Incentive Mechanisms for Crowdsensing: Crowdsourcing With Smartphones

Article *in* IEEE/ACM Transactions on Networking · May 2015

DOI: 10.1109/TNET.2015.2421897

4 authors:

Dejun Yang
Colorado School of Mines
122 PUBLICATIONS **9,265** CITATIONS
SEE PROFILE

Guoliang Xue
Arizona State University
354 PUBLICATIONS **17,378** CITATIONS
SEE PROFILE

Xi Fang
Google Inc.
41 PUBLICATIONS **6,274** CITATIONS
SEE PROFILE

Jian Tang
Syracuse University
232 PUBLICATIONS **11,052** CITATIONS
SEE PROFILE

Incentive Mechanisms for Crowdsensing: Crowdsourcing with Smartphones

Dejun Yang, *Member, IEEE*, Guoliang Xue, *Fellow, IEEE*, Xi Fang, and Jian Tang, *Senior Member, IEEE*

***Abstract*—Smartphones are programmable and equipped with a set of cheap but powerful embedded sensors, such as accelerometer, digital compass, gyroscope, GPS, microphone, and camera. These sensors can collectively monitor a diverse range of human activities and the surrounding environment. Crowdsensing is a new paradigm which takes advantage of the pervasive smartphones to sense, collect, and analyze data beyond the scale of what was previously possible. With the crowdsensing system, a *crowdsourcer* can recruit smartphone *users* to provide sensing service. Existing crowdsensing applications and systems lack good incentive mechanisms that can attract more user participation. To address this issue, we design incentive mechanisms for crowdsensing. We consider two system models: the crowdsourcer-centric model where the crowdsourcer provides a reward shared by participating users, and the user-centric model where users have more control over the payment they will receive. For the crowdsourcer-centric model, we design an incentive mechanism using a Stackelberg game, where the crowdsourcer is the leader while the users are the followers. We show how to compute the *unique Stackelberg Equilibrium*, at which the utility of the crowdsourcer is maximized, and none of the users can improve its utility by unilaterally deviating from its current strategy. For the user-centric model, we design an *auction-based incentive mechanism*, which is *computationally efficient*, *individually rational*, *profitable*, and *truthful*. Through extensive simulations, we evaluate the performance and validate the theoretical properties of our incentive mechanisms.**

***Index Terms*—Crowdsourcing, crowdsensing, incentive mechanism, Stackelberg game**

1. Introduction

The past few years have witnessed the proliferation of smartphones in people's daily lives. With the advent of 4G networks and more powerful processors, the needs for laptops in particular have begun to fade. Smartphone sales passed PCs for the first time in the final quarter of 2010 [13]. According to the International Data Corporation (IDC), there were more than 1.3 billion smartphones shipped worldwide in 2014 [17]. It is estimated that the smartphone users worldwide will total 2.5 billion in 2015 [41].

Nowadays, smartphones are programmable and equipped with a set of cheap but powerful embedded sensors, such as accelerometer, digital compass, gyroscope, GPS, microphone, and camera. These sensors can collectively monitor a diverse range of human activities and the surrounding environment. Smartphones are undoubtedly revolutionizing many sectors of our life, including social networks, environmental monitoring, business, healthcare, and transportation [21].

If all the smartphones on the planet together constitute a single sensing network, it would form the largest sensing network ever. One can leverage millions of personal smartphones and a near-pervasive wireless network infrastructure to sense, collect, and analyze data far beyond the scale of what was possible before, without the need to deploy thousands of static sensors. This new paradigm is commonly referred as *crowdsensing*.

Realizing the great potential of crowdsensing, many researchers have developed various applications and systems, such as Sensorly [36] for making cellular/WiFi network coverage maps, Nericell [27] and VTrack [42] for providing traffic information, PIER [28] for calculating personalized environmental impact and exposure, and Ear-Phone [33] for creating noise maps. For more details on crowdsensing applications, we refer interested readers to the survey paper [21].

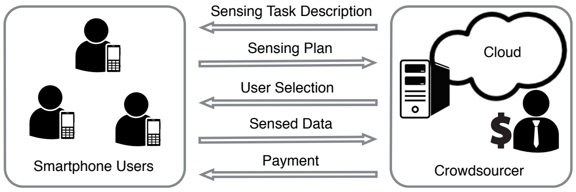


Fig. 1. Crowdsensing system

In the crowdsensing system, as shown in Fig. 1, there is a *crowdsourcer* and a large group of smartphone *users* connected with the crowdsourcer via the cloud. These smartphone users act as sensing service providers. The crowdsourcer recruits smartphone users to provide sensing services.

