of region boundary lines is motivated by the observation that when the transmission powers from APs are different or the RF noises are different due to environment factors, the location of lines defining the region boundaries have to be *shifted* to reflect the actual AP-Sequence in each region. For example, if the transmission power of AP_i becomes greater than that of AP_j, the boundary line $\hat{B}_{ij}$ is more likely to be closer to AP_j than AP_i [1].

6.2.1 Boundary Lines and the Measured RSS

In our system, we utilize the RSS values measured at a few reference locations to identify the correct location of the region boundary lines. The possible location of the region boundary line can be inferred from the measured RSS values at the reference locations. Let us consider the example shown in Fig. 6(a). The measured RSS values from a reference location r are $RSS_2(r) > RSS_1(r)$, therefore the reference location r should belong to a region with AP-Sequence $<2, 1>$. Consequently, the region boundary line should be above r.

In general, RSS measurements at a single reference location are able to infer the location of multiple region boundary lines, because multiple APs can be scanned at the same reference location. Consequently, a few reference location measurements are able to determine the

1. For notation clarity, we use B_{ij} to represent the bisector of AP_i and AP_j in the ideal case, and use $\hat{B}_{ij}$ to denote the boundary line in practice.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2017.2737426, IEEE Transactions on Mobile Computing

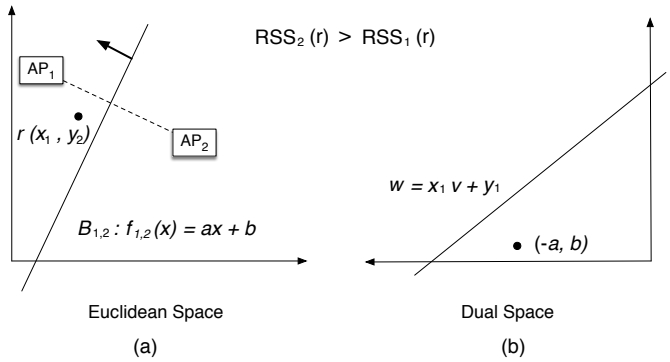


Fig. 6: Infer the location of the boundary line with dual space transformation.

boundary lines to reflect the real-time fingerprint map with AP-Sequence. Our experiment results demonstrate that effective fingerprint map updates and promising localization accuracy can be achieved with as few as 5–10 reference locations for an area over $2100\ m^2$.

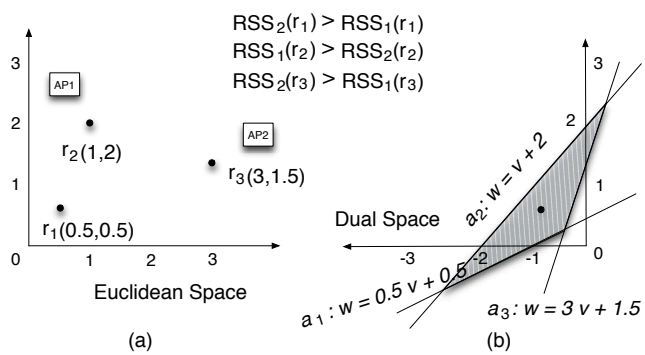


Fig. 7: Estimate the boundary line location

6.2.2 Dual Space Transformation

The measured relative RSS values from APs at individual reference locations can be treated as a constraint on the region boundaries. In the example shown in Fig. 6(a), this constraint is that the boundary line has to be located above the reference location r. In this way, the problem of identifying the location of the region boundary line can be formulated to find a line in the Euclidean space $\hat{B}_{i,j} : y = ax + b$, which satisfies all these constraints. This 2-D search (i.e., with regard to two parameters a and b) imposes significant computation overhead to the localization system, because the entire space may need to be searched in the worst case, i.e., $a, b \in (-\infty, +\infty)$.

We apply the dual space transformation technique [38] on the area of interests to reduce the complexity. Dual space transformation is often used in computational geometry to reduce the search complexity in Euclidean space. Specifically, the dual space transformation in this work transforms the reference locations into lines. As a result, the search for the boundary line in the 2-D Euclidean space can be simplified to search for a point in the dual space and accomplished in $\mathcal{O}(n^2)$ time, as we will see later in this section.

Specifically, the lines and reference locations are transformed as follows in the dual space and vice versa.

$$\begin{array}{rcl} \underline{\text{Euclidean Space}} & & \underline{\text{Dual Space}} \\ B_{ij} : f_{ij}(x) = ax + b & \longleftrightarrow & \beta_{ij} : (-a, b) \\ r_i : (x_i, y_i) & \longleftrightarrow & \alpha_i : w = x_i \cdot v + y_i \end{array} \quad (1)$$

Figure 6(b) shows an example of the dual space transformation for the scenario in Fig. 6(a). The original bisector $B_{1,2} : f_{1,2} = ax + b$ and the reference location $r(x_1, y_1)$ are transformed to a point $(-a, b)$ and a line $w = x_1 v + y_1$ in the dual space respectively.

An important property of this dual space transformation is that it preserves the proximity relationship, i.e., if a line is below/above a reference location in Euclidean space, then its transformed point in the dual space is also below/above the transformed line from that reference location. In Fig. 6, because the boundary line should be above r in the Euclidean space, we know the point in the dual space corresponding to the region boundary line should be above $(-a, b)$ as well.

6.2.3 Identify the Boundary Lines in the Dual Space

The proximity property of dual space allows us to easily determine a feasible boundary line under the constraints imposed by the measured RSS values.

Let us take Fig. 7(a) as an example, where three reference locations ($r_1 = (0.5, 0.5)$, $r_2 = (1, 2)$, and $r_3 = (3, 1.5)$) exist in an area covered by two APs. If the received signal strength at reference location r_1 satisfies $RSS_2(r_1) > RSS_1(r_1)$, then the boundary line should be above r_1. Similarly, as the measured RSS at r_2 and r_3 are $RSS_1(r_2) > RSS_2(r_2)$ and $RSS_2(r_3) > RSS_1(r_3)$, the boundary line should be below r_2 but above r_3. Then the dual space transformation is performed to identify the location of the region boundary.

Figure 7(b) shows the dual space transformation results. According to Equation (1), the three reference locations are mapped to

$$\begin{array}{rcl} r_1 : (0.5, 0.5) & \longleftrightarrow & \alpha_1 : w = 0.5v + 0.5, \\ r_2 : (1.0, 2.0) & \longleftrightarrow & \alpha_2 : w = v + 2, \\ r_3 : (3.0, 1.5) & \longleftrightarrow & \alpha_3 : w = 3v + 1.5. \end{array}$$

According to the proximity property of dual space, if the boundary line $y = ax + b$ is located above r_1 in the Euclidean space, its transformed point $(-a, b)$ in the dual space must also be above the line α_1 in the dual space. Similarly, $(-a, b)$ must be below line α_2 and above α_3. This is represented in Fig. 7(b) as the shaded polygonal area. Then we estimate $(-a, b)$ as the polygon centroid, which can be obtained by existing algorithms [39]. Finally we remap $(-a, b)$ to the Euclidean space and the region boundary line is obtained.

