# Pricing Mechanism for Quality-Based Radio Mapping via Crowdsourcing

Xuhang Ying, Sumit Roy, and Radha Poovendran
Department of Electrical Engineering, University of Washington - Seattle
Email: {xhying, sroy, rp3}@uw.edu

***Abstract*—White Space (WS) Networking crucially relies on the active monitoring of spatio-temporal spectrum usage (to identify WS opportunities). To achieve this, one way is to gather spectrum data via wide-area sensor deployment and construct better Radio Environment Maps (REMs) with spatial models such as Kriging and Gaussian Process (GP). An economically viable alternative is via incentivized crowdsourcing, i.e., outsourcing sensing tasks to mobile users who have sensorized high-end client devices like tablets or smartphones, and providing proper incentives to compensate for users' sensing costs. In crowdsourced REM, features that impact REM performance and economic cost include user locations and the heterogeneity of user devices, which impact data quality and sensing costs. In this work, we emphasize the use of a hardware noise term in the GP model to account for data quality, and adopt mutual information to quantify sampling performance; we further design a pricing mechanism that allows the platform to maximize its expected utility at each stage and send optimal price offers to users sequentially, with joint consideration of sampling value, data quality and cost. We conduct simulations to evaluate the performance. Simulation results show that our mechanism outperforms two baseline mechanisms, and benefits from more users and less hardware noise (i.e., better data quality).**

***Index Terms*—Pricing Mechanism, Crowdsourcing, Radio Mapping, Spatial Interpolation, Gaussian Process, Data Quality.**

## I. Introduction

The rapidly growing demand for wireless services has translated into an increasing need for network capacity and additional spectrum. Although a significant portion of spectrum is already allocated or licensed for various purposes (e.g., TV services, radar), many studies have shown that spectrum is often grossly underutilized. To improve spectrum utilization, it is highly desirable to share locally idle licensed spectrum, called *White Spaces* (WS), with unlicensed users dynamically, while protecting licensed users from harmful interference [1].

To enable WS networking, it is crucial to monitor spectrum usage and identify WS opportunities in both space and time. Current spectrum databases based on empirical radio propagation models are efficient and scalable, but tend to be error-prone, since they do not account for local environments (e.g., trees, buildings) [2], [3]. The above suggests a need for wide-area spectrum sensing with sensors that are able to provide more accurate local RSSI[1] data, which can be used to construct better REMs via model-based spatial interpolation techniques, such as Kriging [2]–[4] or Gaussian process (GP) [5]–[7].

This work was supported by NSF AST award 1443923 under the EARS program.

[1]RSSI stands for received signal strength indicator.

As wide-area sensor deployment could be costly, the economically viable alternative is *incentivized crowdsourcing (or crowd-sensing)*: the platform who wants to acquire spectrum data can outsource sensing tasks to spatially distributed users, who have mobile devices with sensing capability, and pay them appropriately to compensate for consumed resources (e.g., battery, CPU, storage). In this framework, a set of users has a *sampling value* to the platform based on locations and data quality, and a corresponding *total payment* (i.e., the sum of individual payments). The platform aims to maximize the *utility* (i.e., the difference of the total value and payment), and the questions become who to select and how much to pay.

Compared to traditional model-based data acquisition [6], crowdsourced REM is different in the following aspects. First, users possess heterogeneous devices that produce RSSI data of different quality, and the platform has no direct control over user locations. It means that the model needs to explicitly model and incorporate data quality, and user selection is restricted to the set of interested users. Second, unlike typical specialized sensors, user devices are not dedicated to sensing and busy with many other tasks simultaneously. Despite the fact that energy (or battery) costs of a specific type of device are most dominant and deterministic for a given task, users who have the same type of device may incur different additional opportunity costs based on their own device statuses, when they decide to spend resources on sensing and not on others. Therefore, payment determination needs to account for the randomness in users' sensing costs.

Recently, a number of incentive mechanisms have been proposed for general-purpose crowd-sensing [8]–[10], but few focus on spatial-model-based crowd-sensing. In our previous work [11] on this topic, we proposed an auction-based incentive mechanism, but implicitly assumes equally good spectrum data. In this work, we define data quality in terms of hardware noise, and consider pricing mechanisms. Based on reported device types, the platform can estimate noise and cost distributions, and use spatial models to valuate user measurements. Our primary contributions are as follows:

- We explicitly consider hardware noise in the GP model, and adopt mutual information to quantify sampling performance and data quality.
- We design a crowdsourcing system that periodically acquires spectrum data from users to construct REMs, and propose a pricing mechanism that allows the platform to send optimal price offers to users sequentially, with joint consideration of sampling value, data quality and cost.

- We conduct simulations to evaluate the proposed mechanism, and demonstrate its superiority against two variants (one sends the optimal offer to the user with maximum utility, and the other sends a random offer to the user with maximum expected utility). Our results show that the proposed mechanism is affected by the valuation function over noise-aware MI, and benefits from more users and less hardware noise (i.e., better data quality).

The rest of the paper is organized as follows. A review of related works is provided in Section II. We describe our model in Section III and our pricing mechanism in Section IV. We evaluate the proposed mechanism in Section V, and provide the conclusion in Section VI.

## II. Related Works

In spatial sampling or sensing, a natural problem is how to select the optimal set of sample or sensor locations under certain constraints (e.g., cardinality constraint), which is often formulated as an optimization problem, such as minimizing the prediction-error variance as in geo-statistics [12] or maximizing mutual information as in GP-based sensor placement [6]. In such campaigns, sampling errors and noise tend to be systematic, since measurement instruments and sensors are usually made by the same manufacturer and carefully calibrated to minimize variations in hardware. When it comes to crowdsourcing, however, heterogeneous devices tend to behave differently, which should be taken into consideration.