Although there are many applications and systems based on crowdsensing [27, 28, 33, 36, 42], most of them require voluntary participation. While participating in a crowdsensing task, smartphone users consume their own resources such as battery and computing power. In addition, users also expose themselves to potential privacy threats by sharing their sensed data with location tags. Therefore *a user would not be interested in participating in crowdsensing, unless it receives a satisfying reward to compensate its resource consumption and potential privacy breach*. Without adequate user participation, it is impossible for the crowdsensing applications to achieve good service quality, since sensing services are truly dependent on users' sensed data. While many researchers have developed

This is an extended and enhanced version of the paper [45] that appeared in ACM MobiCom 2012. Yang is with Colorado School of Mines, Golden, CO 80401. Xue is with Arizona State University, Tempe, AZ 85287. Tang is with Syracuse University, Syracuse, NY 13244. Email: djyang@mines.edu, xue@asu.edu, jtang02@syr.edu. This research was supported in part by NSF grants 1217611, 1218203, 1420881, 1421685, and 1444059, and ARO grants W911NF-12-1-0470 and W911NF-09-1-0467. The information reported here does not reflect the position or the policy of the federal government.

different crowdsensing applications [8, 22], they either do not consider the design of incentive mechanisms or have neglected some critical properties of incentive mechanisms. To fill this void, we design several incentive mechanisms to motivate users to participate in crowdsensing applications.

We consider two types of incentive mechanisms for a crowdsensing system: *crowdsourcer-centric incentive mechanisms* and *user-centric incentive mechanisms*. In a crowdsourcer-centric incentive mechanism, the crowdsourcer has the absolute control over the total payment to users, and users can only tailor their actions to cater for the crowdsourcer. Whereas in a user-centric incentive mechanism, the roles of the crowdsourcer and users are reversed. To assure itself of the bottom-line benefit, each user announces a *reserve price*, the lowest price at which it is willing to sell a service. The crowdsourcer then selects a subset of users and pay each of them an amount that is no lower than the user's reserve price.

A. Summary of Key Contributions

The following is a list of our main contributions.

- We design incentive mechanisms for crowdsensing, a new sensing paradigm that takes advantage of the pervasive smartphones to scale up the sensed data collection and analysis to a level of what was previously impossible.
- We consider two system models from two different perspectives: the crowdsourcer-centric model where the crowdsourcer provides a fixed reward to participating users, and the user-centric model where users can have their reserve prices for the sensing service.
- For the crowdsourcer-centric model, we design an incentive mechanism using a Stackelberg game. We present an efficient algorithm to compute the *unique Stackelberg Equilibrium*, at which the utility of the crowdsourcer is maximized, and none of the users can improve its utility by unilaterally deviating from its current strategy.
- For the user-centric model, we design an auction-based incentive mechanism, which is computationally efficient, individually-rational, profitable, and truthful.

B. Paper Organization

The remainder of this paper is organized as follows. We first discuss related work in Section 2. In Section 3, we describe the crowdsensing system models, including both the crowdsourcer-centric model and the user-centric model. We then present our incentive mechanisms for these two models in Sections 4 and 5, respectively. We evaluate the performance in Section 6. We conclude this paper in Section 7.

2. Related Work

As one of the first papers, *SenseMart* [5] discussed the issue of incentives in the design of sensing data exchange mechanisms and raised several challenging questions. There are some studies on design recruitment/incentive mechanisms for participatory sensing, which is similar to crowdsensing [8, 11, 22, 26]. In [34], Reddy *et al.* developed recruitment frameworks to enable the crowdsourcer to identify well-suited participants for sensing services. However, they focused only on the user selection, not the incentive mechanism design. In [8], Danezis *et al.* developed a sealed-bid second-price auction to motivate user participation. However, the utility of the crowdsourcer was neglected in the design of the auction. In [22], Lee and Hoh designed and evaluated a reverse auction based dynamic price incentive mechanism, where users can sell their sensed data to the service provider with users' claimed bid prices. However, the authors failed to consider the truthfulness in the design of the mechanism. In [11], Duan *et al.* studied two applications, data acquisition and distributed computing. For data acquisition, they considered a threshold revenue model, where a certain number of smartphone users are required to successfully build the data base. The total reward is shared equally among all participating users. For distributed computing, they designed a contract-based mechanism to decide different task-reward combinations for heterogeneous users. In addition to incentives, Li and Cao [23] considered the privacy protection in the incentive mechanism design. Different objectives have also been considered. For example, Koutsopoulos [19] developed a randomized incentive mechanism to minimize the total payment to the participating users while guaranteeing certain quality of service level. Assuming that the cost distribution is known, Luo *et al.* [26] designed an all-pay auction based incentive mechanism such that the expected profit is maximized and the individual rationality is satisfied. The aforementioned mechanisms are all off-line, the authors in [48] and [50] designed online incentive mechanisms where the crowdsourcer makes decisions instantly upon the user's arrival. In this paper, we do not require the knowledge of the cost distribution, and focus on designing deterministic off-line incentive mechanisms.

Through experiments with real data, Musthag *et al.* [29] empirically compared three different incentive mechanisms and revealed several interesting observations which are beneficial for designing incentive mechanisms with low-cost, high compliance, and high data quality. All the studied mechanisms assume that users' decisions are independent of each other and do not affect others' received rewards. However, the focus of this paper is on designing incentive mechanisms while considering users' strategic decisions.

