Compressive Sensing Based Positioning Using RSS of WLAN Access Points

Chen Feng[1,2], Wain Sy Anthea Au[1], Shahrokh Valaee[1], and Zhenhui Tan[2]

[1] Department of Electrical and Computer Engineering, University of Toronto

[2] State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University

Email: {chenfeng, anthea, valaee}@comm.utoronto.ca, zhhtan@center.njtu.edu.cn

Abstract— **The sparse nature of location finding problem makes the theory of compressive sensing desirable for indoor positioning in Wireless Local Area Networks (WLANs). In this paper, we address the received signal strength (RSS)-based localization problem in WLANs using the theory of compressive sensing (CS), which offers accurate recovery of sparse signals from a small number of measurements by solving an ℓ_1-minimization problem. A pre-processing procedure of orthogonalization is used to induce incoherence needed in the CS theory. In order to mitigate the effects of RSS variations due to channel impediments, the proposed positioning system consists of two steps: coarse localization by exploiting affinity propagation, and fine localization by the CS theory. In the fine localization stage, access point selection problem is studied to further increase the accuracy. We implement the positioning system on a WiFi-integrated mobile device (HP iPAQ hx4700 with Windows Mobile 2003 Pocket PC) to evaluate the performance. Experimental results indicate that the proposed system leads to substantial improvements on localization accuracy and complexity over the widely used traditional fingerprinting methods.**

Keywords- Indoor positioning, Compressive sensing, Affinity propagation, WLANs

I. Introduction

Recent technological achievements have made it feasible to deliver indoor Location-Based Services (LBSs) using Wireless Local Area Network (WLAN) infrastructures, for personal and commercial applications, such as indoor positioning, navigation for the disabled, and location-based security [1][2][3]. However, due to the complexity of indoor environment, it is usually difficult to provide a satisfactory level of accuracy in most applications. Thus, one of the key challenges arises: *how to design an accurate indoor positioning system that can be easily deployed on commercially available mobile devices without any hardware modification.*

Received Signal Strength (RSS)-based localization algorithms have been extensively studied as an inexpensive solution for indoor positioning systems in recent

years [4][5]. Compared with other measurement-based algorithms (e.g., time-of-arrival (TOA) or angle-of-arrival (AOA) measurements of ultra-wideband (UWB) signal) [6], RSS can be easily obtained by a WiFi-integrated mobile device, without any additional hardware modification. However, the key challenge for such systems comes from the variations of RSS due to the radio channel impediments such as shadowing, multipath, and the orientation of the wireless device, etc. This increases the difficulty of designing accurate indoor positioning systems.

In general, two approaches are used by the existing WLAN positioning techniques to model the RSS-position dependency for location estimation. Early positioning systems use a prior theoretical or empirical radio propagation model to formulate the RSS-position relationship [7]. However, these models are unreliable due to the dynamic and unpredictable nature of indoor radio propagation [8]. Therefore, another method known as fingerprinting is proposed [9]. The position of a mobile user is estimated by comparing its online RSS readings with offline observations. One simple solution is the k-nearest neighbor algorithm (kNN), which estimates the mobile user's location by computing the centroid of the k closest neighbors that have the smallest Euclidean distance with respect to the online RSS readings [10]. Such system is easy to implement but the estimation is not very accurate. Another solution to the fingerprinting approach is to solve the problem by a statistical method, in which the probability of each potential position is analyzed using the Bayesian theory or kernel functions [5][11]. However, an explicit formulation of RSS distribution is challenging in real environment and thus, these probabilistic based systems often have high computational complexity.

Compressive sensing (CS) provides a novel framework for recovering signals that are sparse or compressible under a certain basis, with far fewer noisy measurements than that needed by the Nyquist sampling theorem [12][13]. The sparse signal can be reconstructed exactly with high probability by solving an ℓ_1-minimization

problem [14]. The sparse nature of location finding in the spatial domain motivates us to exploit the CS theory for indoor positioning system [15][16], which offers exact deterministic recovery under certain conditions [12].

With these new insights in mind, in this paper, we propose a RSS-based localization scheme composed of two phases: an *offline phase*, during which clustering by affinity propagation is performed to address the effects of RSS variations; and an *online phase*, which consists of a coarse localization stage using cluster matching and a fine localization stage using compressive sensing. In the fine localization stage, AP selection schemes are studied to further improve the accuracy of the estimation. We have implemented the proposed positioning system on a Personal Digital Assistant (PDA, HP iPAQ hx4700 with Windows Mobile 2003 pocket PC) to evaluate the localization accuracy. From the implementation, we have shown that the proposed system is able to estimate the location accurately by using only a small number of RSS measurements. In addition, such system does not require to collect a huge number of RSS time samples, and thus it consumes less memory and computation resources.

The remainder of this paper is organized as follows. Section II sets up the problem that will be used through the paper. Section III proposes the overall positioning system, and describes the interactions between the server and the mobile device. Section IV to Section VI describe the individual blocks of the proposed system, including offline phase using affinity propagation, coarse localization using cluster matching, and fine localization using compressive sensing. The performance is evaluated through simulations and implementations in Section VII. Finally, Section VIII concludes the paper.

II. PROBLEM SETUP

We start with a typical WLAN positioning scenario, where a user carries a mobile device equipped with a WLAN adapter, taking RSS measurements from L WLAN *Access Points* (APs) in an indoor environment. It is noticed that the location of these APs is unknown to the positioning system.

