Steering Crowdsourced Signal Map Construction via Bayesian Compressive Sensing

Suining He and Kang G. Shin
Department of Electrical Engineering and Computer Science,
The University of Michigan, Ann Arbor, MI 48109-2121
Email: {suiningh,kgshin}@umich.edu

***Abstract*—Mobile crowdsensing with increasing pervasiveness of smartphones has enabled a myriad of applications, including urban-scale signal map monitoring and revision. Despite the importance of its quality, due to the large size of a site to cover, dense crowdsourcing is neither cost-effective nor convenient for crowdsourcing participants, making it critical and challenging to balance between signal quality and crowdsourcing cost.**

To address this problem, we propose a novel incentive mechanism, BCCS, based on Bayesian Compressive Crowdsensing (BCS). BCCS iteratively determines the spatial grids for crowdsourcing quality and predicts the remaining unexplored grids for deployment efficiency. BCS returns not only the predicted signal values, but also the confidence intervals for convergence and incentive control. A probabilistic user participation and measurement model is applied for incentive design, which is flexible for crowdsensing deployment. Our extensive evaluation based on two different data sets shows that BCCS achieves much higher prediction accuracy (often by more than 20%) with lower payments to the participants and fewer iterations (often by 30%) than existing solutions.

I. Introduction

The prevalence of smartphones, equipped with various radio and environmental sensors, has enabled various interesting mobile applications of which crowdsensing has recently become very popular, especially for urban-scale signal map monitoring and revision with the emergence of smart city [1], [2]. To enable large-scale monitoring, a commensurate incentive mechanism is essential to motivate layman users/participants for various sensing tasks [3].

Signal quality and *sensing cost* are two inherently conflicting goals when designing a signal map crowdsensing platform. An important question arises: *is it possible to minimize the sensing error while still maintaining a low level of sensing cost?* Three critical challenges need to be addressed before a crowdsourced signal map can be successfully constructed:

- *Missing value inference*: Traditional urban-scale or large-area crowdsensing is often costly, especially for metropolitan areas or large shopping malls. On the other hand, the distribution of crowdsourcing participants over spatial and temporal spaces is uneven, leading to sparse coverage. In order to reduce the total data-collection cost, one may need to conduct sparse crowdsourcing and also infer the missing/unexplored signal values.
- *Crowdsourcing quality estimation*: Quality of the crowdsourced signals is critical for the platform. Thus, besides the cost concern, one should estimate and maintain data quality. Furthermore, such quality estimation also determines the cycles of crowdsourcing, preventing over- and under-sampling.
- *Incentive map determination*: In the platform design, one may need to decide on the spatial distribution of monetary rewards to incentivize the participation in crowdsourcing for better signal coverage and quality.

We design several approaches to *jointly* address the above challenges. In order to reduce the sensing area (cost) needed for task allocation, we propose using Bayesian compressive sensing (BCS) [4] to estimate the unexplored/missing values. Unlike previous compressive sensing approaches, BCS not only predicts the signals, but also provides a confidence interval for quality estimation. Thus, BCS provides flexibility in signal inference and incentive design. Based on the BCS framework, we can also estimate the crowdsensing quality, and determine the incentive distribution map at a lower cost.

Specifically, we propose a signal map crowdsensing framework called BCCS (*B*ayesian *C*ompressive *C*rowd*S*ensing), which iteratively determines signal map, task and incentive distribution. This paper makes three main contributions:

- *Joint Bayesian Compressive Crowdsensing Framework*: We formulate the signal map sensing mechanism into a BCS framework. We leverage the inherent correlation between different measurement points in terms of *spatial*, *signal* and *temporal* dimensions, hence accurately recovering the signals and significantly reducing the data-collection cost.
- *Probabilistic Incentive Design*: In order to address the unknown relationship between monetary reward and crowdsensing process, we characterize the user participation probabilistically without explicit user cost information. This way, BCCS adapts to the participation dynamics with better flexibility.
- *Comprehensive Evaluation*: We conduct extensive evaluation based on urban mesh and indoor WLAN datasets. The results have further shown the effectiveness and applicability of BCCS. Compared to the state-of-the-art algorithms, BCCS significantly reduces the payment and iteration costs (often reducing more than 30%), while with higher missing value estimation accuracy (reducing at least 20% of errors).

Although for the concreteness of evaluation we only utilize RF signals in our data analysis, BCCS is an incentive framework general enough to be applied in various emerging crowdsourced signal map construction scenarios [2], [5], [6].

The remainder of this paper is organized as follows. After

The work reported in this article was supported in part by the National Science Foundation under Grant CNS-1317411.

978-1-5386-4128-6/18/$31.00 ©2018 IEEE

discussing the related work in Section II, we present the system model and preliminaries in Section III. Based on the model, we discuss the signal map crowdsensing framework in Section IV, followed by the probabilistic incentive design in Section V. Then, we present the performance evaluation in Section VI, and finally conclude the paper in Section VII.

II. Related Work

Mobile crowdsensing has been attracting considerable attention in recent years [7], [8]. Pioneered by the authors of [3], [9], smartphone-based crowdsourcing has been studied extensively due mainly to pervasive proliferation of smartphones [7]. Based on the well-studied game theory framework, different auction-based incentive designs have been proposed [10]. By considering diverse properties of bidding, Tang *et al.* proposed a multi-dimensional auction mechanism [11] for crowdsourced mobile video streaming. In order to stimulate both requesters and workers, Jin *et al.* proposed an incentive design based on double auction [12]. These studies of auction usually considered that the thus-determined payments guarantee or ensure users to participate in crowdsourcing. Different from their auction process, we consider more opportunistic and dynamic crowdsensing workers in our formulation, which is important for platform deployment [13].

Signal map construction has also been attracting much attention due to its importance in site spectrum monitoring [1], [14], [15], location-based service (LBS) [13], [16], [17] and network construction [2]. For LBS deployment [18], gamification and online learning were proposed in [13] to motivate the users in crowdsourcing Wi-Fi signals. However, their studies did not consider inference of the missing signals to reduce the overall sensing accuracy and cost. Besides, the game design in their mobility may be oversimplified. Our framework jointly considers the crowdsourcing payment, coverage and signal quality, thus achieving better crowdsensing performance. The incentive design in BCCS is also amendable to those emerging solutions in RF signal map construction to further improve their crowdsourcing quality and deployability [2], [19].

Some more recent studies consider compressive sensing [20] for the allocation of crowdsourcing tasks [21]. Awareness of cost and density has also been incorporated in [22] and [23], respectively. However, they did not consider the corresponding incentive design to improve performance [3], [15]. Compressive sensing was proposed in [24] for air quality monitoring. However, it did not consider the correlations among sample points to refine the quality. In contrast, we leverage the inherent correlations and hence achieve better accuracy.

Furthermore, compared to these traditional compressive sensing techniques [21]–[23], [25], [26] which provide only a single point estimate, our BCCS framework augments it with a full posterior probability density, showing the confidence level. We further utilize these error estimates to improve crowdsensing and steer the sensing cycles. BCCS is also experimentally compared with state-of-the-art algorithms [21], [27]–[29] in missing value inference, and validated with its accuracy and deployment efficiency.

III. System Models & Preliminaries

Table I lists the symbols used in the formulation of BCCS. Presented below is the primer of system model and formulation of BCCS. We first discuss the model of crowdsensing tasks, signal and incentive maps in Section III-A. and then the preliminaries of Bayesian compressive sensing in Section III-B.

TABLE I: Major symbols in BCCS formulation.

