Updating Wireless Signal Map with Bayesian Compressive Sensing

Bo Yang Suining He S.-H. Gary Chan

ABSTRACT

In a wireless system, a signal map shows the signal strength at different locations termed reference points (RPs). As access points (APs) and their transmission power may change over time, keeping an updated signal map is important for applications such as Wi-Fi optimization and indoor localization. Traditionally, the signal map is obtained by a full site survey, which is time-consuming and costly. We address in this paper how to efficiently update a signal map given sparse samples randomly crowdsourced in the space (e.g., by signal monitors, explicit human input, or implicit user participation).

We propose Compressive Signal Reconstruction (CSR), a novel learning system employing Bayesian compressive sensing (BCS) for online signal map update. CSR does not rely on any path loss model or line of sight, and is generic enough to serve as a plug-in of any wireless system. Besides signal map update, CSR also computes the estimation error of signals in terms of confidence interval. CSR models the signal correlation with a kernel function. Using it, CSR constructs a sensing matrix based on the newly sampled signals. The sensing matrix is then used to compute the signal change at all the RPs with any BCS algorithm. We have conducted extensive experiments on CSR in our university campus. Our results show that CSR outperforms other state-of-the-art algorithms by a wide margin (reducing signal error by about 30% and sampling points by 40%).

Categories and Subject Descriptors

C.2.1 [**Network Architecture and Design**]: Wireless Communication

Keywords

Signal map learning; signal change; Bayesian compressive sensing; crowdsourcing; database reconstruction.

1. INTRODUCTION

Received signal strength indicator (RSSI) exhibits spatial variation, which forms the so-called *signal map* (or *signal heat map*). Knowing the signal map in a timely manner is important for many applications. For example, in a Wi-Fi network, the signal of an access point (AP) may change over time due to AP introduction, removal, migration, power adjustment, etc. The system administrator would be interested in the current signal map so as to understand the Wi-Fi coverage, or to adjust/tune Wi-Fi settings. In fingerprint-based indoor localization, keeping its signal database updated would lead to improved localization accuracy.

Signal map is often obtained through site survey, where a professional surveyor walks through the site to measure signal values at many predefined locations termed “reference points” (RPs). As signal may evolve over time, this survey has to be conducted frequently; this is laborious, time-consuming and costly.

We consider in this paper the following problem: given a previously obtained signal map and some sparse signals newly sampled at random points in the site, how can we online update, or *reconstruct*, the signal map (in terms of signal values at RPs)? Addressing this problem can substantially cut the survey cost to keep the signal map up-to-date. As each AP signal may be considered independently, in the following we will focus on a single arbitrary AP in the site.

Signals may be *crowdsourced* in space in several ways. For example, one may install sensors or monitors at different locations which report signals over time. One may also conduct explicit user input, where a dedicated surveyor samples the space at random points. Alternatively, implicit user participation may be used, where naive users may unknowingly report signals measured at their locations (as in some indoor localization system [18, 30]).

Reconstructing signal map given random sparse samples is challenging. Existing learning approaches often assume a certain RSSI signal propagation model (e.g., a log-distance path loss model), line of sight or open space [1, 25], which does not work well in complex indoor environments. Furthermore, as the measured signal is noisy, it is desirable that the reconstructed signal map has confidence interval indicating the likely RSSI range at a point.

We propose Compressive Signal Reconstruction (CSR), a novel online signal learning scheme based on *Bayesian compressive sensing* (BCS). CSR is a generic standalone module which may be integrated with any existing wireless system to keep signal map updated. CSR uses a kernel function to model the signal correlation between any pair of RPs in the signal map (where the correlation value is high if their

ACM ISBN 978-1-4503-2138-9.
DOI: 10.1145/1235

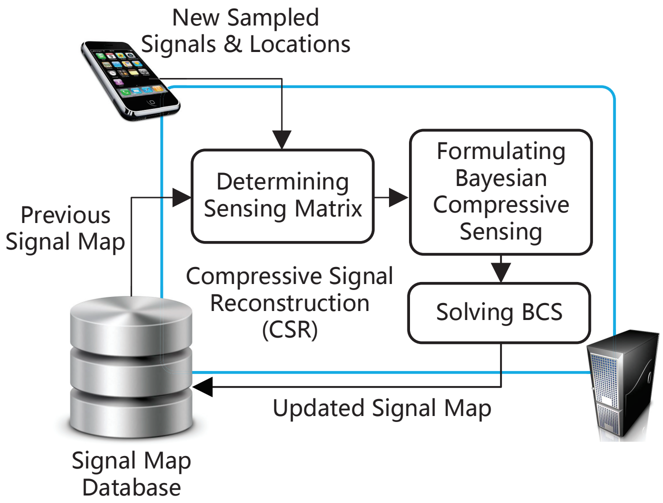
New Sampled Signals & Locations
Previous Signal Map
Determining Sensing Matrix
Formulating Bayesian Compressive Sensing
Compressive Signal Reconstruction (CSR)
Solving BCS
Updated Signal Map
Signal Map Database

Figure 1: System framework of CSR, an online learning module for signal map reconstruction.

locations and RSSIs are similar). A new crowdsourced sample is first mapped to its closest RP location. Such RP is termed the *sample neighbor*. Using the correlations between the sample neighbor and all the other RPs, a *sensing matrix* is then formed for all the samples. Given the sensing matrix, CSR computes the signal change for all the RPs using compressive sensing.

To the best of our knowledge, this is the first work applying BCS to address online signal map learning. Note that CSR is general enough to be applied to any wireless signal, though most of the discussions and experiments in the paper are in the context of Wi-Fi. Unlike other signal reconstruction schemes, the BCS in CSR does not assume any radio propagation model or line-of-sight condition. Furthermore, BCS provides estimation uncertainty, in terms of a confidence bar, for each reconstructed RSSI at an RP.

We show the system framework of CSR in Figure 1. CSR may be executed periodically (say, in every hour) or in fixed batches (after collecting say a certain number of samples). The flow is as follows:

1. *Signal Map Initialization*: The signal map is initialized by an offline site survey. Pairs of <location, RSSI>'s are stored in the database and form the initial signal map which may evolve over time with CSR.

2. *Determining Sensing Matrix for Newly Sampled Signals*: In the online phase, given the location of a newly crowdsourced RSSI, CSR first finds its sample neighbor. Given this RP, CSR computes its correlation with all the other RPs using a kernel function. With multiple samples, a sensing matrix for compressive sensing is then formed based on these correlation values.

3. *Formulating Bayesian Compressive Sensing*: With the sample neighbor, CSR first calculates the RSSI change between the original signal and the newly sampled RSSI. Given all such RSSI changes and the sensing matrix, CSR then formulates the signal map reconstruction into a Bayesian compressive sensing problem.

4. *Solving BCS*: CSR solves the above BCS (via some existing algorithm), and finds all the signal changes at the RPs in the map. The updated <location, RSSI>'s are then returned to the database.

Table 1: Major symbols used in CSR.

Notations	Definitions
M	Number of sample neighbors (the nearest RPs)
N	Number of reference points (RPs) in the site
$(\boldsymbol{y}_t)_{M\times 1}$	$M \times 1$ vector of crowdsourced RSSI samples at time t
$(\boldsymbol{\Delta y}_t)_{M\times 1}$	Vector of crowdsourced signal map change of M RPs at time t
$(\boldsymbol{x}_{t-1})_{N\times 1}$	Vector of RSSIs at N RPs at time $t-1$
$(\boldsymbol{\Delta x}_t)_{N\times 1}$	Vector of full signal map change at N RPs at t
$\boldsymbol{e}_{M\times 1}$	Vector of additive noise in compressive sensing
$\boldsymbol{\Phi}_{M\times N}$	$M \times N$ sensing matrix in compressive sensing
$\boldsymbol{\psi}_m$	Correlation coefficients of the m-th sample neighbor and the N RPs in the signal map
$\boldsymbol{\mu}$	Vector of the mean predicted signal change
$\boldsymbol{\Sigma}$	Variance matrix of predicted signal change
$\boldsymbol{\alpha}$	Vector of reciprocals for variance of predicted $(\boldsymbol{\Delta x}_t)_{N\times 1}$
β	Reciprocal of additive noise e_m variance

We have conducted extensive experiments on CSR in our university campus. Our results validate and confirm that CSR adapts to signal changes for different indoor environments. CSR outperforms other state-of-the-art algorithms such as the Basis Pursuit [11] and the Log-distance Path Loss model [1, 32] with substantially lower RSSI error (by more than 30%) and fewer sampling points (by 40%). By integrating CSR with a Wi-Fi fingerprint-based localization system [3], we also reduce the localization error significantly (by more than 40%).