In incentivized crowdsourcing, different mechanisms are proposed for different settings. In [8], users can choose any subset of tasks with predefined values and ask for minimum payments. Authors proposed a truthful auction-based mechanism that maximizes the platform's utility. In [9], authors defined empirical quality indicator for each user as the deviation from the average of its most recent measurements, and focused on minimizing the total payment to users while meeting a certain quality of service. In [10], authors focused on data quality estimation of uncalibrated devices with expectation maximization algorithm, and proposed a pricing mechanism for general sensing purposes. In this work, we characterize device heterogeneity in terms of data quality (or hardware noise) and sensing costs. We formally incorporate hardware noise into the spatial model for RSSI data, and propose an expected-utility-maximizing pricing mechanism.

## III. Our Model

### A. System Architecture

Fig. 1 illustrates our system, which consists of a centralized server called *platform*, and spatially distributed *mobile users* with sensing capability. Each user knows its current location, whose device is registered and authenticated with the platform.

Data acquisition happens periodically. In each period, users first sign up with the platform and submit location and device type, so that the platform can evaluate the value of data for radio mapping and quantify noise degradation (data quality). Then the platform informs a subset of winners of the sensing task with detailed instructions (e.g., desired center frequency, sampling rate) along with take-it-or-leave-it *price offers*. Note

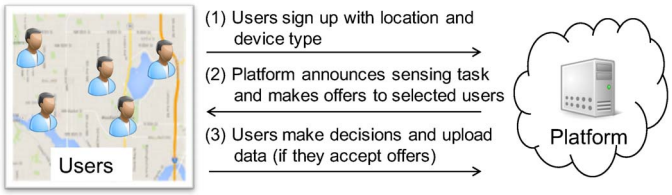



Fig. 1: Pricing architecture for crowdsourced radio mapping.

that offers may be made simultaneously or sequentially. Since each user consumes hardware resources for sensing, it will incur a *sensing cost*, which is privately known only to itself. We assume that all users are *rational* and will accept an offer if the reward is higher than the cost; otherwise, they are free to reject an offer without incurring cost of any kind[2]. Finally, users inform the platform of their decisions, and upload geo-tagged spectrum data to the platform, if they accept offers.

We assume users are of low mobility (e.g., pedestrians) and the displacement between the registered location and the eventual measurement location is small. We also assume users are non-collaborative and honest in following the protocol (e.g., no cheating with location or device type). Considerations of security and privacy enhancement within this framework is left as future work.

### B. Proposed Noise-Aware RSSI Model

We model the front-end RSSI (in dBm) at a point $s \in \mathbb{R}^2$ as a Gaussian random variable $X_s \sim N(\mu_s, \sigma_s^2)$, which is location-dependent and can be decomposed as

$$X(s) = \mu(s) + \delta(s) \tag{1}$$

where $\mu(s)$ captures the RSSI component due to path loss, and $\delta(s)$ represents the shadowing effect, which is normal in the dB scale. When a user $i$ takes a measurement at location $s_i$, additional hardware noise $\epsilon_i$ (in dB) is introduced by the hardware, and the noisy measurement $X_i(s_i)$ is given by

$$X_i(s_i) = X(s_i) + \epsilon_i \tag{2}$$

where $\epsilon_i \sim N(0, \sigma_{\epsilon_i}^2)$, which is independent of the location and other devices. Therefore, $X_i(s_i)$ is a Gaussian random variable with a mean of $\mu_{s_i}$ and a variance of $\sigma_{s_i}^2 + \sigma_{\epsilon_i}^2$. For simplicity, we write $X(s_i)$ as $X_i$ and $X_i(s_i)$ as $\tilde{X}_i$, where "~" is used to emphasize the presence of hardware noise.

As in GP model, the set of noisy measurements at sample locations $\mathcal{S} = \{s_1, ..., s_n\}$ acquired from by $n$ different users within the region of interest constitutes a Gaussian random vector $\tilde{X}_\mathcal{S} = [\tilde{X}_1, ..., \tilde{X}_n]$ with a joint distribution of

$$f(\tilde{\mathbf{x}}_\mathcal{S}) = \frac{1}{(2\pi)^{n/2}|\Sigma_{\tilde{\mathcal{S}}\tilde{\mathcal{S}}}|} e^{-\frac{1}{2}(\tilde{\mathbf{x}}_\mathcal{S} - \mu_\mathcal{S})^T \Sigma_{\tilde{\mathcal{S}}\tilde{\mathcal{S}}}^{-1} (\tilde{\mathbf{x}}_\mathcal{S} - \mu_\mathcal{S})} \tag{3}$$

where $\tilde{\mathbf{x}}_\mathcal{S} = [\tilde{x}_1, ..., \tilde{x}_n]$ is a realization of $\tilde{X}_\mathcal{S}$, $\mu_\mathcal{S} = [\mu_1, ..., \mu_n]$ is the mean vector and $\Sigma_{\tilde{\mathcal{S}}\tilde{\mathcal{S}}}$ is the covariance matrix with the $(i, j)$-th entry being $\text{Cov}(\tilde{X}_i, \tilde{X}_j)$.

Since shadowing is spatially correlated, it may be statistically captured by the covariance between $X_i$ and $X_j$, which

[2]We assume no entry or other overhead costs, i.e., a user does not incur a fee to communicate with the platform, and the communication cost is negligible.

is assumed to be a function of the distance difference, that is, $\text{Cov}(X_i, X_j) = \mathcal{K}(||s_i - s_j||)$, where $\mathcal{K}(\cdot)$ is a parametric kernel or covariance function[3]. Since $\{\epsilon_i\}$ are zero-mean and independent of $\{X_i\}$, we also have $\text{Cov}(\tilde{X}_i, \tilde{X}_j) = \text{Cov}(\tilde{X}_i, X_j) = \text{Cov}(X_i, X_j)$ for $i \neq j$.