There are also many studies investigating the incentive issues in a broader area, crowdsourcing [16, 18, 31, 40, 49]. Zhang and van der Schaar [49] proposed reputation-based incentive mechanisms for crowdsourcing, where participants will earn reputations upon the completion of a task. However, Silberman *et al.* [38] showed the importance of monetary reward compared to other incentives in the crowdsourcing system, as most participants report that they do not take crowdsourcing tasks for fun or kill time. Kamar and Horvitz [18] designed incentive mechanisms for *consensus tasks* that have correct answers to incentivize users for reporting true information. They introduced a novel payment rule, called consensus prediction rule, for evaluating the users' reports in order to determine the payments. Nath *et al.* [31] focused on incentive mechanisms design to minimize the total cost or minimize the total time for executing the task. However, they considered sybilproofness, budget balance, contribution rationality, collapse-proofness, but not truthfulness. Goel *et al.* [16] developed a truthful mechanism, TM-Uniform, for crowdsourcing markets with a budget constraint. They proved that TM-Uniform is budget feasible, individually rational, truthful, and is 3-approximate compared to the optimum solution. However, they assumed that each user is only allowed to work on one task. Therefore, the task assignment process is basically a matching between tasks and users. Using a different approach, Singla and Krause [40] presented a novel, no-regret posted price mechanism in stochastic online settings. Different from our models, in a stochastic online setting, users arrive one at a time, and the system needs to make the decision about the user selection and payment while users are arriving.

Incentive mechanism design was also studied for other networking problems, such as spectrum trading [15, 43, 46, 52], routing [51], and cooperative communications [3, 44]. However none of them can be directly applied to crowdsensing applications, as they all considered properties specifically pertain to the studied problems. Note that the Stackelberg game model in [3] is similar to our crowdsourcer-centric model, but consists of multiple leaders and multiple Stackelberg equilibria.

3. System Model and Problem Formulation

We use Fig. 1 to aid our description of the crowdsensing system. The system consists of a *crowdsourcer*, which resides in the cloud and consists of multiple sensing servers, and many smartphone *users*, which are connected to the crowdsourcer via the cloud. As a potential future direction, the multi-crowdsourder model might be studied using the similar approach as in [6, 47]. The crowdsourcer first publicizes the sensing tasks. Assume that there is a set $\mathcal{U} = \{1, 2, \ldots, n\}$ of smartphone users interested in participating in crowdsensing after reading the sensing task description, where $n \geq 2$. A user participating in crowdsensing will incur a *cost*, to be elaborated later. Therefore it expects a *payment* in return for its service. Taking cost and payment into consideration, each user makes its own sensing plan, which could be the sensing time or the reserve price for selling its sensed data, and submits it to the crowdsourcer. After collecting the sensing plans from users, the crowdsourcer computes the payment for each user and sends the payments to the users. The chosen users will conduct the sensing tasks and send the sensed data to the crowdsourcer. This completes the crowdsensing process.

The crowdsourcer is only interested in maximizing its own utility. Since smartphones are owned by different individuals, it is reasonable to assume that users are *selfish* but *rational*. Hence each user only wants to maximize its own utility, and will not participate in crowdsensing unless there is sufficient incentive. The focus of this paper is on the design of incentive mechanisms that are simple, scalable, and have provably good properties. Other issues in the design and implementation of the whole crowdsensing system is out of the scope of this paper. Please refer to MAUI [7] for energy saving issues, PRISM [9] for application developing issues, and PEPSI [10] and TP [35] for privacy issues.

We study two models: *crowdsourcer-centric* and *user-centric*. In the crowdsourcer-centric model, the sensing plan of an interested user is in the form of its sensing time. A user participating in crowdsensing will earn a payment that is no lower than its cost. However, it needs to compete with other users for a fixed total payment. In the user-centric model, each user asks for a price for its service. If selected, the user will receive a payment that is no lower than its asked price. Unlike the crowdsourcer-centric model, the total payment is not fixed for the user-centric model. Hence, the users have more control over the payment in the user-centric model. These two models address two different but complementary scenarios. The crowdsourcer-centric model is for the scenario where users' contribution to the crowdsensing application can be modeled as a continous variable. Whereas, the user-centric model is for the scenario where the whole crowdsesning task can be divided into small individual tasks.

Table I lists frequently used notations.

TABLE I
Frequently used notations

Notation	Description
$\mathcal{U}, \mathcal{U}_{-i}$	set of users and set of users except user i
n	number of users
R	reward of the crowdsourcer
t_i, t, t_{-i}	sensing time/strategy of user i, strategy profile of all users, strategy profile excluding user i's strategy
κ_i	cost unit of user i
$\beta_i(t_{-i})$	best response of user i given t_{-i}
$\bar{u}_i, \bar{u}_0$	utility function of user i and the crowdsourcer in the crowdsourcer-centric model
λ	system parameter in $\bar{u}_0$
Γ, Γ_i, τ_j	set of tasks, set of user i's tasks and task
m	number of tasks
ν_j	value of task τ_j
c_i, b_i	cost and bid of user i
$\mathcal{S}$	set of selected users
p_i	payment to user i
$v(\mathcal{S})$	total value of the tasks by $\mathcal{S}$
$\tilde{u}_i, \tilde{u}_0$	utility function of user i and the crowdsourcer in the user-centric model

A. Crowdsourcer-Centric Model

In this model, there is only one sensing task. The crowdsourcer announces a total reward $R > 0$, motivating n users to participate in crowdsensing, while each user decides its level of participation based on the reward.