Due to the time varying characteristic of radio propagation caused by multipath, shadowing, and interference, a training-based method known as fingerprinting is widely used in positioning systems in WLANs [9]. During an offline phase, time samples of RSS readings are collected at known locations, which are referred to as the *Reference Points* (RPs). The τth time sample of RSS reading recorded for AP i at RP j is denoted as $\{\psi_{i,j}(\tau), \tau = 1, ..., q, q > 1\}$, with q being the total number of time samples collected. The average of these time samples is computed and stored in a database, known as a *radio map* on the server. Such radio map gives a sufficient representation of the spatial

RSS properties in the given environment. The radio map[2] database is represented by $\mathbf{\Psi}$:

$$\mathbf{\Psi} = \begin{pmatrix} \psi_{1,1} & \psi_{1,2} & \cdots & \psi_{1,N} \\ \psi_{2,1} & \psi_{2,2} & \cdots & \psi_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ \psi_{L,1} & \psi_{L,2} & \cdots & \psi_{L,N} \end{pmatrix} \quad (1)$$

where $\psi_{i,j} = \frac{1}{q}\sum_{\tau=1}^{q}\psi_{i,j}(\tau)$ is the average of RSS readings over time domain from AP i at RP j, for $i = 1, 2, ..., L$, and $j = 1, 2, ..., N$. L is the total number of APs and N is the total number of RPs. The columns of $\mathbf{\Psi}$, radio map vectors, can be referred to as $\boldsymbol{\psi}_j = [\psi_{1,j}, \psi_{2,j}, \cdots, \psi_{L,j}]^T, j = 1, 2, ..., N$, which represent the RSS readings at each RP j, where the superscript T denotes transposition. In addition, the server also stores the variance of these time samples, which can be used for selecting proper APs in the fine localization stage (see Section VI-A). For each RP j, $j = 1, 2, ...N$, a variance vector is defined as

$$\mathbf{\Delta}_j = [\Delta_{1,j}, \Delta_{2,j}, \cdots, \Delta_{L,j}]^T \quad (2)$$

where $\Delta_{i,j} = \frac{1}{q-1}\sum_{\tau=1}^{q}(\psi_{i,j}(\tau) - \psi_{i,j})^2$ is the unbiased estimate of the variance of RSS time samples for AP i at RP j.

The radio map is then the table of $(x_j, y_j; \boldsymbol{\psi}_j, \mathbf{\Delta}_j), j = 1, ..., N$, where (x_j, y_j) is the coordinates of the jth RP. If no RSS reading is found for an AP at a RP, the corresponding RSS entity in the radio map is set to a small value (-110dBm in our case) which implies zero power reading.

III. OVERALL INDOOR POSITIONING SYSTEM

Different from traditional fingerprinting approaches, in which a large number of RSS time samples are required to formulate the RSS distribution, the CS-based localization scheme, proposed in this paper, offers accurate indoor location estimation from far fewer number of RSS measurements by solving a linear program, while no RSS distribution is needed. To mitigate the effects of RSS deviations, a coarse localization using clustering by *affinity propagation* [17] is introduced before applying the CS-based localization scheme to weed out outliers, increase the accuracy and reduce the communication costs.

The overview of the proposed indoor positioning system is shown in Fig. 1. It consists of two phases: an offline phase which collects the radio map database and performs clustering using affinity propagation; and an online phase which performs the localization. During the online phase, the system consists of two stages: coarse localization using cluster matching and fine localization using compressive sensing. The individual blocks of the

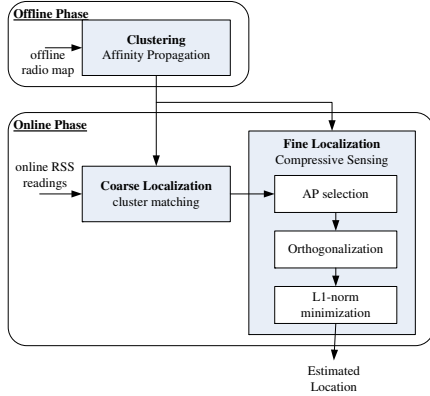system model in Fig. 1 will be explained in details in the following sections.

Fig. 1.   Block diagram of the proposed indoor localization system.

Fig. 2.   Interaction between the server and the mobile device during the offline phase.

When the server for the system is first set up, an offline stage, also referred to as the training period, is carried out to collect the radio map database for the area of interest. The mobile device collects a small number of RSS time samples from APs at specific RPs and then passes the information to the server. The server creates the radio map matrix $\mathbf{\Psi}$ according to (1) and records the coordinates and the variance vectors of the corresponding RPs. Then, affinity propagation is operated on the radio map database through message exchanges between neighboring RPs at the server, until cluster information are generated (see Section IV). Fig. 2 shows the operations for the system during the offline phase.

The actual localization of the mobile device takes place in the online phase. Fig. 3 shows the interaction between the mobile device and the server for location estimation. First, the mobile device collects the online RSS readings. Then, the device communicates with the server to perform the coarse localization by cluster matching as described in Section V. This stage helps to reduce the area of interest into a smaller group of RPs and thus reduce the complexity as well as the computation time for the following localization stage. The fine localization stage is explained in Section VI. It can accurately estimate the device's location by using the result from the previous stage to obtain relevant radio map database from the server for the CS-based position estimation.