In a system with n reference locations, we need a computation time of $\mathcal{O}(n)$ to transform the n sets of measured RSS values to the constraints on the region boundary lines, and another $\mathcal{O}(n)$ time is needed to transform the reference locations to the corresponding

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2017.2737426, IEEE Transactions on Mobile Computing

lines in the dual space. Then based on these mapped lines in the dual space, a time of $\mathcal{O}(n^2)$ is needed to identify the points in the dual space that define the polygon in which $(-a, b)$ is located (e.g., the shaded area in Fig. 7(b)). Finally, an $\mathcal{O}(n)$ time is needed to identify the polygon centorid [39]. As a result, we can identify the boundary of the regions formed by two APs with a time of $\mathcal{O}(n^2)$. Fortunately, the total number of regions in the fingerprint map is bounded by $\mathcal{O}(m^4)$, where m is number of APs [30]. Therefore, the region-partitioning algorithm has polynomial time running complexity.

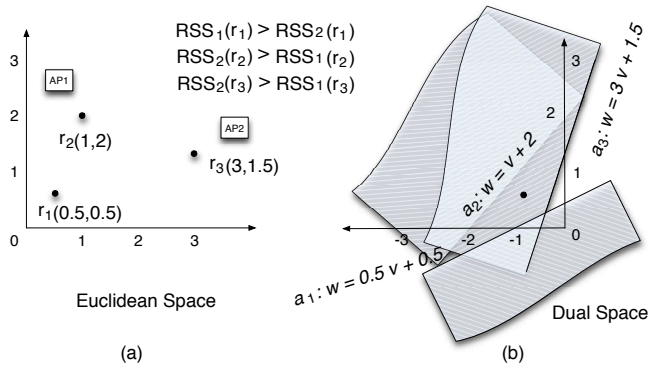


Fig. 8: A case where no boundary line satisfying all measurements exists.

Because lines are essentially an approximated representation of the region boundaries, it is possible that a boundary line $y = ax+b$ satisfying all the measurements does not exist in practice. Figure 8 illustrates an example of such conflictions, where the received RSS values show that $RSS_1(r_1) > RSS_2(r_1)$, $RSS_2(r_2) > RSS_1(r_2)$, and $RSS_2(r_3) > RSS_1(r_3)$. These indicate that the boundary line $y = ax + b$ should be located below r_1, above r_2, and above r_3. Transforming the three constraints into the dual space, we have the three shaped area as shown in Fig. 8(b). To identify the region boundary line satisfying all the three constraints, we need to find a point in the dual space that is covered by all these regions, which clearly does not exist in this example.

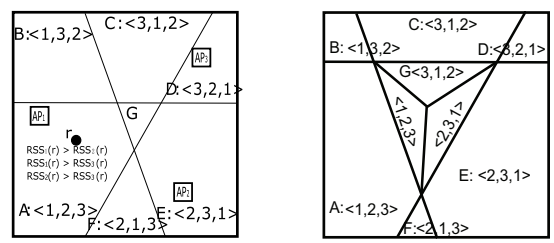
It is easy to detect such conflicts in the dual space. For a particular region boundary, if there is no overlapping area that satisfies all the measurements, the conflict has occurred. To resolve this conflict, we first identify all the overlapping regions in the dual space that are formed based on the RSS measurements from more than one reference locations, and then use the number of reference locations as the weight of these regions. Finally, we select the center of mass of all these weighted regions as our best approximation of $(-a, b)$. Note we can continuously improve the area partitioning when new reference locations become available.

7 System Implementation

AP-sequence fingerprint localization system is running on a server and the user application is implemented on Android devices. The server performs AP-Sequence assignment, localization, and fingerprint map-update. The user application scans RSS and upload it for localization and it also search for any RSS peak as an organic landmark.

7.1 Region Partitioning

The location information and SSIDs of APs, which are required to construct AP-Sequence fingerprint map, were provided by our campus IT managers. We first use this information to generate bisector lines for every pair of APs on the area of interest by the method introduced in Section 5. Initially, site surveyors collect RSS readings near the APs. Based this initial measurement set, we apply the dual space transformation (described in Sec. 6.2.2) to obtain the boundary lines, and thus the area is partitioned into a set of regions.



(a) Partitioned area (b) Divide G further

Fig. 9: AP-Sequence assignment example

7.2 AP-Sequence Assignment

After partitioning the area into regions, we assign an unique AP-Sequence to each region according to the relative position of the region to each of its boundaries. As shown in Fig. 9(a), the reference location r has an AP-Sequence of $<1, 2, 3>$ because in the corresponding region A, we have $RSS_1(r) > RSS_2(r)$, $RSS_1(r) > RSS_3(r)$, and $RSS_2(r) > RSS_3(r)$ by checking the relative position of r for each of its boundary lines. We can assign AP-Sequences to other regions (e.g., regions B-F as shown in Fig. 9(a)) in a similar manner, except for region G. In region G, we have $RSS_1(r) > RSS_2(r)$, $RSS_2(r) > RSS_3(r)$, and $RSS_3(r) > RSS_1(r)$, which is a conflicting sequence. Such conflicted regions exist because these regions are not partitioned in an exact manner, i.e., region boundaries are approximated by lines. We handle each conflicted region by first dividing it into non-conflicting sub-regions, and then assigning AP-Sequence to each sub-region. Specifically, for each sub-region, we assign it an AP-Sequence that is identical to the one assigned to its adjacent non-conflicting region, as in Fig. 9(b). Because all the regions are always convex polygon, the conflicting regions can be divided into a set of triangles by first identifying the centroid of the region and then connecting every vertex of the region to its centroid. Each triangle must share an edge with one adjacent region so that each triangle can use the AP-Sequence of its adjacent region.