Notations	Definitions
M	Number of sparsely crowdsourced samples
N	Number of grids to be crowdsourced
t	Time stamp of the t-th cycle for crowdsensing
ω_j	Selection indicator variable of grid j
$\boldsymbol{s}^t$	Entire signal map in the site at time t
$\Delta\boldsymbol{s}^t$	Set of signal-map differences at time t
z_i	Crowdsourced signals at the i-th location
s_j	Signal measurement at the j-th location/grid
$\boldsymbol{\Phi}$	Projection matrix between sparse and entire maps
(l_i^x, l_i^y)	2-D (x, y)-coordinates of location i
$c^d(i,j)$	Spatial correlation between locations i and j
$c^s(i,j)$	Signal-space correlation between locations i and j
$[\boldsymbol{\pi}, \eta]$	Hyperparameters for BCS learning
c_u	Crowdsensing cost of user u
$\boldsymbol{\nu}$	Ratio of signal differences $\Delta\boldsymbol{s}$ vs. the rewards $\mathbf{R}$
κ	Confidence level for error estimation
$\boldsymbol{\tau}_t$	Selected grid for incentive update at time t

A. Signal Crowdsensing & Incentive Map

We consider the spatial crowdsourcing, given the incentives and tasks distributed, as well as a crowd of rational smartphone users. The crowdsourcing tasks are distributed over a spatial map. Suppose at each sensing cycle, a total of M crowdsourced signals are uploaded, forming an M-dimensional vector $\boldsymbol{z} \triangleq [z_1, \ldots, z_M]^T$. Also, each z_i $(1 \leq i \leq M)$ is tagged with a location coordinate $\boldsymbol{l}_i = (l_i^x, l_i^y)$.

As the site usually covers a large area, we discretize it into finite (say, N) *grids* of an equal size. We consider the measurements inside each grid as uniformly distributed while varying over different grids. Note that the actual grid size can be customized based on specific applications and signal quality requirement. For example, one may set a mile-level grid size for city-scale air quality monitoring while a smaller size for street noise monitoring. In indoor RF spectrum monitoring, meter-level grids can be sufficient. We assume that the grid granularity is given beforehand.

Given discretization into N sensing grids, each grid j $(1 \leq j \leq N)$ of 2-D coordinate $\boldsymbol{l}_j = \left(l_j^x, l_j^y\right)$ is tagged with a certain signal value s_j. The entire *signal map* is modeled as $\boldsymbol{s} \triangleq [s_1, \ldots, s_N]^T$, and the estimated one as $\widehat{\boldsymbol{s}}$. Given estimation $\widehat{\boldsymbol{s}}$ and ground-truth $\boldsymbol{s}$, we define an error function $\Xi \triangleq \mathcal{E}(\widehat{\boldsymbol{s}}, \boldsymbol{s})$ between the predicted and the actual signal map in order to estimate the prediction quality. As ground-truth $\boldsymbol{s}$ may not always be available, one may need to consider use of subset samples $\boldsymbol{s}'$ as a replacement and for cross validation [26].

During the crowdsensing in BCCS, we consider a sufficient number of crowdsourcing workers within the site, and a worker can easily finish an allocated task without leaving a grid. The incentive distribution for the users is characterized by an *incentive map*.

Definition 1: Incentive map: $\mathbf{R} \triangleq [r_1, \ldots, r_N]$ which characterizes the spatial distribution of monetary rewards corresponding to N grids in $\boldsymbol{s}$. Each grid j $(1 \leq j \leq N)$ of 2-D coordinate $\boldsymbol{l}_j = \left(l_j^x, l_j^y\right)$ is associated with a certain amount of

monetary reward r_j. It describes the corresponding payment for the signal-collection task.

B. Primer of Bayesian Compressive Sensing (BCS)

Compressive sensing (CS) is a novel signal reconstruction method which has been widely used in image processing, networking, traffic monitoring and sensor networks [30]–[33]. For ease of applying CS to crowdsensing, we use the concept of projection matrix.

Definition 2: Projection matrix: For a spatial crowdsensing task with M sparse measurements and N grids to be reconstructed ($M \ll N$), the sensing projection matrix of correlations, or *projection matrix*, is denoted as $\mathbf{\Phi} \in \mathbb{R}^{M\times N}$. The entry at the i-th row and the j-th column of $\mathbf{\Phi}$, denoted as $\psi(i,j)$, represents the inherent correlation between the i-th crowdsourced signal with the one in grid j of the entire map.

The objective of traditional CS problem formulation is to find a sparse solution $\boldsymbol{s} \in \mathbb{R}^N$ in the under-determined linear equation $\boldsymbol{z} = \mathbf{\Phi}\boldsymbol{s}$, given the measurement vector $\boldsymbol{z} \in \mathbb{R}^M$ and $M \times N$ projection matrix $\mathbf{\Phi}$ ($M \ll N$). Specifically, the CS framework can be formulated as

$$\arg\min_{\boldsymbol{s}\in\mathbb{R}^N} \|\boldsymbol{s}\|_0, \quad \text{s.t. } (\boldsymbol{z})_{M\times 1} = (\mathbf{\Phi})_{M\times N}(\boldsymbol{s})_{N\times 1}. \tag{1}$$

Directly solving the above l_0 optimization problem in Eq. (1) is NP-complete [4]. Alternatively, the CS theory reveals that a computationally tractable optimization problem based on l_1 norm yields an equivalent solution [4]. Considering the potential signal noise and Gaussian likelihood [15], we further model the sensing process as

$$\boldsymbol{z} = \mathbf{\Phi}\boldsymbol{s}+\boldsymbol{e}, \quad \mathcal{P}(\boldsymbol{z}|\boldsymbol{s},\sigma^2) = \frac{1}{(2\pi\sigma^2)^{\frac{M}{2}}}\exp\left(-\frac{\|\boldsymbol{z}-\mathbf{\Phi}\boldsymbol{s}\|^2}{2\sigma^2}\right), \tag{2}$$

where σ represents the inherent unknown noise variance of $\boldsymbol{e}$. Therefore, Bayesian compressive sensing (BCS) can be formulated. For each estimation of $\boldsymbol{s}$, we infer not only the value but also the confidence level (in terms of variance). Based on the above, we define the *entire signal-map construction algorithm* for the problem formulation.

Definition 3: Entire signal-map reconstruction algorithm: A reconstruction algorithm $\mathcal{F}(\boldsymbol{z},\mathbf{\Phi})$ returns the entire signal map estimate $\widehat{\boldsymbol{s}}$ which is close to original $\boldsymbol{s}$, given the sparsely crowdsourced signals $\boldsymbol{z}$ and the projection matrix $\mathbf{\Phi}$, i.e.,

$$\mathcal{F}(\boldsymbol{z},\mathbf{\Phi}) = \widehat{\boldsymbol{s}} \approxeq \boldsymbol{s}. \tag{3}$$

IV. Signal Map Crowdsensing via BCS Framework

We present the basic problem formulation and propose the signal-map crowdsourcing based on Bayesian Compressive Sensing (BCS). We first present the crowdsensing problem in Section IV-A. Then, we discuss how to infer the missing values via BCS in Section IV-B, followed by a solution to the problem in Section IV-C.

A. Basic Problem Formulation

The basic crowdsourcing problem in BCCS can be formulated as a fine-grained grid selection problem. Let ω_j be the indicator variable ($\omega_j \in \{0,1\}$) at grid j, i.e., $\omega_j = 0$ if grid j is not selected, and $\omega_j = 1$ otherwise. Also, let r_j be the crowdsourcing cost (payment) at grid j, while

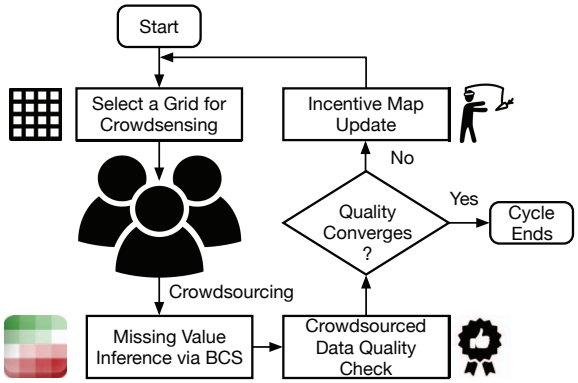


Fig. 1: The basic system workflow illustration of BCCS.