The *sensing plan of user* i is represented by $t_i \geq 0$, the time duration it is willing to provide the sensing service. We assume that each participating user is available during the task, and all times in this period are equally valuable to the crowdsourcer. Our results can be easily extended to the model where the crowdsourcer weighs users' data differently. By setting $t_i = 0$, user i indicates that it will not participate in crowdsensing. The *sensing cost of user* i is $\kappa_i t_i$, where $\kappa_i \in \Theta$ is its *unit cost*, and $\Theta = \{\theta_1, \theta_2, \ldots, \theta_l\}$ is the set of unit costs. We assume that the crowdsourcer knows Θ and the probability distribution of users with unit cost θ_j for each $\theta_j \in \Theta$. This can be learnt from analyzing the historical data. Let n_j denote the number of users with unit cost θ_j for each $\theta_j \in \Theta$. Thus $n = \sum_{\theta_j \in \Theta} n_j$. Assume that the reward received by user i is proportional to

t_i. Then the *utility of user i* is

$$\bar{u}_i = \frac{t_i}{\sum_{j\in\mathcal{U}} t_j} R - t_i\kappa_i, \tag{3.1}$$

i.e., reward minus cost. Note that our results do not straightforwardly extend to the nonlinear cost functions. Based on the above user utility function, all users with the same unite cost would choose the same sensing plan. The *utility of the crowdsourcer* is

$$\bar{u}_0 = g(\tilde{t}_1, \tilde{t}_2, \ldots, \tilde{t}_l; n_1, n_2, \ldots, n_l) - R, \tag{3.2}$$

where $\tilde{t}_j$ is the sensing time of users with unit cost θ_j for each $\theta_j \in \Theta$, and $g(\tilde{t}_1, \tilde{t}_2, \ldots, \tilde{t}_l; n_1, n_2, \ldots, n_l)$ is the crowdsourcer's valuation function of users' sensing time. We will later show that users with the same unit cost choose the same sensing time, i.e., $t_i = t_j$ if $\kappa_i = \kappa_j$. We assume that $g(0, \ldots, 0; 0, \ldots, 0) = 0$ and $g(\tilde{t}_1, \tilde{t}_2, \ldots, \tilde{t}_l; n_1, n_2, \ldots, n_l)$ is a strictly concave function in variables $\tilde{t}_1, \tilde{t}_2, \ldots, \tilde{t}_l$ for any fixed $n_1, n_2, \ldots, n_l$ and also monotonically increasing in each $\tilde{t}_j$ for each $\theta_j \in \Theta$. This assumption is realistic and general, which has been adopted in the literature [24, 25].

Under this model, the objective of the crowdsourcer is to decide the optimal value of R so as to maximize (3.2), while each user $i \in \mathcal{U}$ selfishly decides its sensing time t_i to maximize (3.1) for the given value of R. Since no rational user is willing to provide service for a negative utility, user i shall set $t_i = 0$ when $R \le \kappa_i \sum_{j\neq i\in\mathcal{U}} t_j$.

B. User-Centric Model

In this model, the crowdsourcer announces a set $\Gamma = \{\tau_1, \tau_2, \ldots, \tau_m\}$ of *tasks* for the users to select. Each $\tau_j \in \Gamma$ has a *value $\nu_j > 0$ to the crowdsourcer*. Each user i selects a subset of tasks $\Gamma_i \subseteq \Gamma$ based on its preference. Accordingly, user i also has an *associated cost c_i*, which is private and only known to itself. User i then submits the task-bid pair (Γ_i, b_i) to the crowdsourcer, where b_i, called user i's *bid*, is the *reserve price* user i wants to sell its service for. Note that it is possible that $\Gamma_i \cap \Gamma_j \neq \emptyset$ for two users i and j. This is because users select their task set based on their own schedules and arrangements. Take the Cellular Signal Coverage application [36] as an example, where the tasks are to measure the cellular signals around specific locations. Users may select their sets of tasks based on their daily routes from home to work. Upon receiving the task-bid pairs from all the users, the crowdsourcer selects a subset $\mathcal{S}$ of users as *winners* and determines the *payment p_i* for each winning user i. The *utility of user i* is

$$\tilde{u}_i = \begin{cases} p_i - c_i, & \text{if } i \in \mathcal{S}, \\ 0, & \text{otherwise.} \end{cases} \tag{3.3}$$

The *utility of the crowdsourcer* is

$$\tilde{u}_0 = v(\mathcal{S}) - \sum_{i\in\mathcal{S}} p_i, \tag{3.4}$$

where $v(\mathcal{S}) = \sum_{\tau_j\in\cup_{i\in\mathcal{S}}\Gamma_i} \nu_j$. The union operation on the task sets is to avoid duplications. Note that $v(\mathcal{S})$ can be generalized to other functions, as shown in Section 5.

For the user-centric model, we aim to design an incentive mechanism satisfying the following four desirable properties:

- **Computational Efficiency**: The outcome of the auction can be computed in polynomial time.
- **Individual Rationality**: Each participating user will have a non-negative utility by submitting its true valuation.
- **Profitability**: The value brought by the winners should be no less than the total payment paid to them.
- **Truthfulness**: No bidder can improve its utility by submitting a bid different from its true valuation (which is cost in this paper), no matter what others submit.

The importance of the first three properties is obvious, because they together assure the feasibility of the incentive mechanism. Being truthful, the incentive mechanism can eliminate the fear of market manipulation and the overhead of strategizing over others for the participating users.

