Towards Crowdsourced Signal Map Construction via Implicit Interaction of IoT Devices

Suining He S.-H. Gary Chan
Department of Computer Science and Engineering, The Hong Kong University of Science and Technology
Email: {sheaa, gchan}@cse.ust.hk

***Abstract*—Site survey cost has become a major deployment challenge for large-scale wireless systems. Some previous approaches require explicit user inputs to label the locations of crowdsourced signals, which is inconvenient in practice. Others may rely on intensively-calibrated specialized sensors. Furthermore, the locations of crowdsourced signals are usually random and sparse, making it difficult to build a complete signal map.**

To address these, we propose Surecose, a novel signal map construction system by implicit crowdsourcing and interaction of low-cost cooperative sensors. Surecose is initialized with the light-weight sparse signal map collection. Having the background application on, the naive users are implicit surveyors unknowingly uploading crowdsourced signals. Phones may detect each other (mobile sensors), or beacons (fixed sensors). To label their locations, Surecose measures the mutual sensor proximity of the IoT devices. Via joint location labeling and sensor calibration, Surecose transparently locates the users despite devices used. Then Surecose constructs the database with a directionally and environmentally adapted Gaussian process, which scales the labeled locations and RSSIs anywhere into a complete signal map. Extensive experiments in diverse environments, including a campus hall, an international airport and a premium shopping mall, have validated that Surecose can adaptively and accurately construct the Wi-Fi signal map (often more than 20% reduction in RSSI construction error), with low survey cost (around 70% survey reduction) and little human intervention.

***Index Terms*—Sensor interaction; implicit crowdsourcing; signal map construction; automatic location labeling; sensor calibration; adaptive directional signal regression.**

I. Introduction

In many wireless systems, received signal strength indicator (RSSI) exhibits spatial variation, forming the so-called signal map (or the heat map) in the site. Knowledge of the signal map is essential for many applications. For instance, in a WLAN network, the signals of the access points (AP) may vary with respect to wall partition and furniture obstruction. System administrators would be interested in the signal map so as to understand the installed Wi-Fi coverage, or to adjust/tune Wi-Fi settings. In fingerprint-based localization, constructing its signal database would be essential for its deployment.

Despite its wide application, traditional signal map construction is often conducted via the laborious and tedious *site survey* or so-called *fingerprinting*. For spacious indoor sites like the international airport or the large shopping mall, traditional survey is costly and infeasible.

This work was supported, in part, by Guangdong Provincial Department of Science and Technology (GDST16EG04 2016A050503024) and Natural Science Foundation of Guangdong Province (2014A030313154).

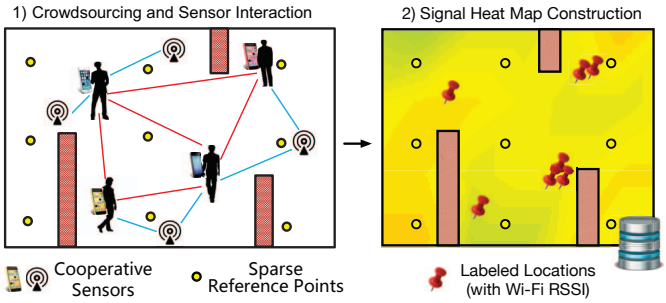

Fig. 1: Illustration of (1) automatic location labeling with crowdsourcing and sensor interaction, and (2) adaptive signal map construction.

Survey reduction may be achieved via low-cost crowdsourcing. A signal map can be generated by the participatory sensing by naive ILBS users. However, in most of the recent studies, two major challenges have not been well addressed before the crowdsourcing system is deployable: 1) how to *estimate the locations of these RSSI vectors without intrusive user intervention*: Traditional fingerprinting is to manually associate (or *label*) the signals with their ground-truth locations (say, reference points or RPs). We cannot always expect naive users to explicitly input her/his locations, which, however, is often assumed by many existing crowdsourcing-based systems; 2) how to *construct a complete signal map* of the site: Crowdsourced locations may be randomly and sparsely distributed, rather than in regular grid for practical use. *Adaptively scaling* the crowdsourced data to a complete signal map (of reasonable coverage and quality) will definitely benefit the deployment of large-scale wireless systems.

To address above issues, we illustrate in Figure 1 a new framework for survey reduction and signal map construction [1]. We only need sparse signal map (with data points or RPs in large grid size) beforehand via simple and fast site survey. Beyond that, recent boom in Internet of things (IoT) has enabled the communication and cooperation (interaction) among various off-the-shelf IoT devices. These sensors can be either fixed (like iBeacons and Wi-Fi sniffers), or mobile (peer smartphones) with the users. They interactively measure the mutual proximity. These proximity readings and crowdsourced RSSIs jointly help label these user locations without their explicit location inputs. Afterwards, given the RSSIs and estimated locations crowdsourced *anywhere*, we can *scale* them into a signal map of the site by a certain learning and

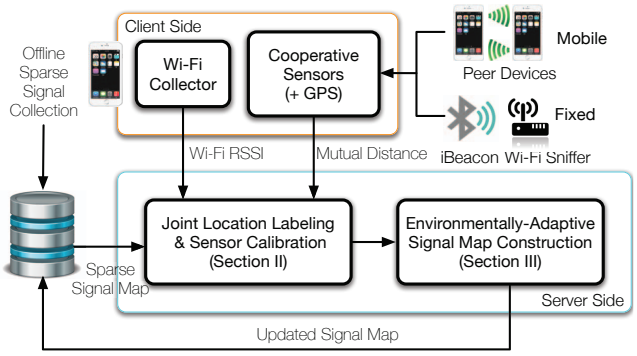

Fig. 2: System flow of our proposed survey-reduction framework.

prediction approach. The initial sparse signal map is then further *updated* and *improved* into a more complete one.

Two important issues have to be addressed such that above framework can be realized: 1) Location labeling needs to be implicit and IoT devices are heterogeneous in signals. How to *jointly* address both is important for the signal map labeling. 2) Survey environments may be complex and partitioned. How to achieve *adaptive* and accurate signal map prediction is critical.

To address above, we propose ***Surecose***, a novel ***su***rvey ***re***duction and signal map construction system with implicit ***c***rowds***o***urcing and cooperative ***se***nsors. Our major contributions are in three folds as follows:

- *Joint Location Labeling & Sensor Calibration* for Crowdsourced Signals: To automatically label the locations of the crowdsourced RSSIs, we propose a novel joint optimization-based algorithm which localizes all the targets given their RSSIs and mutual distances. To our best knowledge, Surecose is the first framework which jointly calibrates the received signal strength (RSS) of interactive sensors such that the target locations are estimated implicitly despite different IoT devices used.
- *Environmentally-Adaptive Signal Map Construction*: The estimated user locations (with radio signals) are usually sparse and random in the site. Without any preprocessing, they cannot scale to the complete signal map for practical use. To address this, we adapt the Gaussian process (GP) to construct signal points *anywhere*. Based on the labeled crowdsrouced signals, Surecose adaptively predicts signals (virtual signal points) at the regular grids of RPs, and builds up the complete signal database beyond the sparse one. Our adapted GP considers the directions of the received signals with respect to the APs, which adapts to the complex and partitioned indoor environments.
- *Extensive Experiment & Prototype Studies*: We have conducted large-scale experiments in distinct environments, including our HKUST campus hall, the Hong Kong International Airport (HKIA) and a premium shopping mall of the Hong Kong Olympian City (HKOC). Results have shown that Surecose adapts to different signal environments and accurately constructs the signal map with low initial cost (*often achieving more than* 70% *workload reduction in offline survey cost*).