Conditioned on a set of noisy measurements $\tilde{\mathbf{x}}_{\mathcal{S}}$, the front-end RSSI $X_y$ at an unmeasured location $y$ is a Gaussian with a conditional mean $\mu_{y|\tilde{\mathcal{S}}}$ and variance $\sigma^2_{y|\tilde{\mathcal{S}}}$,

$$\begin{aligned}\mu_{y|\tilde{\mathcal{S}}} &= \mu_y + \Sigma_{y\tilde{\mathcal{S}}}\Sigma^{-1}_{\tilde{\mathcal{S}}\tilde{\mathcal{S}}}(\tilde{\mathbf{x}}_{\mathcal{S}} - \mu_{\mathcal{S}}) \qquad (4)\\ &= \mu_y + \Sigma_{y\mathcal{S}}\left(\Sigma_{\mathcal{S}\mathcal{S}} + D_{\tilde{\mathcal{S}}}\right)^{-1}(\tilde{\mathbf{x}}_{\mathcal{S}} - \mu_{\mathcal{S}})\\ \sigma^2_{y|\tilde{\mathcal{S}}} &= \sigma^2_y - \Sigma_{y\tilde{\mathcal{S}}}\Sigma^{-1}_{\tilde{\mathcal{S}}\tilde{\mathcal{S}}}\Sigma_{\tilde{\mathcal{S}}y} \qquad (5)\\ &= \sigma^2_y - \Sigma_{y\mathcal{S}}\left(\Sigma_{\mathcal{S}\mathcal{S}} + D_{\tilde{\mathcal{S}}}\right)^{-1}\Sigma_{\mathcal{S}y}\end{aligned}$$

where $\Sigma_{\mathcal{S}\mathcal{S}}$ is the covariance matrix for $X_{\mathcal{S}}$, $\Sigma_{y\mathcal{S}}$ is the covariance vector with each entry for each $u \in \mathcal{S}$ which has a value of $\mathcal{K}(||y - s_i||)$, and $D_{\tilde{\mathcal{S}}}$ is an $m$-by-$m$ diagonal matrix with $d_i = \sigma^2_{\epsilon_i}$.

### C. Sampling Design

In crowdsourced radio mapping, measurements are noisy and only available at user locations $\mathcal{S}$, but we may be interested in (front-end) RSSI prediction at a different set of unmeasured locations $\mathcal{U}$ (e.g., a discretization of a desired region). A natural question is that given user locations and data quality, how the platform should choose the best subset $\mathcal{A} \subseteq \mathcal{S}$ that optimizes the sampling performance subject to certain constraints (e.g., cardinality constraint on $\mathcal{A}$), which is referred to as *sampling design* in spatial statistics. One popular metric is called *mutual information* (MI), which quantifies the reduction of uncertainty about the estimates of interested locations given selected sampling locations. Given the noise-aware RSSI data model, our MI metric is given by

$$MI(\tilde{\mathcal{A}}) = I(X_{\mathcal{U}}; X_{\tilde{\mathcal{A}}}) = H(X_{\mathcal{U}}) - H(X_{\mathcal{U}}|X_{\tilde{\mathcal{A}}}) \qquad (6)$$

where $H(X_{\mathcal{V}}|X_{\tilde{\mathcal{A}}})$ is conditional entropy that captures prediction uncertainty in $X_{\mathcal{U}}$, given by

$$H(X_{\mathcal{U}}|X_{\tilde{\mathcal{A}}}) = -\int f(\mathbf{x}_{\mathcal{U}}, \tilde{\mathbf{x}}_{\mathcal{A}}) \log f(\mathbf{x}_{\mathcal{U}}|\tilde{\mathbf{x}}_{\mathcal{A}}) d\mathbf{x}_{\mathcal{U}} d\tilde{\mathbf{x}}_{\mathcal{A}} \qquad (7)$$

### D. User Model

Consider $n$ users at locations $\mathcal{S} = \{s_1, ..., s_n\}$. The sensing cost of user $i$ is modeled as a random variable $C_i$, which is private and independent of each other. The cost distribution (PDF) $f_i(c_i)$ is determined by the device type. In reality, it is very likely that data quality of a device is positively related to its hardware cost and sensing cost, as a consequence. By knowing the device type, we assume the platform can accurately estimate the cost distribution and noise characteristics. Since a user may or may not accept an offer, we define $Y_i$ for user $i$ who receives an offer $p_i$,

$$Y_i = \begin{cases} 1, & \text{if } c_i < p_i, \text{ and user } i \text{ accepts the offer} \\ 0, & \text{if } c_i \geq p_i, \text{ and user } i \text{ rejects the offer} \end{cases} \qquad (8)$$

where $\Pr(Y_i = 1|p_i) = \int_0^{p_i} f_i(c_i) dc_i = F_i(p_i)$.

[3]In this study, we assume $\mathcal{K}(\cdot)$ is both stationary and isotropic [6]. However, our following discussions are applicable to general kernel functions.

### E. Platform Model

In each period, the platform selects a subset of users $\mathcal{A} \subseteq \mathcal{S}$, and determines a price vector $\mathbf{p} = [p_1, ..., p_m]$, where $m = |\mathcal{A}|$. The platform has the knowledge of current user locations, data quality, cost distributions, and the spatial model (estimated from previous measurements). Denote the set of users who accept offers as $\mathcal{B} \subseteq \mathcal{A}$. Then the *utility* of the platform is given by

$$u(\mathcal{B}, \mathbf{p}) = v(MI(\tilde{\mathcal{B}})) - \sum_{i \in \mathcal{B}} p_i \qquad (9)$$

where $v(\cdot)$ is the valuation function of the platform over MI. From the economical perspective, the first term is the achieved revenue, the second is the total cost, and their difference is the resulting profit.