4. Incentive Mechanism for Crowdsourcer-Centric Model

We model the crowdsourcer-centric incentive mechanism as a *Stackelberg game* [14], which we call the IMCC game. There are two stages in this mechanism: In the first stage, the crowdsourcer announces its reward R; in the second stage, each user strategizes its sensing time to maximize its own utility. Therefore the crowdsourcer is the *leader* and the users are the *followers* in this Stackelberg game. Meanwhile, both the crowdsourcer and the users are *players*. The *strategy of the crowdsourcer* is its reward R. The *strategy of user i* is its working time t_i. Let $t = (t_1, t_2, \ldots, t_n)$ denote the *strategy profile* consisting of all users' strategies. Let t_{-i} denote the strategy profile excluding t_i. As a notational convention, we write $t = (t_i, t_{-i})$. Note that this incentive mechanism is similar to the lottery-based mechanisms in [24, 25]. The difference lies in the assumption that a large number of users are considered and thus each user's strategy is negligible comparing to the total strategy. In addition, the objective in [24, 25] is to choose the mechanism parameters to achieve the socially optimal level. While in our model, the crowdsourcer is only interested in maximizing its own utility.

Note that the second stage of the IMCC game can be considered a non-cooperative game, which we call the Sensing Time Determination (STD) game. Given the IMCC game formulation, we are interested in the following questions:

Q1: For a given reward R, is there a set of stable strategies in the STD game such that no user has anything to gain by unilaterally changing its current strategy?

Q2: If the answer to Q1 is affirmative, is the stable strategy set unique? When it is unique, users will be guaranteed to select the strategies in the same stable strategy set.

Q3: How can the crowdsourcer select the value of R to maximize its utility in (3.2)?

The stable strategy set in Q1 corresponds to the concept of *Nash Equilibrium* (NE) in game theory [14].

Definition 1 (Nash Equilibrium): A set of strategies $(t_1^{ne}, t_2^{ne}, \ldots, t_n^{ne})$ is a Nash Equilibrium of the STD game if for any user i,

$$\bar{u}_i(t_i^{ne}, t_{-i}^{ne}) \ge \bar{u}_i(t_i, t_{-i}^{ne}),$$

for any $t_i \ge 0$, where $\bar{u}_i$ is defined (3.1).

The *existence* of an NE is important, since an NE strategy profile is stable (no player has an incentive to make a unilateral change) whereas a non-NE strategy profile is unstable. The *uniqueness* of NE enables the crowdsourcer to predict the behaviors of the users and thus to select the optimal value of R. Therefore the answer to Q3 depends heavily on those to Q1 and Q2. The optimal solution computed in Q3 together with the NE of the STD game constitutes a solution to the IMCC game, called *Stackelberg Equilibrium.*

In Section 4-A, we prove that **for any given $R > 0$, the STD game has a unique NE**, and present an efficient algorithm for computing the NE. In Section 4-B, we prove that **the IMCC game has a unique Stackelberg Equilibrium**, and present an efficient algorithm for computing it.

A. User Sensing Time Determination

We first introduce the concept of *best response strategy.*

Definition 2 (Best Response Strategy): Given t_{-i}, a strategy is user i's best response strategy, denoted by $\beta_i(t_{-i})$, if it maximizes $\bar{u}_i(t_i, t_{-i})$ over all $t_i \ge 0$.

Based on the definition of NE, every user is playing its best response strategy in an NE. From (3.1), we know that $t_i \le \frac{R}{\kappa_i}$ because $\bar{u}_i$ will be negative otherwise. To study the best response strategy of user i, we compute the derivatives of $\bar{u}_i$ with respect to t_i:

$$\frac{\partial \bar{u}_i}{\partial t_i} = \frac{-Rt_i}{(\sum_{j\in\mathcal{U}} t_j)^2} + \frac{R}{\sum_{j\in\mathcal{U}} t_j} - \kappa_i, \tag{4.1}$$

$$\frac{\partial^2 \bar{u}_i}{\partial t_i^2} = -\frac{2R\sum_{j\in\mathcal{U}_{-i}} t_j}{(\sum_{j\in\mathcal{U}} t_j)^3} < 0. \tag{4.2}$$

Since the second-order derivative of $\bar{u}_i$ is negative, the utility $\bar{u}_i$ is a *strictly concave function* in t_i. Hence given any $R > 0$ and any strategy profile t_{-i} of the other users, the best response strategy $\beta_i(t_{-i})$ of user i is unique, if it exists. If the strategy of all other user $j \ne i$ is $t_j = 0$, then user i does not have a best response strategy, as it can have a utility arbitrarily close to R, by setting t_i to a sufficiently small positive number. Thus we are only interested in the case when $\sum_{j\in\mathcal{U}_{-i}} t_j > 0$. Setting the first derivative of $\bar{u}_i$ to 0, we have

$$\frac{-Rt_i}{(\sum_{j\in\mathcal{U}} t_j)^2} + \frac{R}{\sum_{j\in\mathcal{U}} t_j} - \kappa_i = 0. \tag{4.3}$$

Solving for t_i in (4.3), we obtain

$$t_i = \sqrt{\frac{R\sum_{j\in\mathcal{U}_{-i}} t_j}{\kappa_i}} - \sum_{j\in\mathcal{U}_{-i}} t_j. \tag{4.4}$$

If the RHS (right hand side) of (4.4) is positive, it is also the best response strategy of user i, due to the concavity of $\bar{u}_i$. If the RHS of (4.4) is less than or equal to 0, then user i does not participate in the crowdsensing by setting $t_i = 0$ (to avoid a deficit). Hence we have

$$\beta_i(t_{-i}) = \begin{cases} 0, & \text{if } R \le \kappa_i \sum_{j\in\mathcal{U}_{-i}} t_j; \\ \sqrt{\frac{R\sum_{j\in\mathcal{U}_{-i}} t_j}{\kappa_i}} - \sum_{j\in\mathcal{U}_{-i}} t_j, & \text{otherwise.} \end{cases} \tag{4.5}$$

These analyses lead to the following algorithm for computing an NE of the SDT game.