The overall system flow of Surecose is shown in Figure 2. In the offline (initialization) phase, Surecose is initiated with a sparse signal map. $\langle \text{RSSI}, \text{Location} \rangle$'s are sparsely collected and stored in the signal database. Then in the online (data collection) phase, smartphones of the naive users (*client side*) implicitly sample RSSIs in the WLAN (say, Wi-Fi). Meanwhile, the clients may detect proximity from IoT devices (fixed or mobile), or obtain occasional GPS fixes (with confidence range) as locational constraints. Sampled data may be stored in the client phones and uploaded later to *server side*. *Joint Location Labeling & Sensor Calibration* module (Section II) locates the crowdsourcing users, and implicitly calibrates the sensor readings. Surecose associates the crowdsourced location labels with the Wi-Fi RSSI vectors. Finally, given the crowsourced inputs and RPs, *Environmentally-adaptive Signal Map Construction* (Section III) returns the newly constructed signal map to the database, and updates the sparse one. With the crowdsourcing and transparent sensor interaction, Surecose gradually builds up the signal map with low cost.

This paper is organized as follows. In Section II, we first present joint location labeling and sensor calibration to estimate crowdsourcing locations. Given crowdsourced data, we present how to adaptively construct the signal map for different environments in Section III. Experimental results are illustrated in Section IV. Afterwards, we discuss the related works in Section V, and finally conclude in Section VI.

II. Joint Location Labeling & Sensor Calibration

The first task of Surecose is to label locations of the crowdsourced signals, given only a sparse signal map and mutual sensor proximity. We propose a novel joint optimization scheme to locate the users automatically despite different devices used. We first present the preliminaries of the formulation (Section II-A), followed by the novel objective function and constraints (Section II-B). Finally, we present the formulation and its complexity analysis (Section II-C).

A. Preliminaries of Signals & Location Estimation

Recall that Surecose is initialized with a sparse signal map (say, with Q reference points on regular grid) in the site, which can be obtained by surveyors via low-cost site survey (say, fingerprinting [2]). The survey grid size of reference points (RPs) can be large, say, in 10 to 15 m, depending on the site size and expected location labeling accuracy. Denote f_q^l as the mean of RSSI samples (dBm) at RP q from AP l, and we form the reference vector at RP q as

$$\boldsymbol{F}_q = \left[f_q^1, f_q^2, \ldots, f_q^L \right], \tag{1}$$

whose 2-D coordinate in the indoor map is denoted as $\mathbf{s}_q$.

Then in the online signal map construction phase, the users (targets) implicitly contribute their Wi-Fi RSSI signals. Surecose labels the locations of these mobile users (targets). Let $\mathbf{V} = \{1, 2, \ldots, M\}$ be the index of crowdsourcing users, and $\widehat{\mathbf{x}}_m, m \in \mathbf{V}$ be the location of user m to be estimated. Let t_m^l be the RSSI (dBm) at target m from AP l. Then we have the target RSSI vector at m as

$$\boldsymbol{T}_m = \left[t_m^1, t_m^2, \ldots, t_m^L \right], \tag{2}$$

which is fed to our joint location labeling and sensor calibration, and later used in the signal map construction.

RPs $\mathbf{s}_q$'s in the sparse signal map are used to locate the mobile users. Each sparse RP is given a weight variable ω_{mq} in the formulation. Specifically, we estimate the user m's location with weights $\{\omega_{mq}\}$ and all RPs $\{\mathbf{s}_q\}$, i.e.,

$$\widehat{\mathbf{x}}_m = \sum_{q=1}^{Q} \omega_{mq}\mathbf{s}_q, \quad \text{where } \sum_{q=1}^{Q} \omega_{mq} = 1, \quad \omega_{mq} \geq 0, \quad (3)$$

for $\forall q \in \{1, 2, \ldots, Q\}$. In other words, the better the RP is in satisfying the objectives and constraints in the formulated optimization problem, the higher value ω_{mq} is given, and the more likely the user is estimated near this RP.

B. Objective Function & Constraints

The objectives of our location labeling are in two folds: 1) *minimizing distance errors* of cooperative sensors (measured between users' mobile sensors, or between mobile and fixed sensors); 2) *maximizing the similarity matching* between the target RSSIs and the sparse signal map. As solving the bi-objective optimization is difficult, we keep one objective and set another as the constraint to satisfy both criteria above [3].

In terms of mutual distance, we consider the relatively small proximity (say, less than 8 m) for cooperative sensor communication in Surecose, which provides more reliable indication of nearby mobile or fixed devices. Let P_{mn} be the *received signal strength* (RSS in dBm) between IoT devices m and n. Similar to [4], [5], the relationship of their mutual distance δ_{mn} and the RSS value P_{mn} can be approximated as

$$\delta_{mn} = \alpha_{mn}P_{mn} + \beta_{mn}, \ \alpha_{mn} < 0, \ \beta_{mn} < 0. \quad (4)$$

Many empirical studies [4], [5] have shown that in the indoor LOS conditions with short range, a simple linear law may offer sufficient match with experimental data instead of the standard log-distance path loss model. This can be partly related to wave guiding effects characterizing indoor propagation along corridors and narrow space [5]. We set the bound constraints as $\alpha_{min} \leq \alpha_{mn} \leq \alpha_{max}, \beta_{min} \leq \beta_{mn} \leq \beta_{max}$. Calibrating $\boldsymbol{\theta}$ helps Surecose to adapt to different cooperative IoT devices.

For each user m, we denote the index set of her/his detected devices (including mobile sensors of peer phones and fixed sensors) as Ω_m. Note that for the sensors $n \in \Omega_m$, some may be fixed sensors with known locations, while others may be from mobile peer smartphones with location $\mathbf{x}_n$ to be estimated. Our first objective of the location labeling in Surecose is to find the weights of RPs, ω_{mq}'s, and parameters $\boldsymbol{\theta}$, which jointly minimize the distance errors, i.e.,

$$\underset{\{\omega_{mq}\},\boldsymbol{\theta}}{\arg\min} \sum_{m=1}^{M} \sum_{n\in\Omega_m} \Theta_{mn} \left(\|\widehat{\mathbf{x}}_m - \mathbf{x}_n\|_2 - \delta_{mn}\right)^2, \quad (5)$$

where $\Theta_{mn} = 1$ if two sensors have mutual distance measurement and 0 otherwise. Note that in practice, the users usually form a *graph* which is not fully connected and the total computation is not high.

To meanwhile maximize the signal matching for location estimations (the second objective), we further constrain weights ω_{mq}'s. Specifically, let $cos(\boldsymbol{T}_m, \boldsymbol{F}_q)$ be *cosine similarity* between $\boldsymbol{T}_m$ and $\boldsymbol{F}_q$ (the larger, the more similar). For each m, we characterize the matching between target signals and the sparse RPs (signal points) by weighted sum of cosine similarity, i.e., $sim(\mathbf{T}_m) = \sum_{q=1}^{Q} cos(\boldsymbol{T}_m, \boldsymbol{F}_q) \cdot \omega_{mq}$ $(0 \leq sim(\mathbf{T}_m) \leq 1)$. We set a lower bound constraint as

$$\sum_{m=1}^{M} sim(\mathbf{T}_m) \geq \gamma \cdot M \quad (0 \leq \gamma \leq 1), \quad (6)$$

where γ is a tunable parameter (evaluated in Section IV) representing the tradeoff between mapping accuracy and computation efficiency.