Without prior knowledge of user decisions, the platform has to select users based on the *expected utility* (EU), defined as

$$EU(\mathcal{A}, \mathbf{p}) = \sum_{\mathcal{B} \subseteq \mathcal{A}} u(\mathcal{B}, \mathbf{p}) \cdot \Pr(\mathcal{B}, \mathbf{p}) \qquad (10)$$

where $\Pr(\mathcal{B}, \mathbf{p}) = \Pi_{i \in \mathcal{B}} F_i(p_i) \cdot \Pi_{j \in \mathcal{A} \setminus \mathcal{B}}(1 - F_j(p_j))$, since each user makes an independent decision. Then the platform's objective can be formulated as

$$\max_{\mathcal{A} \subseteq \mathcal{S}, \mathbf{p}} EU(\mathcal{A}, \mathbf{p}) \qquad (11)$$

## IV. Quality-Based Pricing Schemes

Essentially, the platform aims to jointly maximize the expected utility $EU(\mathcal{A}, \mathbf{p})$ in the discrete domain of $\mathcal{A}$ as well as in the continuous domain of $\mathbf{p}_{\mathcal{A}}$. Unfortunately, it can be very difficult to choose $\mathcal{A}$ and $\mathbf{p}$ simultaneously. On the one hand, it is intractable to compute the objective function $U(\cdot)$, since there are up to $2^{|\mathcal{A}|}$ possible decision combinations for a given $\mathcal{A}$ (especially when $|\mathcal{A}|$ is large). On the other hand, there are up to $2^n$ different combinations of $\mathcal{A}$ (or $\binom{n}{k}$ for the cardinality constraint $|\mathcal{A}| = k$).

Although the sampling method may be deployed for computing $EU(\cdot)$ as in influence maximization [13], it could be very time-consuming and thus not scalable in practice. Therefore, we are interested in sequential offering in this paper, where the platform make offers one by one based on the knowledge of previous responses, which may be more feasible and scalable in the real world.

### A. Computing Marginal Noise-Aware MI

We use $m_{\tilde{\mathcal{A}}}(\tilde{z})$ to denote the marginal contribution in noise-aware MI of an additional noisy measurement $\tilde{z}$, given the set of collected measurements $\tilde{\mathcal{A}}$. For simplicity, we denote $H(X_{\mathcal{U}}|X_{\mathcal{A}})$ as $H(\mathcal{H}|\mathcal{A})$. Then we have

$$\begin{aligned} m_{\tilde{\mathcal{A}}}(\tilde{z}) &= MI(\tilde{\mathcal{A}} \cup \{z\}) - MI(\tilde{\mathcal{A}}) \qquad (12)\\ &= H(\mathcal{U}) - H(\mathcal{U}|\tilde{\mathcal{A}} \cup \{\tilde{z}\}) - \left[H(\mathcal{U}) - H(\mathcal{U}|\tilde{\mathcal{A}})\right]\\ &= H(\mathcal{U}|\tilde{\mathcal{A}}) - H(\mathcal{U}|\tilde{\mathcal{A}} \cup \{\tilde{z}\})\\ &= H(\mathcal{U} \cup \tilde{\mathcal{A}}) - H(\tilde{\mathcal{A}}) - \left[H(\mathcal{U} \cup \tilde{\mathcal{A}} \cup \{\tilde{z}\}) - H(\tilde{\mathcal{A}} \cup \{\tilde{z}\})\right]\\ &= H(\tilde{z}|\tilde{\mathcal{A}}) - H(\tilde{z}|\tilde{\mathcal{A}} \cup \mathcal{U}) \end{aligned}$$

where $H(\tilde{z}|\tilde{\mathcal{A}})$ is the (differential) entropy of a Gaussian random variable $X_{\tilde{z}}$ provided $X_{\tilde{\mathcal{A}}}$, which is given by

$$H(\tilde{z}|\tilde{\mathcal{A}}) = H(X_{\tilde{z}}|X_{\tilde{\mathcal{A}}}) = \frac{1}{2}\log(2\pi e\sigma^2_{\tilde{z}|\tilde{\mathcal{A}}}) \tag{13}$$

From Eq. 6, we can further compute

$$\begin{aligned}\sigma^2_{\tilde{z}|\tilde{\mathcal{A}}} &= \sigma^2_z + \sigma^2_{\epsilon_z} - \Sigma_{\tilde{z}\tilde{\mathcal{A}}}\Sigma^{-1}_{\tilde{\mathcal{A}}\tilde{\mathcal{A}}}\Sigma_{\tilde{\mathcal{A}}\tilde{z}} \\ &= \sigma^2_z + \sigma^2_{\epsilon_z} - \Sigma_{z\mathcal{A}}\left(\Sigma_{\mathcal{A}\mathcal{A}} + D_{\tilde{\mathcal{A}}}\right)^{-1}\Sigma_{\mathcal{A}z}\end{aligned} \tag{14}$$

where $D_{\tilde{\mathcal{A}}}$ is a diagonal matrix with $d_i = \sigma^2_{\epsilon_i}$ for $i \in \mathcal{A}$.

### B. Proposed Pricing Mechanism

Our pricing mechanism is described in Algorithm 1, where the platform always sends the next offer to the best user that maximizes expected utility at the corresponding optimal price, based on the knowledge of previous responses. Note that we omit the "~" notation for convenience in the algorithm.