$\Xi = [\xi_1, \ldots, \xi_N]$ and Γ be the signal map error estimate (*i.e.,* at each grid j we infer the error as ξ_j) and the corresponding required error constraint, respectively.

In order to balance between signal map quality and budgetary concerns, we dynamically choose the grids to minimize the signal difference while meeting the budgetary constraints. In other words, we steer workers towards those grids with the minimum total crowdsensing cost (ω_j in the budget objective), subject to sensing recovery results $\widehat{\boldsymbol{s}}$ (*i.e.,* missing value inference) and the estimated signal difference requirement Γ (*i.e.,* quality constraint), *i.e.,*

$$\begin{aligned} \arg\min_{\{\omega_j\}} \quad & \sum_{j=1}^{N} r_j\omega_j, \\ \text{s.t.} \quad & \widehat{\boldsymbol{s}} = \mathcal{F}(\boldsymbol{z},\mathbf{\Phi}), \quad \omega_j \in \{0,1\}, \quad \forall j \in \{1,\ldots,N\}, \\ & \Xi = \mathcal{E}(\widehat{\boldsymbol{s}},\boldsymbol{s}'), \quad \sum_{j=1}^{N}\xi_j\omega_j \le \Gamma. \end{aligned} \tag{4}$$

In fact, the problem in Eq. (4) is non-trivial to solve for the following reasons. 1) *Unknown sensing cost for each user*: The crowdsensing cost incurred by each participant may not always be available in practice, making a decision on r_j difficult. 2) *Confidence in terms of estimation*: In order to stop the crowdsensing cycles properly, one may need to estimate the confidence level of the constructed signal map.

To perform the above optimization and address these two issues, we design an iterative and interactive scheme based on BCS, as illustrated in Fig. 1. The basic idea consists of three steps as follows.

1) *Incentive map publication & crowdsensing*: We first publish the spatial incentive map for crowdsourcing users. Given the monetary rewards, users collect the signal measurements and upload them to the platform.
2) *Missing value inference*: Given the sparsely-crowdsourced signals, the platform infers the missing values based on Bayesian compressive sensing. This way we reduce the number of grids to be crowdsourced.
3) *Confidence estimation & stopping*: Via the confidence measurements, the crowdsensing platform estimates the level of quality satisfaction and determines ongoing crowdsourcing towards 1), or a stop.

B. Missing Value Inference & Signal Reconstruction

Signal map crowdsourcing is generally sparse and dynamic in nature. Therefore, we leverage BCCS to address the sparse-

ness, while BCCS dynamically constructs the entire signal map based on the signal revision.

Missing value inference can be formulated as a sparse BCS problem. The crowdsourced signal map is generally and inherently sparse. Due to the latent spatial-temporal correlations inside the map, each observation can be considered as the linear weighted average of many other sparsely distributed samples. Formally, the main problem of missing value inference is *how to decompose the known signals* ($\boldsymbol{z}$) to *derive those of other grids which are not sampled* ($\boldsymbol{s}$) based on *certain projection of correlations* ($\boldsymbol{\Phi}$), *i.e.,*

$$\boldsymbol{z} = \boldsymbol{\Phi}\boldsymbol{s}. \tag{5}$$

According to some recent studies, $\boldsymbol{s}$'s are generally sparse for many important signal map inference problems (including RF [18] and air quality signals [26]). Further discussions on other signals' sparsity will be part of our future work. Via the BCS formulation we then recover those missing values (probabilistically) in the signal map.

Specifically, we model the missing value inference function $\mathcal{F}(\boldsymbol{z}, \boldsymbol{\Phi})$ as a dynamic signal inference problem with respect to time t (the corresponding crowdsourced sample and signal map are thus attached with a time stamp). Given the sparsely-measured signal and the projection matrix, we recover the entire signal map accompanied with a certain confidence interval. We then find the difference of signal maps at time t and $t-1$ to monitor the crowdsourcing process.

The projection matrix characterizes the inherent relationships between the sparsely-sampled signals, z_i's, and those in the entire signal map, s_j's. We consider a general framework design for signal map crowdsourcing, including the projection matrix customization. Given the spatial complexity and temporal dynamics of signal map, different designs and metrics can be embedded into our BCCS framework in a hybrid way.

We present three important correlations considered within our projection matrix $\boldsymbol{\Phi}$:

- *Temporal correlation*: At each sensing cycle or time t, BCCS compares crowdsourced $\boldsymbol{z}^t$ against the previously available signal map $\boldsymbol{s}^{t-1}$ for each correlation element in $\boldsymbol{\Phi}$. This way, we embed within $\boldsymbol{\Phi}$ the *temporal correlation* between two consecutive sensing cycles, which is essential for large-scale dynamic signal map construction [21], especially for RF, air pollution and noise map.
- *Spatial correlation*: The closer two locations (of z_i and s_j), the more correlated two samples are. We characterize the spatial correlation using the Euclidean distance, *i.e.,* $\|\boldsymbol{l}_i - \boldsymbol{l}_j\|_2 = \sqrt{(l_i^x - l_j^x)^2 + (l_i^y - l_j^y)^2}$, and then have
$$c^{\mathrm{d}}(i,j) \triangleq \exp\left(-\frac{\|\boldsymbol{l}_i - \boldsymbol{l}_j\|_2}{\sigma_d^2}\right), \tag{6}$$
between the geographic locations in the map. We normalize all $c^d(i,j)$'s into the range of $[0,1]$.
Note that other spatial geographic metrics, including Manhattan distance ($\|\boldsymbol{l}_i - \boldsymbol{l}_j\|_1 = |l_i^x - l_j^x| + |l_i^y - l_j^y|$) for a metropolitan area, can be easily applied within framework of BCCS. Then, we may have
$$c^{\mathrm{d}}(i,j) \triangleq \exp\left(-\frac{\|\boldsymbol{l}_i - \boldsymbol{l}_j\|_1}{\sigma_d^2}\right), \tag{7}$$
- *Signal correlation*: The more similar two measurements in the signal space, the more correlated two samples. We can either characterize the spatial correlation based on kernel-based [5] signal value difference of z_i and s_j, given by
$$c^{\mathrm{s}}(i,j) \triangleq \exp\left(-\frac{\|z_i - s_j\|^2}{\sigma_s^2}\right), \tag{8}$$
or the cosine similarity between two high-dimensional signal vectors (say, of H dimensions), *i.e.,*
$$c^{\mathrm{s}}(i,j) \triangleq \frac{\vec{\boldsymbol{z}}_i \cdot \vec{\boldsymbol{s}}_j}{\|\vec{\boldsymbol{z}}_i\| \cdot \|\vec{\boldsymbol{s}}_j\|}, \quad \vec{\boldsymbol{z}}_i \in \mathbb{R}^H, \quad \vec{\boldsymbol{s}}_j \in \mathbb{R}^H, \tag{9}$$
depending on the input form of collected data. Similar to $c^d(i,j)$, all $c^s(i,j)$'s are later normalized. For deployment, we suggest use of Eq. (8) for signal maps with 1-D value measurements, including air quality and noise level. If the signal map consists of multi-dimensional measurements (for example, each grid j may receive an RF signal vector $\vec{\boldsymbol{s}}_j$ from H transceivers), we suggest use of Eq. (9) for more comprehensive correlation characterization.

In order to jointly consider the latter two correlation metrics, we further calculate each of the items inside our projection matrix $\boldsymbol{\Phi}$, denoted as $\psi(i,j)$, by their weighted average, *i.e.,*
$$\psi(i,j) \triangleq \varphi \cdot c^d(i,j) + (1-\varphi) \cdot c^s(i,j), \tag{10}$$
where φ ($0 \le \varphi \le 1$) represents a weighted trade-off between the aforementioned spatial and signal correlations. In summary, *jointly* considering the inherent *temporal* (t), *spatial* ($c^{\mathrm{d}}(i,j)$) and *signal* ($c^s(i,j)$) correlations makes BCCS more robust to the environmental dynamics in the signal map.