Algorithm 1: Computation of the NE

```
1  Sort the unit costs in $\Theta$, $\theta_1 < \theta_2 < \cdots < \theta_l$;
2  $\Theta_w \leftarrow \emptyset$;
3  Let $j \in [1, l]$ be the smallest such that $\sum_{k=1}^{j} n_k \ge 2$;
4  $j \leftarrow j + 1$;
5  while $j \le l$ and $\theta_j < \frac{\theta_j + \sum_{k=1}^{j-1} n_k\theta_k}{\sum_{k=1}^{j-1} n_k}$ do
6      if $n_j > 0$ then $\Theta_w \leftarrow \Theta_w \cup \{\theta_j\}$ ;
7      $j \leftarrow j + 1$;
8  end
9  $n_0 \leftarrow \sum_{\theta_k\in\Theta_w} n_k$;
10 foreach $\theta_j \in \Theta_w$ do
11     $\tilde{t}_j = \frac{(n_0-1)R}{\sum_{\theta_k\in\Theta_w} n_k\theta_k}\left(1 - \frac{(n_0-1)\theta_j}{\sum_{\theta_k\in\Theta_w} n_k\theta_k}\right)$;
12 end
13 $\mathcal{S} \leftarrow \{i | \kappa_i \in \Theta_w\}$;
14 foreach $i \in \mathcal{U}$ do
15     if $i \in \mathcal{S}$ then $t_i^{ne} = \tilde{t}_j$, such that $\kappa_i = \theta_j$ ;
16     else $t_i^{ne} = 0$ ;
17 end
18 return $t^{ne} = (t_1^{ne}, t_2^{ne}, \ldots, t_n^{ne})$
```

<u>**Remark:**</u> The crowdsourcer needs to compute only Θ_w (Lines 1–12). The rest of Algorithm 1 (Lines 13–18) is for the purpose of proving the existence and uniqueness of the NE. In addition, essentially, $t_i^{ne} = \frac{(|\mathcal{S}|-1)R}{\sum_{j\in\mathcal{S}}\kappa_j}\left(1 - \frac{(|\mathcal{S}|-1)\kappa_i}{\sum_{j\in\mathcal{S}}\kappa_j}\right)$ for any $i \in \mathcal{S}$ by substituting θ_k with corresponding κ_j in $\tilde{t}_j$.

The following theorem shows that Algorithm 1 computes the unique NE of the STD game.

Theorem 1: Let $R > 0$ be given. Let $\bar{t} = (\bar{t}_1, \bar{t}_2, \ldots, \bar{t}_n)$ be the strategy profile of an NE for the STD game, and let $\bar{\mathcal{S}} = \{i \in \mathcal{U} | \bar{t}_i > 0\}$. We have

1) $|\bar{\mathcal{S}}| \ge 2$.
2) $\bar{t}_i = \begin{cases} 0, & \text{if } i \notin \bar{\mathcal{S}}; \\ \frac{(|\bar{\mathcal{S}}|-1)R}{\sum_{j\in\bar{\mathcal{S}}}\kappa_j}\left(1 - \frac{(|\bar{\mathcal{S}}|-1)\kappa_i}{\sum_{j\in\bar{\mathcal{S}}}\kappa_j}\right), & \text{otherwise.} \end{cases}$
3) If $\kappa_q \le \max_{j\in\bar{\mathcal{S}}}\{\kappa_j\}$, then $q \in \bar{\mathcal{S}}$.
4) Assume that the users are ordered such that $\kappa_1 \le \kappa_2 \le \cdots \le \kappa_n$. Let h be the largest integer in $[2, n]$ such that $\kappa_h < \frac{\sum_{j=1}^{h}\kappa_j}{h-1}$. Then $\bar{\mathcal{S}} = \{1, 2, \ldots, h\}$.

These statements imply that Algorithm 1 computes the unique NE of the STD game. In addition, the time complexity of Algorithm 1 is $O(n \log n)$.

Proof: We first prove 1). Assume that $|\bar{\mathcal{S}}| = 0$. User 1 can increase its utility from 0 to $\frac{R}{2}$ by unilaterally changing its sensing time from 0 to $\frac{R}{2\kappa_1}$, contradicting the NE assumption. This proves that $|\bar{\mathcal{S}}| \ge 1$. Now assume that $|\bar{\mathcal{S}}| = 1$. This means $\bar{t}_k > 0$ for some $k \in \mathcal{U}$, and $\bar{t}_j = 0$ for all $j \in \mathcal{U}_{-k}$. According to (3.1) the current utility of user k is $R - \bar{t}_k\kappa_k$. User k can increase its utility by unilaterally changing its sensing time from $\bar{t}_k$ to $\frac{\bar{t}_k}{2}$, again contradicting the NE assumption. Therefore $|\bar{\mathcal{S}}| \ge 2$.