**Algorithm 1:** Proposed mechanism

**input :** $\mathcal{S}$ – user locations, $\mathcal{U}$ – locations of interest, $\mathcal{K}(\cdot)$ – kernel, $v(\cdot)$ – valuation function, $\{F_i(c_i)\}$ – cost distributions, $\{\tilde{\sigma}_i^2\}$ – noise variances

**output:** $\mathcal{A}$ – sent offers, $\mathcal{B}$ – accepted offers, $\mathbf{p}$ – prices

1 $\mathcal{A} \leftarrow \emptyset$, $\mathcal{B} \leftarrow \emptyset$, $\mathbf{p}$; // $\mathcal{B}$ is the set of accepted users
2 $U_{\mathcal{A}} \leftarrow v(MI(\mathcal{B})) - \sum_{i\in\mathcal{B}} p_i$; // Current utility
3 **while** $\mathcal{A} \neq \mathcal{S}$ **do**
4  // Price determination for each user
5  **foreach** $i \in \mathcal{S} \setminus \mathcal{A}$ **do**
6  $m_i \leftarrow MI(\mathcal{B}\cup\{i\}) - MI(\mathcal{B})$;
7  $p_i^* \leftarrow \arg\max_{p_i\in[0,v(m_i)]}(v(m_i)-p_i)\cdot F_i(p_i)$;
8  $EU^*_{\mathcal{A}\cup\{i\}} \leftarrow U_{\mathcal{A}} + (v(m_i)-p_i^*)\cdot F_i(p_i^*)$
9  // User selection
10  $\mathcal{T} \leftarrow \mathcal{S}\setminus\mathcal{A}$;
11  **while** $\mathcal{T} \neq \emptyset$ **do**
12  $j \leftarrow \arg\max_{i\in\mathcal{T}} EU^*_{\mathcal{A}\cup\{i\}}$;
13  **if** $EU^*_{\mathcal{A}\cup\{j\}} - U_{\mathcal{A}} > \gamma$ **then**
14  $\mathcal{A} \leftarrow \mathcal{A}\cup\{j\}$, $\mathbf{p} \leftarrow [\mathbf{p},\ p_j^*]$;
15  Send an offer $p_j^*$ to user $j$, and observes $y_j$;
16  **if** $y_j = 1$ **then**
17  $\mathcal{B} \leftarrow \mathcal{B}\cup\{j\}$;
18  **Break**;
19  **else**
20  $\mathcal{T} \leftarrow \mathcal{T}\setminus\{j\}$;
21  **else**
22  **return** $(\mathcal{A},\mathcal{B},\mathbf{p})$;
23 **return** $(\mathcal{A},\mathcal{B},\mathbf{p})$;

*1) Price determination (Lines 6-8):* Given the current set of sent offers $\mathcal{A}$ and the set of accepted offers $\mathcal{B} \subseteq \mathcal{A}$, the current (conditional) utility is given by $U_{\mathcal{A}} = v(MI(\tilde{\mathcal{B}})) - \sum_{i\in\mathcal{B}} p_i$. Based on whether user $i \in \mathcal{S}\setminus\mathcal{A}$ accepts an offer $p_i$, the resulting utility may change or stay the same,

$$U_{\mathcal{A}\cup\{i\}}(p_i|\mathcal{B}) = \begin{cases} U_{\mathcal{A}} + v(m_{\tilde{\mathcal{B}}}(\tilde{i})) - p_i, & \text{if } c_i \le p_i \\ U_{\mathcal{A}}, & \text{if } c_i > p_i \end{cases} \tag{15}$$

where $m_{\tilde{\mathcal{B}}}(\tilde{i})$ is the marginal gain in noise-aware MI if user $i$ is successfully recruited (Line 6). For an offer to be meaningful, $p_i$ should be no greater than $m_{\tilde{\mathcal{B}}}(\tilde{i})$ (and also in the domain of $C_i$). Accounting for user $i$'s uncertainty in accepting the offer, we can compute the expected utility of sending an offer to user $i$, which is a function of $p_i$ and given by

$$\begin{aligned}EU_{\mathcal{A}\cup\{i\}}(p_i|\mathcal{B}) &= U_{\mathcal{A}} + [v(m_{\tilde{\mathcal{B}}}(\tilde{i})) - p_i]\cdot F_i(p_i) \\ &= U_{\mathcal{A}} + [v(m_{\tilde{\mathcal{B}}}(\tilde{i})) - p_i]\cdot\int_0^{p_i} f_i(c_i)dc_i\end{aligned} \tag{16}$$

For each user $i$, the goal is to compute $p_i^*$ that maximizes $EU_{\mathcal{A}\cup\{i\}}(p_i|\mathcal{B})$ (Line 7). Note that if $f_i(p_i)$ is differentiable and non-increasing in the domain of $p_i$, then $EU_{\mathcal{A}\cup\{i\}}(p_i|\mathcal{B})$ is a concave function in $p_i$, and efficient algorithms (e.g., gradient descent) can be used to compute $p_i^*$ [14]. Indeed, we can check its second-order derivate,

$$EU''_{\mathcal{A}\cup\{i\}}(p_i|\mathcal{B}) = -2f_i(p_i) + [v(m_{\tilde{\mathcal{B}}}(\tilde{i})) - p_i]\cdot f_i'(p_i)$$

Since $f_i(p_i) \ge 0$, $m_{\tilde{\mathcal{B}}}(\tilde{i}) \ge p_i$ and $f_i'(p_i) \le 0$, we have $EU''_{\mathcal{A}\cup\{i\}}(p_i|\mathcal{B}) \le 0$. One example is uniform distribution. In general, we may search for the optimal price exhaustively.