Note that our scheme provides a generic framework for crowdsensing and missing value inference. Therefore, other metrics characterizing correlations [34] can be easily applied inside matrix $\boldsymbol{\Phi}$. We also experimentally evaluate the correlation setting of φ in Section VI. After calculation of Eq. (10), we normalize $\psi(i,j)$ w.r.t. each row i in $\boldsymbol{\Phi}$, *i.e.,* $\sum_{j=1}^{N} \psi(i,j) = 1, \forall i \in \{1,\ldots,M\}$.

Based on the BCS framework, at sensing cycle or time t, the linear relationship between the i-th sparsely crowdsourced sample, z_i^t, and those inside the entire signal map, $\boldsymbol{s}^t$, is considered as
$$z_i^t = \boldsymbol{\psi}_i \boldsymbol{s}^t + \epsilon_i, \quad i \in \{1,\ldots,M\} \tag{11}$$
where $\boldsymbol{\psi}_i$ is the i-th row (vector) of $(\boldsymbol{\Phi})_{M\times N}$. Let $\boldsymbol{e} = [\epsilon_1,\ldots,\epsilon_M]^T$ be a 1-D vector consisting of M noise elements ϵ_i's. Then, the entire relationship is given by
$$\left(\boldsymbol{z}^t\right)_{M\times 1} = (\boldsymbol{\Phi})_{M\times N}\left(\boldsymbol{s}^t\right)_{N\times 1} + (\boldsymbol{e})_{M\times 1}, \tag{12}$$
where the projection matrix is
$$(\boldsymbol{\Phi})_{M\times N} = \begin{bmatrix} \boldsymbol{\psi}_1 \\ \vdots \\ \boldsymbol{\psi}_M \end{bmatrix} = \begin{bmatrix} \psi(1,1) & \ldots & \psi(1,N) \\ \vdots & \ddots & \vdots \\ \psi(M,1) & \ldots & \psi(M,N) \end{bmatrix}. \tag{13}$$
Based on the above, taking $\boldsymbol{z}^t$ and $\boldsymbol{\Phi}$ into account, the signal map construction is to *find* $\boldsymbol{s}^t$ *such that the posterior likelihood* $\mathcal{P}(\boldsymbol{s}^t|\boldsymbol{z}^t,\sigma^2)$ *is maximized*, i.e.,
$$\arg\max_{\boldsymbol{s}^t} \quad \mathcal{P}(\boldsymbol{s}^t|\boldsymbol{z}^t,\sigma^2), \quad \text{s.t.} \quad \boldsymbol{s}^t \sim \mathcal{P}(\boldsymbol{s}^t). \tag{14}$$
Our formulation considers the measurement at grid j as
$$s_j \sim \mathcal{N}(\bar{s}_j, \sigma^2). \tag{15}$$
Note that other probabilistic distributions can be easily applied within the BCS framework [15], [35]. We also have
$$\mathcal{P}(\boldsymbol{z}^t|\boldsymbol{s}^t,\sigma^2) = \frac{1}{(2\pi\sigma^2)^{\frac{M}{2}}} \exp\left(-\frac{\|\boldsymbol{z}^t - \boldsymbol{\Phi}\boldsymbol{s}^t\|^2}{2\sigma^2}\right). \tag{16}$$
In summary, we denote the estimated mean and covariance of posterior signal measurements $\boldsymbol{s}^t$ as $\bar{\boldsymbol{S}} \in \mathbb{R}^N$ and $\boldsymbol{\Sigma} \in \mathbb{R}^{N\times N}$.

Then, we consider the posterior distribution of $\boldsymbol{s}^t$ as

$$\mathcal{P}(\boldsymbol{s}^t|\boldsymbol{z}^t,\sigma^2)\sim\mathcal{N}(\bar{\boldsymbol{S}},\boldsymbol{\Sigma}). \tag{17}$$

Given the crowdsourced signals and predicted missing values via BCS, the signal difference between two consecutive time-stamps is also calculated and monitored. A large difference implies that inference errors exist due to insufficient crowdsourcing or measurement noise, and the signal map has not yet converged. Specifically, the entire signal map difference between time-stamps t and $t-1$ is modeled linearly as

$$\boldsymbol{\Delta s}^t=|\boldsymbol{s}^t-\boldsymbol{s}^{t-1}|. \tag{18}$$

In BCCS, we first find $[\bar{\boldsymbol{S}},\boldsymbol{\Sigma}]$, and then $\boldsymbol{\Delta s}^t$ in Eq. (18) can be determined. The calculated signal map change $\boldsymbol{\Delta s}^t$ and the confidence level $\boldsymbol{\Sigma}$ are used for later incentive map update (to be discussed in Section V).

C. Hyperparameter Estimation Solving BCS

We briefly discuss how to solve the formulated BCS problem. Our basic idea is to introduce some hyperparameters for ease of transformation, jointly maximize their likelihood with input data, and finally estimate $\boldsymbol{s}$ and $\boldsymbol{e}$. In other words, we model the learning problem of BCS into a *hyperparameter estimation* form [4], [15].

Specifically, for each signal s_j at grid j to be estimated, we introduce the precision Π_j of Gaussian density (as prior knowledge), *i.e.,* the reciprocal or inverse of variance σ_j^2, and $\eta=\sigma^{-2}$, *i.e.,* the precision of the external measurement noise $\boldsymbol{e}$. We also define a vector $\boldsymbol{\pi}=[\Pi_1,\ldots,\Pi_N]$ for ease of presentation, and have

$$\mathcal{P}(\boldsymbol{s}|\boldsymbol{\pi})=\prod_{j=1}^{N}\mathcal{N}(\bar{s}_j,\Pi_j^{-1}). \tag{19}$$

Given $\boldsymbol{z}^t$, $\boldsymbol{\pi}$ and η, we denote analytical results of the posterior mean and covariance [15], [35] for $\boldsymbol{s}^t$ as $\bar{\boldsymbol{S}}$ and $\boldsymbol{\Sigma}$ (as Section IV-B), which are given as

$$\bar{\boldsymbol{S}}\triangleq\eta\boldsymbol{\Sigma}\boldsymbol{\Phi}^T\boldsymbol{z}^t, \tag{20}$$

where the covariance is given by

$$\boldsymbol{\Sigma}\triangleq(\eta\boldsymbol{\Phi}^T\boldsymbol{\Phi}+\boldsymbol{\mathcal{D}})^{-1}, \tag{21}$$

and diagonal matrix $\boldsymbol{\mathcal{D}}=diag(\Pi_1,\ldots,\Pi_N)$.

The estimation of BCS hyperparameters is usually achieved via relevance vector machine [4], which derives parsimonious solution for the model via Bayesian inference. Specifically, the hyperparameter $[\boldsymbol{\pi},\eta]$ can be estimated via the maximum likelihood of $\log p(s|\boldsymbol{\pi},\eta)$ [15], [35], which iteratively updates $\boldsymbol{\pi}$ and η to achieve overall expectation maximization.

At each iteration k, we find new values for each of hyperparameters $[\boldsymbol{\pi},\eta]$ ($\forall j\in\{1,\ldots,N\}$) by

$$\Pi_j^k\triangleq\frac{1-\Pi_j^{k-1}\boldsymbol{\Sigma}_{jj}^k}{\left(\bar{\boldsymbol{S}}_j^k\right)^2}, \tag{22}$$

and

$$\frac{1}{\eta_j^k}\triangleq\frac{\left|\left|\boldsymbol{z}^t-\boldsymbol{\Phi}\bar{\boldsymbol{S}}^k\right|\right|_2^2}{M-\sum_{j=1}^{N}(1-\Pi_j^{k-1}\boldsymbol{\Sigma}_{jj}^k)}, \tag{23}$$

where $\bar{\boldsymbol{S}}^k$ and $\boldsymbol{\Sigma}^k$ are given by Eqs. (20) and (21), respectively. In summary, we empirically initialize the values of $[\boldsymbol{\pi}^0,\eta^0]$, then calculate $[\bar{\boldsymbol{S}}^k,\boldsymbol{\Sigma}^k]$ based on Eqs. (20) and (21), and iteratively update $[\boldsymbol{\pi}^k,\eta^k]$ until their final convergence. The number of iterations can be customized according to the time allowed and expected task granularity. In practice, we observe that BCCS converges quickly after only a few iterations (usually less than 10), which is applicable for many large-scale deployments.