Note that the larger $sim(\mathbf{T}_m)$ is, the more matched the estimated target location (Equation (3)) are with the RPs nearby (those of high ω_{mq}). Equation (6) ensures that the weighted signal similarity between the target RSSIs and the sparse signal map is lower bounded. The optimizer enlarges ω_{mq}'s of RPs with more similar signals to satisfy this bound, and our second objective is hence jointly satisfied. With Equations (5) and (6), the distance errors and the signal matching are jointly considered in Surecose to label the crowdsourcing locations.

Besides, given all Q RPs in the sparse signal map, as the crowdsourcing users are more likely surrounded by sparse RPs than exactly on one of them, we constrain the location estimations to be between RPs, i.e.,

$$\omega_{mq} \leq \lambda, \quad \lambda \triangleq \frac{\max cos(\boldsymbol{T}_m, \boldsymbol{F}_q)}{\sum_{q=1}^{Q} cos(\boldsymbol{T}_m, \boldsymbol{F}_q)}, \quad (7)$$

which adaptively prevents large weight assignment (i.e., with some ω_{mq} close to 1) on an RP. Note that other signal similarity metrics (say, Euclidean distance) can be also applied.

Users may occasionally obtain a GPS location fix $\widehat{\mathbf{x}}_{mg}$ and a confidence range δ_{mg} (for example, near the building gates or close to outdoor). It may provide an optional constraint (a rough region), i.e., $\|\widehat{\mathbf{x}}_m - \widehat{\mathbf{x}}_{mg}\|_2^2 \leq \delta_{mg}^2$.

C. Problem Formulation & Complexity Analysis

Our novel *joint* location labeling and sensor calibration finds the crowdsourced locations such that the mutual distance errors are minimized given signal similarity constraints, i.e.,

$$\begin{array}{ll} \underset{\{\omega_{mq}\},\boldsymbol{\theta}}{\arg\min} & \text{Objective } (5), \\ \text{s.t.} & \text{Constraints } (3),\ (4),\ (6),\ (7). \end{array} \quad (8)$$

The above formulation can be efficiently solved by some convex optimization solvers [3]. $\{\omega_{mq}\}$'s are then used for location labeling by Equation (3). Labeling crowdsourced locations is conducted automatically at the background (might be returned to users as byproducts), and ⟨RSSI, location⟩'s are used in the next phase to construct the complete signal maps.

We briefly analyze the computational complexity. Given M crowdsourcing users and Q sparse RPs, Formulation (8) takes $\mathcal{O}(Q^3M^3)$ [3]. The location labeling and sensor calibration can be conducted at the server side and therefore the user

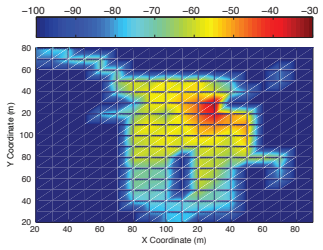

Fig. 3: Directional difference in signal propagation (campus, RSSI in dBm).

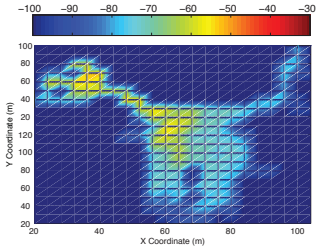

Fig. 4: Spatial difference in signal propagation (campus, RSSI in dBm).

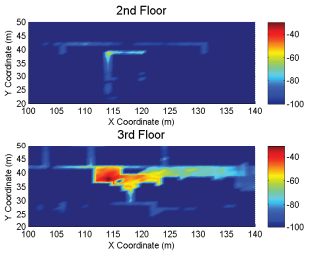

Fig. 5: Floor difference in signal propagation (campus, RSSI in dBm).

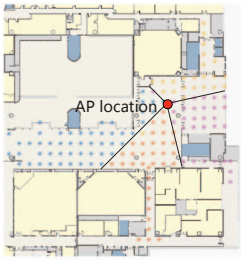

Fig. 6: RPs of an AP clustered by affinity propagation clustering w.r.t. directions (campus).

experience at mobiles is not affected. The computation can be further reduced via region partitioning and mapping [6].

III. Environmentally-Adaptive Signal Map Construction

The second task here is to construct the virtual signal points as the RPs of regular grids, and form a complete signal map for wireless applications. It is because the labeled crowdsourced locations are usually random and sparse, which may not cover all the areas of interest in the site. Using a regular grid increases the scalability of survey reduction so that the indoor areas (including locations of the predefined RPs) unexplored in crowdsourcing can be effectively covered by the signal map. Surecose formulates an adaptive Gaussian process (GP) to construct the complete signal map. We further preprocess the data so that GP predicts more adaptively the signal points according to the radio direction and regions.

In Section III-A, we first identify the heterogeneity (directional and regional) of signal propagation in complex environment, and how to preprocess input data (i.e., ⟨RSSI, location⟩'s) correspondingly. Then we show in Section III-B how we regress the RSSI spatial distribution based on GP, taking into account the signal properties at various directions and environments. We then discuss the hyperparameter learning in GP and provide complexity analysis in Section III-C.

A. Environmentally-Adaptive Data Preprocessing

In our deployment, we have observed the signal propagation is environmentally heterogeneous due to complex indoor settings. We first characterize the directional and regional radio propagation via some signal maps. Figures 3, 4 and 5 show the signal maps of 3 different APs at our university atrium hall. We collect totally 520 RPs in 5 m density for this $2,000$ m^2 survey site. We can observe that signal path loss around the red peak (potential AP location) is markedly anisotropic (heterogeneous) w.r.t. directions (Figure 3), while showing shows various distribution patterns across corridors (Figure 4) and floors (Figure 5). If we only uniformly regress and predict signals despite the directions and regions [7], [8], [9], large errors still exist as RF propagation parameters may still vary accordingly. We hence need to consider above properties in construct a signal map reflecting the ground-truth distribution.

To address this, we first estimate the AP locations like [7], based on the input signals (i.e., sparse signal points $\boldsymbol{F}_q$'s and the crowdsourced $\boldsymbol{T}_m$'s) which detect this AP. Given the relative directions of these signal points w.r.t. the AP location, we apply affinity propagation clustering [10] to adaptively partition the input locations into multiple clusters. This efficient clustering algorithm requires no explicit cluster number like k-means clustering. Hence we can cluster the data into groups of adaptive size for each AP. Figure 6 shows 4 groups of sparse RPs in the offline phase are formed w.r.t. their relative directions from the estimated AP peak (red dot).

Suppose after the clustering, the signal data of each AP indexed by l, including sparse signal points and the crowdsourced ones, are partitioned into C^l sets (data are first partitioned at each floor if collected in a multi-storey building). Each set of input locations $\mathbf{X}_c^l$ and their corresponding RSSI vectors $\boldsymbol{F}_c^l$, indexed by c ($1 \leq c \leq C^l$), are fed to the GP for database construction. Then at each RP in grid, the AP RSSI is predicted locally based on the signals in the corresponding cluster (with similar propagation direction) at that floor.