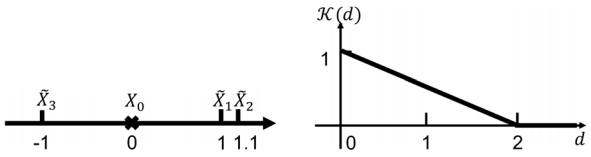
*2) User selection (Lines 10-22):* The algorithm considers users in $\mathcal{S}\setminus\mathcal{A}$ in descending order of $EU^*_{\mathcal{A}\cup\{i\}}$ (Lines 10-12). If the user being considered contributes at least $\gamma$ in terms of marginal expected utility (Line 13), where $\gamma$ is a preset threshold (e.g., 0.01), the platform sends the optimal offer to that user and waits for its response (Lines 14-15). If it accepts, the platform needs to re-evaluate remaining users and re-compute optimal prices, based on the updated information (Lines 16-18); otherwise, the next best user will receive an offer immediately (Line 20). The algorithm is terminated if i) there are no remaining users to send offers to, or ii) none of the remaining users can bring non-trivial marginal expected utility to the platform.

*3) Complexity Analysis:* The computational complexity of the proposed mechanism is $\mathcal{O}(n^2)$, since it takes $\mathcal{O}(n)$ to determine the optimal price for each remaining user and the platform may select up to $n$ users in the worst case. The inner while-loop does not require re-computation of optimal prices, and is dominated by the for-loop. Note that the complexity of $\mathcal{O}(n^2)$ is very conservative, since the algorithm may terminate much earlier based on on system configurations.

### C. Example

We take the example in Fig. 2 to illustrate our quality-based pricing mechanism. In this example, there are three users located at $s_1 = 1$, $s_2 = 1.1$ and $s_3 = -1$ (Fig. 2a), whose measured RSSI values are denoted as $\tilde{X}_1$, $\tilde{X}_2$ and $\tilde{X}_3$, respectively. The platform aims to predict the front-end RSSI $X_0$ at $s = 0$. In other words, we have $\mathcal{S} = \{\tilde{X}_1, \tilde{X}_2, \tilde{X}_3\}$ and $\mathcal{U} = \{X_0\}$. We assume users' costs are uniformly distributed – $C_1 \sim U[0.5, 1]$, $C_2 \sim U[0.5, 1]$ and $C_3 \sim U[0.7, 1.2]$; their data quality is also different – $\sigma^2_{\epsilon_1} = 0.4$, $\sigma^2_{\epsilon_2} = 0.1$ and $\sigma^2_{\epsilon_3} = 0.2$. The kernel/covariance function $\mathcal{K}(d) = -\frac{1}{2}d + 1$, where $d$ is the distance between two locations (Fig. 2b). The valuation function is assumed to be $v(x) = 10\cdot x$.

At initialization, $\mathcal{A} = \mathcal{B} = \emptyset$ and $U_{\mathcal{A}} = 0$. The platform needs to determine the first offer. First, the platform considers




(a) (b)

Fig. 2: (a) Example topology. (b) Kernel function $\mathcal{K}(\cdot)$

user 1. Since we have $\sigma^2_{\tilde{X}_1|\emptyset} = \tilde{\sigma}^2_1 = \mathcal{K}(0) + \sigma^2_{\epsilon_1} = 1.4$ and $\sigma^2_{\tilde{X}_1|X_0} = \sigma^2_{\tilde{X}_1} - \sigma_{\tilde{X}_1,X_0} \cdot (\sigma^2_{X_0})^{-1} \cdot \sigma_{\tilde{X}_1,X_0} = 1.15$, the marginal MI gain of user 1 is $m_\emptyset(\tilde{X}_1) = \frac{1}{2}(\log \sigma^2_{\tilde{X}_1|\emptyset} - \log \sigma^2_{\tilde{X}_1|X_0}) = 0.0984$. The expected utility of sending an offer to user 1 is $EU(p_1) = U_\emptyset + (10 \cdot 0.0984 - p_1) \cdot \frac{p_1 - 0.5}{1-0.5}$. Maximizing $EU(p_1)$ (quadratic in $p_1$) results in $p_1^* = 0.742$ and $EU(p_1^*) = \mathbf{0.117}$.

Similarly, we have $p_2^* = 0.755$ and $EU(p_2^*) = \mathbf{0.135}$, as well as $p_3^* = 0.935$ and $EU(p_3^*) = \mathbf{0.110}$. Since user 2 leads to the maximum expected utility, the platform will send a price offer $p_2^* = 0.755$ to user 2.

Assume user 2 accepts the offer. Then we have $\mathcal{A} = \mathcal{B} = \{\tilde{X}_2\}$, $U_\mathcal{A} = v(MI(\{\tilde{X}_2\})) - p_2^* = 10 \times 0.101 - p_2^* = 0.255$. The platform needs to determine the second offer. It first consider user 1. Since $\sigma^2_{\tilde{X}_1|\tilde{X}_2} = 0.580$ and $\sigma^2_{\tilde{X}_1|\{\tilde{X}_2,X_0\}} = 0.564$, we have $m_{\tilde{\mathcal{A}}}(\tilde{X}_1) = \frac{1}{2}(\log \sigma^2_{\tilde{X}_1|\tilde{X}_2} - \log \sigma^2_{\tilde{X}_1|\{\tilde{X}_2,X_0\}}) = 0.014$. However, as $10 \cdot m_{\tilde{\mathcal{A}}}(\tilde{X}_1) - p_1 < 0$ for $p_1 \in [0.5, 1]$, the platform will not benefit from sending an offer to user 1 at all. Similarly, we have $p_3^* = 1.085$ and $EU(p_3^*) = \mathbf{0.296}$ for user 3. Therefore, the platform will send a price offer $p_3^* = 1.085$ to user 3 and waits for its response.