V. Probabilistic Incentive Design

Given the missing value inference, we design the incentive map update using a probabilistic user measurement model. We first discuss opportunistic user modeling and reward design of an incentive map in Section V-A. We then present how to properly stop crowdsensing cycles in Section V-B, followed by the overall complexity analysis in Section V-C.

A. User Modeling & Reward Design in Incentive Map

The incentive map and payment distribution consider the participants' behavior, given the monetary incentives [24]. In case of mobile crowdsensing, smartphones or other hand-held devices are resource-constrained due to limited battery, device capacity or communication bandwidth. Thus, the participation level of a user u is correlated with his/her sensing cost c_u [24].

Given an estimated signal map, we determine the incentive map in order to further trigger the crowdsourcing process. Specifically, the payment update is to determine the incentive within each grid such that the expected signal quality can be maximized. A user u's decision on participating data collection in a grid j is considered as a random indicator variable I_u. We consider $I_u=1$ if r_j is less than c_u of user u, and $I_u=0$ otherwise. Clearly, a user won't be steered if his sensing cost exceeds payment. Let $\mathcal{P}_j=\mathcal{P}(I_u=1)$ be the probability distribution that user u participates in sensing in grid j, and we further consider the user participation probabilistically.

Due to privacy concerns [26], it is usually difficult to obtain the cost information from each user [21], [24]. So, we model the user's participation probabilistically. In particular, the probability of I_u given payment r_j is modeled as

$$\mathcal{P}(I_u|r_j)=\begin{cases}1-\exp(-\rho r_j), & \text{if } I_u=1;\\ \exp(-\rho r_j), & \text{if } I_u=0,\end{cases} \tag{24}$$

where $\rho>0$ characterizes the user sensitivity towards a reward. Note that the distribution function in Eq. (24) is an increasing concave probabilistic function in range $[0,1]$, showing a diminishing or saturating return towards a given amount of reward [36]. In other words, it strictly monotonically increases with the payments at a lower rate. Suppose there are U_j users in grid j. We consider the obtained number of samples at grid j is equal to $N_j=\sum_{u=1}^{U_j}I_u$.

The probability of getting λ_j measurements at grid j is modeled by a Binomial distribution $N_j\sim\mathcal{B}(U_j,\mathcal{P}_j)$, *i.e.,*

$$\mathcal{P}(N_j=\lambda_j)=\frac{U_j!}{\lambda_j!(U_j-\lambda_j)!}\mathcal{P}_j^{\lambda_j}\left(1-\mathcal{P}_j^{\lambda_j}\right)^{U_j-\lambda_j}. \tag{25}$$

Using this distribution, we can steer the participation of users in the crowdsensing based on r_j. We consider for each grid j sufficient measurements as $\tilde{\lambda}$, and let total probability of obtaining fewer than $\tilde{\lambda}$ signals as $\mathcal{P}(N_j<\tilde{\lambda})=\sum_{\lambda_j=0}^{\tilde{\lambda}-1}\mathcal{P}(N_j=\lambda_j)$. To ensure the acquisition of minimum $\tilde{\lambda}$ measurements, the rewards distributed at each grid are steered via the following probability bound θ, *i.e.,*

$$1-\mathcal{P}(N_j<\tilde{\lambda})\geq\theta. \tag{26}$$

Based on this as well as Eqs. (24) and (25), we can directly find the reward value r_j that steers crowdsensing in grid j. Then,

Algorithm 1: BCCS: Incentive design for signal map construction based on *B*ayesian *C*ompressive *C*rowd*S*ensing.

Input: M crowdsourced samples, spatial distribution of rewards $\mathbf{R}$, previous signal map $\boldsymbol{s}^t$ at time t with N grids.
Output: Reconstructed grids $\boldsymbol{\tau}_{t+1}$ to be selected for next crowdsensing cycle.

$\boldsymbol{\tau}_{t+1} \leftarrow \boldsymbol{\tau}_t$; /* Grid selection */
for $g \leftarrow 1$ *to* G **do**
  /* Ratio of differences vs rewards */
  $\boldsymbol{\nu} \leftarrow abs(\Delta \boldsymbol{s})/\mathbf{R}$;
  $\tau_g \leftarrow \arg\max_{j \notin \boldsymbol{\tau}_{t+1}} \nu_j$; /* Maximum ratio */
  $\boldsymbol{\tau}_{t+1} \leftarrow \boldsymbol{\tau}_{t+1} \bigcup \tau_g$; /* Selected grid */
  solve the BCS problem in Equation (14);
  /* Estimate signal map difference */
  $\boldsymbol{\Delta s} \leftarrow \boldsymbol{s}^{t+1} - \boldsymbol{s}^t$;
  /* Cross validation & error estimate) */
  if *iscrossval*($\boldsymbol{s}^t$) **then**
    break; /* Satisfy Eq.(27) in Sec.V-B */
  end
end
return $\boldsymbol{\tau}_{t+1}$;

the system displays the corresponding monetary incentives $\mathbf{R}$ on the spatial or geographic map.

Finally, we dynamically update the monetary incentive map by selecting the grid $\boldsymbol{\tau}$, based on the absolute signal map difference $\Delta \boldsymbol{s}^t$ vs. the reward r_j in order to balance between these two critical factors [24]. In other words, the grid with more signal difference and less cost is critical for the platform, and is more likely to be selected. Therefore, at each time t, the grid with the maximum ratio of absolute signal difference to reward (platform cost) is selected for another crowdsensing cycle and further incentive update. This way, we can balance between signal map quality and crowdsourcing cost.

B. Crowdsourcing Cycle Stopping Criterion

Determining the stop condition of iterative cycles is critical for a crowdsensing platform [26]. An early stop may lead to insufficient data collection for a signal map of quality, while a late stop wastes resources and money. In practice, the platform has no knowledge of the ground truth estimation, and hence it is challenging to design the error function for the determination of convergence.

As mentioned earlier, BCS not only provides the estimated values, but also the corresponding estimation variance. We may further utilize this as the confidence interval to decide on the termination of the task-allocation process (*i.e.,* error estimate $\Xi = \mathcal{E}(\widehat{\boldsymbol{s}}, \boldsymbol{s})$, and requirement Γ in Eq. (4)). We integrate this with k-fold cross-validation [37] to evaluate the trained model.

Specifically, at each iteration, BCCS picks Q samples as the target data from all (M) samples for cross validation. The remaining ($M - Q$) samples are used for training. Then, BCS returns the confidence levels at different locations over the target confidence level. BCCS repeats the above process. We then compute the average and standard deviation of the prediction variances in the entire signal map, and check whether it is within a certain confidence interval. In particular, BCCS checks if the average signal difference between the Q ($Q < M$) re-sampled values and the predicted ones by BCS is within a predefined range:

$$\frac{1}{Q}\sum_{j=1}^{Q} |s_j - \widehat{s}_j| < \kappa \cdot \sigma, \tag{27}$$

where κ is a certain tunable parameter controlling the confidence level, and σ is the average standard deviation of signal map estimate obtained from the BCS framework (*i.e.,* $\boldsymbol{\Sigma}$). If Eq. (27) is satisfied, we terminate the entire signal map construction, else we continue the crowdsourcing iteration. Via the k-fold cross validation, BCCS also prevents the overfitting in terms of model estimation [26], and also estimates the confidence level to determine iteration stop.