B. Signal Prediction via Adaptive Gaussian Process

After grouping the data into different clusters, we present how to scale them into a signal map. Given training data $\mathbf{X}_c^l$ and $\boldsymbol{F}_c^l$ of AP l, we are to predict the RSSIs at any arbitrary location $\mathbf{x}_*$ in a grid (new RPs). We first regress the parameters in a path loss model. Then given the signal propagation model and crowdsourced RSSI samples, the Gaussian process predicts the RSSIs at predefined regular grids to form the virtual signal points. Surecose then stores them in database for ILBS or signal map monitoring. Note that in our GP, the path loss is used to *reflect overall signal propagation trend*, while the additional terms upon the path loss *capture local signal distribution* in complex none-line-of-sight (NLoS) environment. Therefore, GP is adaptive to the complex NLOS indoor environments with wall partitioning [11], [1].

We first regress the path loss model based on input signals to reflect the overall signal trend. GP utilizes RSSIs in $\boldsymbol{F}_c^l$ to regress the signal propagation parameters and AP l's 2-D location $\mathbf{x}^l$. Given the input $< \mathbf{X}_c^l, \boldsymbol{F}_c^l >$, denote $m(\mathbf{X}_c^l)$ as the predicted mean RSSI using log-distance path loss model. Based on the path loss model, for each $\mathbf{x}^* \in \mathbf{X}_c^l$, we have

$$m(\mathbf{x}^*) = P_c^l - 10\Gamma_c^l \log_{10}\left(\frac{\|\mathbf{x}^* - \mathbf{x}^k\|_2}{d_0}\right), \qquad (9)$$

where P_c^l and Γ_c^l ($P_c^l < 0$ and $\Gamma_c^l > 0$) are the parameters for signal propagation within the RP cluster c. $d_0 = 1$ m in

our setting. In regressing the path loss model, let $\mathbf{e}_c^l$ be the difference between measured RSSIs $\boldsymbol{F}_c^l$ and $m(\mathbf{X}_c^l)$, i.e.,

$$\mathbf{e}_c^l = \left| \boldsymbol{F}_c^l - m(\mathbf{X}_c^l) \right|. \tag{10}$$

GP regression here is to find P_c^l, Γ_c^l and $\mathbf{x}^l$ such that the sum of differences with all inputs is minimized. This can be solved via traditional gradient descent method [3].

After regressing the path loss model, we form GP to predict RSSIs at unexplored locations of interest (new RPs). Let N_c^l be the number of all the signal points in cluster c for AP l. Based on GP, Wi-Fi RSSIs at different locations are considered as correlated [12], and the covariance between two signal points $f_i^l, f_j^l \in \boldsymbol{F}_c^l$ depends on the distance of i and j. Based on the GP formulation [12], we have the covariance as

$$cov(\mathbf{x}_i, \mathbf{x}_j) = (\tau_c^l)^2 \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|_2^2}{2(g_c^l)^2} \right), \tag{11}$$

where the hyperparameters $(\tau_c^l)^2$ and g_c^l are the signal variance and scaling factors, respectively. $cov(\mathbf{x}_i, \mathbf{x}_j)$ means that the closer two locations (say, the crowdsourced RPs) are in physical space, the more correlated their signals would be. Considering the RSSI noise, we further formulate the covariance function between f_i^l and f_j^l as

$$cov(f_i^l, f_j^l) = cov(\mathbf{x}_i, \mathbf{x}_j) + (\sigma_c^l)^2 \Delta_{ij}, \tag{12}$$

where Δ_{ij} is 1 if $i = j$ and 0 otherwise. Given AP l and signal point locations $\mathbf{X}_c^l = \left[\mathbf{x}_1^l, \ldots, \mathbf{x}_{N_c}^l\right]$, covariance of the observed RSSIs $\boldsymbol{F}_c^l = \left[f_1^l, \ldots, f_{N_c}^l\right]$ becomes [12]

$$\mathbf{K}_{\boldsymbol{F}} = \mathbf{K}_c^l + (\sigma_c^l)^2 \mathbf{I}, \tag{13}$$

where $\mathbf{K}_c^l$ is the $N_c^l \times N_c^l$ covariance matrix of the input locations, i.e., $\mathbf{K}_c^l[i,j] = cov(\mathbf{x}_i, \mathbf{x}_j)$.

Then posterior of the predicted RSSI $f_{\mathbf{x}_*}^l$ at a location $\mathbf{x}_*$ is distributed with mean $\mu_{\mathbf{x}_*}^l$ and covariance $\sigma_{\mathbf{x}_*}^l$, i.e.,

$$p\left(f_{\mathbf{x}_*}^l | \mathbf{x}_*, \mathbf{X}_c^l, \boldsymbol{F}_c^l\right) = \mathcal{N}\left(\mu_{\mathbf{x}_*}^l, (\sigma_{\mathbf{x}_*}^l)^2 + (\sigma_c^l)^2\right), \tag{14}$$

where $(\sigma_c^l)^2$ represents their corresponding observation noise. Then the predicted mean RSSI, denoted as $\mu_{\boldsymbol{x}_*}^l$ (dBm), and signal variance $(\sigma_{\boldsymbol{x}_*}^l)^2$ (dB) are given by

$$\begin{aligned} \mu_{\mathbf{x}_*}^l &= m(\mathbf{x}_*) - cov(\mathbf{x}_*, \mathbf{X}_c^l)^T \mathbf{K}_{\boldsymbol{F}}^{-1} \mathbf{e}, \\ (\sigma_{\mathbf{x}_*}^l)^2 &= cov(\mathbf{x}_*, \mathbf{x}_*) - cov(\mathbf{x}_*, \mathbf{X}_c^l)^T \mathbf{K}_{\boldsymbol{F}}^{-1} cov(\mathbf{x}_*, \mathbf{X}_c^l). \end{aligned} \tag{15}$$

Existing wireless applications require a complete signal map. We find the unexplored locations between sparse points and form new RPs as $\mathbf{x}_*$'s in Equation (15). Say, given 10 m preliminary survey density, we complement the predefined grids into 5 m, and predict virtual RF RSSIs upon them to form the signal map. These new $\mathbf{x}_*$'s are also clustered as those in Section III-A. Then $\mathbf{x}_*$'s are fed to the corresponding GP for that cluster (Equation (15)) to predict $f_{\mathbf{x}_*}^l$.

Finally, we obtain $\left[\mu_{\mathbf{x}_*}^l, (\sigma_{\mathbf{x}_*}^l)^2\right]$, and store them in the database as the predicted signal mean and variance. As the RSSIs are dynamically fed by crowdsourced data stream, for each signal f_q^l at RP q, we use the autoregressive moving average to update signals, i.e., $f_q^l = \rho f_q^l + (1-\rho)\mu_q^l$ $(0 \le \rho \le 1)$, and reduce temporal fluctuation. In our experiment, ρ is set as 0.5 empirically. RSSI device dependency can be further calibrated via linear regression or maximum likelihood [4]. Via GP, we predict the RSSIs on dense and regular RPs beyond the initial sparse ones. If a new AP is detected, we can add it into our database.