After receiving user 3's response, the platform will determine the next offer. If we go through the same process to evaluate user 1, we will find that the platform will not gain anything from sending user 1 an offer, regardless of user 3's decision. Therefore, the algorithm will terminate and the corresponding $(\mathcal{A}, \mathcal{B}, \mathbf{p})$ will be returned.

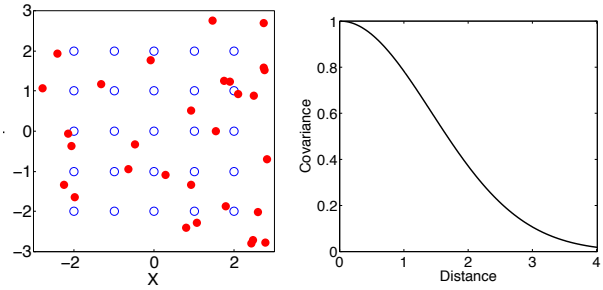
## V. Performance Evaluation

In this section, we compare the proposed mechanism with two baseline mechanisms, and study the impact of the number of users, the valuation function and noise (data quality) on the proposed mechanism.

### A. Simulation Setup

As shown in Fig. 3a, the desired region is discretized into a total of 25 locations[4]. The total number of users is $n$, and each user $i$ has an independent uniform cost distribution, i.e., $C_i \sim U[\underline{c}_i, \underline{c}_i + 1]$, where $\underline{c}_i$ is drawn at the start of each experiment from a uniform distribution $U[0.5, 1]$. We assume the relationship between noise variance and sensing cost is given by $\sigma^2_{\epsilon_i} = 1 - \underline{c}_i$. In other words, $\sigma^2_{\epsilon_i}$ is uniformly distributed over $[0, 0.5]$. The covariance function is given by $\mathcal{K}(d) = \exp(-\frac{d^2}{\theta^2})$ (Fig. 3b), where $\theta = 2$, and the valuation function is $v(x) = \alpha \cdot x$, where $\alpha$ is a constant representing the value per unit of MI. The threshold $\gamma$ for marginal expected utility is 0.01. All results were averaged over 100 experiments.

[4]The resolution is chosen to reduce the amount of time to compute MI. The granularity can definitely be improved in practice, and efficient algorithms have been proposed to compute MI more efficiently [6].




(a) (b)

Fig. 3: (a) Sample topology – an 6km-by-6km region that is quantized into a total of 25 locations (in blue circles) and has 30 randomly distributed users (in red dots). (b) Covariance function (Gaussian) $\mathcal{K}(d) = -\exp(\frac{d^2}{\theta^2})$, where $\theta = 2$.

### B. Results

*1) Comparison with Baseline Mechanisms:* In this simulation, we consider the following two baseline mechanisms:

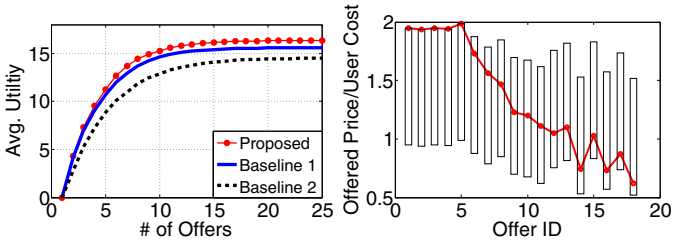
(i) Baseline 1 – a mechanism that computes optimal prices (as in Algorithm 1), but greedily selects the next user with *maximum possible utility*;

(ii) Baseline 2 – a mechanism that selects *random prices*, but greedily selects the next user with maximum expected utility (as in Algorithm 1).

We set $n$ to 40 and $\alpha$ to 3. As in Fig. 4a, the average utility increases but at a slower rate when more offers are sent out for all three mechanisms. Since all three mechanisms stop sending offers when no remaining user can contributed non-trivial positive (expected) utility, the average utility does not increase after a certain point. Besides, we can see that the proposed mechanism outperforms the first baseline mechanism, because it takes into account of the possibility of an offer rejection. It also performs better than the second baseline mechanism by choosing a price smartly instead of blindly.

Taking a closer look at the pricing strategy of the proposed mechanism (Fig. 4b), we can see that the mechanism is very generous in making initial offers to guarantee successful recruitment, because initial samples will have very high values to the platform. But it prefers to take its chances for following offers so as to balance sampling performance and cost.

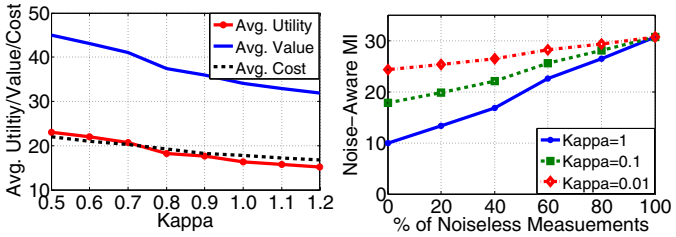
*2) Impact of Noise:* To study the impact of noise (i.e., data quality), we set $n$ to 40, $\alpha$ to 3, and scaled the noise variance for each user by $\kappa$ (varied from 0.5 to 1.2), i.e., $\sigma'^2_{\epsilon_i} = \kappa \cdot \sigma^2_{\epsilon_i}$. As shown in Fig. 5a, hardware noise has an adverse impact on the average utility, as expected. Since our metric is noise-aware, it decreases when noise increases, and so do sampling values. The average cost decreases because offers with lower prices are made, and users are more likely to reject them. But the decreasing rate of the average cost is smaller than that of the average value, which causes the average utility to decrease.