We next prove 2). Let $n_0 = |\bar{\mathcal{S}}|$. Since we already proved that $n_0 \ge 2$, we can use the analysis of (4.3) at the beginning of this section, with t replaced by $\bar{t}$, and $\mathcal{S}$ replaced by $\bar{\mathcal{S}}$. Considering that $\sum_{j\in\mathcal{U}} \bar{t}_j = \sum_{j\in\bar{\mathcal{S}}} \bar{t}_j$, we have

$$\frac{-R\bar{t}_i}{(\sum_{j\in\bar{\mathcal{S}}} \bar{t}_j)^2} + \frac{R}{\sum_{j\in\bar{\mathcal{S}}} \bar{t}_j} - \kappa_i = 0, \quad i \in \bar{\mathcal{S}}. \tag{4.6}$$

Summing up (4.6) over the users in $\bar{\mathcal{S}}$ leads to $n_0R - R = \sum_{j\in\bar{\mathcal{S}}} \bar{t}_j \cdot \sum_{j\in\bar{\mathcal{S}}} \kappa_j$. Therefore we have

$$\sum_{j\in\bar{\mathcal{S}}} \bar{t}_j = \frac{(n_0-1)R}{\sum_{j\in\bar{\mathcal{S}}} \kappa_j}. \tag{4.7}$$

Substituting (4.7) into (4.6) and considering $\bar{t}_j = 0$ for any $j \in \mathcal{U} \setminus \bar{\mathcal{S}}$, we obtain the following:

$$\bar{t}_i = \frac{(n_0-1)R}{\sum_{j\in\bar{\mathcal{S}}} \kappa_j} \left(1 - \frac{(n_0-1)\kappa_i}{\sum_{j\in\bar{\mathcal{S}}} \kappa_j}\right) \tag{4.8}$$

for every $i \in \bar{\mathcal{S}}$. This proves 2).

We then prove 3). By definition of $\bar{\mathcal{S}}$, we know that $\bar{t}_i > 0$ for every $i \in \bar{\mathcal{S}}$. From (4.8), $\bar{t}_i > 0$ implies $\frac{(n_0-1)\kappa_i}{\sum_{j\in\bar{\mathcal{S}}} \kappa_j} < 1$. Therefore we have

$$\kappa_i < \frac{\sum_{j\in\bar{\mathcal{S}}} \kappa_j}{|\bar{\mathcal{S}}| - 1}, \forall i \in \bar{\mathcal{S}}, \tag{4.9}$$

and thus

$$\max_{i\in\bar{\mathcal{S}}} \kappa_i < \frac{\sum_{j\in\bar{\mathcal{S}}} \kappa_j}{|\bar{\mathcal{S}}| - 1}. \tag{4.10}$$

Assume that $\kappa_q \leq \max_{j\in\bar{\mathcal{S}}}\{\kappa_j\}$ but $q \notin \bar{\mathcal{S}}$. Since $q \notin \bar{\mathcal{S}}$, we know that $\bar{t}_q = 0$. The first-order derivative of $\bar{u}_q$ with respect to t_q when $t = \bar{t}$ is

$$\frac{R}{\sum_{j\in\bar{\mathcal{S}}} \bar{t}_j} - \kappa_q = \frac{\sum_{j\in\bar{\mathcal{S}}} \kappa_j}{n_0 - 1} - \kappa_q > \max_{i\in\bar{\mathcal{S}}}\{\kappa_i\} - \kappa_q \geq 0. \tag{4.11}$$

This means that user q can increase its utility by unilaterally increasing its sensing time from $\bar{t}_q$, contradicting the NE assumption of $\bar{t}$. This proves 3).

Finally, we prove 4). Statements 1) and 3) imply that $\bar{\mathcal{S}} = \{1, 2, \ldots, q\}$ for some integer q in $[2, n]$. From (4.9), we conclude that $q \leq h$. Assume that $q < h$. Then we have $\kappa_{q+1} < \frac{\sum_{j=1}^{q+1} \kappa_j}{q}$, which implies $\frac{\sum_{j=1}^{q} \kappa_j}{q-1} - \kappa_{q+1} > 0$. Hence the first order derivative of $\bar{u}_{q+1}$ with respect to t_{q+1} when $t = \bar{t}$ is $\frac{\sum_{j=1}^{q} \kappa_j}{q-1} - \kappa_{q+1} > 0$. This contradiction proves $q = h$. Hence we have proved 4), as well as the theorem.