C. Hyperparameter Estimation & Complexity Analysis

Before GP is applied, the hyperparameters for the GP, $\boldsymbol{\theta}_c^l = \{\tau_c^l, \sigma_c^l, g_c^l\}$, are calculated by maximum likelihood estimation (MLE) [12]. Wi-Fi signals in the crowdsourced and sparse signal points within the cluster c are used as training samples, which are considered as jointly Gaussian distributed [12], i.e.,

$$\boldsymbol{F}_c^l \sim \mathcal{N}(m(\mathbf{X}_c^l), \mathbf{K}_{\boldsymbol{F}}), \tag{16}$$

and the log-likelihood of all input RSSI signals $\boldsymbol{F}_c^l$ with hyperparameters $\boldsymbol{\theta}_c^l$ is given by

$$\log p\left(\boldsymbol{F}_c^l | \mathbf{X}_c^l, \boldsymbol{\theta}_c^l\right) = -\frac{1}{2}\mathbf{e}^T \mathbf{K}_{\boldsymbol{F}}^{-1} \mathbf{e} - \frac{1}{2}\log|\mathbf{K}_{\boldsymbol{F}}| - \frac{N_c^l}{2}\log 2\pi.$$

Based on the MLE, we minimize the error function (log-likelihood) from input data $\boldsymbol{F}_c^l$, i.e.,

$$E(\boldsymbol{F}_c^l | \mathbf{X}_c^l, \boldsymbol{\theta}_c^l) \triangleq -\log p(\boldsymbol{F}_c^l | \mathbf{X}_c^l, \boldsymbol{\theta}_c^l). \tag{17}$$

The above MLE problem can also be efficiently solved using gradient-descent algorithm [3]. By calculating the closed forms of gradient decent, we can find the corresponding $\boldsymbol{\theta}_c^l$ and feed them to the GP model. After that, we can predict the signals at new data points correspondingly.

We briefly summarize complexity of signal map construction. Clustering N^l signal points for each AP l takes $\mathcal{O}((N^l)^2)$ [10]. Given N_c^l data points in the cluster c, the calibrated mean function takes $\mathcal{O}(N_c^l)$. Each iteration in the gradient descent (hyperparameter estimation) takes $\mathcal{O}((N_c^l)^3)$ due to the matrix inversion ($\mathbf{K}_{\boldsymbol{F}}^{-1}$). Empirically, gradient descent usually takes only 5 iterations. The GP prediction can efficiently be conducted at the server side, after users upload their data. Given N_u^l unexplored locations for RSSI construction, the signal prediction takes $\mathcal{O}(N_c^l N_u^l)$. To summarize, given L APs and C clusters, the overall complexity of our GP signal map construction is $\mathcal{O}(LC((N_c^l)^3 + N_c^l N_u^l))$.

IV. Illustrative Experimental Evaluation

In this section, we first describe our experimental settings (Section IV-A). Then we illustrate the evaluation results in these sites to validate performance of Surecose (Section IV-B).

A. Experimental Settings & Comparison Schemes

Figure 7 shows the test site of a $5,000$ m^2 campus atrium (HKUST). From the floor map we can observe clear wall partitioning and none-line-of-sight (NLoS) measurements are expected. Locations of Wi-Fi sniffers and iBeacons are also shown in the map. We filter away the mobile APs tethered by smartphones, merge virtual APs, and focus on the officially deployed ones to construct signal map. These Wi-Fi APs in the site are installed by independent bodies of the site, and we know neither their transmission power nor actual locations.

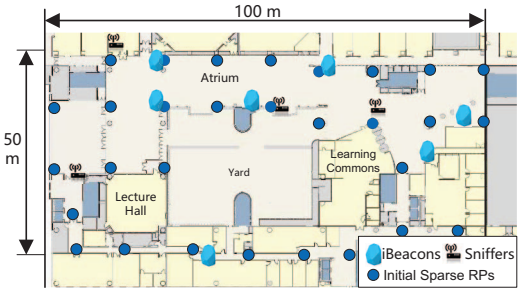

Fig. 7: The map of our university atrium, where blue dots represent the sparse RPs (signal points) with grid size of 10 m.

Besides four Open-WRT-based sniffers adapted from TP-Link routers, we also deploy several iBeacons (with TI CC2541). Compared with Wi-Fi sniffers, iBeacon has smaller coverage and faster signal degradation, which helps differentiate the user location. We combine iBeacons and Wi-Fi sniffers to balance between accuracy and site coverage. Note that the installation locations of the sniffers and beacons can follow the existing WLAN indoors for well-planned coverage. Figure 8 shows the experimental devices including iBeacons, and a connected sniffer (with a laptop server) adapted from a commercial Wi-Fi router (TP-Link). Like the sniffers, iBeacons are similarly attached with the wall. HTC One X and Google Nexus 5 are used for the implicit device interaction (within 15 m) of the naive volunteer users [13].

The experiment consists of two phases: *the preliminary site survey* (only around 15 min over 30 sparse RPs by two dedicated surveyors) and the *crowdsourced RSSI collection* (by cooperative sensors). As density of preliminary signal map and cooperative sensors is low in our setting, the initial cost of Surecose is very small. In the preliminary signal collection, we conduct sparse site survey of 10 m grid size (i.e., the distance between two neighboring RPs) to collect Wi-Fi RSSI vectors. During the preliminary signal collection, a surveyor stands on each RP and collects 15 RSSI samples when he/she is facing each of the four directions (east, south, west and north). In the online phase, ten users' RSSIs and cooperative sensor information are utilized to construct the 126 signal points of 3 m grid size, which cover the site of interest. Note that both phases are conducted at working hours with crowds nearby, and noisy measurements are expected in the data.

Besides crowdsourced data from users and the preliminary sparse signal map, we collect following signals in evaluation:

1) *Comparison data* (signal map construction evaluation): the ground-truth Wi-Fi RSSI vectors at locations which are not explored in sparse site survey. We evaluate the signal map construction quality based on the Wi-Fi RSSI prediction error (or the mean error at all RPs), which is given by the absolute difference between ground-truth RSSIs and the predicted (virtual) or regressed ones.
2) *Query data* (online positioning evaluation): We collect Wi-Fi RSSI vectors from random locations after each signal map construction. We use them as location queries and locate them with the constructed signal map. Totally 100 RSSI vectors are collected as query data.

To further validate the Wi-Fi signal map construction quality and the location labeling accuracy, we leverage traditional localization and signal prediction schemes as comparison:

- *WKNN* [14]: a weighted k nearest-neighbor localization algorithm based on traditional signal collection. Signal vectors are compared based on Euclidean distance.
- *EZLoc* [9]: which uses the Wi-Fi RSSIs at known and unknown locations to linearly regress the path-loss model and form the signal database [4]. The signal propagation from APs are leveraged as constraints over target location. Then EZLoc locates the target position by minimizing the distance errors from all detected APs.
- *Matrix Completion* (MC) [15]: which leverages the matrix completion algorithm to recover the missing signal measurement of signal map in the site [15].
- *Linear Signal Regression* (LDPL) [4], [16], [17]: which uniformly predicts RSSIs of all propagation directions and environments through least square linear regression of the log-distance path loss (LDPL) model.

During location labeling, we also compare Surecose with the *minimum mean squared error* (MMSE) algorithm [18], which minimizes distance errors from cooperative sensors to estimate locations. Given the constructed signal map by Surecose, WKNN is also used to validate the signal map quality.