To validate the higher stability of AP-Sequence over traditional absolute RSS based fingerprinting, we collect RSS values from 8 APs for a period of about 30 minutes

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2017.2737426, IEEE Transactions on Mobile Computing

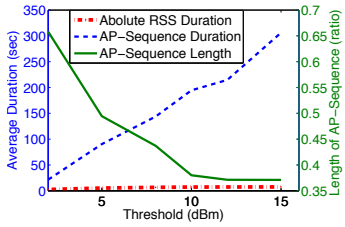
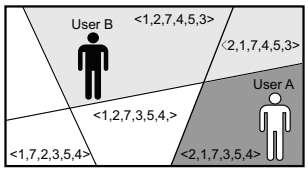


Fig. 10: Stability of AP-Sequence



(a) User and AP-sequences

User A		User B	
RSS_2	-52	RSS_2	-62
RSS_1	-62	RSS_1	-63
RSS_7	-75	RSS_7	-74
RSS_3	-82	RSS_4	-80
RSS_5	-82	RSS_5	-82
RSS_4	-120	RSS_3	-120

(b) Measurement

Fig. 11: AP-seq. localization example

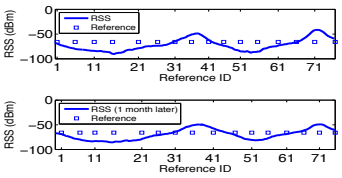


Fig. 12: RSS peaks at different time

with a sampling frequency of 0.25 Hz. For the absolute RSS based fingerprinting, we refer the fingerprint as *changed* if the difference between two consecutive RSS values from either of the two APs is larger than the threshold value. On the other hand, we refer the AP-Sequence based fingerprint as *changed* if the relative order of the RSS values from a *reliable* AP pair are changed for two consecutive samples. The stability of the two fingerprinting mechanisms is shown in Fig. 10, where the dash blue line indicates duration of AP-Sequence. We observed that the duration of absolute RSS based fingerprint was less than 10sec while AP-Sequence lasted more than few hundreds of second. It is interesting to note that increasing the threshold value increases the duration of AP-Sequence linearly. In contrast, the average length of AP-Sequence decreases fast. Its length reduce to 40% of its original length when the threshold increase to 10dBm. The original length of AP-Sequnce is eight since 8 APs were used for this experiment. The length of AP-Sequence affects the localization accuracy so we used 10dBm for the threshold value during our evaluation.

7.3 Localization using AP-Sequence

After assigning an AP-Sequence to each region, we now present a novel localization scheme that utilizes such assigned AP-Sequences. With the assigned unique AP-Sequence of each region, to determine a user's location, we first obtain a measured AP-Sequence by comparing the RSS values among different pairs of APs. (similar to how we assign an AP-Sequence as discussed in Section 7.2). Then the location is estimated to be at the center of the region with the AP-Sequence that most closely matches to the measured AP-Sequence. A straightforward approach is to apply sequence matching algorithms such as longest common sequence. However, these approaches may not generate reliable localization as they do not consider the stability of measured RSS, especially when only one single scan is available.

To achieve stable and effective localization, we present a novel sequence matching scheme that takes real-time RSS variations into consideration. Even though RSS values are not stable over time, our empirical experiments show that the AP-Sequence of an AP pair typically remains stable if their measured RSS difference is greater than 10 dBm. Similar observations are reported in [5], [40]. Therefore for a given AP pair, if their measured RSS difference is greater than 10 dBm, we define it as a reliable AP pair. Our sequence matching scheme compares the number of reliable AP pairs appeared in the AP-Sequence of individual regions.

Figure 11 illustrates an example on the AP-Sequence matching with two users. The area is partitioned into regions and each assigned an AP-Sequence. The RSS values of each AP measured by user A and B are shown in Fig. 11(b). For APs that cannot be scanned at a reference location, we use a low RSS value of -120 dBm to complete the sequence. For user A, her reliable AP-Sequence pair is $\{(2,1), (2,7), (2,3), (2,5), (2,4), (1,7), (1,3), (1,5), (1,4), (7,4), (3,4), (5,4)\}$. Then we match these individual AP pairs to the regions. The region with AP-Sequence $<2,1,7,3,5,4>$ contains 12 matching AP pairs, while regions with AP-Sequence $<1, 2, 7, 4, 5, 3>$, $<1, 2, 7, 3, 5, 4>$, $<1, 7, 2, 3, 5, 4>$, and $<2, 1, 7, 4, 5, 3>$ contains 9, 11, 10, and 10, respectively. Therefore user A is localized to the center of region $<2,1,7,3,5,4>$. It is possible that a user may have multiple best matching regions. For example, user B's reliable AP pairs are $\{(2,7), (2,4), (2,5), (2,3), (1,7), (1,4), (1,5), (1,3), (7,3), (4,3), (5,3)\}$, and both regions with AP-Sequence $<1,2,7,4,5,3>$ and $<2, 1, 7, 4, 5, 3>$ contain 11 matching pairs. To resolve this conflict, we localize B to the center of the two regions.

7.4 Fingerprint Map-Update

Channel quality and transmission powers of APs change overtime. Because of that, the fingerprint map also has to change overtime. Since our fingerprint map construction requires only a small number of reference locations, a site surveyor can periodically collect RSS scans from a few random reference locations. Alternatively, we can leverage the participatory-sensing-based approaches [1], [2], where participators upload their RSS scans along with their ground truth locations. This is a fitting approach for updating our fingerprint map since device heterogeneity has almost no effect on our localization accuracy. However, both of these approaches does not scale well since utilizing the site surveyor for long period is not cost effective and it is difficult to provide sufficient incentives to participators for their long term contribution.

In order to scale, the indoor localization system should be fully self-contained after the initial setup. To address this challenges, we utilize the concept of *organic landmarks* in our system design. Beside the naturally existing organic landmarks [3], we also utilized RSS peaks as another landmark. A distinctive RSS peak can be observed easily when a user walks across the coverage of

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2017.2737426, IEEE Transactions on Mobile Computing

any APs. In Fig. 12, we show the received RSS values of an AP when a user moves along a trajectory over a month's time. The location of RSS peak is very reliable and unaffected by dynamic power control. Similar observations have been reported several recent works, such as Walkie-Markie [41]. Furthermore, the RSS peaks are also distributed quite evenly since the APs are well spread over the indoor area. Therefore, using RSS peaks as the landmarks is effective in fingerprint map-update.

The procedure of map-update follows two phases: peak discovery and peak scan. In the peak discovery phase, the site surveyor explores the locations of these RSS peaks. Generally, one RSS peak is detected for each AP. But if there are multiple peaks for a same AP only the strongest peak (one closest to the location of AP) is used as a landmark. In the peak scan phase, the peak is continuously searched from RSS scans of all users by WiFi-Mark Detection [41], while providing the indoor navigation service to the users. When the peak of some AP is detected it is compared with stored peak signals of corresponding APs. If it is a good match the RSS scans of all APs at that peak location is used in the region partitioning for updating the fingerprint map. Therefore, this unsupervised fingerprint map update incurs no additional cost except for the initial fingerprint map.

8 Experiment

Experiments are conducted at two different sites **I** and **II**, shown in the Figure 13 and Figure 14 respectively. These two sites are chosen to demonstrate the effect of environmental noises on the performance of AP-Sequence.