This paper is organized as follows. We first discuss the background and preliminary of CSR in Section 2. After that, the signal map reconstruction problem and determining sensing matrix are presented in Section 3. Then the algorithm of CSR in solving BCS is discussed in Section 4. We finally illustrate our experimental results in Section 5, followed by conclusion in Section 6.

2. BACKGROUND & PRELIMINARY

We review previous work in this section, first on survey reduction and signal map construction with crowdsourcing (Section 2.1), followed by the algorithms in signal reconstruction (Section 2.2). Finally we review the basic principles of Bayesian compressive sensing (Section 2.3). We show the important symbols used in this paper in Table 1.

2.1 Survey Reduction with Crowdsourcing

In order to save time and labor in the signal map construction, a number of crowdsourcing approaches have been proposed [14, 17]. Recent crowdsourcing studies focus on explicit user participation and implicit update through extra-infrastructure.

Users may explicitly input her/his current locations and RSSIs to update the signal map [7, 9, 19]. In practice, such explicit participation may be inconvenient for users.

To address this, implicit approaches have been studied, which make use of a wide range of extra devices or sensing techniques such as RFID [16], anchor nodes [4, 21], inertial navigation systems [35], ambient sensors [2], and Wi-Fi sniffers [24]. These infrastructures estimate user locations and their RSSIs. Then the systems generates the updated signal map.

Our signal map reconstruction CSR is orthogonal to above works in the following aspects: 1) CSR focuses on reconstructing the outdated signal map regardless of how the signals are crowdsourced. Explicit user input or implicit sensor-assisted crowdsourcing in the above studies can be easily integrated with CSR to achieve survey reduction. For example, crowdsourced <RSSI, location>'s are accumulated through above approaches and then serve as the input for our CSR. 2) CSR does not assume, and hence is not restricted to any sensor measurement model or crowdsourcing user mobility like [9, 10]. Therefore, it is more general and can be applied in more complex indoor environments.

2.2 Signal Map Reconstruction Algorithms

Signal reconstruction has been widely studied in recent years [22]. Gaussian process (GP) [1, 15] has been proposed for signal map construction. Early works like [15] utilize GP to reconstruct signal map. Later works utilize GP to model spatial distribution in the site [1]. In contrast to GP, our work on Bayesian compressive sensing does not assume any propagation model. Therefore, it is more general and can be applied in more complex indoor environments.

More recent works study signal reconstruction using matrix completion [20, 28, 29]. Some have studied using matrix completion to construct missing RSSI values in signal map [20]. The work in [29] also utilizes matrix completion in order to reduce site survey. Given limited site survey where not all AP signals are covered, these works consider the signal map in the site as an incomplete matrix and recover the unknown matrix elements, i.e., the unknown AP signals through the matrix completion. However, the matrix completion constructs the signal map in a deterministic manner. Our BCS considers the uncertainty in signals and is more robust towards the noisy measurements. Furthermore, CSR provides the confidence interval of estimation, which can be used in many probabilistic applications such as the Horus localization system [36]. Note that, in contrast to the work in [10], CSR does not assume any knowledge on environmental factors (building structure change, AP power alteration, etc.) which cause the signal map change; it updates the signal map simply based on the signal difference. Therefore, it is general enough to apply in complex signal environments.

2.3 Compressive Sensing

Compressive Sensing (CS) is a novel signal reconstruction approach which takes in far fewer samples than required in traditional Nyquist paradigm [11]. It has been applied in a wide range of areas including signal processing and image reconstruction.

The objective of CS framework is to find a sparse solution $\boldsymbol{x} \in \mathbb{R}^N$ of linear equation $\boldsymbol{y} = \boldsymbol{\Phi x}$, where the measurement vector $\boldsymbol{y} \in \mathbb{R}^M$ and the $M \times N$ sensing matrix $\boldsymbol{\Phi}$ are known. Note that $M \leq N$, i.e., the linear system is underdetermined. Given the vector $\boldsymbol{x}$, let ℓ_0-norm be the number of nonzero entries of $\boldsymbol{x}$. One simple way to pose a CS framework is to solve the following optimization problem [13],

$$\hat{\boldsymbol{x}} = \underset{\boldsymbol{x} \in \mathbb{R}^n}{\arg\min} \|\boldsymbol{x}\|_0, \quad \text{subject to} \quad \boldsymbol{y} = \boldsymbol{\Phi x}. \tag{1}$$

As above ℓ_0-minimization problem is NP-hard, an alternative solution is to use ℓ_1-norm of $\boldsymbol{x}$ (i.e., $\|\boldsymbol{x}\|_1 = \Sigma_{n=1}^N |\boldsymbol{x}_n|$) under a sufficient constraint Restricted Isometry Property (RIP) [8] based on the convex optimization. Many other optimization-based greedy algorithms [31] are developed to reconstruct sparse signal $\boldsymbol{x}$, such as OMP [34] and CoSaMP [27].

In practice, the CS measurements $\boldsymbol{y}$ are corrupted by environmental noises, denoted as $\boldsymbol{e} \in \mathbb{R}^M$. This noise can be approximated as a zero-mean Gaussian distribution with unknown variance σ^2. Then the CS model can be expressed as $\boldsymbol{y} = \boldsymbol{\Phi x} + \boldsymbol{e}$. We can consider the Gaussian likelihood function of signal $\boldsymbol{y}$ given $\boldsymbol{x}$ and $\boldsymbol{e}$, i.e.,

$$p\left(\boldsymbol{y}|\boldsymbol{x}, \sigma^2\right) = \frac{1}{(2\pi\sigma^2)^{m/2}} \exp\left(-\frac{\|\boldsymbol{y} - \boldsymbol{\Phi x}\|^2}{2\sigma^2}\right). \tag{2}$$

Given $\boldsymbol{\Phi}$ and measurements $\boldsymbol{y}$, and considering prior knowledge about sparse vector $\boldsymbol{x}$ and noise variance σ^2, the above CS reconstruction becomes a Bayesian learning problem, or the so-called *Bayesian compressive sensing* (BCS). Particularly, BCS aims to seek the full posterior values of $\boldsymbol{x}$ and σ^2, given the corresponding prior knowledge and the new measurements.

BCS outperforms the traditional deterministic CS by the following two aspects:

- Traditional CS requires RIP in order to be solvable, which may not be satisfied in many application scenarios [26]. Unlike traditional CS, BCS only relies on the priori statistical sparse properties of signals, which can be easily satisfied for signal map reconstruction.
- Traditional CS outputs only the reconstructed signal map. BCS further provides confidence levels in terms of error bars for reconstructed signals [26], which can be further utilized in other application scenarios such as probabilistic localization algorithms [36].

3. PROBLEM OF SIGNAL MAP LEARNING

We first present the signal map reconstruction problem based on compressive sensing (CS) in Section 3.1. Given this problem and CS formulation, we then present in Section 3.2 how to determine coefficients of the sensing matrix.

3.1 Signal Map Reconstruction Problem

Suppose there are N RPs in the site map for an AP, which are labeled by $n \in \{1, 2, \ldots, N\}$. We show in Figure 2 a site with RPs (locations may not be regular). The signal of the n-th RP at time $t-1$ is given by

$$\boldsymbol{f}_{t-1}^n = ([L_x^n, L_y^n], x_{t-1}^n), \tag{3}$$

where x_{t-1}^n represents RSSI of that RP. The RSSIs of the entire signal map can then be represented as an N-dimensional vector $\boldsymbol{x}_{t-1} \in \mathbb{R}^N$, or $\boldsymbol{x}_{t-1} = [x_{t-1}^1, \ldots, x_{t-1}^N]^T$.