We have also conducted studies in the premium shopping mall (HKOC, $25,000$ m^2) and the international airport (HKIA, $10,000$m^2). In the mall, the initial sparse site survey is conducted on 27 RPs in 8 m grid, followed by signal map construction over 94 RPs in 4 m grid with Surecose. In the airport, 74 RPs in 10 m grid are initially collected, and then signal map construction is conducted over 278 RPs on 5 m grid. In these two sites, we also collect Wi-Fi RSSIs in 4 m and 5 m density respectively (upon the constructed RP locations) as comparison data to evaluate the virtual signal map prediction quality. We can detect around 24 APs on average at each sparse RP on campus, while 48 APs in the mall and 50 APs in the airport (as shown in Figure 9). Most of detected APs at each RP are in fact weak. We also show the linear approximation of RSS-proximity relationship between the mobile IoT devices from iBeacons in Figure 10. Given extensive empirical studies, we set by default $[\alpha_{min}, \alpha_{max}] = [-3.5, -2.5]$, $[\beta_{min}, \beta_{max}] = [-60, -55]$ and $\gamma = 0.94$ in Equations (4) and (6) .

B. Illustrative Experimental Results

Figure 11 shows the CDF of location labeling error (query data, campus) with different algorithms given only sparse signal map. As Surecose jointly optimizes the estimations fusing the reference signals and cooperative sensor distances, it can achieve much higher accuracy than WKNN, EZLoc and sensor-based MMSE. Surecose adapts the device parameters to those shown in Figure 10. WKNN relies on the nearest neighbor matching, which is often prone to signal noise. EZLoc and MMSE only consider the proximity from APs

Fig. 8: Illustration of the beacon and Wi-Fi sniffer device settings.

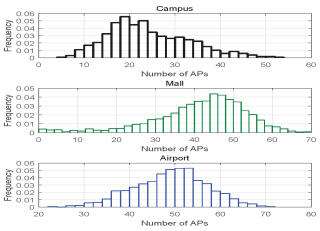

Fig. 9: Illustration of the detected AP histogram in three experimental sites.

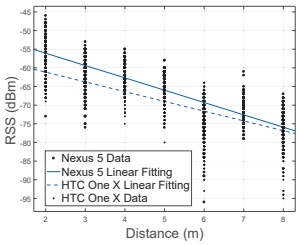

Fig. 10: Linear approximation of the RSS-proximity relationship for the IoT devices.

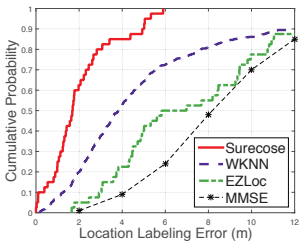

Fig. 11: CDF comparison of location labeling errors (m) (query data, campus).

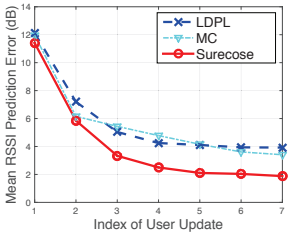

Fig. 12: Mean Wi-Fi RSSI prediction error (dB) vs. update index (comparison data, campus).

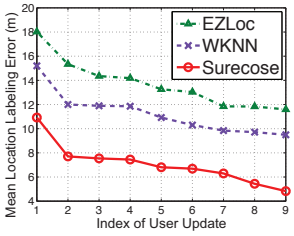

Fig. 13: Mean location labeling errors (m) vs. user update index (query data, campus).

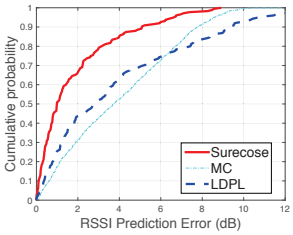

Fig. 14: CDF of Wi-Fi RSSI prediction errors after convergence (comparison data, campus).

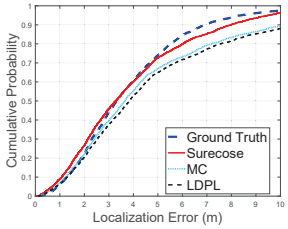

Fig. 15: CDF of WKNN localization errors with different databases (query data, campus).

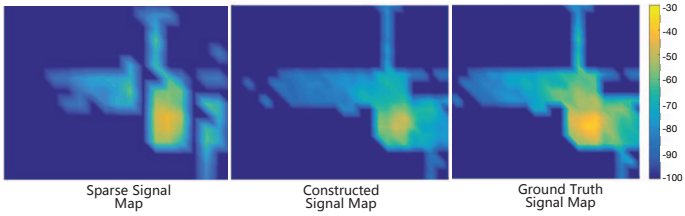

Fig. 16: Visualization of different signal maps (RSSI in dBm) from a Wi-Fi AP in our university campus (AP MAC: 00:08:30:70:18:90).

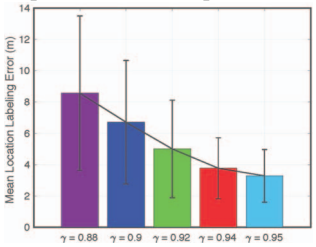

Fig. 17: Mean location labeling errors in meters (w/ std) vs. γ.

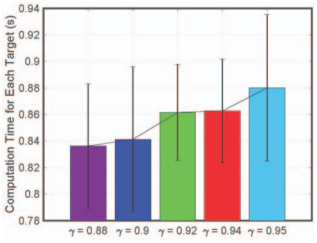

Fig. 18: Mean location labeling time in seconds (w/ std) vs. γ.

or beacons in localization. By fusing mutual distances, signal map matching and device adaptation, we mitigate the noise and reduce location labeling errors.

Given that the labeled user locations may be randomly distributed, Surecose further scales the data into a complete signal map. Figure 12 shows the mean Wi-Fi RSSI prediction error (dB) against temporal updates (comparison data). As more signals are uploaded, RSSI prediction error gradually decreases as Surecose learns the signal spatial distribution. After a few rounds of updates (say, 5), Surecose already captures the RSSIs and the prediction error converges. As Surecose considers regional and directional heterogeneity of signal propagation, it achieves much lower prediction errors (more than 20% reduction) than LDPL and MC. With the predicted RSSIs, the signal map is scaled from the crowdsourced one, and 70% offline survey cost (*from* 126 *RPs to* 30 *RPs*) is reduced compared with traditional signal collection.

Given RSSI update in the signal map, we show in Figure 13 the improvement on online location labeling, i.e., the mean labeling errors (query data) versus index of temporal user updates. It shows the decrease of errors in all the algorithms with more incoming signals, as the signals crowdsourced from the users gradually accumulate and a more complete signal map is constructed. As more Wi-Fi RSSIs are crowdsourced by the LBS users, Surecose gradually improves the signal map quality and locates the following users with better accuracy.

Figure 14 shows the CDF of Wi-Fi RSSI prediction errors (dB) with different schemes after 5 signal updates (RSSI error convergence point) in Figure 12. We also conduct linear regression and matrix completion with uniform consideration at all propagation directions as benchmark. Surecose outperforms the traditional LDPL and MC, as Surecose considers more adaptively the signals w.r.t. the complex environments. MC requires inherently redundant AP RSSI information for prediction, which is not always satisfied in practical sparse crowdsourcing. Given the constructed databases, we evaluate its benefit for existing wireless applications. We further show the performance of WKNN localization in Figure 15. We can see that in locating the query data the database constructed by Surecose outperforms the others using LDPL and MC.