IV. AFFINITY PROPAGATION FOR CLUSTERING IN OFFLINE PHASE

Since the indoor propagation channel varies over time, online RSS measurements may deviate from those stored in the radio map database. Fig. 4 gives an example
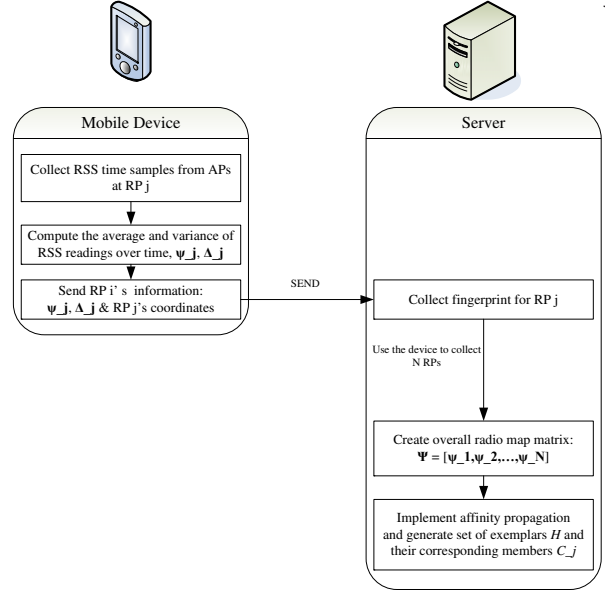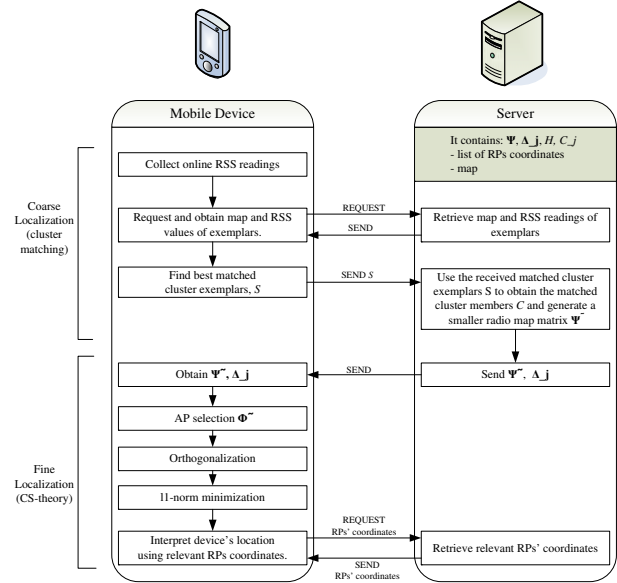

Fig. 3.   Interaction between the server and the mobile device during the online phase.

of the histogram of RSS variation with time at a certain RP. In order to mitigate the effects of such deviations and to remove potential outliers, a coarse localization by exploiting affinity propagation is performed to confine the localization problem into a smaller region, a subset of RPs that have similar RSS readings to the online measurements, where the fine localization can then be applied. Instead of randomly selecting K initial exemplars in the K-means clustering algorithm [18], the affinity
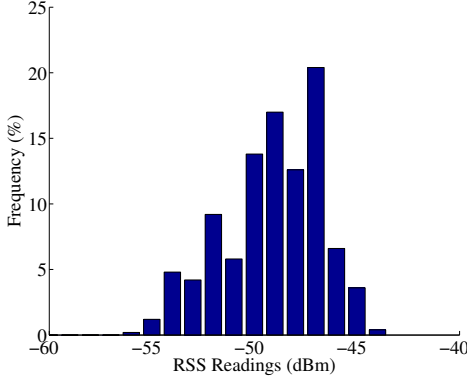
Fig. 4. An example of RSS variation with time at a certain reference point.

propagation algorithm takes a set of real numbers, known as *preference* (p) for each RPs as input, so that RPs with larger values are more likely to be chosen as exemplars. This algorithm can generate better clusters because of its initialization-independent property [17] and thus, it is used in our coarse localization stage.

Affinity propagation takes an input measures of similarity between pairs of RPs. The pairwise similarity $s(i, j)$ indicates how well the RP j is suited to be the exemplar for RP i. Based on the observation from Fig. 4, we can assume that the radio map vector for each RP j is indeed $\boldsymbol{\psi}_j + \boldsymbol{\delta}_j$, where $\boldsymbol{\delta}_j$ is the measurement noise. We also assume that $\boldsymbol{\delta}_j$ is nearly Gaussian distributed, so that the Euclidean distance between two RPs is a good decision rule. Thus, we define the similarity function to be the squared Euclidean distance between the RSS readings of the two RPs, that is

$$s(i, j) = -\|\boldsymbol{\psi}_i - \boldsymbol{\psi}_j\|^2, \forall i, j \in \{1, 2, ..., N\} \quad (3)$$

There are two kinds of real-valued messages transmitted between the RPs for affinity propagation: the *responsibility message* to decide which RPs are exemplars and the *availability message*, to decide to which cluster an RP belongs.

- the *responsibility message* $r(i, j)$, sent from RP i to candidate exemplar RP j, reflects the accumulated evidence for how well-suited RP j is to serve as the exemplar for RP i, taking into account of other potential exemplars j' for RP i, that is

$$r(i, j) = s(i, j) - \max_{j' \neq j} \{a(i, j') + s(i, j')\} \quad (4)$$

where $i \neq j$, and $s(i, j)$ is the similarity between RP i and RP j and $a(i, j)$ is the *availability message* defined later. In our implementation, we define the self-responsibility $r(i, i)$, which is known as *preference* (p) as the median of the input similarities,

resulting in a moderate number of clusters, that is[4]

$$p = \mathrm{median}\{s(i, j), \forall i, j \in \{1, 2, ..., N\}\} \quad (5)$$

- the *availability message* $a(i, j)$, sent from candidate exemplar RP j to RP i, reflects the accumulated evidence for how appropriate it would be for RP i to choose RP j as its exemplar, taking into account the support from other RPs that RP j should be an exemplar:

$$a(i, j) = \min \left\{ 0, r(j, j) + \sum_{i' \neq i, j} \max\{0, r(i', j)\} \right\} \quad (6)$$

The self-availability $a(j, j)$ reflects accumulated evidence that RP j is an exemplar, based on the positive responsibilities sent to it from other RPs:

$$a(j, j) = \sum_{i' \neq j} \max\{0, r(i', j)\} \quad (7)$$

The messages are passed recursively between neighboring RPs and the above updating rules are followed until a good set of exemplars and corresponding clusters emerges. This process is conducted at the server when the fingerprints are collected during the offline phase. Let $\mathcal{H}$ be the set of exemplars and for each RP $j \in \mathcal{H}$, let $\mathcal{C}_j$ denote the set of RPs for which RP j is an exemplar. Using the set of exemplars and their corresponding radio map vector, we will propose a coarse localization procedure to select the best clusters that match the online RSS observation, and then the RPs of these candidate clusters will be used to localize the mobile device during fine localization stage.