Over a period of time, RSSIs $\boldsymbol{x}_{t-1}$ of the signal map evolves to $\boldsymbol{x}_t$. The N-dimensional vector of *full signal changes* on all N RPs from $t-1$ to t are given by

$$\boldsymbol{\Delta x}_t = \boldsymbol{x}_t - \boldsymbol{x}_{t-1}, \tag{4}$$

where $\boldsymbol{\Delta x}_t = [\Delta x_t^1, \ldots, \Delta x_t^N]^T$. The objective of the signal map reconstruction problem is to estimate $\boldsymbol{\Delta x}_t$.

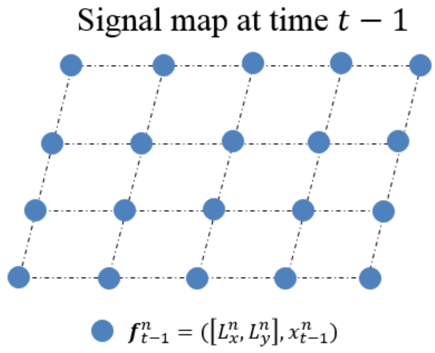

Figure 2: The signal map at time $t-1$. Each RP contains $\boldsymbol{f}_{t-1}^n = ([L_x^n, L_y^n], x_{t-1}^n)$.

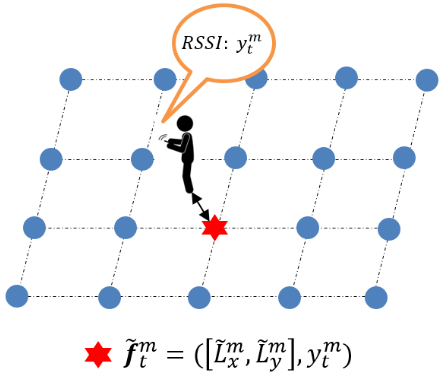

Figure 3: Finding the nearest RP as one crowdsourced sample neighbor.

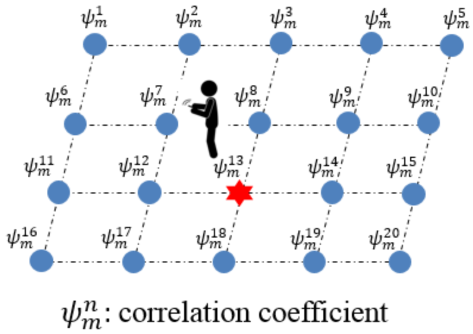

Figure 4: The correlation coefficients between the sample neighbor and all RPs.

In the online reconstruction phase, a batch of new samples are crowdsourced and aggregated from time $t-1$ to t. Each new sample is first mapped to its physically nearest RP (with the smallest Euclidean distance on the 2-D map), which is the red point shown in Figure 3. If there are multiple new crowdsourced samples mapped to the same RP, their RSSIs are averaged and stored for that RP. After that, the batch of new samples are mapped to M different RPs, forming the M crowdsourced sample neighbors (M RPs which are nearest to the crowdsourced samples). Each sample neighbor contains the new signal sample y_t^m. These are the sparse sampling points on the whole signal map.

Specifically, similar to $\boldsymbol{f}_{t-1}^n$, we define the m-th crowdsourced sample neighbor at time t as

$$\tilde{\boldsymbol{f}}_t^m = ([\tilde{L}_x^m, \tilde{L}_y^m], y_t^m), \tag{5}$$

where its coordinates $[\tilde{L}_x^m, \tilde{L}_y^m]$ are the location of the nearest RP in the map. The crowdsourced RSSIs are aligned as $\boldsymbol{y}_t \in \mathbb{R}^M$, or $\boldsymbol{y}_t = [y_t^1, \ldots, y_t^M]^T$. The RSSI change on the m-th sample neighbor can be calculated as the RSSI difference between new sample y_t^m crowdsourced at time t, and the previous RSSI $\tilde{x}_{t-1} \in \boldsymbol{x}_{t-1}$ of the nearest RP on the previous signal map, i.e.,

$$\Delta y_t^m = y_t^m - \tilde{x}_{t-1}. \tag{6}$$

Then the vector $\boldsymbol{\Delta y}_t \in \mathbb{R}^M$ represents the total *crowdsourced signal changes*, i.e., $\boldsymbol{\Delta y}_t = [\Delta y_t^1, \ldots, \Delta y_t^M]^T$. We consider that each RSSI in $\boldsymbol{y}_t$ has measurement noise $e_m \sim \mathcal{N}(0, \sigma^2)$.

Given a previous signal map $\boldsymbol{x}_{t-1}$ and sparse crowdsourced samples $\boldsymbol{y}_t$, we first determine the crowdsourced signal changes $\boldsymbol{\Delta y}_t$ at the sample neighbors (the nearest RPs). Then the signal map reconstruction learns the full signal changes $\boldsymbol{\Delta x}_t$ based on these sparse $\boldsymbol{\Delta y}_t$. The key problem of signal map reconstruction is how to determine $\boldsymbol{\Delta x}_t$ given only $\boldsymbol{\Delta y}_t$. Similar to compressive sensing, we consider that an individual crowdsourced signal change Δy_t^m can be formulated as a subsample from the full signal changes $\boldsymbol{\Delta x}_t$. Then we can leverage the compressive sensing to recover the entire signal map from these sparse samples.

Specifically, for each sample neighbor $\tilde{f}_t^m$, we find its correlation with all other RPs in the site, as illustrated in Figure 4. Let ψ_m^n be the sampling or sensing coefficient between crowdsourced signal change Δy_t^m and the n-th full signal change Δx_t^n. For each individual crowdsourced signal change Δy_t^m, we consider the linear sampling relationship between it and the full signal changes $\boldsymbol{\Delta x}_t$ in the entire signal map as

$$\begin{aligned} \Delta y_t^m &= \sum_{n=1}^{N} \psi_m^n \Delta x_t^n + e_m \\ &= \boldsymbol{\psi}_m \boldsymbol{\Delta x}_t + e_m, \end{aligned} \tag{7}$$

where

$$\boldsymbol{\psi}_m = \left[\psi_m^1, \quad \psi_m^2, \quad \ldots, \quad \psi_m^n, \quad \psi_m^N\right], \tag{8}$$

and

$$\sum_{n=1}^{N} \psi_m^n = 1, \quad 0 \leq \psi_m^n \leq 1. \tag{9}$$

In other word, the crowdsourced signal change is considered as the weighted sum of full signal changes at all RPs. We consider that the higher ψ_m^n is, the more likely that the crowdsourced signal change is near the n-th RP and their signals are correlated. Such a formulation is valid as significant signal map change happens in a close region (due to, for example, crowds of people and new wall partition).

To summarize, for M crowdsourced signal changes $\boldsymbol{y}_t$, the whole subset sampling relationship with full signal changes $\boldsymbol{\Delta x}_t$ is given by

$$(\boldsymbol{\Delta y}_t)_{M\times 1} = \boldsymbol{\Phi}_{M\times N} \left(\boldsymbol{\Delta x}_t\right)_{N\times 1} + \boldsymbol{e}_{M\times 1}, \tag{10}$$

where the sensing matrix Φ is

$$\begin{aligned} \boldsymbol{\Phi} &= \left[\boldsymbol{\psi}_1, \quad \boldsymbol{\psi}_2, \quad \ldots, \quad \boldsymbol{\psi}_m, \quad \boldsymbol{\psi}_M\right]^{\intercal} \\ &= \begin{bmatrix} \psi_1^1 & \ldots & \psi_1^N \\ \vdots & \ddots & \vdots \\ \psi_M^1 & \ldots & \psi_M^N \end{bmatrix}_{M\times N} . \end{aligned} \tag{11}$$

Our signal map reconstruction problem finally becomes *how to find* $\boldsymbol{\Delta x}_t$ *in Equation (10) given* $\boldsymbol{\Delta y}_t$ *and* $\boldsymbol{\Phi}$. Then given the full signal changes, we construct the full signal map $\boldsymbol{x}_t$ as

$$\boldsymbol{x}_t = \boldsymbol{x}_{t-1} + \boldsymbol{\Delta x}_t. \tag{12}$$

3.2 Forming the Sensing Matrix

In this section, we discuss how to determine sensing matrix $\boldsymbol{\Phi}$ in above BCS formulation. Recall that for each new sample, we find in the signal map the nearest RP to form the sample neighbor $\widetilde{\boldsymbol{f}}_t^m$. Then we need to find other RPs in the signal map whose signals are correlated with this crowdsourced sample neighbor (the nearest RP). We consider that two RPs are highly correlated based on the following two criteria:

- *Criterion $\mathcal{I}$. Small physical distance from the sample neighbor $\widetilde{\boldsymbol{f}}_t^m$*: The physical distance $\mathcal{D}_m^n$ between $\widetilde{\boldsymbol{f}}_{t-1}^m$ and $\boldsymbol{f}_{t-1}^n$ is calculated by
$$\mathcal{D}_m^n = \sqrt{(\widetilde{L}_x^m - L_x^n)^2 + (\widetilde{L}_y^m - L_y^n)^2}. \tag{13}$$
The smaller distance $\mathcal{D}_m^n$ is, the signals at these two locations may be more correlated as they have similar distances from APs and their surrounding environments are similar [1]. After calculating the distance for each RP, we normalize each $\mathcal{D}_m^n$ by $\mathcal{D}_m^n = \mathcal{D}_m^n / \left(\sum_{n=1}^N \mathcal{D}_m^n\right)$.

- *Criterion $\mathcal{II}$. Small difference between RSSIs at two reference points (RPs)*: Absolute RSSI difference $\mathcal{R}_m^n$ between $\widetilde{\boldsymbol{f}}_{t-1}^m$ and $\boldsymbol{f}_{t-1}^n$ is calculated by
$$\mathcal{R}_m^n = |\widetilde{x}_{t-1}^m - x_{t-1}^n|. \tag{14}$$
The smaller RSSI difference $\mathcal{R}_m^n$, the more likely that these two locations share similar signal change in the signal space of the new signal map [29]. Similar to $\mathcal{D}_m^n$, we normalize each $\mathcal{R}_m^n$ by $\mathcal{R}_m^n = \mathcal{R}_m^n / \left(\sum_{n=1}^N \mathcal{R}_m^n\right)$.

Combining above criteria, we implement RBF kernel function [12] to evaluate correlation between the sample neighbor and other RPs. Formally, the correlation between $\widetilde{\boldsymbol{f}}_{t-1}^m$ and $\boldsymbol{f}_{t-1}^n$ is given by

$$s_m^n \triangleq \exp\left\{ -\eta\Big[\big(\theta\mathcal{D}_m^n\big)^2 + \big((1-\theta)\mathcal{R}_m^n\big)^2\Big]\right\}, \tag{15}$$

where parameter η represents the sensitivity and θ ($0 \leq \theta \leq 1$) represents the weight between physical distance $\mathcal{D}_m^n$ and RSSI difference $\mathcal{R}_m^n$ (both η and θ are determined empirically and will be described in our experiment). For an individual crowdsourced signal change Δy_t^m, each of the correlation coefficients $\boldsymbol{s}_m^n$ is normalized as

$$\psi_m^n = \frac{s_m^n}{\sum_{n=1}^N s_m^n}. \tag{16}$$

To summarize, given a signal map $\boldsymbol{x}_{t-1}$ and RP locations, we map the new samples $\boldsymbol{y}_t$ to their corresponding nearest RPs. Then we find the crowdsourced signal changes $\boldsymbol{\Delta y}_t$ at those sample neighbors. Then based on $\boldsymbol{\Delta y}_t$ and sensing matrix $\boldsymbol{\Phi}$, through some compressive sensing algorithm we find the full signal changes $\boldsymbol{\Delta x}_t$.

4. BCS FOR SIGNAL RECONSTRUCTION

Based on the problem in Equation (10), in this section we present how to formulate BCS for signal reconstruction. In Section 4.1, we present how to formulate BCS to learn signal changes. Then in Section 4.2, we discuss how to solve the compressive sensing, followed by complexity analysis in Section 4.3.

4.1 Formulating BCS

In the following, we first discuss the probabilistic preliminaries in our BCS formulation. Firstly, RSSI value of each RP at time $t-1$ is modeled as a Gaussian distribution, i.e.,

$$x_{t-1}^n \sim \mathcal{N}\left(\bar{x}_{t-1}^n, (\sigma_{t-1}^n)^2\right), \tag{17}$$

where $\bar{x}_{t-1}^n$ represents mean value and $(\sigma_{t-1}^n)^2$ is RSSI variance. Then RSSI change $\Delta x_t^n = x_t^n - x_{t-1}^n$ from time $t-1$ to t is also a Gaussian distribution, i.e.,

$$\Delta x_t^n \sim \mathcal{N}\left(\Delta\bar{x}_t^n, (\sigma_{t-1}^n)^2 + (\sigma_t^n)^2\right), \tag{18}$$

where $\Delta\bar{x}_t^n$ represents mean value of Δx_t^n. $\boldsymbol{\Delta x}_{t-1}$ can be considered as prior distribution of $\boldsymbol{\Delta x}_t$. Based on Equation (2), we consider

$$p\left(\boldsymbol{\Delta y}_t | \boldsymbol{\Delta x}_t, \sigma^2\right) = \frac{1}{(2\pi\sigma^2)^{\frac{M}{2}}} \exp\left(-\frac{\|\boldsymbol{\Delta y}_t - \boldsymbol{\Phi}\boldsymbol{\Delta x}_t\|^2}{2\sigma^2}\right). \tag{19}$$

From Bayesian point of view, the objective is to find full posterior density function of $\boldsymbol{\Delta x}_t$, given prior distribution of full signal changes $p(\boldsymbol{\Delta x}_t)$ and the crowdsourced signal changes $\boldsymbol{\Delta y}_t$. The signal map reconstruction in CSR becomes a problem of finding $\boldsymbol{\Delta x}_t$ in order to

$$\begin{aligned} \underset{\boldsymbol{\Delta x}_t}{\arg\max} \quad & p\left(\boldsymbol{\Delta x}_t | \boldsymbol{\Delta y}_t, \sigma^2\right), \\ \text{subject to} \quad & \boldsymbol{\Delta x}_t \sim p(\boldsymbol{\Delta x}_t). \end{aligned} \tag{20}$$

Based on Equations (17) and (19), posterior distribution of $\boldsymbol{\Delta x}_t$ is also Gaussian distribution, i.e.,

$$p(\boldsymbol{\Delta x}_t | \boldsymbol{\Delta y}_t, \sigma^2) \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}), \tag{21}$$

where $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ are the mean of posterior full signal changes $\boldsymbol{\Delta x}_t$ and corresponding covariance, respectively. Once $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ are determined, full signal values are updated by Equation (12). The variance $\boldsymbol{\Sigma}$, representing the confidence level, is also stored for later use.

4.2 Solving BCS

In this section, we present how to learn posterior $p(\boldsymbol{\Delta x}_t)$ based on sparse crowdsourced signal changes $\boldsymbol{\Delta y}_t$ and prior distribution of $\boldsymbol{\Delta x}_t$ and σ^2. Let α_n be the reciprocal of $(\sigma_{t-1}^n)^2 + (\sigma_t^n)^2$, and $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_n]$. Given $\Delta x_t^n \sim \mathcal{N}(\Delta\bar{x}_t^n, \alpha_n^{-1})$, we have

$$p\left(\boldsymbol{\Delta x}_t | \boldsymbol{\alpha}\right) = \prod_{n=1}^N \mathcal{N}\left(\Delta\bar{x}_t, \alpha_n^{-1}\right). \tag{22}$$

Let β be the precision of noise $\boldsymbol{e}$ (i.e., $\beta = \sigma^{-2}$). Initial $\boldsymbol{\Delta x}_0$ and β can be either extracted from the initial version of signal map at time 0, or be simply defined as general prior probability distribution such as hierarchical prior [6].