For over one month, users with different smartphones walked around the corridor during both day time and night time. Each time, they followed same trajectory to maintain evaluation consistency. Their ground truth were manually recorded whenever they passed by the reference points on their mobile application. In the site **I**, we found 19 APs (eight of them are private APs) and total of 365 reference points were regularly placed along the corridor with interspacing distance of around 1m. In the site **II**, we found 29 APs (all are part of one WLAN) and total of 78 reference points were regularly placed along the corridor with interspacing distance of around 3m.

For evaluating overhead and reliability of AP-Sequence, its performance is compared with FreeLoc [5] and Horus [23]. For many cases, localization accuracy of Horus is omitted since its performance was nowhere near comparable to both AP-Sequence and FreeLoc. Each bar in the figure is an average of 20 random trials and each error bar is indicating the 90th percentile localization accuracy. For comparison fairness, a center of mass of the top three ranked locations is used as an estimated position for both Horus and FreeLoc. If the estimated position is not found we use the most recent estimation instead.

Our objective here is to show the fundamental performance of AP-Sequence fingerprint localization. There are

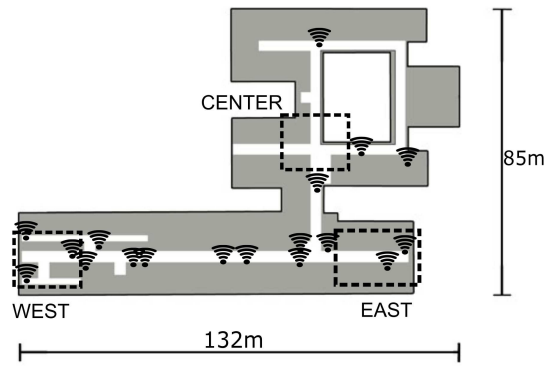


Fig. 13: Site I is a complex environment with L-shaped corridor. Grey areas represent private offices and inaccessible rooms. Most walls are concretes and bricks

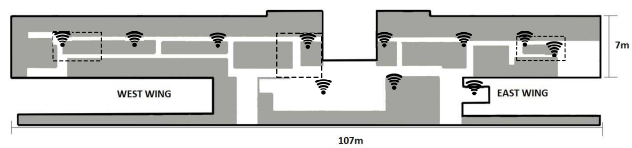


Fig. 14: Site II is an office like environment with almost straight corridor connecting east and west wing. Grey areas represent office tables and small rooms. It is mostly an open space and most of the walls are made out of glasses and plasterboards.

many position enhancing techniques like time averaging and limiting feasible positions based on user's movement. But we did not apply them here since they often hide the real behavior of core localization techniques. Nevertheless, such enhancing techniques should be applied in practice since they would certainly improve the performance of any indoor localization techniques.

Table 1 shows a default setting of all the experiments, unless stated otherwise.

TABLE 1: Default Experiment Setting

Configurations	Site I	Site II
Size of Area	132m × 85m	107m × 7m
Size of Training Points	365/365	78/78
Number of APs	19/19	29/29
Number of Scans	2	2
Localization Time	24 hrs	24 hrs
Device	Mi Note 4G	Sony LT26w

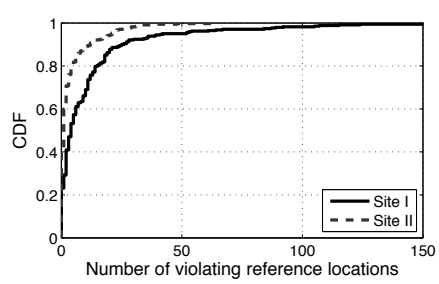
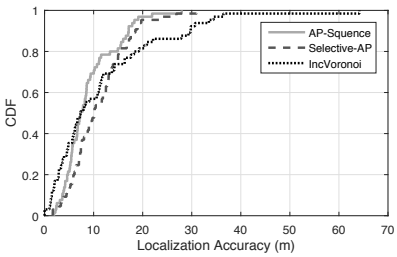
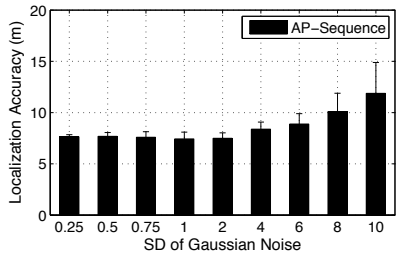


Fig. 15: CDF plot showing the number of misclassified reference locations in AP-Sequence fingerprint map at both testing sites

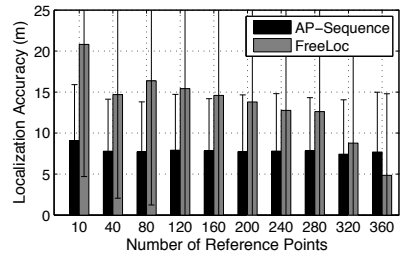
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2017.2737426, IEEE Transactions on Mobile Computing



(a) Impact of calibration

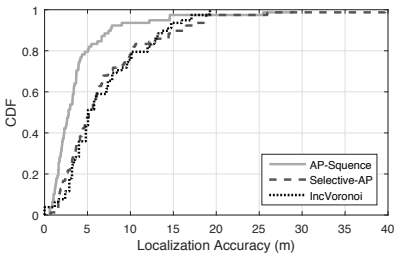


(b) Effect of AP location error

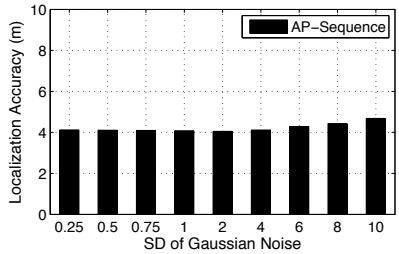


(c) Impact of number of reference points

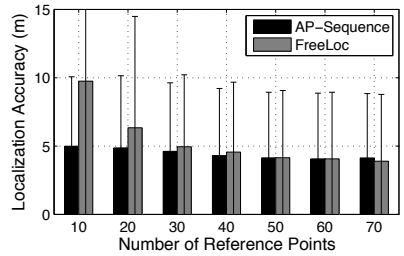
Fig. 16: Site I



(a) Impact of calibration



(b) Effect of AP location error



(c) Impact of number of reference points

Fig. 17: Site II

• **Impact of Calibration.** The core motivation of AP-Sequence is to reduce the overhead in constructing the WiFi-based fingerprint map. But, low calibration of AP-Sequence may compromise on accurate modeling of complex path loss by a set of linear boundaries. We demonstrate the performance of AP-Sequence by conducting the experiments in Site I and Site II. In the experiments, we have observed that radio signals propagate more irregularly in Site I due to thick walls and its complex layout. Therefore, as shown as Figure 15, an average of 11 references in Site I were misclassified by the linear boundaries calibrated by AP-Sequence. On the other hand, only about 4 references were misclassified from Site II. Also, AP-Sequence fingerprint map in site I contained many larger regions compared to Site II since many private APs were placed unevenly round the testing Site I. Clearly, simple layouts like Site II is favorable to AP-Sequence.