V. COARSE LOCALIZATION STAGE BY CLUSTER MATCHING

During the online phase, an online RSS measurement vector, denoted as

$$\boldsymbol{\psi}_r = [\psi_{1, r}, \cdots, \psi_{L, r}] \quad (8)$$

where $\{\psi_{k, r}, k = 1, ..., L\}$ is the average online RSS readings over time from AP k, is collected by the mobile device at unknown location. A coarse localization is then operated on the device by downloading RSS values of the exemplars from the server, $\Psi_{\mathcal{H}} = \{\psi_j | \forall j \in \mathcal{H}\}$ and computing the similarity between its current RSS readings and each exemplar to decide which cluster it belongs to. Instead of selecting one cluster, we keep a few set of best-matched clusters for the following fine localization to avoid the edge problem, which can lead to inaccurate estimation when the location of the mobile device is at the boundaries of clusters. We define $\mathcal{S}$ as the set of best-matched exemplars which have the highest similarities with the online readings. We also define $\mathcal{C}$

as the set of cluster members whose exemplars appear in $\mathcal{S}$,

$$\mathcal{S} = \{j : s(r, j) > \alpha, j \in \mathcal{H}\} \quad (9)$$

$$\mathcal{C} = \bigcup_{j \in \mathcal{S}} \mathcal{C}_j \quad (10)$$

where $s(r, j) = \|\boldsymbol{\psi}_r - \boldsymbol{\psi}_j\|^2$ and α is a pre-defined threshold to obtain a moderate number of clusters in $\mathcal{S}$. In order to limit the number of matched clusters, we set α to be a high fixed percentage of the maximum similarity difference, that is

$$\alpha = 0.95 \max_{l \in \mathcal{H}} \left\{ s(r, l) - \min_{j \in \mathcal{H}} s(r, j) \right\} \quad (11)$$

The mobile device determines the matched exemplars set $\mathcal{S}$, and then requests the server to transmit the part of the database corresponding to $\mathcal{C}$. Then the device downloads from the server the $L \times \tilde{N}$ matrix $\tilde{\boldsymbol{\Psi}}$, with $\tilde{N} = |\mathcal{C}|$, which corresponds to the RPs that appear in $\mathcal{C}$, namely,

$$\tilde{\boldsymbol{\Psi}} = [\boldsymbol{\psi}_j, \forall j \in \mathcal{C}]. \quad (12)$$

The matrix $\tilde{\boldsymbol{\Psi}}$ will be used by the following fine localization stage.

VI. FINE LOCALIZATION STAGE BY COMPRESSIVE SENSING

The localization problem setup in Section II has a sparse nature, as the position of the mobile user is unique in the discrete spatial domain at a certain time. Ideally, assuming the mobile user is located at one of the RPs, the user's location can be formulated as a 1-sparse vector, $\boldsymbol{\theta}$, where $\boldsymbol{\theta}$ is a $\tilde{N} \times 1$ vector with all elements equal to zero except $\theta(n) = 1$, where n is the index of the RP at which the mobile user is located, namely:

$$\boldsymbol{\theta} = [0, ..., 0, 1, 0, ..., 0]^T \quad (13)$$

The online RSS reading measured by the mobile device can then be expressed as:

$$\boldsymbol{y} = \boldsymbol{\Phi}\tilde{\boldsymbol{\Psi}}\boldsymbol{\theta} + \boldsymbol{\varepsilon} \quad (14)$$

where $\tilde{\boldsymbol{\Psi}}$ is defined in (12), and $\boldsymbol{\varepsilon}$ is an unknown measurement noise. The $M \times L$ matrix $\boldsymbol{\Phi}$ is an AP selection operator as defined in the following subsection. Note that the measurement vector $\boldsymbol{y}$ relates to the online RSS reading (8), as

$$\boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{\psi}_r \quad (15)$$

Next we discuss how $\boldsymbol{\Phi}$ can be selected.

A. Access Point Selection

Due to the wide deployment of APs, the total number of detectable APs, L is generally much greater than that required for positioning, which leads to biased estimates and redundant computations [5]. This motivates the use of AP selection techniques to select a subset of available APs used for positioning. According to (1), the set of APs covering the RPs can be denoted as $\mathcal{L}$, with $|\mathcal{L}| = L$. The objective of AP selection is to determine a set $\mathcal{M} \subseteq \mathcal{L}$ such that $|\mathcal{M}| = M \leq L$.

This process is carried out by using the AP selection matrix $\boldsymbol{\Phi}$. Each row of $\boldsymbol{\Phi}$ is a $1 \times L$ vector with all elements equal to zero except $\phi(\ell) = 1$, where ℓ is the index of the AP that is selected for positioning:

$$\boldsymbol{\phi}_m = [0, ..., 0, 1, 0, ..., 0], \forall m = 1, 2, ..., M. \quad (16)$$

Here we investigate two different approaches to select $\boldsymbol{\Phi}$.