Posterior distribution of full signal changes $\boldsymbol{\Delta x}_t$ and parameters $\boldsymbol{\alpha}$, and β is denoted as $p(\boldsymbol{\Delta x}_t, \boldsymbol{\alpha}, \beta | \boldsymbol{\Delta y}_t)$, given crowdsourced signal changes $\boldsymbol{\Delta y}_t$. Then

$$p(\boldsymbol{\Delta x}_t, \boldsymbol{\alpha}, \beta | \boldsymbol{\Delta y}_t) = p(\boldsymbol{\Delta x}_t | \boldsymbol{\alpha}, \beta, \boldsymbol{\Delta y}_t) p(\boldsymbol{\alpha}, \beta | \boldsymbol{\Delta y}_t). \tag{23}$$

In above Equation (23), the posterior distribution $p(\boldsymbol{\Delta x}_t | \boldsymbol{\alpha}, \beta, \boldsymbol{\Delta y}_t)$

can be computed analytically [33] as

$$p(\boldsymbol{\Delta x}_t|\boldsymbol{\alpha},\beta,\boldsymbol{\Delta y}_t)=\frac{p(\boldsymbol{\Delta y}_t|\boldsymbol{\Delta x}_t,\beta)p(\boldsymbol{\Delta x}_t|\boldsymbol{\alpha})}{p(\boldsymbol{\Delta y}_t|\boldsymbol{\alpha},\beta)}$$
$$=(2\pi)^{-\frac{N}{2}}|\boldsymbol{\Sigma}|^{-\frac{1}{2}}\exp\left(-\frac{1}{2}(\boldsymbol{\Delta x}_t-\boldsymbol{\mu})^{\intercal}\boldsymbol{\Sigma}^{-1}(\boldsymbol{\Delta x}_t-\boldsymbol{\mu})\right),$$

where posterior mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$ are

$$\begin{aligned}\boldsymbol{\mu}&=\beta\boldsymbol{\Sigma}\boldsymbol{\Phi}^{\intercal}\boldsymbol{\Delta y}_t,\\ \boldsymbol{\Sigma}&=(\beta\boldsymbol{\Phi}^{\intercal}\boldsymbol{\Phi}+\boldsymbol{A})^{-1},\end{aligned}\tag{24}$$

and $\boldsymbol{A}=diag(\alpha_1,\alpha_2\ldots\alpha_n\ldots,\alpha_N)$.

As $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ rely on parameters $\boldsymbol{\alpha}$ and β, we now need to find $\boldsymbol{\alpha}$ and β to maximize posterior distribution $p(\boldsymbol{\alpha},\beta|\boldsymbol{\Delta y}_t)$, i.e., the second term on right side of Equation (23).

We apply the type II maximum likelihood [33] to estimate the above parameters. Let Σ_{nn} be the n-th diagonal value of $\boldsymbol{\Sigma}$, i.e., the n-th signal RSSI change variance as shown in Equation (24). We can obtain the updated α_n^{new} as

$$\alpha_n^{new}=\frac{1-\alpha_n\Sigma_{nn}}{\mu_n^2},\quad n\in\{1,2\ldots N\},\tag{25}$$

where μ_n is the n-th entry of vector $\boldsymbol{\mu}$. Similar to α_n^{new}, we have

$$\beta^{new}=\frac{M-Tr(\boldsymbol{I}-\boldsymbol{A\Sigma})}{\|\boldsymbol{\Delta y_t}-\boldsymbol{\Phi\mu}\|_2^2},\tag{26}$$

where $Tr(\boldsymbol{A})$ is defined as the trace (the sum of diagonal elements) of matrix $\boldsymbol{A}$.

Given the initial values of $\boldsymbol{\alpha}$ and β, $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ can be calculated through Equation (24), and then $\boldsymbol{\alpha}$ and β can be further iteratively updated by Equation (25) and (26) until convergence.

To summarize, to reconstruct the signal map $\boldsymbol{x}_t$ at time t, BCS learns the full signal changes $\boldsymbol{\Delta x}_t$ based on the sample neighbors $\boldsymbol{y}_t$ and the previous signal map $\boldsymbol{x}_{t-1}$ at time $t-1$. After finding $\boldsymbol{\Delta x}_t$, CSR returns the updated $\boldsymbol{x}_t$ to signal map database for other applications such as Wi-Fi monitoring or indoor localization.

4.3 Complexity Analysis

We briefly analyze the reconstruction complexity as follows. At time $t-1$, we have a signal map for each AP and we consider $\mathcal{O}(N)$ RPs to be reconstructed at time t. Suppose there are M' pairs of <RSSI, location>'s fed to CSR and they are mapped to M crowdsourced sample neighbors ($M'\geq M$) in $\mathcal{O}(M'N)$ time. During each reconstruction (i.e., at time t), M crowdsourced RSSI samples are accumulated. Determining sensing matrix $\boldsymbol{\Phi}$ for M crowdsourced samples takes $\mathcal{O}(MN)$ (Section 3.2). It takes $\mathcal{O}(M)$ to find the crowdsourced signal changes $\boldsymbol{\Delta y}_t$ (Section 3.1). In our experiment, when determining $\boldsymbol{\Delta x}_t$ for signal map reconstruction, we leverage variational Bayesian algorithm [5,23], which takes $\mathcal{O}(M^3)$ (Section 4.2).

To summarize, reconstructing the signal map of one AP over N RPs in CSR takes

$$\mathcal{O}(M'N+M^3).\tag{27}$$

5. EXPERIMENTAL EVALUATION

In this section, we present the experimental evaluation over CSR in our HKUST campus. In Section 5.1, we show the experimental settings and performance metrics of CSR. Then in Section 5.2, we evaluate the performance of CSR compared with state-of-the-art algorithms.

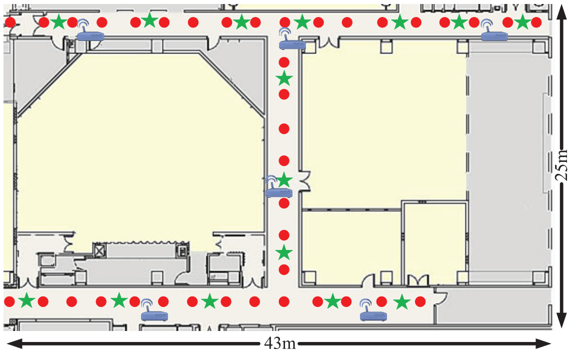

Figure 5: Floor map of the corridor area for experiment (red points are RPs, green stars illustrate crowdsource points). The deployed APs are also marked on the map.

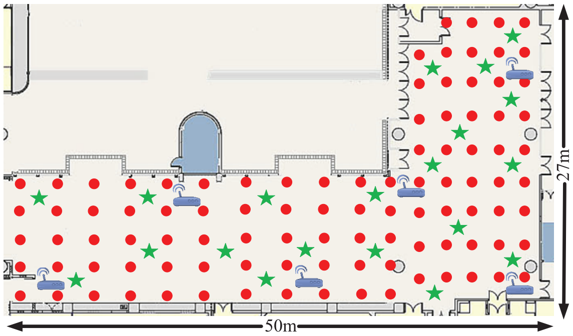

Figure 6: Floor map of the hallway area for experiment (red points are RPs, green stars illustrate crowdsource points). The deployed APs are also marked on the map.

5.1 Settings and Performance Metrics

We conduct all experiments on two indoor scenarios on our university campus. One is a typical corridor environment (43×25 m^2) with many wall partitions, whose floor plan is shown in Figure 5. Another one is a spacious hallway (50×27 m^2), whose floor plan is shown in Figure 6.

Besides environmental disturbance such as metal objects, crowds of people, etc., we also consider the AP power alteration in our experiment to enrich the signal map change studies. In Figures 5 and 6, we show the locations of deployed APs. 6 TP-Link TL-MR3020 APs are installed in each site. We manually adjust the transmission power of 3 APs and move physical locations of other 3 APs over time ($3\sim 5$ meters apart from previous locations). In this way, we simulate the AP transmission power adjustment (due to firmware update or hardware degradation, etc.) and AP physical location changes (due to introduction/removal of wall partition or decoration, etc.). We take into account these possible factors on signal map change, and do not differentiate them in our signal map reconstruction for generality. Meanwhile, the official APs deployed by our university may also experience unknown signal fluctuation or alteration. In our experiment, we evaluate the RSSI reconstruction error with both our own APs and the official ones deployed by the university.