C. Summary & Overall Complexity Analysis

We summarize BCCS in Alg. 1. Given monetary incentives, crowdsourcing users first collect signals, and then, via compressive sensing BCCS infers the missing values at unexplored locations. Based on cross validation and estimated confidence inference, BCCS then infers the potential error and signal differences. It also updates the incentive maps and triggers another cycle of crowdsensing, if needed. Note that the entire sensing can be also initialized by some random sampling, sparse sensor uploads or simple model regression [21]. The signal map is then iteratively constructed by BCCS.

We briefly analyze the computation complexity of BCCS as follows. According to the variational Bayesian algorithm [38], finding $\boldsymbol{\Delta s}_t$ for the signal map recovery takes $\mathcal{O}(M^3)$. Incentive map update takes $\mathcal{O}(kM^3)$, which is dominated by the k-fold cross-validation of signal error estimation. Note that the cross-validation can be conducted concurrently and independently, and therefore the overall computation time becomes small, enabling its practical deployment.

VI. Performance Evaluation

We now evaluate the performance of BCCS using two signal map data sets. Specifically, we first present the comparison metrics and evaluation setups in Section VI-A, and then the illustrative results based on the data sets in Section VI-B.

A. Comparison Metrics & Evaluation Setups

We utilize the following two typical crowdsourced data sets for the evaluation of BCCS.

- *Urban mesh signal map* [39]: which characterizes spatial distribution of RSSI (received signal strength indicator) in an urban environment. It can be utilized for mesh access point (AP) localization and urban location-based service [2]. For evaluation, we split the 7 km^2 urban area into grids of 100 m $\times$ 100 m.
- *Indoor Wi-Fi signal map* [40]: which characterizes the spatial distribution of indoor Wi-Fi WLAN RSSIs. It can be utilized for indoor site monitoring and localization service. For evaluation, we, by default, split the approximately $40,000$ m^2 site into grids of 3 m $\times$ 3 m.

Unless stated otherwise, we use the following parameters by default: $\theta = 0.95$ in Eq. (26). $G = \log\lfloor\sqrt{N}\rfloor$ in Alg. 1. $\sigma_d = \sigma_s = 1.25$ in Eqs. (6) and (8). $\tilde{\lambda} = 15$ in Eq. (26). We consider sensing cost linearly proportional to sample size. For urban-scale signal map construction, we utilize Eq. (8)

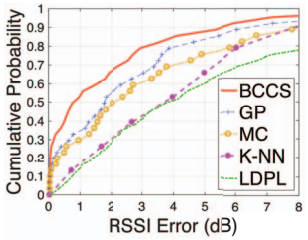


Fig. 2: CDF of signal map missing value inference error (dB, indoor).

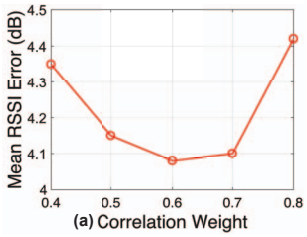
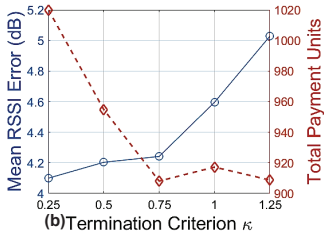


Fig. 3: (a) Mean RSSI prediction error (dB, indoor) vs. correlation weight φ; (b) RSSI error and total payment units vs. parameter κ.



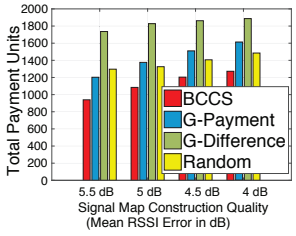


Fig. 4: Total payment evaluation for indoor signal map crowdsourcing.

to characterize signal correlation. In terms of indoor signal map construction, we calculate the correlation $c^s(i,j)$ with Eq. (9) due to its high dimensions. We simulate 18 users in the crowdsourcing. The spatial coordinates of the samples are labeled via GPS or manual inputs. We conduct the data analytics and system evaluation on a PC with 3.4GHz i7-6700 CPU and 16.0GB RAM. At each iteration, $k = 5$ and 20% (*i.e.,* $Q = 0.2M$) of the crowdsourced samples are used for k-fold cross-validation (Eq. (27)).

In terms of missing value inference, we compare the performance of BCCS with the following state-of-the-art algorithms:

- *K-NN* [21], which estimates the missing values based on the weighted average of the K nearest neighbors. In our evaluation, we empirically set $K = 5$ for estimation.
- *Gaussian process* (GP) [27], [28], which models the spatial distribution of signals into Gaussian processes. Based on the kernel functions, GP computes the missing signals.
- *Matrix completion* (MC) [29], which fills in or recovers the missing entries of a partially observed matrix.

Besides the above approaches, we also compare BCCS with the basic linear interpolation (based on log-distance path loss model or *LDPL*). In terms of grid selection, task allocation and payment determination, we compare BCCS with the following mechanisms:

- *Random*, which *randomly* selects the sensing grid at each iteration of crowdsourcing until convergence.
- *G-Payment*, which *greedily* selects the grid with the lowest reward (payment cost) until convergence.
- *G-Difference*, which *greedily* selects the grid with the largest signal difference/change for crowdsourcing.

We compare the schemes using the following metrics:

1) *Mean prediction accuracy*: We calculate the mean RSSI error (difference between ground-truth and prediction) of a signal map, *i.e.,* $\frac{\|\boldsymbol{s}-\widehat{\boldsymbol{s}}\|}{N}$ (dB) among all the grids in a site.
2) *Total crowdsourcing payment units*: We calculate the total amount of rewards in all the grids, which simulates the cost for the crowdsourcing platform.
3) *Number of crowdsourcing iterations*: which characterizes the time consumed for the entire crowdsourcing until convergence. The fewer crowdsourcing cycles, the more efficient the platform in signal map construction.

We leverage the walking traces of the users for evaluation (more sophisticated user mobility modeling is part of our future work). We emulate the crowdsensing process based on these random and sparse traces. As the users walk, RSSIs are collected. We will later visualize the RSSI distribution to illustrate the dynamic signal map construction.

B. Evaluation Results

We first illustrate the quality of indoor signal map construction in terms of missing value inferencing errors. Fig. 2 shows the CDF of signal prediction error at all grids. K-NN and MC only consider the spatial correlation, which may not really reflect the inherently sophisticated signal distribution. LDPL and GP assume a certain path loss model during signal regression. They do not consider the inherent prior correlation between signals, thus leading to low inference accuracy in complex environments. In contrast, BCCS utilizes the adaptive formulation without assuming any signal propagation models, hence adapting to the signal map dynamics. We can observe that BCCS outperforms the other schemes (often by more than 25%) thanks to the probabilistic formulation of BCS signal inference, which is more robust to the inherently noisy measurements.

We also evaluate the effect of the important parameters for BCCS. We first evaluate the mean missing value inference performance vs. the correlation weights in Section IV-B. Fig. 3(a) shows the missing value inferencing accuracy vs. the parameter φ in Eq. (10). We can observe that as the parameter grows, the inferencing accuracy first increases and then decreases. Clearly, a performance tradeoff between the spatial and signal correlations accounts for such a trend. As the figure shows, the sensing correlation in terms of geographical space indicates a stronger influence over the missing value inference, implying the first law of geography. We balance these two important factors based on the above observations.

Fig. 3(b) shows mean RSSI errors and crowdsourcing payment units vs. the parameter κ (Eq. (27)) which determines the termination of crowdsensing process. Note that the stricter the criterion (*i.e.,* smaller κ), the more the measurements collected, leading to smaller errors but higher payments. Clearly, there is a tradeoff between sensing cost and accuracy. In our evaluation, we choose $\kappa = 0.75$ by default.