- *Strongest APs* (S) [19]
  In this approach, the set of APs with the highest RSS readings is selected, arguing that the strongest APs provide the highest probability of coverage over time. Here, the measurement vector (8) is sorted in the decreasing order of RSS readings, and the access points corresponding to the least indices are used. Since $\boldsymbol{\Phi}$ is created based on the current online measurement vector $\boldsymbol{y}$, this criterion may create different $\boldsymbol{\Phi}$ at different runs.

- *Fish Criterion* (F) [20]
  The Fish criterion is used to quantify the discrimination ability for each AP across RPs. By comparing the matric ξ_i, $i = 1, ..., L$, defined as

$$\xi_i = \frac{\sum_{j \in \mathcal{C}} (\psi_{i,j} - \bar{\psi}_i)^2}{\sum_{j \in \mathcal{C}} (\Delta_{i,j})} \quad (17)$$

  where $\bar{\psi}_i = \frac{1}{\tilde{N}} \sum_{j \in \mathcal{C}} \psi_{i,j}$ and the variance vectors, $\{\boldsymbol{\Delta}_j, \forall j \in \mathcal{C}\}$, are retrieved from the server. The access points with the highest ξ_i are selected for fine localization. The denominator of ξ_i ensures that RSS values do not vary much over time so that the offline and online values are similar; while the numerator represents the discrimination ability of each AP by evaluating the strength of variations of mean RSS across RPs.

B. Orthogonalization and Signal Recovery using ℓ_1-minimization

Since $\boldsymbol{\Phi}$ and $\tilde{\boldsymbol{\Psi}}$ are in general coherent in the spatial domain, which violates the incoherence requirement for the CS theory, an orthogonalization process is applied to induce such property [15] [16]. The required steps to recover the sparse vector $\boldsymbol{\theta}$ is stated below.

The measurement vector $\boldsymbol{y}$ obtained from (15) is pre-processed by an orthogonalization operator, $\mathbf{T}$, such that $\boldsymbol{z} = \mathbf{T}\boldsymbol{y}$. Such operator is defined as

$$\mathbf{T} = \mathbf{Q}\mathbf{R}^{\dagger} \quad (18)$$

where $\mathbf{R} = \mathbf{\Phi}\tilde{\mathbf{\Psi}}$, and $\mathbf{Q} = \mathrm{orth}(\mathbf{R}^T)^T$, where $\mathrm{orth}(\mathbf{R})$ is an orthogonal basis for the range of $\mathbf{R}$, and $\mathbf{R}^\dagger$ is a pseudo-inverse of matrix $\mathbf{R}$. It is proven in our previous work [15] [16] that if M is in the order of $\log(\tilde{N})$, $\boldsymbol{\theta}$ can be fully recovered from $\mathbf{z}$ with very high probability, through the following ℓ_1-minimization program.

$$\hat{\boldsymbol{\theta}} = \arg \min_{\boldsymbol{\theta} \in R^{\tilde{N}}} \| \boldsymbol{\theta} \|_1, \ \ s.t. \ \mathbf{z} = \mathbf{Q}\boldsymbol{\theta} + \boldsymbol{\varepsilon}'. \qquad (19)$$

where $\boldsymbol{\varepsilon}' = \mathbf{T}\boldsymbol{\varepsilon}$, and $\| . \|_1$ is the ℓ_1-norm.

On a special note, the complexity of the ℓ_1-minimization algorithm grows proportional to the dimension of the location vector over potential RPs, $\boldsymbol{\theta}$. Thus, the coarse localization stage, which reduces the location problem into a subset of $\tilde{N}$ RPs over all N RPs ($\tilde{N} < N$), also helps to relive the computational burden for solving the ℓ_1-minimization problem, in addition to removing outliers.

We argue in [15] [16] that if the mobile user is located at one of the RPs, the recovered position is almost exact. However, the mobile user may not be necessary located at an RP. In such cases, the recovered location $\hat{\boldsymbol{\theta}}$ is not an exact 1-sparse vector, but with a few non-zero coefficients [15]. In order to compensate for the error induced by the grid assumption, a post-processing procedure is conducted. We choose the dominant coefficients in $\hat{\boldsymbol{\theta}}$ whose values are above a certain threshold λ, and take the centroid of these RPs as the location indicator. Let $\mathcal{R}$ be the set of all indexes of the elements of $\hat{\boldsymbol{\theta}}$ such that

$$\mathcal{R} = \{n | \hat{\theta}(n) > \lambda\}. \qquad (20)$$

These are potential candidate points for the estimate of the location of the mobile user. The device will then request the server to retrieve the corresponding coordinates, $\{(x_i, y_i) | \forall i \in \mathcal{R}\}$. The location of mobile user can be estimated by finding the centroid of these candidate points, that is

$$(\hat{x}, \hat{y}) = \frac{1}{|\mathcal{R}|} \sum_{i \in \mathcal{R}} (x_i, y_i). \qquad (21)$$

VII. EXPERIMENTS AND RESULTS

This section provides details on the experimental evaluation of the proposed positioning system using both simulations and implementations. Real data were obtained from an office building as well as a large shopping center. Specifically, the experiments were first carried out on a 12m × 36m area of the fourth floor of an eight-story building (Bahen Center at University of Toronto), which is comparable to those reported in [7], [9], and [5]. A total of 17 APs were detected throughout the spatial domain. A PDA (HP iPAQ hx4700 with Windows Mobile 2003 pocket PC) was used to measure WLAN signal strength value, and a software was developed in Visual Studio C# to implement the CS-based localization system on the device. The RSS values were collected on the device by using the open source library OpenNetCF, which provides access to MAC address and RSS values of WLAN APs [21]. The RSS observations from 17 APs were recorded for a period of 20 seconds (one reception per second) over 118 RPs with an average grid spacing of 1 m during the offline phase. Besides the office building, we also tested the system in a public area, the Bayview Village shopping center in Toronto. In our experiments, the orientation of the PDA remained the same during both the offline phase and the online phase.