To construct an initial signal map, we conduct RSSI col-

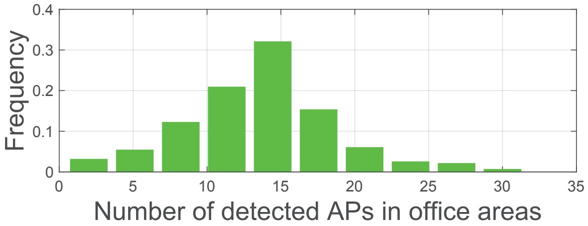

Figure 7: Number of detected APs at each of the RPs in the campus corridor.

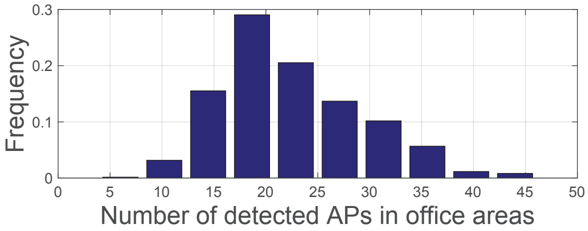

Figure 8: Number of detected APs at each of the RPs in the campus hallway.

lection on 38 RPs in corridor and 86 RPs in hallway. The density of RPs (the distance between two neighboring RPs) is 3 m in both sites. At each RP, at each of the 4 directions (north, south, west and east) we form 1 RSSI vector. Then we have respectively 152 and 344 RSSI vectors in these two sites. We utilize Xiaomi Red Mi 2 for data collection. Red dots in Figure 5 and 6 represent locations of RPs. Figures 7 and 8 show the frequency histogram of detected APs at the RP in the corridor and the hallway environments, respectively.

Besides the initial signal map, we randomly collect RSSI samples every 0.5 h to simulate the crowdsourcing process. In Figures 5 and 6, the ground-truth locations of crowdsourced RSSIs are illustrated as green stars. The RSSI crowdsourcing is conducted in the corridor between April 7th and April 25th, 2016, while in the hallway it is conducted between February 24th and February 29th, 2016. Meanwhile, a manual site survey is conducted every day and the collected fingerprints are used as ground truth to evaluate the reconstruction accuracy.

We present the performance metrics as follows. For each AP, we denote the RSSI at the n-th RP in the previous signal map at time $t-1$ as x_{t-1}^n. At time t, let the ground-truth RSSI at the n-th RP as $\dot{x}_t^n$. Meanwhile, we denote the reconstructed RSSI through a certain scheme as x_t^n at that RP. Then for each RP n, the *reconstruction error* (dB) is defined as the absolute difference between the ground-truth RSSI and the reconstructed one at time t, i.e.,

$$e_{rec}^n = |\dot{x}_t^n - x_t^n|.$$

If e_{rec}^n is close to 0, the reconstructed RSSI closely match the ground-truth one. We also calculate the mean reconstruction error over all RPs covered by the APs. To illustrate the signal change, we also evaluate the RSSI difference between the current and previous signals when no signal map reconstruction is conducted, i.e., $e_{true}^n = |\dot{x}_t^n - x_{t-1}^n|$. We calculate the cumulative density function (CDF) of the reconstruction error e_{rec}^n on all RPs of all APs in each survey site.

We also compare CSR with the following state-of-art reconstruction algorithms:

- *Basis Pursuit (BP) Reconstruction*: BP algorithm is one of deterministic reconstruction methods for compressive sensing framework [11]. Similar to matrix completion method, BP uses convex optimization to predict the signals.
- *Log-distance Path Loss (LDPL) Reconstruction*: In LDPL reconstruction, propagation model is widely used to predict signals [1,32]. In our experiment, we implement least squares to estimate the parameters in the propagation model and then predict RSSI in the site.

An initial signal map is used as the "*Without Reconstruction*" case to illustrate the actual change of signal map RSSIs. To evaluate the performance in dynamic environment, we also vary the number of crowdsourced samples and location input error among crowdsourced signals.

To evaluate the benefits over WLAN application, we incorporate CSR into a Wi-Fi fingerprint-based indoor localization system [3]. Experiments are also conducted at the same corridor shown in Figure 5. The initial fingerprint database or the outdated signal map was constructed in September 2015. To update fingerprints, in the first week of April 2016 we randomly collect new samples at sparse locations covering 25% of the site. We append RSSI readings of newly detected APs and remove those APs which are no longer detected. Given the fingerprints updated by CSR, a weighted k-nearest neighbors algorithm (WKNN) is applied to estimate the user locations. The *localization error* is calculated by the Euclidean distance between the ground-truth location and the estimated one.

5.2 Illustrative Experimental Results

Figure 9 shows the mean reconstruction error versus the parameter η, given a fixed θ (say, 0.6 in our case) in corridor area. η adjusts the correlation sensitivity between two RPs. When η is small (say, 1×10^6), correlation between other RPs and the nearest RP of crowdsourced signals tends to be stronger. In this case, the signals at different RPs are too sensitive to the change of the nearest sample neighbors and therefore the reconstruction error is high. When η further increases, the correlation tends to be weaker and the signal map cannot be fully reconstructed. In our experiment, we set $\eta = 2.5 \times 10^6$ as the optimal value.

Figure 10 shows the mean reconstruction error versus the parameter θ, given a fixed η (say, 2.5×10^6 in our case). θ represents the weight assigned to the relative physical distance between two RPs. When θ is small (say, 0.4), correlation between RPs with similar RSSI values is strong, while correlation between RPs with similar physical locations is strong given large θ (say, 0.8). We observe that the reconstruction accuracy benefits from a slight bias towards the relative physical distance. It is because the RP locations and their mutual distances are more likely to be preserved under dynamic signal map change, keeping the RPs correlated in signal space. However, the small signal difference should be also considered as these two RPs are likely to be correlated in the signal space. Otherwise the reconstruction error becomes higher given too large θ. In our experiment, we choose $\theta = 0.6$ as the optimal case.

Figure 11 shows the reconstruction process on an individual RP by CSR in the corridor environment. We can

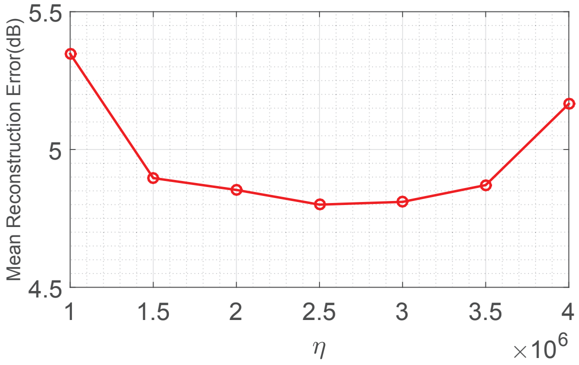

Figure 9: Mean reconstruction error (dB) versus parameter η in corridor.

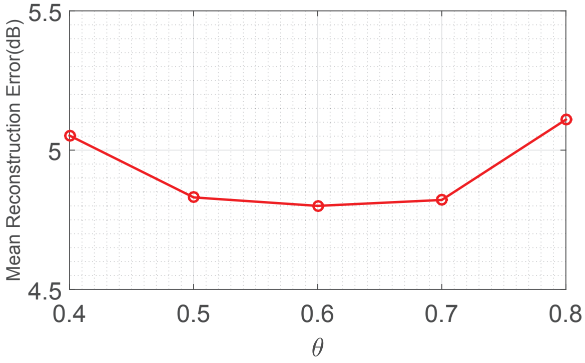

Figure 10: Mean reconstruction error (dB) versus parameter θ in corridor.

observe that as more crowdsourced signals are fed to the CSR, the smaller RSSI reconstruction error will be. We can observe that after several updates (say, 5 in our case) the reconstructed RSSI closely matches the ground-truth one.

Figure 12 illustrates the signal map reconstruction results of an AP using CSR in the corridor environment. We also show the prediction error bars in the figure. We leverage 30% crowdsourced samples in the whole site. In this figure, black crosses represent the original and outdated RSSIs, while the blue diamonds are the ground-truth signals. Their difference in signal level indicates that the signal map has changed. The signal map is then updated by CSR, and the red circles are the adapted RSSI values. We can observe that the predicted RSSIs (red circles) closely match ground truth (blue diamonds). It shows that CSR successfully adapts the signal map towards the ground truth, significantly reduces the difference with the altered signal map.