Fig. 4 shows the total payments of different schemes assigned to the crowdsourcing workers. The higher accuracy required, the more payments needed for fine-grained sampling. Other schemes do not consider this balancing between the signal map quality and the crowdsensing cost, hence leading to low utilization of the crowdsourcing budget. Note that G-Difference greedily recovers signals of grids with large differences without cost concerns, thus incurring higher sensing costs than others. However, BCCS considers how to best utilize the crowdsourced signals to minimize the RSSI prediction errors, and a sufficient amount of data for the entire signal map construction. Thanks to the proposed compressive crowdsensing, BCCS maintains the constructed signal map quality, reduces the unnecessary measurements, and hence lowers the

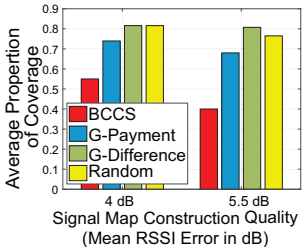


Fig. 5: Crowdsourced grid coverage proportion (indoor).

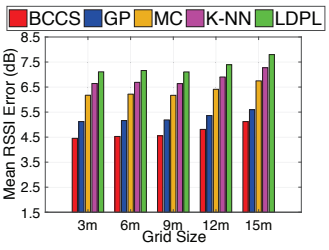


Fig. 6: Mean RSSI prediction errors vs. grid size (indoor).

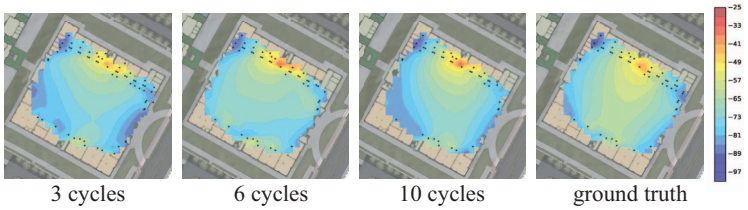


Fig. 7: RSSI visualization of a dynamic signal map construction process (indoor). We observe the map after 10 cycles resembles the ground truth.

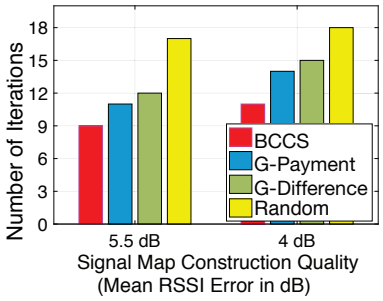


Fig. 8: Number of sensing cycles in the signal map crowdsourcing (indoor).

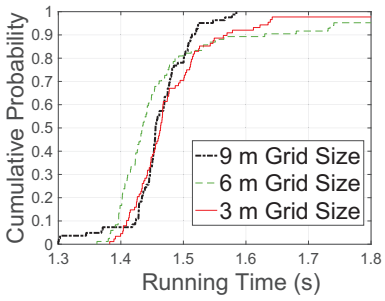


Fig. 9: CDF of missing inference running time for signal crowdsourcing (indoor).

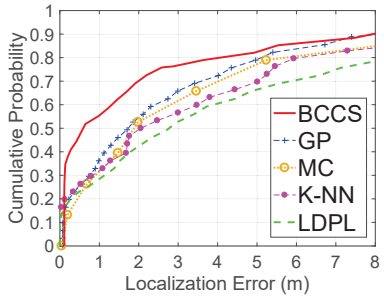


Fig. 10: CDF of localization errors with different signal maps (indoor).

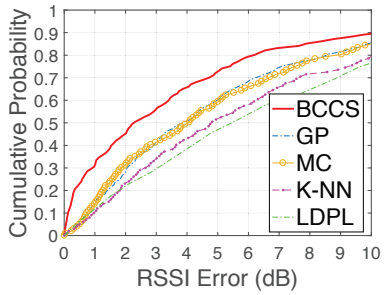


Fig. 11: CDF of mesh signal map missing value inference error (urban).

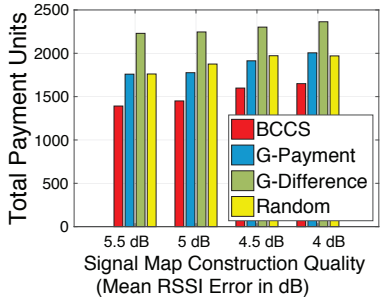


Fig. 12: Total payment evaluation for mesh signal crowdsourcing (urban).

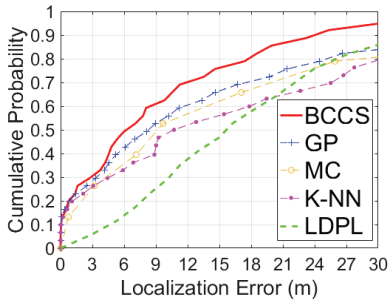


Fig. 13: CDF of access point localization errors with different signal maps (urban).

overall crowdsourcing cost by more than 30%.

Similarly, Fig. 5 shows the coverage proportion (*i.e.,* the ratio of the selected grid number to the total) of selected grids during the crowdsensing process. The fewer grids selected, the lower costs incurred. The number of sensing grids of BCCS is smaller than the other schemes.

Fig. 6 shows the mean missing value inference errors (in dB) of BCCS w.r.t. the grid sizes. Clearly, the prediction granularity is highly correlated with the designed grid size. In other words, the denser the grid size, the more information available for missing value inference, and hence the better the constructed signal map. From this figure, we can see that BCCS is adaptive to various sensing grid sizes.

To visualize the construction process, Fig. 7 shows the dynamic signal maps of an AP w.r.t. crowdsensing cycles, where BCCS is shown to dynamically build up the signal map and the thus-constructed one is very close to the real one.

Fig. 8 shows the number of iterations during the crowdsensing. Clearly, the smaller signal prediction error is required, the more iterations are needed before convergence. Compared to other state-of-the-art schemes, as BCCS jointly maintains the signal prediction accuracy and low cost, the number of iterations during crowdsourcing is significantly reduced, thus minimizing the overall deployment cost of BCCS. Under different signal prediction criteria, BCCS is shown to outperform other schemes by a large margin. Besides Fig. 8, we show in Fig. 9 the CDFs of computation time in each sensing cycle of BCCS signal map construction. Due to its sparse design, BCCS is efficient. We have also observed that the most time-consuming part of BCCS lies in the k-fold cross-validation. Clearly, as the grid size increases, the computation overhead decreases. As the sparse reconstruction with correlations markedly reduces reliance on the data amount, we can observe that BCCS remains to be computationally efficient even with dense grids.

To further assess the signal map quality, we make use of an indoor localization algorithm (say, K-nearest neighbors estimation [16]) to find the locations of signals. We calculate the Euclidean distance (as shown inside Eq. (6)) between the ground-truth and estimated locations. Fig. 10 shows the localization errors given the constructed signal maps with different algorithms. Via better signal map construction, the localization accuracy with BCCS outperforms those of previous state-of-the-art schemes.

We have also conducted extensive simulation over the wireless mesh signals collected in an urban area. Figs. 11 and 12 show the signal prediction accuracy as well as the overall payment cost, respectively. One can observe from both figures that BCCS still outperforms the other algorithms by a large margin. Note that due to less wall partitions outdoor, the GP and LDPL suffer less from none-line-of-sight measurements.

Hence, in terms of missing value inference, their performance gap from that of BCCS is slightly smaller than the indoor case. Fig. 13 also shows the mesh network AP localization (via some well-known access point estimation algorithm [2], [40]) given the constructed signal maps. We have also conducted similar evaluations like those for indoor scenarios. The other evaluation results are qualitatively similar, and due to page limit we omit them here.

VII. Conclusion

In this paper, we have proposed a novel mechanism, called BCCS, to incentivize the signal map construction via joint formulation of crowdsourcing and Bayesian compressive sensing. By novel Bayesian Compressive Sensing (BCS), BCCS infers the missing values during sparse crowdsensing and reduces the platform sensing cost. Furthermore, based on the returned Bayesian confidence level, BCCS dynamically determines the incentive distribution to steer crowdsourcing users for quality measurements. Via a more incentivized interaction between crowdsourcing platform and workers, BCCS predicts the missing values with much lower signal map prediction error and construction cost. Our extensive evaluation has validated the effectiveness and applicability of BCCS. In future, we plan to further validate our BCCS based on other datasets (including urban air quality and noise level).