Both the simulation and experimental implementation on the PDA used the collected radio map database for their localization estimation. The two evaluation methods are different in the way they use the online RSS observations. In *simulations*, the 118 RPs with prior known positions were chosen as the testing points. We then added white Gaussian noise to the observation vector $\mathbf{y}$ to evaluate the system's performance under different signal-to-noise ratio (SNR). In the *implementation* study, the online observations were collected on a different day by the device at unknown locations as the testing points to evaluate the actual performance of the system under time-varying environment.

In the following subsections, the performance of the localization system is evaluated by the localization error, which is measured by averaging the Euclidean distance between the estimated locations of the mobile user and its actual location over the testing points. Section VII-A to Section VII-D provide both simulation results and the corresponding implementation results for the two phase localization scheme.

A. Offline Stage: Clustering by Affinity Propagation

This subsection evaluates the performance of clustering using affinity propagation with respect to the specified parameter (preference, p), by comparing with the K-means clustering algorithm.

In order to mitigate RSS variations and to remove potential outliers for coarse localization, affinity propagation is used on the overall radio map to generate exemplars and their corresponding clusters during the offline phase at the server side. Fig. 5 demonstrates the number of generated clusters with respect to the preference parameter (p) specified for the affinity propagation algorithm. It is shown that a larger p results in a smaller number of clusters. Since all the RPs have the same probability for being an exemplars, we first set the preference p as the median of the input similarities to generate a moderate number of clusters. We will tune the parameter shortly for a better performance.
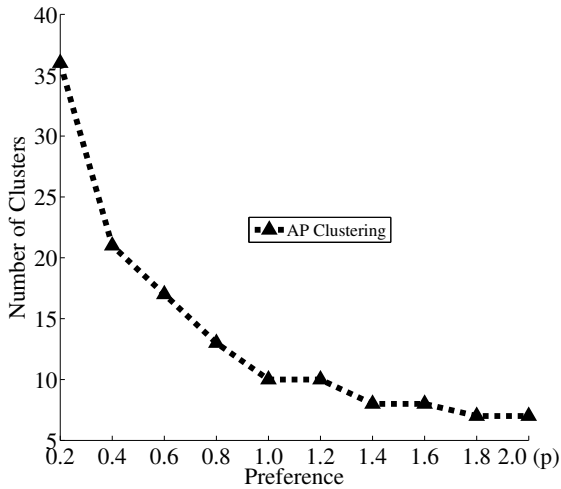
Fig. 5. The number of clusters with respect to parameter p.

Fig. 6 shows the clustering result on the PDA. Each circle represents one RP collected on the map, and each color represents one cluster. It shows that the 118 RPs are divided into 10 clusters, and RPs belonging to the same cluster are geographically close to each other; this allows the clusters to be used for removing outliers in coarse localization.

Fig. 6. The clustering result on the PDA, using affinity propagation in lab environment, (10 clusters are generated).

The performance of clustering is first evaluated by the clustering accuracy, which is defined as the ratio of the number of testing data matched to the correct cluster shown in Fig. 6 to the overall number of testing data. Fig. 7 compares the affinity propagation with the K-means clustering approach in terms of the clustering accuracy, with respect to the number of clusters. Both the simulation results and the implementation results show that affinity propagation outperforms the K-means approach, especially for large database where a large number of clusters is generated.
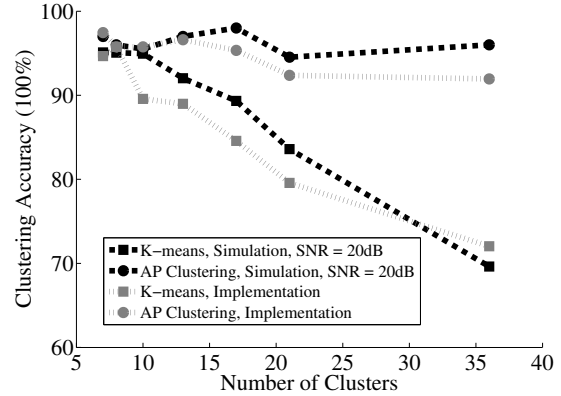

Fig. 7. The clustering accuracy with respect to the number of clusters, using the affinity propagation and the K-means approaches.

B. Online Stage: Coarse Localization

This subsection evaluates the performance of coarse localization using cluster matching, with respect to the number of clusters generated. Other than the clustering accuracy, the performance in terms of the localization error is studied as well. In simulations, we reduce the SNR to 10 dB to evaluate the system.

Fig. 8 illustrates the corresponding results for both simulation and implementation. The coarse localization scheme operating on clusters generated by affinity propagation achieves a better performance than that generated by the K-means clustering approach. The experimental results in Fig. 8 also show that the proposed algorithm outperforms K-means when the number of clusters is more than 15. Thus, we tune the preference parameter p into $0.4p$ to achieve 21 clusters as shown in Fig. 5 for our following experiments.
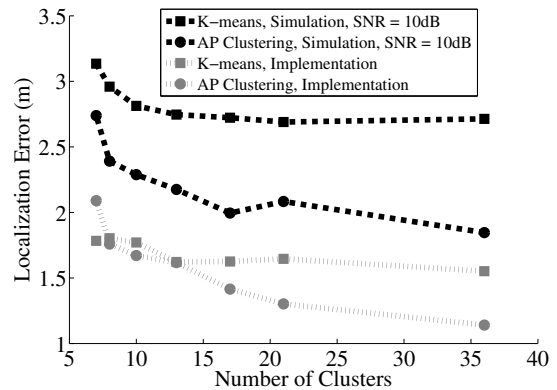

Fig. 8. The experimental result of coarse localization, localization error with respect to the number of clusters generated by affinity propagation, and K-means clustering.