Figure 16 and Figure 17 show AP-Sequence with small calibration can provide 7.52m average localization accuracy at site I and 4.09m average localization accuracy at site II, respectively. As expected, the localization accuracy of AP-Sequence at site I is lower than the site II for all cases but it could still provide a room level accuracy.

Although AP-Sequence reduces the overhead associated with calibration, it is not a calibration-free approach. Therefore, we compare AP-Sequence with two calibration-free approaches, Selective-AP [27] and IncVoronoi [28], and show the trade-off between small calibration overhead and expected localization performance gain. During the tests, user defined parameters of Selective-AP and IncVoronoi were manually tuned until they provide best possible localization accuracy on Site I and Site II. For example, a constant C used in IncVoronoi for its obstacle handling was set to 1 on Site I and 0 on Site II — applying obstacle handling did not improve IncVoronoi's accuracy on Site II, likely because many walls on Site II being the light-partitioning walls rather than thick concrete walls.

Figure 16(a) and 17(a) show the localization accuracy of AP-Sequence, IncVoronoi, and Selective-AP. AP-Sequence outperforms Selective-AP by up to 33% and 50% on Site I and II, respectively, and outperforms IncVoronoi by up to 57% on Site II. This is because (i) Selective-AP assumes ideal path loss model and represents every boundary between each pair of APs as a bisector, and (ii) IncVoronoi handles obstacles only as good as the building layout information. However, RF signal undergoes complex path loss even in environment like Site II, which is of many open space and with only light partitioning between the rooms — considering walls as homogeneous objects cannot capture such complex signal path loss. Figure 16(a) shows IncVoronoi suffers large localization errors as large as 30m, likely because Site I has many walls with varying thickness and materials which can not be modeled as a homogeneous obstacles neither. That said, small calibration is needed to uncover the effect of dynamic indoor environment on RF signal propagation, as with AP-Sequence.

• **Impact of AP Position Accuracy.** Next, AP-Sequence uses locations of APs in the process of generating AP-Sequence fingerprint map. Therefore, performance of AP-Sequence clearly depends on the accuracy of AP locations. In order to test its effect, Gaussian noise was added to the location of each AP before generating the AP-Sequence fingerprint map. Figure 16(b) and Figure 17(b) shows that AP-Sequence is robust against this AP location error until standard deviation (SD) of Gaussian noise is larger than 4 (equivalent to noise variance of 16m).

• **Impact of Number of Reference Points.** In this experiment, a subset of reference points was randomly

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2017.2737426, IEEE Transactions on Mobile Computing

selected for generating fingerprint maps of AP-Sequence and FreeLoc. Figure 16(c) and Figure 17(c) show that AP-Sequence can achieves a localization accuracy of 7.6m at site I and 5m at site II by utilizing just 11% and 14% of its total reference points, respectively. Whereas, FreeLoc suffers greatly when available reference points are low. It only outperforms AP-Sequence when about 90% of reference points are available for generating its fingerprint map. This is expected for FreeLoc since an unknown position is mapped to its nearest reference point during its localization phase. Therefore, its localization accuracy depends on the granularity of reference points covering the area.

Actually, FreeLoc outperforming AP-Sequence is an artifact of how localization test was performed. Since reference points were used for both generating the fingerprint map and testing for localization, FreeLoc could perfectly localize a position with zero error while AP-Sequence can never achieves such accuracy since it uses center of the regions as an estimated position. Therefore, this type of localization test clearly favors the FreeLoc. For fair comparison, we made a small modification to AP-Sequence. Instead of using a center of the region, we allowed AP-Sequence to use an average location of reference points as an estimated position if some of the reference points are present in the region. With this modification, we noticed that AP-Sequence can always outperform FreeLoc. This result is not present in the graph since it is unlikely to happen in practice.

Being able to generate, maintain, and update fingerprint map, is a key feature of AP-Sequence which differentiates it from the rest of fingerprint-based localizations. Therefore, it is ideal to combine AP-Sequence with crowdsourcing based training techniques such as ZEE [2].

• **Impact of Selected Reference Points.** In order to show the effect of displacement of reference points used for generating fingerprint map, we built 4 different fingerprint maps using RSS reading collected from 4 different regions in Site I and Site II. Three fingerprint maps were calibrated with RSS readings collected from 3 different sets of 20 reference points, each from east wing, west wing, and centre, respectively. Another fingerprint maps were calibrated with RSS readings collected from 29 reference points locating closest to 29 APs. This set is denoted as near APs and it shows an impact of uniformly distributed reference points since APs are generally well distributed around the indoor area. Using the RSS values collected from these four sets of references, we have generated 4 different fingerprint maps for each Site I and Site II. Each fingerprint map was evaluated by applying it to a same a trace file containing RSS value from every training points for localization.

Figure 18 shows that the displacement of reference points affects the localization accuracy. When the reference points were all clustered around the east side of both sites the average localization accuracy of our approach was the worst. This is because some APs on the west wing were never detected at some reference points on east wing. The center set provides better localization accuracy than west or east set since the reference points at the center can detect most of the APs and their measured signal strengths were generally strong and unaffected by the channel noise and the multipath. Uniformly selecting the reference points provides good localization accuracy on both site. Note that the the uniform selection, albeit performs the best in the four reference points distributions, may still suffer from conflictions — some RSS readings at reference points from east wing conflict with RSS readings at reference points from the center, thus obfuscating AP-Sequence to find the correct locations of boundaries in the fingerprint map. We will elaborate on identifying the optimal set of reference points in our future work, as will be explained in Sec. 9.

• **Impact of Size of RSS Scan.** The heavy training is the main overhead of any fingerprint-based localizations like FreeLoc and Horus. They require a large number of reference points to cover the area of an interest and a large number of scans at each reference point for building an accurate fingerprint map.
To show the impact of number of scans, we collect 1000 WiFi scans from every reference location at site **II**. The average RSS and standard deviation are estimated from those scans and then used to build the fingerprint map of Horus. The average RSS is used for building the fingerprint map of FreeLoc and AP-Sequence as well.