References

[1] X. Zhou, Z. Zhang *et al.*, "Practical conflict graphs in the wild," *IEEE/ACM TON*, vol. 23, no. 3, pp. 824–835, June 2015.

[2] Z. Li, A. Nika *et al.*, "Identifying value in crowdsourced wireless signal measurements," in *Proc. WWW*, 2017, pp. 607–616.

[3] D. Yang, G. Xue *et al.*, "Crowdsourcing to smartphones: Incentive mechanism design for mobile phone sensing," in *Proc. ACM MobiCom*, 2012, pp. 173–184.

[4] S. Ji, Y. Xue *et al.*, "Bayesian compressive sensing," *IEEE TSP*, vol. 56, no. 6, pp. 2346–2356, June 2008.

[5] J. Wang, N. Tan *et al.*, "WOLoc: WiFi-only outdoor localization using crowdsensed hotspot labels," in *Proc. IEEE INFOCOM*, 2017.

[6] B. Zhou, M. Elbadry *et al.*, "BatMapper: Acoustic sensing based indoor floor plan construction using smartphones," in *Proc. ACM MobiSys*, 2017, pp. 42–55.

[7] F. Restuccia, S. K. Das *et al.*, "Incentive mechanisms for participatory sensing: Survey and research challenges," *ACM TOSN*, vol. 12, no. 2, pp. 13:1–13:40, Apr. 2016.

[8] S. Liu, Z. Zheng *et al.*, "Context-aware data quality estimation in mobile crowdsensing," in *Proc. IEEE INFOCOM*, 2017.

[9] D. Yang, G. Xue *et al.*, "Incentive mechanisms for crowdsensing: Crowdsourcing with smartphones," *IEEE/ACM TON*, vol. 24, no. 3, pp. 1732–1744, June 2016.

[10] M. H. Cheung, F. Hou *et al.*, "Make a difference: Diversity-driven social mobile crowdsensing," in *Proc. IEEE INFOCOM*, 2017.

[11] M. Tang, S. Wang *et al.*, "MOMD: A multi-object multi-dimensional auction for crowdsourced mobile video streaming," in *Proc. IEEE INFOCOM*, 2017.

[12] H. Jin, L. Su *et al.*, "CENTURION: Incentivizing multi-requester mobile crowd sensing," in *Proc. IEEE INFOCOM*, 2017.

[13] R. Kawajiri, M. Shimosaka *et al.*, "Steered crowdsensing: Incentive design towards quality-oriented place-centric crowdsensing," in *Proc. ACM UbiComp*, 2014, pp. 691–701.

[14] Y. Zhao, W. Li *et al.*, "Quantized conflict graphs for wireless network optimization," in *Proc. IEEE INFOCOM*, 2015, pp. 2218–2226.

[15] B. Yang, S. He *et al.*, "Updating wireless signal map with Bayesian compressive sensing," in *Proc. ACM MSWiM*, 2016, pp. 310–317.

[16] C. Wu, Z. Yang *et al.*, "Static power of mobile devices: Self-updating radio maps for wireless indoor localization," in *Proc. IEEE INFOCOM*, 2015, pp. 2497–2505.

[17] S. He, W. Lin *et al.*, "Indoor localization and automatic fingerprint update with altered AP signals," *IEEE TMC*, vol. 16, no. 7, pp. 1897–1910, July 2017.

[18] C. Feng, W. S. A. Au *et al.*, "Received-signal-strength-based indoor positioning using compressive sensing," *IEEE TMC*, vol. 11, no. 12, pp. 1983–1993, Dec 2012.

[19] Y. Kim, Y. Chon *et al.*, "Mobile crowdsensing framework for a large-scale Wi-Fi fingerprinting system," *IEEE Pervasive Computing*, vol. 15, no. 3, pp. 58–67, July 2016.

[20] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theor.*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[21] L. Wang, D. Zhang *et al.*, "CCS-TA: Quality-guaranteed online task allocation in compressive crowdsensing," in *Proc. ACM UbiComp*, 2015, pp. 683–694.

[22] L. Xu, X. Hao *et al.*, "More with less: Lowering user burden in mobile crowdsourcing through compressive sensing," in *Proc. ACM UbiComp*, 2015, pp. 659–670.

[23] X. Hao, L. Xu *et al.*, "Density-aware compressive crowdsensing," in *Proc. ACM/IEEE IPSN*, 2017, pp. 29–39.

[24] T. Liu, Y. Zhu *et al.*, "Incentive design for air pollution monitoring based on compressive crowdsensing," in *Proc. IEEE GLOBECOM*, 2016, pp. 1–6.

[25] R. K. Rana, C. T. Chou *et al.*, "Ear-phone: An end-to-end participatory urban noise mapping system," in *Proc. ACM/IEEE IPSN*, 2010, pp. 105–116.

[26] L. Wang, D. Zhang *et al.*, "Sparse mobile crowdsensing: Challenges and opportunities," *IEEE Commun. Mag.*, vol. 54, no. 7, pp. 161–167, July 2016.

[27] F. Yin and F. Gunnarsson, "Distributed recursive Gaussian Processes for RSS map applied to target tracking," *IEEE J-STSP*, vol. 11, no. 3, pp. 492–503, April 2017.

[28] H. C. Yen and C. C. Wang, "Cross-device Wi-Fi map fusion with Gaussian processes," *IEEE TMC*, vol. 16, no. 1, pp. 44–57, Jan. 2017.

[29] S. Nikitaki, G. Tsagkatakis *et al.*, "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.

[30] M. Roughan, Y. Zhang *et al.*, "Spatio-temporal compressive sensing and internet traffic matrices (extended version)," *IEEE/ACM TON*, vol. 20, no. 3, pp. 662–676, June 2012.

[31] Y. Zhu, Z. Li *et al.*, "A compressive sensing approach to urban traffic estimation with probe vehicles," *IEEE TMC*, vol. 12, no. 11, pp. 2289–2302, Nov. 2013.

[32] L. Kong, M. Xia *et al.*, "Data loss and reconstruction in sensor networks," in *Proc. IEEE INFOCOM*, April 2013, pp. 1654–1662.

[33] J. Wang, S. Tang *et al.*, "Data gathering in wireless sensor networks through intelligent compressive sensing," in *Proc. IEEE INFOCOM*, 2012, pp. 603–611.

[34] C. Meng, H. Xiao *et al.*, "Tackling the redundancy and sparsity in crowd sensing applications," in *Proc. ACM SenSys*, 2016, pp. 150–163.

[35] M. E. Tipping, "Sparse Bayesian learning and the relevance vector machine," *J. Mach. Learn. Res.*, vol. 1, pp. 211–244, Sep. 2001.

[36] P. Micholia, M. Karaliopoulos *et al.*, "Mobile crowdsensing incentives under participation uncertainty," in *Proc. ACM MSCC*, 2016, pp. 29–34.

[37] R. Kohavi *et al.*, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *Proc. IJCAI*, vol. 14, no. 2, 1995, pp. 1137–1145.

[38] S. Li, X. Tao *et al.*, "A variational Bayesian EM approach to structured sparse signal reconstruction," in *Proc. IEEE VTC-Fall*, Sept. 2014, pp. 1–5.

[39] C. Phillips, R. Senior *et al.*, "Robust coverage and performance testing for large area networks," in *Springer AccessNets*, 2008.

[40] J. Torres-Sospedra, R. Montoliu *et al.*, "UJIIndoorLoc: A new multi-building and multi-floor database for WLAN fingerprint-based indoor localization problems," in *Proc. IPIN*, 2014, pp. 261–270.