Figure 16 visualizes the Wi-Fi signal heat map of an AP (MAC: 00:08:30:70:18:90) before and after the signal map construction by Surecose. Compared with the sparse one (left), the signal map constructed by Surecose (middle) resembles the ground truth (right), which also matches the results in Figures 12 and 14. These figures have visualized the adaptive signal map construction through Surecose crowdsourcing.

Figure 17 shows the mean location labeling error (m) with standard deviation (std) versus γ in Surecose (campus query data; sparse signal map in 10 m density). As γ increases, the

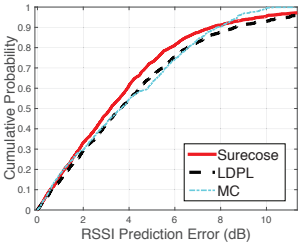

Fig. 19: CDF of Wi-Fi RSSI prediction errors (shopping mall, comparison data).

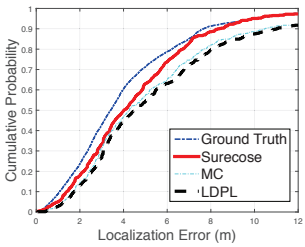

Fig. 20: CDF of WKNN errors with databases formed by different schemes (mall, query data).

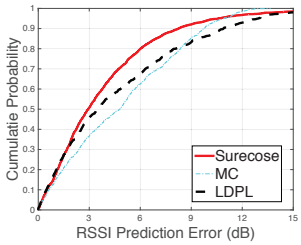

Fig. 21: CDF of Wi-Fi RSSI prediction errors (airport, comparison data).

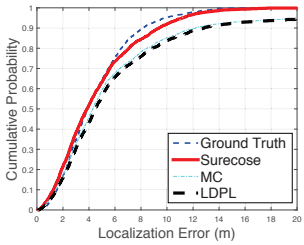

Fig. 22: CDF of WKNN errors with databases formed by different schemes (airport, query data).

accuracy grows, as higher constraint filters wrong RP mappings in localization. We can also observe that the accuracy with $\gamma = 0.95$ is close to that of $\gamma = 0.94$ (red), showing the diminishing improvement in performance. Figure 18 shows the corresponding mean location labeling time (s) with standard deviation versus γ (query data). We observe that as γ increases the computation time grows mainly because a higher threshold leads to more search over RPs in locating the target, which is of higher accuracy but longer computation time. To balance efficiency and accuracy, we set $\gamma = 0.94$ by default.

We have also conducted similar studies in the shopping mall and the airport. As the location labeling results are qualitatively similar, we do not repeat them for brevity. We focus on the signal map evaluation at these two sites.

Figure 19 shows the RSSI prediction error (dB) in the constructed signal map of the shopping mall. Surecose accurately predicts Wi-Fi RSSIs using comparison data. Given databases generated by different schemes, Figure 20 shows the CDF of WKNN localization errors (query data). Compared with the database constructed by LDPL and MC, the one generated by Surecose is much closer to that using ground-truth (i.e., combining sparse signal map and comparison data). It is mainly because Surecose captures not only the general trends of radio propagation, but also the local signal patterns (including NLOS) in complex indoor partitions. Hence it constructs more accurate signal maps (by around 20% reduction in RSSI errors) and lower error in WKNN is observed given the database generated by Surecose.

Similar to the mall, we validate its signal prediction accuracy and localization error in the airport in Figures 21 and 22. Note that the signal noise in the airport is much larger than that on the campus and the shopping mall, as the airport is larger with more pedestrians nearby. With adaptive GP, Surecose outperforms the LDPL and MC in signal prediction accuracy. Given the high accuracy in RSSI prediction, Surecose successfully reduces the offline site survey by more than 70%, i.e., *from* 94 *RPs to* 27 *RPs* in the shopping mall, and *from* 274 *RPs to only* 74 in the airport. Such reduction is important for large-scale wireless applications in these spacious sites.

We summarize the GP regression results and signal properties in these three sites. The airport, shopping mall and the campus are distinct in Wi-Fi RSSI propagation model parameters and signal noise. We first compare the different learned RSSI propagation model parameters (mean constant

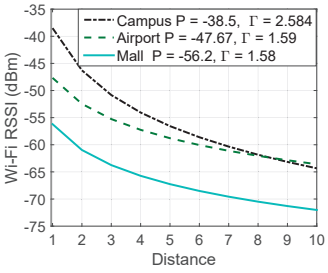

Fig. 23: Learned Wi-Fi RSSI model parameters in these sites.

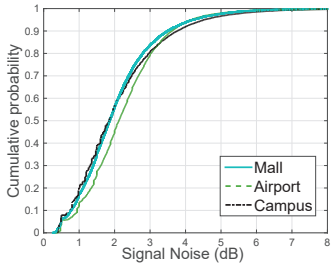

Fig. 24: CDF of Wi-Fi signal noise (dB) in these sites.

loss P^l and mean path loss component Γ^l in Equation (9) for all APs). Figure 23 shows the difference in the signal propagation in three sites. Due to different indoor wall partitions, the path loss component Γ may vary at different sites. RSSI on campus enjoys faster attenuation as there are more wall partitions in campus offices and corridors. Faster attenuation indicates more spatially distinguishable wireless features, which may benefit the fingerprint-based localization in general. Compared with the airport and the mall, our Surecose achieves larger performance improvement gap in the university campus environment. It is mainly because the complex partitioned environment on campus leads to more difficult prediction for LDPL and MC than the large open space. Due to higher Wi-Fi RSSI noise in the airport (Figure 24), larger errors in localization and RSSI prediction is expected there (Figures 21 and 22) than in the other two sites. Despite these, Surecose is overall robust against the noise and achieves high adaptivity.

V. Related Work

To construct the signal map, some recent works study using specialized infrastructures to collect signals. ARIADNE [18] investigates using specialized Wi-Fi monitors and ray-tracing to construct the signal map. [19] proposes further using RFID and environment sensors to monitor the signal map. More recent works like [20] utilize installed cameras to track the people and construct the Wi-Fi signal map. Different from above works, our Surecose is a novel joint (user) location labeling and sensor calibration framework based on existing low-cost IoT devices which have been pervasively deployed. To our best knowledge, it is the first work using implicit IoT interaction to build the signal map. Furthermore, Surecose utilizes the more advanced Gaussian process (GP) to predict the signal map. Unlike the deterministic ray-tracing, simple path loss model regression and matrix completion [15], Surecose

considers local signal distributions probabilistically and adapts to complex environment with NLoS measurements [7], [11]. With the proposed novel data clustering and preprocessing, it is more adaptive to different signal environments than traditional GP schemes. Despite the context in Wi-Fi RSSI for ease of prototyping, Surecose can be easily extended to other emerging signals, including CSI, FM and geomagnetism [20].