Figure 19 shows the effect of number of scans. AP-Sequence and FreeLoc are almost unaffected by the number of scans because they both use relative differences between two RSS values from a pair of APs. Whereas, Horus uses absolute RSS values. Therefore, its performance clearly depends on the number of scans. Although Horus starts to outperform FreeLoc and comparable to AP-Sequence only when there are more than 250 scans at every reference point, its localization accuracy reduces otherwise.

• **Reliability of Fingerprint Map Over Time.** Performances of any WiFi fingerprint based localization depends on the environment. Specifically, it depends on the time of localization, the number of available APs, and device heterogeneity. To show the impact of the time of localization, we have conducted experiments for over a month at site **II**. The fingerprint map is generated only once from the RSS scans collected from our initial set of experiments. The same devices are used for both training and localization.

Figure 20 shows the performance of AP-Sequence and FreeLoc at different times. Interestingly, AP-Sequence maintains its localization accuracy of 4m for about a week and then reduces to above 6m after a month. FreeLoc also shows a similar behavior but its accuracy degrade more gradually after a day (24 hrs). From this result, it is evident that RSS noises or temporal variation within 24 hrs period is small and it is not sufficient to change the relative RSS order of two APs. Therefore, both AP-Sequence and FreeLoc tolerates the temporal variation for about a week.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2017.2737426, IEEE Transactions on Mobile Computing

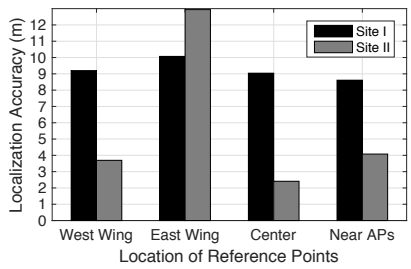


Fig. 18: Impact of reference point.

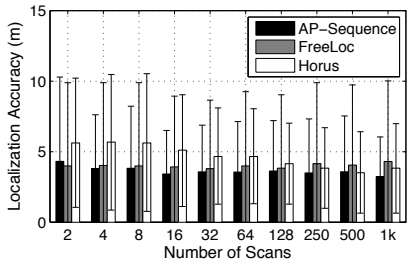


Fig. 19: Impact of number of scans.

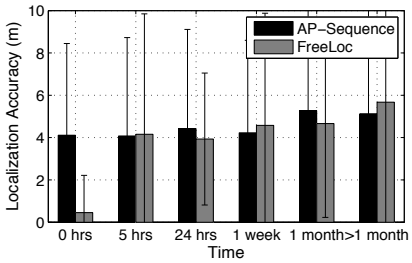


Fig. 20: Performance overtime.

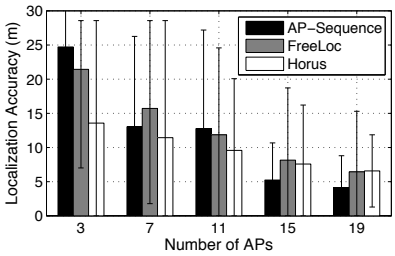


Fig. 21: Impact of the number of available APs.

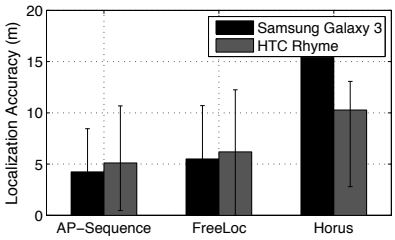


Fig. 22: Impact of heterogeneous devices.

We have also observed that transmission powers of some APs are automatically adjusted several times a day due to dynamic power control mechanism. But the results shows both AP-Sequence and FreeLoc show some robustness against this dynamic power control. The performance of both AP-Sequence and FreeLoc is effected only when the significant relative ordering of AP pairs are changed. This means there is a significant change in the environment or transmission power due to dynamic power control.

Another noticeable difference between FreeLoc and AP-Sequence is at time 0 in Figure 20. At this time, an average localization error of FreeLoc is close to zero. Whereas, the average localization error of AP-Sequence is 4.05m. That is when the users are localized right after generating the fingerprint map. This is expected outcome since a linear boundary in the AP-Sequence fingerprint map is only an approximation of a real boundary of arbitrary shapes. During the process of generating its fingerprint map, it loses some information about the radio signal and its correlation to the specific reference point. But, after just 5 hours later, this benefit fade away and the average localization error of FreeLoc increase to 4m. This shows that the short temporal variation of radio signal has a significant impact on the localization accuracy of FreeLoc also.

Here we would like to highlight that the performance of FreeLoc is comparable to AP-Sequence only when a full set of reference points is utilized for generating its fingerprint map. In order to maintain its localization accuracy, FreeLoc would need to update its fingerprint map daily. This is not a very practical solution since it would need to train entire reference points again. On the other hand, AP-Sequence only needs to train about 10% of reference points to maintain its localization accuracy.

• **Impact of Available APs.** To show the impact of available APs in the environment, we randomly removed some APs and averaged 10 random trials for each case. In site **II**, 11 APs are from the same floor and 8 APs are from different floors. APs from different floors are projected vertically down or up to our testbed. Figure 21 shows the performance of AP-Sequence, FreeLoc, and Horus under different number of APs. The effect of available APs is rather significant for all three designs. When there are 19 APs available, all of them could maintain the average localization accuracy of 6m and below. However, when there are less than 10 APs, none of them could correctly localize the users. One interesting observation is that the performance of AP-Sequence and FreeLoc is highly depending on the available APs. When there are enough APs available, its localization accuracy is better than Horus. On the other hand, when the number of available APs is low, the performance of AP-Sequence is the worst. This is because localization accuracy of AP-Sequence highly depends on the length of relative ordering of APs. If it is as low as 3, it can not generate sufficient number of regions in the fingerprint map which results in a poor performance. FreeLoc also suffers for the same reason but it is still able to differentiate one reference point from the other with a few strong APs. When number of APs increases above 7 AP-Sequence quickly outperforms FreeLoc since number of regions in the AP-Sequence fingerprint map increases exponentially. Horus is better than both AP-Sequence and FreeLoc in case of low number of APs. This is because absolute RSS values of same AP can still be different at different locations due to other channel effects like multipath and interference.

• **Impact of Heterogeneous Devices.** To show the impact of heterogeneous devices, a user carries three different brands of smartphones. The WiFi fingerprint map is generated using LG Nexus4 smartphone. Figure 22 shows the impact due to heterogeneous devices on the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2017.2737426, IEEE Transactions on Mobile Computing

three schemes. As seen, the localization accuracies of AP-Sequence and FreeLoc is almost the same for both devices while the localization accuracy of Horus is very different between two devices. FreeLoc can tolerate the device heterogeneity because it focuses on the overall relationship between RSS from APs rather than individual RSS values [5]. AP-Sequence can also tolerate the device heterogeneity due to the same reason as FreeLoc since AP-Sequence also focuses on the relative RSS rather than absolute RSS values.