Figure 13 shows the CDF of signal map reconstruction error (e^n_{rec} in dB) using LDPL, BP and CSR in the corridor environment. We also show the actual RSSI change (e^n_{true}) in the signal map, denoted as "Without Update". CSR improves the mean reconstruction error by around 29% (from 6.6 dB to 4.7 dB) in the corridor. Given the same number of crowdsourced samples (only 40% samples in the whole site), we can observe that CSR outperforms LDPL and BP schemes. It is because unlike LDPL reconstruction the BCS in CSR does not rely on any radio propagation model or line-of-sight condition. Furthermore, the probabilistic formulation in CSR tolerates the signal noise and is more robust than the deterministic BP scheme.

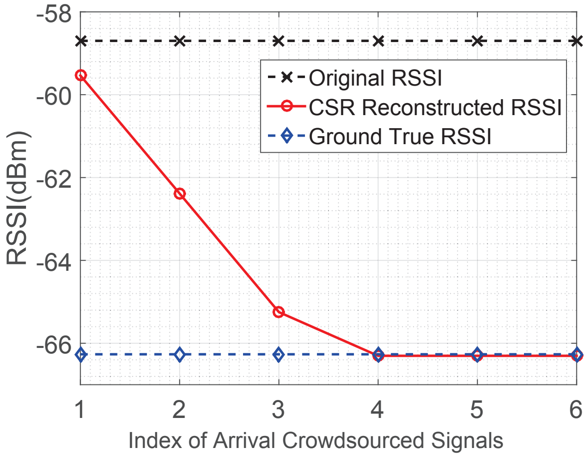

Figure 11: Reconstructed RSSI of an AP versus the index of crowdsourced signals at a single RP.

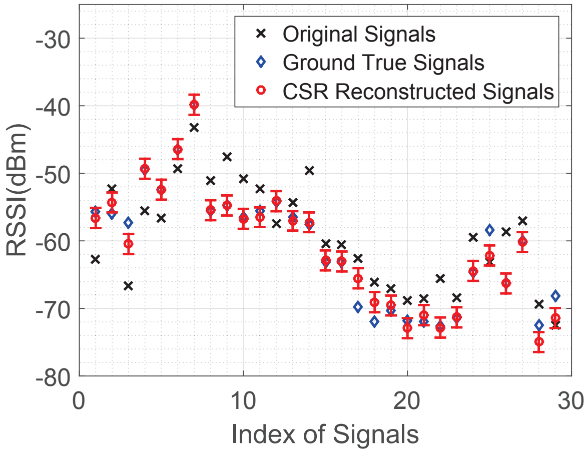

Figure 12: RSSI reconstruction results using CSR for one AP at different RPs.

Figure 14 shows the mean signal map reconstruction error (dB) versus different number of crowdsourced signals in the corridor environment. At the beginning, as no crowdsourced data are fed, the RSSI reconstruction error is high for all schemes. As more crowdsourced samples are given, the RSSI reconstruction accuracy increases. We can observe that CSR already achieves high RSSI reconstruction accuracy even given 40% of all crowdsourced samples. In our deployment setting, we set 40% by default to achieve balance between the RSSI error and the crowdsourcing efficiency.

Figure 15 shows the mean signal map reconstruction error (dB) versus the location input error in the corridor environment. We simulate the scenario when the location of the RSSI vector is fed to CSR. We consider a Gaussian location error upon the ground-truth coordinate. Clearly, the RSSI reconstruction error increases as larger location input error is added. We can observe that even under large location input error (say, up to 6 m), CSR can still achieve better reconstruction accuracy than the other schemes.

Figure 16 shows the CDF of signal map reconstruction error in the hallway environment (see Figure 6). Similar to Figure 13, CSR achieves much lower RSSI reconstruction error compared with LDPL and BP. As the signal reconstruc-

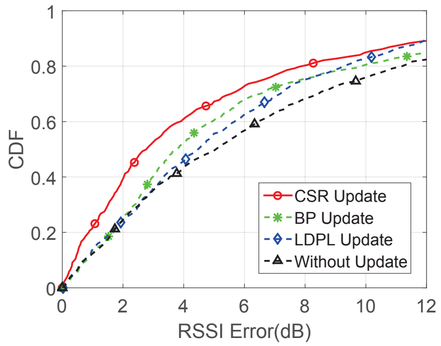

Figure 13: Cumulative probability of RSSI reconstruction error (dB) in corridor.

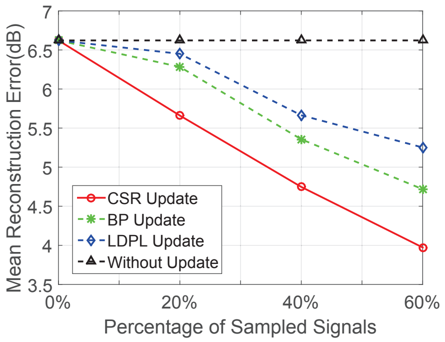

Figure 14: Mean reconstruction error versus percentage of samples used in corridor.

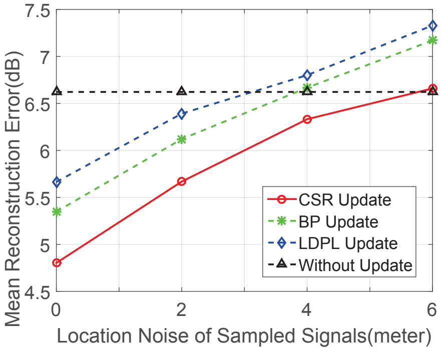

Figure 15: Mean reconstruction error versus location input error in corridor.

tion results are qualitatively similar to those in the corridor, for brevity we do not repeat them here.

In the following, we present the improvement in Wi-Fi fingerprint-based localization accuracy using CSR. Figure 17 shows the CDF of localization errors using the Wi-Fi fingerprints updated by different signal map reconstruction schemes. We can observe that CSR improves the localization error by at least 40%, and the updated localization accuracy with CSR is close to that with manual site survey (the ground-truth signal map). It shows that CSR effectively updates the Wi-Fi fingerprint database and improves the adaptivity of indoor localization systems towards the environmental change.

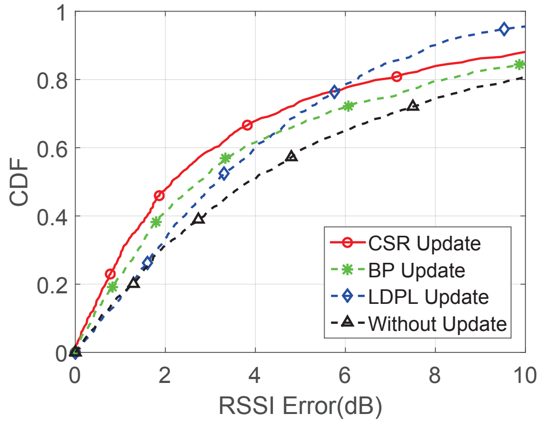

Figure 16: Cumulative probability of RSSI reconstruction error (dB) in the hallway.

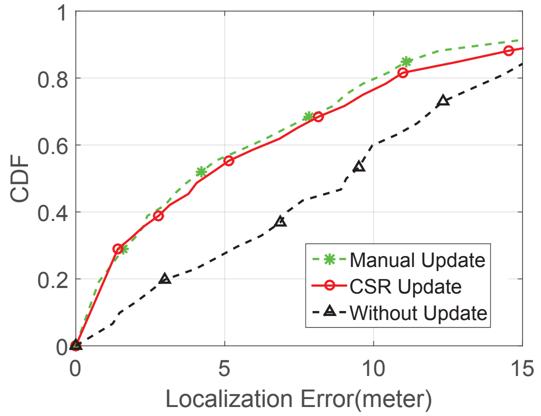

Figure 17: Cumulative errors of the Wi-Fi fingerprint-based localization system integrated with CSR.

6. CONCLUSION

In this paper, we propose Compressive Signal Reconstruction (CSR), a novel signal map online learning scheme using Bayesian compressive sensing (BCS) and crowdsourcing. Via the BCS formulation, CSR not only finds the signal changes for radio map update, but also provides error bars for confidence inference. Besides, CSR does not rely on any radio propagation model, and can be easily integrated with any wireless applications such as Wi-Fi monitoring or indoor localization systems.