As mentioned in Section V, the similarity function can be used to reduce the number of APs used in the

location estimation. Since clustering reduces the area of interest into a subset $\mathcal{C}$, the dimension of sparse signal in the CS algorithm is also reduced to $\tilde{N}$. This allows the system to reduce the number of APs required for accurate location recovery. Fig. 9 shows the localization error as a function of the number of APs used in the algorithm. Here we have assumed that the strongest APs are used. As illustrated in Fig. 9, only 5 APs are needed to achieve about 1.4 m error when 21 clusters are generated by affinity propagation, and 1.9 m error under 10 clusters. However, 17 APs are needed to achieve the same level of accuracy if no clustering scheme is applied.
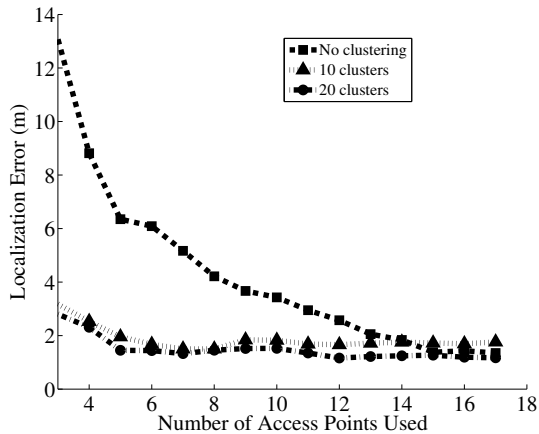

Fig. 9.   The implementation result of reducing the number of APs.

C. Online Stage: Fine Localization

Fig. 10 compares the performance of different schemes of AP selection, in terms of localization error. Implementation result shows that AP selection using the Fish criterion achieves better performance than the strongest APs selection.

Also, we compare the proposed positioning system with the traditional fingerprinting approaches, known as the kNN and the kernel-based methods [5], in terms of the localization accuracy. The proposed positioning system used affinity propagation to generate 21 clusters during the offline phase, and then performed coarse localization by cluster matching, followed by a fine localization stage consisting of AP selection by the Fish criterion and an ℓ_1-minimization algorithm in the online phase. Fig. 11 shows the implementation result on the PDA. As noticed that once the number of measurements exceeds the minimum bound as required by the CS theory, our proposed positioning system achieves the best performance among the three approaches. The proposed system implementation on the PDA leads to the location estimate error improvement of 0.8m (37%) and 1.3m
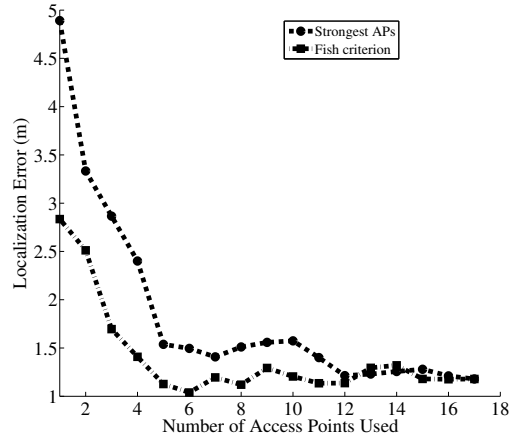

Fig. 10.   The Implementation result of AP selection, using the strongest APs, and the Fish criterion.

(48%) over that of the kNN and the Kernel-based methods, respectively, when 5 APs are used.
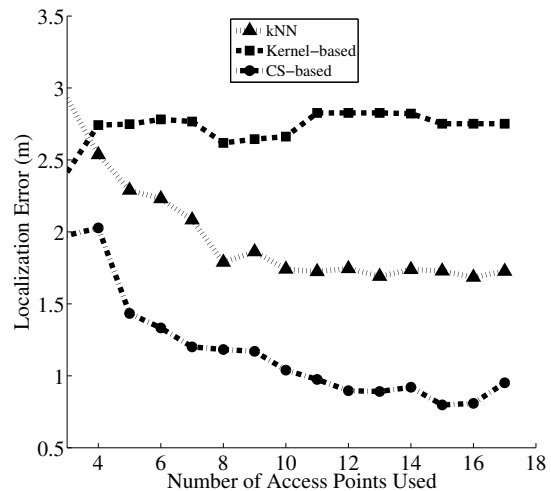

Fig. 11.   Comparison of localization accuracy among different positioning systems in the lab environment.

D. Implementation Result in Public Area

We also tested the system in the Bayview Village shopping center in Toronto. A total of 69 APs were detected throughout a $40,000$ square foot area, and RSS readings at 202 RPs were collected.

Fig. 12 shows the corresponding clustering result on the PDA, where 15 clusters were generated by affinity propagation. Fig. 13 compares the performance of the proposed positioning system with the kNN and the Kernel-based methods in the Bayview Village. It shows

that the proposed system leads to an approximate location estimate error improvement of 0.8m (20%) and 1.1m (26%) over the kNN and the Kernel-based methods, respectively, when 5 APs are used. The redundancy can be addressed by AP selection, in which only a small number of APs is used for location estimation and thus helps to reduce the computation time.
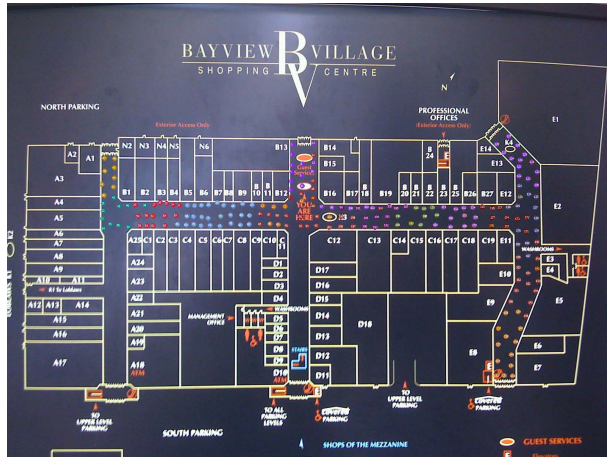