Exploiting user feedbacks for signal map construction has gained attention recently. OIL [21] proposes an organic fingerprinting which requires explicit and intrusive user location inputs. SLAM [8] conducts simultaneous localization and map construction, which is orthogonal to our studies here. Surecose focuses on updating sparse signal map given users' data and locations. SLAM, on the other hand, can serve as the initial indoor map inputs for Surecose. Zee [22] considers occasional access to GPS, and uses inertial motion sensors or INS to locate users. Similarly, WILL [23], MRI [16] and TransitLabel [24] study using INS to track users for signal collection. Their location labeling accuracy largely relies on extensively fine-grained inertial sensor calibration. In contrast to theirs, Surecose does not rely on intensively-calibrated INS, while its highly adaptive scheme (joint location labeling and sensor calibration) supports more pervasive deployment.

Interaction of IoT devices has recently attracted much attention. Social-Loc [13] and Montage [25] utilize the proximity between users to conduct human location estimations. Unlike above only considering specific user localization, we focus on reducing signal map survey cost with IoT device interaction. Discussions on synchronization, privacy and energy efficiency of sensors have been studied in [13], [25], [26], which are orthogonal and amendable to our studies here.

VI. Conclusion

Signal map construction is important for many wireless systems. To construct the initial signal map, previous approaches often assume explicit user feedback or intensively-calibrated sensors to label the signal locations. Furthermore, as signals are often crowdsourced at different random locations, scaling them into a complete signal map is not easy.

We propose Surecose, a novel IoT system for implicit signal map construction. Naive users unconsciously collects radio signals via cooperative sensing. Mutual sensor distances and Wi-Fi RSSIs are fed to joint location labeling and sensor calibration, which transparently locates these users without explicit intrusive inputs. Given the randomly and sparsely crowdsourced data, Gaussian process adaptively predicts RSSIs anywhere in the site. We only need a sparse initial signal map and low-cost sensors to construct the signal map, instead of dense and laborious site survey. Extensive experiments in three distinct indoor sites have shown that Surecose can adaptively construct the signal database with low survey cost (around 70% labor reduction) and little human intervention.

References

[1] C. Gao and R. Harle, "Easing the survey burden: Quantitative assessment of low-cost signal surveys for indoor positioning," in *Proc. IPIN*, Oct 2016, pp. 1–8.

[2] S. He and S.-H. G. Chan, "Wi-Fi fingerprint-based indoor positioning: Recent advances and comparisons," *IEEE Communications Surveys Tutorials*, vol. 18, no. 1, pp. 466–490, Firstquarter 2016.

[3] S. P. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[4] L. Li, G. Shen, C. Zhao, T. Moscibroda, J.-H. Lin, and F. Zhao, "Experiencing and handling the diversity in data density and environmental locality in an indoor positioning service," in *Proc. ACM MobiCom*, 2014, pp. 459–470.

[5] F. Montorsi, F. Pancaldi, and G. M. Vitetta, "Map-aware models for indoor wireless localization systems: An experimental study," *IEEE TWC*, vol. 13, no. 5, pp. 2850–2862, May 2014.

[6] S. He, J. Tan, and S.-H. G. Chan, "Towards area classification for large-scale fingerprint-based system," in *Proc. ACM UbiComp*, 2016, pp. 232–243.

[7] M. M. Atia, A. Noureldin, and M. J. Korenberg, "Dynamic online-calibrated radio maps for indoor positioning in wireless local area networks," *IEEE TMC*, vol. 12, no. 9, pp. 1774–1787, 2013.

[8] B. Ferris, D. Fox, and N. D. Lawrence, "WiFi-SLAM using Gaussian process latent variable models." in *Proc. IJCAI*, 2007, pp. 2480–2485.

[9] K. Chintalapudi, A. Padmanabha Iyer, and V. N. Padmanabhan, "Indoor localization without the pain," in *Proc. ACM MobiCom*, 2010, pp. 173–184.

[10] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315, no. 5814, pp. 972–976, 2007.

[11] S. Kumar, R. M. Hegde, and N. Trigoni, "Gaussian process regression for fingerprinting based localization," *Ad Hoc Networks*, vol. 51, pp. 1 – 10, 2016.

[12] C. E. Rasmussen, *Gaussian Processes for Machine Learning*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2006.

[13] J. Jun, Y. Gu, L. Cheng, B. Lu, J. Sun, T. Zhu, and J. Niu, "Social-Loc: Improving indoor localization with social sensing," in *Proc. ACM SenSys*, 2013, pp. 14:1–14:14.

[14] P. Bahl and V. N. Padmanabhan, "RADAR: An in-building RF-based user location and tracking system," in *Proc. IEEE INFOCOM*, vol. 2, 2000, pp. 775–784.

[15] S. Nikitaki, G. Tsagkatakis, and P. Tsakalides, "Efficient multi-channel signal strength based localization via matrix completion and Bayesian sparse learning," *IEEE TMC*, vol. 14, no. 11, pp. 2244–2256, Nov 2015.

[16] H. Shin, Y. Chon, Y. Kim, and H. Cha, "MRI: Model-based radio interpolation for indoor war-walking," *IEEE TMC*, vol. 14, no. 6, pp. 1231–1244, June 2015.

[17] F. Lemic, V. Handziski, G. Caso, P. Crombez, L. De Nardis, A. Wolisz, T. Van Haute, and E. De Poorter, "Toward extrapolation of WiFi fingerprinting performance across environments," in *Proc. ACM HotMobile*, 2016, pp. 69–74.

[18] Y. Ji, S. Biaz, S. Pandey, and P. Agrawal, "ARIADNE: A dynamic indoor signal map construction and localization system," in *Proc. ACM MobiSys*, 2006, pp. 151–164.

[19] Y.-C. Chen, J.-R. Chiang, H.-h. Chu, P. Huang, and A. W. Tsui, "Sensor-assisted Wi-Fi indoor location system for adapting to environmental dynamics," in *Proc. ACM MSWiM*, 2005, pp. 118–125.

[20] S. Papaioannou, H. Wen, Z. Xiao, A. Markham, and N. Trigoni, "Accurate positioning via cross-modality training," in *Proc. ACM SenSys*, 2015, pp. 239–251.

[21] J.-G. Park, B. Charrow, D. Curtis, J. Battat, E. Minkov, J. Hicks, S. Teller, and J. Ledlie, "Growing an organic indoor location system," in *Proc. ACM MobiSys*, Jun. 2010, pp. 271–284.

[22] A. Rai, K. K. Chintalapudi, V. N. Padmanabhan, and R. Sen, "Zee: Zero-effort crowdsourcing for indoor localization," in *Proc. ACM MobiCom*, 2012, pp. 293–304.

[23] C. Wu, Z. Yang, Y. Liu, and W. Xi, "WILL: Wireless indoor localization without site survey," *IEEE TPDS*, vol. 24, no. 4, pp. 839–848, 2013.

[24] M. Elhamshary, M. Youssef, A. Uchiyama, H. Yamaguchi, and T. Higashino, "TransitLabel: A crowd-sensing system for automatic labeling of transit stations semantics," in *Proc. ACM MobiSys*, 2016, pp. 193–206.

[25] L. Zhang, K. Liu, Y. Jiang, X. Y. Li, Y. Liu, and P. Yang, "Montage: Combine frames with movement continuity for realtime multi-user tracking," in *Proc. IEEE INFOCOM*, April 2014, pp. 799–807.

[26] H. Liu, J. Yang, S. Sidhom, Y. Wang, Y. Chen, and F. Ye, "Accurate WiFi based localization for smartphones using peer assistance," *IEEE TMC*, vol. 13, no. 10, Oct 2014.