We have deployed CSR in our university campus. Extensive experiments show that CSR achieves more than 30% improvement in RSSI reconstruction accuracy compared with other state-of-the-art schemes. We have shown that CSR is also robust to location input errors (as much as 6 meters) in the crowdsourced signals. Furthermore, we have also integrated CSR with existing Wi-Fi fingerprint-based indoor localization systems. Given the changed signal map, CSR effectively reconstructs the Wi-Fi fingerprint database and improves the localization accuracy by at least 40%.

7. REFERENCES

[1] M.M. Atia, A. Noureldin, and M.J. Korenberg. Dynamic online-calibrated radio maps for indoor positioning in wireless local area networks. *IEEE Trans. Mobile Computing*, 12(9):1774–1787, Sept 2013.

[2] M. Azizyan, I. Constandache, and R. R. Choudhury. SurroundSense: Mobile phone localization via ambience fingerprinting. In *Proc. ACM MobiCom*, pages 261–272, 2009.

[3] P. Bahl and V.N. Padmanabhan. RADAR: An in-building RF-based user location and tracking system. In *Proc. IEEE INFOCOM*, volume 2, pages 775–784, 2000.

[4] Walter Balzano, Aniello Murano, and Fabio Vitale. WiFACT – wireless fingerprinting automated continuous training. In *Proc. IEEE AINA*, pages

75–80, 2016.

[5] M. J. Beal. Variational algorithms for approximate Bayesian inference. *PhD Thesis*, pages 1–281, 2003.

[6] C. M. Bishop and M. E Tipping. Variational relevance vector machines. In *Proc. UAI*, pages 46–53. Morgan Kaufmann Publishers, 2000.

[7] P. Bolliger. Redpin - adaptive, zero-configuration indoor localization through user collaboration. In *Proc. MELT*, pages 55–60. ACM New York, NY, USA, 2008.

[8] E. J. Candes and T. Tao. Decoding by linear programming. *IEEE Trans. Information Theory*, 51(12):4203–4215, 2005.

[9] K. Chang and D. Han. Crowdsourcing-based radio map update automation for Wi-Fi positioning systems. In *Proc. ACM GeoCrowd*, pages 24–31. ACM New York, NY, USA, 2014.

[10] Yi-Chao Chen, Ji-Rung Chiang, Hao-hua Chu, Polly Huang, and Arvin Wen Tsui. Sensor-assisted Wi-Fi indoor location system for adapting to environmental dynamics. In *Proc. ACM MSWiM*, pages 118–125, 2005.

[11] D. L. Donoho. Compressed sensing. *IEEE Trans. Information Theory*, 52(4):1289–1306, 2006.

[12] Ke-Lin Du and M. N. S. Swamy. Radial basis function networks. In *Neural Networks and Statistical Learning*, pages 299–335. Springer London, 2014.

[13] M. F. Duarte and Y. C. Eldar. Structured compressed sensing: From theory to applications. *IEEE Trans. Signal Processing*, 59, 2011.

[14] Moustafa Elhamshary, Moustafa Youssef, Akira Uchiyama, Hirozumi Yamaguchi, and Teruo Higashino. TransitLabel: A crowd-sensing system for automatic labeling of transit stations semantics. In *Proc. ACM MobiSys*, 2016.

[15] B. Ferris, D. Fox, and N. Lawrence. WiFi-SLAM using gaussian process latent variable models. In *Proc. IJCAI*, pages 2480–2485. Morgan Kaufmann Publishers Inc. San Francisco, CA, USA, 2007.

[16] M. Gunawan, B. Li, T. Gallagher, and G. Retscher. A new method to generate and maintain a WiFi fingerprinting database automatically by using RFID. In *Proc. IPIN*, pages 1–6. IEEE, 2012.

[17] B. Guo, C. Chen, D. Zhang, Z. Yu, and A. Chin. Mobile crowd sensing and computing: when participatory sensing meets participatory social media. *IEEE Communications Magazine*, 54(2):131–137, February 2016.

[18] Suining He, Bo Ji, and S.-H. Gary Chan. Chameleon: Survey-free updating of fingerprint database for indoor localization. *IEEE Pervasive Magazine*, to appear.

[19] A. K. M. M. Hossain and W. S. Soh. A survey of calibration-free indoor positioning systems. *Computer Communications*, 66(15):1–13, 2015.

[20] Y. Hu, W. Zhou, Z. Wen, Y. Sun, and B. Yin. Efficient radio map construction based on low-rank approximation for indoor positioning. *Mathematical Problems in Engineering*, 2013, 2013.

[21] Maissa Ben Jamâa, Anis Koubâa, and Yasir Kayani. EasyLoc: RSS-based localization made easy. *Procedia Computer Science*, 10:1127–1133, 2012.

[22] Qideng Jiang, Yongtao Ma, Kaihua Liu, and Zhi Dou. A Probabilistic Radio Map Construction Scheme for Crowdsourcing-Based Fingerprinting Localization. *IEEE Sensors Journal*, 16(10):3764–3774, 2016.

[23] Shaoyang Li, Xiaoming Tao, and Jianhua Lu. A variational Bayesian em approach to structured sparse signal reconstruction. In *Proc. IEEE VTC Fall*, pages 1 – 5. IEEE, 2014.

[24] C. Luo, L. Cheng, M. C. Chan, Y. Gu, J. Li, and M. Zhong. Pallas: Self-bootstrapping fine-grained passive indoor localization using WiFi monitors. *IEEE Trans. Mobile Computing*, 2016.

[25] Halgurd S. Maghdid, Ihsan Alshahib Lami, Kayhan Zrar Ghafoor, and Jaime Lloret. Seamless Outdoors-Indoors Localization Solutions on Smartphones. *ACM Computing Surveys*, 48(4), 2016.

[26] Andrea Massa, Paolo Rocca, and Giacomo Oliveri. Compressive sensing in electromagnetics - a review. *IEEE Antennas and Propagation Magazine*, 57(1):224–238, 2015.

[27] D Needell and J A Tropp. CoSaMP: Iterative signal recovery from incomplete and inaccurate samples. *Applied and Computational Harmonic Analysis*, 26(3):301–321, 2009.

[28] S. Nikitaki, G. Tsagkatakis, and P. Tsakalides. Efficient training for fingerprint based positioning using matrix completion. In *Proc. EUSIPCO*, pages 195–199, Aug 2012.

[29] S. Nikitaki, G. Tsagkatakis, and P. Tsakalides. Efficient multi-channel signal strength based localization via matrix completion and Bayesian sparse learning. *IEEE Trans. Mobile Computing*, 14(11):2244–2256, Nov 2015.

[30] Jun-geun Park, Ben Charrow, Dorothy Curtis, Jonathan Battat, Einat Minkov, Jamey Hicks, Seth Teller, and Jonathan Ledlie. Growing an organic indoor location system. In *Proc. ACM MobiSys*, pages 271–284. ACM New York, NY, USA, 2010.

[31] Saad Qaisar, Rana Muhammad Bilal, Wafa Iqbal, Muqaddas Naureen, and Sungyoung Lee. Compressive Sensing: From Theory to Applications, A Survey. *Journal of Communications and Networks*, 15(5):443–456, 2013.

[32] Hyojeong Shin, Yohan Chon, Yungeun Kim, and Hojung Cha. MRI: Model-based radio interpolation for indoor war-walking. *IEEE Trans. Mobile Computing*, 14(6):1231–1244, 2015.

[33] Michael E Tipping. Sparse Bayesian learning and the relevance vector machine. *The Journal of Machine Learning Research*, pages 211–244, 2001.

[34] Joel a. Tropp and Anna C. Gilbert. Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Trans. Information Theory*, 53(12):4655–4666, 2007.

[35] C. Wu, Z. Yang, and Y. Liu. Smartphones based crowdsourcing for indoor localization. *IEEE Trans. Mobile Computing*, 14(2):444–457, Feb 2015.

[36] Moustafa Youssef and Ashok Agrawala. The Horus WLAN location determination system. In *Proc. ACM MobiSys*, pages 205–218, 2005.