Fig. 12. The clustering result on the PDA using affinity propagation in the Bayview Village, (15 clusters are generated).
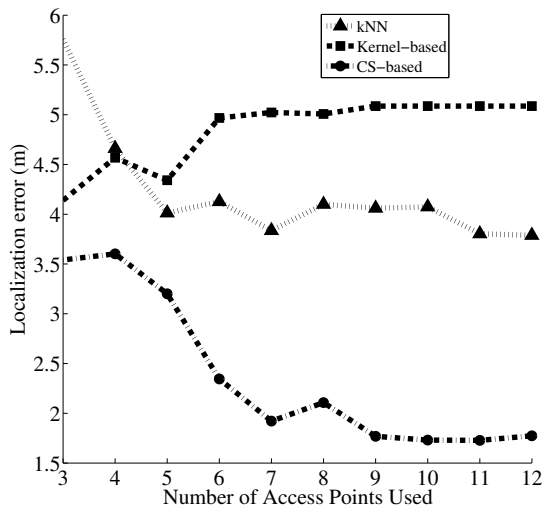

Fig. 13. Comparison of localization accuracy among different positioning systems in the Bvayview Village.

VIII. CONCLUSION

In this paper, we have proposed an indoor positioning system based on compressive sensing in WLANs. The intuition behind this technique is that location estimation is a sparse problem and thus according to the CS theory, the location can be well recovered from

only a small number of noisy measurements through an ℓ_1-minimization program. We have used a coarse localization to compensate for the complex radio channel. Experimental results demonstrate that the proposed two-stage localization method leads to substantial improvements on localization accuracy and reduces complexity over the widely used traditional fingerprinting methods.

REFERENCES

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Communication Magazine*, vol. 40, no. 8, pp. 102–114, August 2002.

[2] N. Patwari, J. N. Ash, and S. Kyperountas, "Locating the nodes: Cooperative localization in wireless sensor networks," *IEEE Signal Processing Magazine*, pp. 54–69, July 2005.

[3] M. Rodriguez, J. Favela, E. Martinez, and M. Munoz, "Location-aware access to hospital information and services," *IEEE Trans. Information Technology in Biomedicine*, vol. 8, no. 4, pp. 448–455, 2004.

[4] G. Sun, W. G. J. Chen, and K. R. Liu, "Signal processing techniques in network-aided positioning: A survey of state-of-the-art positioning designs," *IEEE Signal Processing Magazine*, vol. 22, no. 4, pp. 12–23, 2005.

[5] A. Kushki, N. Plataniotis, and A. N. Venetsanopoulos, "Kernel-based positioning in wireless local area networks," *IEEE Trans. on Mobile Computing*, vol. 6, no. 6, June 2007.

[6] X. Li and K. Pahlavan, "Super-resolution toa estimation with divversity for indoor geolocation," *IEEE Trans. Wireless Comm.*, vol. 3, no. 1, pp. 224–234, 2004.

[7] P. Bahl and V. Padmanabhan, "Radar: An in-building rf-based user location and tracking system," *Proc. INFOCOM*, vol. 2, pp. 775–784, 2002.

[8] A. Goldsmith, *Wireless Communications*, 1st ed. Cambridge University Press, 2005.

[9] K. Kaemarungsi and P. Krishnamurthy, "Modeling of indoor positioning systems based on location fingerprinting," *Proc. INFOCOM*, vol. 2, pp. 1012–1022, 2004.

[10] B. Li, J. Salter, A. G. Dempster, and C. Rizos, "Indoor positioning techniques based on wireless lan," *Auswireless Conference*, 2006.

[11] R. Singh, L. Macchi, C. Regazzonik, and K. Plataniotis, "A statistical modelling based location determination method using fusion in wlan," *Proc. Int'l Workshop Wireless Ad-Hoc Networks*, 2005.

[12] J. C. Emmanuel and B. W. Michael, "An introduction to compressive sampling," *IEEE Signal Processing Magazine*, pp. 21–30, March 2008.

[13] J. Romberg, "Imaging via compressive sampling," *IEEE Signal Processing Magazine*, pp. 14–20, March 2008.

[14] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal on Scientific Computing*, vol. 20, no. 1, pp. 33–61, 1998.

[15] C. Feng, S. Valaee, and Z. H. Tan, "Multiple target localization using compressive sensing," *IEEE Globecom*, 2009.

[16] ——, "Localization of wireless sensors using compressive sensing for manifold learning," *IEEE The 20th Personal, Indoor and Mobile Radio Communications Symposium, PIMRC*, 2009.

[17] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315, no. 1, February 2007.

[18] E. Gokcay and J. Principe, "Information theoretic clustering," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, no. 2, pp. 158–172, February 2002.

[19] M. Youssef, A. Agrawala, and A. U. Shankare, "Wlan location determination via clustering and probability distributions," *Proc, First IEEE Int'l Conf, Pervasive Computing and Comm*, pp. 143–155, 2003.

[20] J. Shawe-Taylor and N. Cristianini, *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004.

[21] "Opennetcf shared source projects," 2009. [Online]. Available: http://www.opennetcf.com/FreeSoftware/tabid/84/Default.aspx