During this experiment, we also observe that the performance of both AP-Sequence and FreeLoc are affected by which device is used for generating WiFi fingerprint map. If a device with low antenna sensitivity, like HTC Rhyme smartphone, is used for generating the fingerprint map, localization accuracy reduces noticeably. This is because RSS from some APs are not detected by this smartphone and the effect of it is like having low number of available APs. In order to test the effect of this, we have generated a fingerprint map with 4 devices: Samsung Galaxy, HTC One X, MotoXT910, and SonyLT26w. At each reference location the RSS reading from one of the device was chosen randomly. In this case, the average localization of AP-Sequence and FreeLoc were 5.03m and 4.49m, respectively. This result indicates that the effect of device heterogeneity is negligible as long as a good mix of devices are used for map calibration.

9 Further Discussions

Here we present a few further discussions on AP-Sequence.

• **Limitation.** The low overhead of AP-Sequence is achieved based on the availability of AP locations, although it does not require their precise locations. AP-Sequence does not provide a way of identifying the minimum set of reference locations that is necessary to achieve certain localization accuracy — it only guarantees a low number of initial calibrations will provide higher returns in indoor localization accuracy. Also, we represent the boundaries between pairs as lines, which may not always hold, e.g., in case of a pair of APs having large difference in their transmission powers.

• **Targeted Market.** In the domain of WiFi based fingerprint localization, its initial high overhead and map maintenance cost was its main draw back. Calibration-free approaches do overcome this draw back by trading the localization accuracy or by demanding precise information about the indoor environment. AP-Sequence sits between this calibration-free approach and full scale fingerprint localization with the high overhead. Our main contribution is that we provide an efficient way of utilizing the calibration overhead in constructing (and maintaining) the fingerprint map.

• **Future Directions.** AP-Sequence achieves low-cost WiFi fingerprinting. We will further improve it by (i) exploring the identification of the optimal set of reference points, whose necessity has been experimentally shown in Sec. 8, (ii) investigating its potential integration with state-of-the-art indoor localization techniques such as [27], [28].

10 Conclusion

We design AP-Sequence, a robust metric for fingerprinting in indoor localization, which reduces implementation overheads while ensuring high localization accuracy. A key idea behind AP-Sequence is to use a robust fingerprinting like the relative RSS differences among different APs and utilize region-partitioning technique with AP-Sequence. We implemented an AP-Sequence based indoor localization and have extensively evaluated its performance over long-term experiments.

Acknowledgment This research was supported in part by the state key laboratory of industrial control technology with Grant ICT1600206, NSF with grants CNS-1539047, CNS-1652669, and CNS-1527727.

References

[1] Z. Yang, C. Wu, and Y. Liu, "Locating in fingerprint space: wireless indoor localization with little human intervention," in *MobiCom'12*, 2012.
[2] A. Rai, K. K. Chintalapudi, V. N. Padmanabhan, and R. Sen, "Zee: zero-effort crowdsourcing for indoor localization," in *MobiCom'12*, 2012.
[3] H. Wang, S. Sen, A. Elgohary, M. Farid, M. Youssef, and R. R. Choudhury, "No need to war-drive: unsupervised indoor localization," in *MobiSys'12*, 2012.
[4] H. Liu, Y. Gan, J. Yang, S. Sidhom, Y. Wang, Y. Chen, and F. Ye, "Push the limit of WiFi based localization for smartphones," in *MobiCom'12*, 2012.
[5] S. Yang, P. Dessai, M. Verma, and M. Gerla, "FreeLoc: Calibration-free crowdsourced indoor localization," in *INFOCOM'13*, 2013.
[6] S. Sen, J. Lee, K.-H. Kim, and P. Congdon, "Avoiding multipath to revive inbuilding WiFi localization," in *MobiSys'13*, 2013.
[7] C. Wu, Z. Yang, Z. Zhou, Y. Liu, and M. Liu, "DorFin: WiFi fingerprint-based localization revisited," *CoRR*, 2013.
[8] P. Bolliger, "Redpin - adaptive, zero-configuration indoor localization through user collaboration," in *MELT'08*, 2008.
[9] J. g. Park, D. Curtis, S. Teller, and J. Ledlie, "Implications of device diversity for organic localization," in *INFOCOM'11*, 2011.
[10] J. Ledlie, J. g. Park, D. Curtis, A. Cavalcante, L. Camara, A. Costa, and R. Vieira, "Mole: A scalable, user-generated wifi positioning engine," in *IPIN '11*, 2011.
[11] E. H. John Krumm, Gerry Cermak, "Rightspot: A novel sense of location for a smart personal object," in *UbiComp'03*, 2003.
[12] M. B. Kjrgaard, "Indoor location fingerprinting with heterogeneous clients," *Pervasive and Mobile Computing*, vol. 7, no. 1, pp. 31–43, 2011.
[13] A. Subramanian, P. Deshpande, J. Gaojgao, and S. Das, "Drive-by localization of roadside wifi networks," in *INFOCOM'08*, 2008.
[14] A. W. T. Tsui, W.-C. Lin, W.-J. Chen, P. Huang, and H.-H. Chu, "Accuracy performance analysis between war driving and war walking in metropolitan wi-fi localization," *IEEE Transactions on Mobile Computing*, vol. 9, no. 11, pp. 1551–1562, 2010.
[15] A. Kosba, A. Saeed, and M. Youssef, "Rasid: A robust wlan device-free passive motion detection system," in *PerCom'12*, 2012.
[16] J. Jun, Y. Gu, L. Cheng, B. Lu, J. Sun, T. Zhu, and J. Niu, "Social-Loc: Improving indoor localization with social sensing," in *SenSys'13*, 2013.
[17] J. Xiong and K. Jamieson, "Arraytrack: A fine-grained indoor location system," in *NSDI'13*, 2013.
[18] J. Krumm and K. Hinckley, "The NearMe wireless proximity server," in *UbiComp'04*, 2004.
[19] S. Kumar and R. M. Hegde, "An efficient compartmental model for real-time node tracking over cognitive wireless sensor networks," *IEEE Transactions on Signal Processing*, vol. 63, no. 7, pp. 1712–1725, 2015.
[20] K. Chintalapudi, A. Padmanabha Iyer, and V. N. Padmanabhan, "Indoor localization without the pain," in *MobiCom'10*, 2010.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2017.2737426, IEEE Transactions on Mobile Computing

[21] D. Turner, S. Savage, and A. C. Snoeren, "On the empirical performance of self-calibrating wifi location systems," in *LCN'11*, 2011.