(a) (b)

Fig. 4: (a) Comparison between the proposed mechanism and two baseline mechanisms. (b) Prices offered by the platform (in red dots) via the proposed mechanism compared to users' cost distributions (in rectangles) in one experiment.

We are also interested in the impact of noise existence on the noise-aware MI. We selected a sample set of 11 noisy measurements, and varied the percentage of noiseless measurements from 0% (all noisy) to 100% (all noiseless). As shown in Fig. 5b, the noise-aware MI increases when there are more noiseless measurements. When $\kappa = 1$ (i.e., $\sigma^2_{\epsilon_i} \sim U[0, 0.5]$), MI is 30.65 with no noisy measurements, almost three times that achieved with all noisy measurements, which is 9.93. When $\kappa = 0.01$, noise variances become very small (less than 0.005), but the difference in MI due to noise is as large as 6.34, which is not negligible. Therefore, the existence of noise (even if it is small) has a significant impact on the noise-aware MI and thus the sampling value.

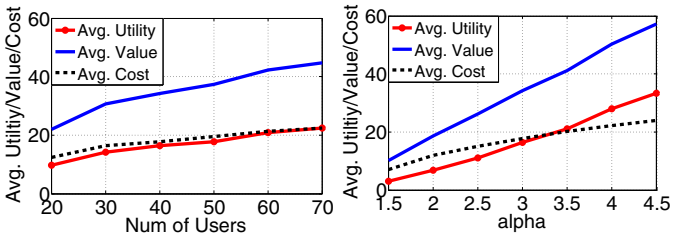



(a) (b)

Fig. 5: (a) Impact of noise variances on mechanism performance. Note that $\kappa$ is a constant that scales noise variances, i.e., $\sigma'^2_{\epsilon_i} = \kappa \cdot \sigma^2_{\epsilon_i}$ for each user $i$. (b) Noise-aware MI versus the percentage of noiseless measurements for different $\kappa$ values. The same set of 11 noisy measurements is used.

*3) Impact of $n$ and $\alpha$:* We first set $\alpha$ to 3 and varied $n$ from 20 to 70 with an increment of 10. As in Fig. 6a, the average utility tends to increase proportionally as the number of users (or user density given a fixed region) increases. This is mainly because as the competition among users becomes more intense, there will be more users with higher sampling values, as well as ones with lower costs, and the platform has greater freedom to select the next user.

Then we set $n$ to 40 and varied $\alpha$ from 1.5 to 4.5 with an increment of 0.5. As illustrated in Fig. 6b, when $\alpha$ increases, the average utility tends to increase proportionally. So does the average value and cost, but the former increases at a larger rate. This is making sense, because now the platform values each unit of noise-aware MI much more, and would like to pay more for each user. As a consequence, users are more likely to accept offers and the utility becomes larger.




(a) (b)

Fig. 6: Impact of (a) number of users and (b) $\alpha$ (i.e., value per unit of noise-aware MI) on mechanism performance.

## VI. Conclusion

In this paper, we captured the heterogeneity of user devices in crowdsourced REM in terms of data quality (hardware noise) and sensing costs. We added hardware noise in the GP-based RSSI model, and adopted noise-aware MI to quantify sampling performance. We further proposed a pricing mechanism that allows the platform to maximize its expected utility and send optimal offers to users one by one, with joint consideration of sampling value, data quality and cost.

We evaluated the proposed mechanism via simulations, and demonstrate its superiority against two baseline mechanisms. Our results show that the proposed mechanism is affected by the valuation function over noise-aware MI, and benefits from more users and less hardware noise (i.e., better data quality).

## References


[1] FCC, "Second report and order and memorandum opinion and order," Nov. 2008.

[2] A. Achtzehn *et al.*, "Improving accuracy for TVWS geolocation databases: Results from measurement-driven estimation approaches," in *in Proc. 7th IEEE DYSPAN*, 2014.

[3] X. Ying *et al.*, "Revisiting TV coverage estimation with measurement-based statistical interpolation," in *COMSNETS 2015*, Jan. 2015, pp. 1–8.

[4] N. Cressie, *Statistics for spatial data*. John Wiley & Sons, 1991.

[5] A. Deshpande *et al.*, "Model-driven data acquisition in sensor networks," in *Proceedings of the Thirtieth International Conference on Very Large Data Bases - Volume 30*, Toronto, Canada, 2004.

[6] A. Krause *et al.*, "Near-optimal sensor placements in gaussian processes: Theory, efficient algorithms and empirical studies," *J. Mach. Learn. Res.*, Jun. 2008.

[7] J. Fink and V. Kumar, "Online methods for radio signal mapping with mobile robots," in *ICRA 2010*, May 2010.

[8] D. Yang *et al.*, "Crowdsourcing to smartphones: Incentive mechanism design for mobile phone sensing," in *Proc. of the 18th Mobicom 2012*. New York, NY, USA: ACM, 2012.

[9] I. Koutsopoulos, "Optimal incentive-driven design of participatory sensing systems," in *INFOCOM 2013*, Apr. 2013.

[10] D. Peng *et al.*, "Pay as how well you do: A quality based incentive mechanism for crowdsensing," in *Proc. of the 16th MobiHoc*, 2015.

[11] X. Ying, S. Roy, and R. Poovendran, "Incentivizing crowdsourcing for radio environment mapping with statistical interpolation," in *Dynamic Spectrum Access Networks (DySPAN)*, 2015.

[12] J. W. van Groenigen *et al.*, "Constrained optimisation of soil sampling for minimisation of the kriging variance," *Geoderma*, Jan. 1999.

[13] D. Kempe *et al.*, "Maximizing the spread of influence through a social network," in *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2003.

[14] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.