[22] Q. Zhang, Z. Zhou, W. Xu, J. Qi, C. Guo, P. Yi, T. Zhu, and S. Xiao, "Fingerprint-free tracking with dynamic enhanced field division," in *INFOCOM'15*, 2015.

[23] M. Youssef and A. Agrawala, "The Horus WLAN location determination system," in *MobiSys'05*, 2005.

[24] A. Haeberlen, E. Flannery, A. M. Ladd, A. Rudys, D. S. Wallach, and L. E. Kavraki, "Practical robust localization over large-scale 802.11 wireless networks," in *MobiCom'04*, 2004.

[25] J. Koo and H. Cha, "Autonomous construction of a WiFi access point map using multidimensional scaling," in *Pervasive'11*, 2011.

[26] S. He, T. Hu, and S.-H. G. Chan, "Contour-based trilateration for indoor fingerprinting localization," in *SenSys '15*, 2015.

[27] R. Liu, C. Yuen, J. Zhao, J. Guo, R. Mo, V. N. Pamadi, and X. Liu, "Selective ap-sequence based indoor localization without site survey," in *IEEE 83rd Vehicular Technology Conference (VTC Spring)*, 2016.

[28] R. Elbakly and M. Youssef, "A robust zero-calibration rf-based localization system for realistic environments," in *SECON '16*, 2016.

[29] Z. Zhong and T. He, "Achieving range-free localization beyond connectivity," in *SenSys'09*, 2009.

[30] B. K. Kiran Yedavalli, "Sequence-based localization in wireless sensor network," *IEEE Transactions on Mobile Computing*, vol. 7, no. 1, pp. 81 – 94, 2008.

[31] P. Bahl and V. Padmanabhan, "Radar: an in-building RF-based user location and tracking system," in *INFOCOM'00*, 2000.

[32] "Cisco wireless mesh access points, design and deployment guide," http://www.cisco.com/en/US/products/ps8368.

[33] G. Lui, T. Gallagher, B. Li, A. Dempster, and C. Rizos, "Differences in RSSI readings made by different Wi-Fi chipsets: A limitation of WLAN localization," in *ICL-GNSS'11*, 2011.

[34] "Procurve networking by hp planning a wireless network," http://www.hp.com/rnd/pdfs/802.11technicalbrief.pdf.

[35] Z. Zhang, X. Zhou, W. Zhang, Y. Zhang, G. Wang, B. Y. Zhao, and H. Zheng, "I am the antenna: Accurate outdoor AP location using smartphones," in *MobiCom'11*, 2011.

[36] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*. MIT Press, 2001.

[37] L. W. Barclay, *Propagation of Radiowaves*. The Institution of Engineering and Technology, 2002.

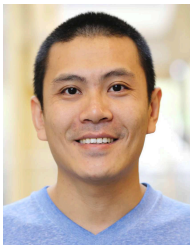
[38] M. de Berg, O. Cheong, M. van Kreveld, and M. Overmars, *Computational Geometry, Algorithms and Applications*. Springer, 2008.

[39] J. R. A., *Advanced Euclidean Geometry*. Dover, 2007.

[40] S. Guo, Z. Zhong, and T. He, "FIND: Faulty node detection for wireless sensor networks," in *SenSys'09*, 2009.

[41] G. Shen, Z. Chen, P. Zhang, T. Moscibroda, and Y. Zhang, "Walkie-Markie: Indoor pathway mapping made easy," in *NSDI'13*, 2013.

Junghyun Jun is research engineer for Trimble New Zealand. Previously, he was an assistant professor at Indian Institute of Technology, Ropar. Prior to this he served as a Postdoc Research Associate at Singapore University of Technology and Design. He received PhD from University of Cincinnati in Computer Science and Engineering, 2011. He received his MS in Electrical Engineering-System from USC Viterbi School of Engineering, 2005. He is a recipient of the best paper/demo awards from WCNC'13 and IPSN'12 (runner-up).



Liang He is an Assistant Professor at University of Colorado Denver, CO. He worked as a research fellow at The University of Michigan at Ann Arbor during 2015-2017, as a research scientist at Singapore University of Technology & Design during 2012–2014, and as a research assistant at University of Victoria during 2009–2011. His research interests include CPSes, cognitive battery management, and networking. He has been a recipient of the best paper/poster awards of MobiSys'17, QShine'14, WCSP'11, and GLOBECOM'11, and a best paper candidate of GLOBECOM'14.

Yu Gu is a lead engineer for the Visa data platform. Previously he was a research scientist for IBM Watson Health/IBM Research Austin from 2014 to 2017, and an assistant professor at Singapore University of Technology and Design from 2010 to 2014. He received his PhD from the University of Minnesota, Twin Cities in 2010. He is author and co-author of over 100 peer-reviewed papers, with the best paper/demo awards from BuildSys'14 (runner-up), GlobeCom'14, QShine'14, WCNC'13, IPSN12 (runner-up), MSN'11 and SenSys'10 (runner-up).

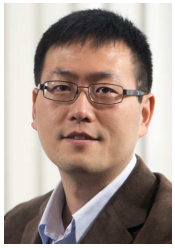
Wenchao Jiang is currently a Ph.D. candidate University of Minnesota, Twin Cities. Prior to this, he was a research assistant at Singapore University of Technology and Design (SUTD). He received Bachelor of Science (2011) and Master of Science (M.Sc.), Computer Science (2014) at Shanghai Jiao Tong University, China.

Gaurav Kushwaha joined the B.Tech program at Department of Computer Science and Engineering, IIT Ropar in 2012. His research interests fall mainly in the field of wireless sensor networks, algorithm design, machine learning and its applications

Vipin A joined the B.Tech program at Department of Computer Science and Engineering, IIT Ropar in 2012. His research interests lies in the field of mobile computing, computer networks, software testing and debugging.

Long Cheng received the Ph.D. degree from Beijing University of Posts and Telecommunications, China in 2012. He is currently a postdoctoral researcher at the State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, China. His research interests include network security and forensics, wireless sensor networks, cyber-physical systems, mobile and pervasive computing. He is a member of ACM and IEEE.

Cong Liu received the Ph.D. degree in computer science from the University of North Carolina at Chapel Hill, USA. He is currently an Assistant Professor with the Department of Computer Science, University of Texas at Dallas, USA. His current research interests include real-time and embedded systems, cyber-physical systems, and mobile and cloud computing. He was a recipient of the Best Student Paper Award from IEEE RTSS'09, and IEEE RTCSA'11.



Ting Zhu is now an assistant professor in the Department of Computer Science and Electrical Engineering at the University of Maryland, Baltimore County, USA. He was an assistant professor at the State University of New York, Binghamton during 2011-2014. He received his Ph.D. from the Department of Computer Science and Engineering at the University of Minnesota in 2010.