The running time of Algorithm 1 is dominated by sorting, and thus is $O(n \log n)$. ■

B. Crowdsourcer Utility Maximization

According to the above analysis, the crowdsourcer, which is the leader in the Stackelberg game, knows that there exists the unique NE for the users for any given value of R. Hence the crowdsourcer can maximize its utility by choosing the optimal R. Plugging $\tilde{t}_j$ computed by Algorithm 1 into (3.2), we have

$$\bar{u}_0 = g(X_1R, X_2R, \ldots, X_lR; n_1, n_2, \ldots, n_l) - R, \tag{4.12}$$

where

$$X_j = \begin{cases} \frac{(n_0-1)}{\sum_{\theta_k\in\Theta_w} n_k\theta_k} \left(1 - \frac{(n_0-1)\theta_j}{\sum_{\theta_k\in\Theta_w} n_k\theta_k}\right), & \theta_j \in \Theta_w, \\ 0, & \theta_j \notin \Theta_w. \end{cases} \tag{4.13}$$

Theorem 2: There exists the unique Stackelberg Equilibrium (R^*, t^{ne}) in the IMCC game, where R^* is the unique maximizer of the crowdsourcer's utility in (4.12) over $R \in [0, \infty)$, Θ_w and t^{ne} are given by Algorithm 1 with R^*.

Proof: Since $g(\tilde{t}_1, \tilde{t}_2, \ldots, \tilde{t}_l; n_1, n_2, \ldots, n_l)$ is a strictly concave function in variables $\tilde{t}_1, \tilde{t}_2, \ldots, \tilde{t}_l$ for any fixed $n_1, n_2, \ldots, n_l$, $g(X_1R, X_2R, \ldots, X_lR; n_1, n_2, \ldots, n_l)$ is a strictly concave function of R for any fixed $n_1, n_2, \ldots, n_l$. Therefore the utility $\bar{u}_0$ in (4.12) is a strictly concave function of R for $R \in [0, \infty)$. Since the value of $\bar{u}_0$ in (4.12) is 0 for $R = 0$ and goes to $-\infty$ when R goes to ∞, it has a unique maximizer R^* that can be efficiently computed using either bisection or Newton's method [2]. ■

5. Incentive Mechanism for User-Centric Model

Auction theory [20] is the perfect theoretical tool to design incentive mechanisms for the user-centric model. We propose a reverse auction based incentive mechanism for the user-centric model. An *auction* takes as input the *bids* submitted by the users, selects a subset of users as *winners*, and determines the *payment* to each winning user.

A. Auctions Maximizing Crowdsourcer Utility

Our first attempt is to design an incentive mechanism maximizing the utility of the crowdsourcer. Now designing an incentive mechanism becomes an optimization problem, called *User Selection* problem: Given a set $\mathcal{U}$ of users, select a subset $\mathcal{S}$ such that $\tilde{u}_0(\mathcal{S})$ is maximized over all possible subsets. In addition, it is clear that $p_i = b_i$ to maximize $\tilde{u}_0(\mathcal{S})$. The utility $\tilde{u}_0$ then becomes

$$\tilde{u}_0(\mathcal{S}) = v(\mathcal{S}) - \sum_{i\in\mathcal{S}} b_i. \tag{5.1}$$

To make the problem meaningful, we assume that there exists at least one user i such that $\tilde{u}_0(\{i\}) > 0$.

Unfortunately, as shown in [45], it is NP-hard to find the optimal solution to the User Selection problem.

Since it is unlikely to find the optimal subset of users efficiently, we turn our attention to the development of approximation algorithms. To this end, we take advantage of the *submodularity* of the utility function.

Definition 3 (Submodular Function): Let $\mathcal{X}$ be a finite set. A function $f : 2^{\mathcal{X}} \mapsto \mathbb{R}$ is submodular if

$$f(\mathcal{A} \cup \{x\}) - f(\mathcal{A}) \geq f(\mathcal{B} \cup \{x\}) - f(\mathcal{B}),$$

for any $\mathcal{A} \subseteq \mathcal{B} \subseteq \mathcal{X}$ and $x \in \mathcal{X} \setminus \mathcal{B}$, where $\mathbb{R}$ is real number set.

We now prove the submodularity of the utility $\tilde{u}_0$.

Lemma 1: The utility $\tilde{u}_0$ is submodular.

Proof: By Definition 3, we need to show that

$$\tilde{u}_0(\mathcal{S} \cup \{i\}) - \tilde{u}_0(\mathcal{S}) \geq \tilde{u}_0(\mathcal{T} \cup \{i\}) - \tilde{u}_0(\mathcal{T}),$$

for any $\mathcal{S} \subseteq \mathcal{T} \subseteq \mathcal{U}$ and $i \in \mathcal{U} \setminus \mathcal{T}$. It suffices to show that $v(\mathcal{S} \cup \{i\}) - v(\mathcal{S}) \geq v(\mathcal{T} \cup \{i\}) - v(\mathcal{T})$, since the second term in $\tilde{u}_0$ can be subtracted from both sides. Considering $v(\mathcal{S}) = \sum_{\tau_j \in \cup_{i\in\mathcal{S}}\Gamma_i} \nu_j$, we have

$$\begin{aligned} v(\mathcal{S}\cup\{i\}) - v(\mathcal{S}) &= \sum_{\tau_j\in\Gamma_i\setminus\cup_{j\in\mathcal{S}}\Gamma_j} \nu_j && (5.2)\\ &\geq \sum_{\tau_j\in\Gamma_i\setminus\cup_{j\in\mathcal{T}}\Gamma_j} \nu_j && (5.3)\\ &= v(\mathcal{T}\cup\{i\}) - v(\mathcal{T}). && (5.4)\end{aligned}$$

Therefore $\tilde{u}_0$ is submodular. *As a byproduct, we proved that v is submodular as well.* ■

When the objective function is submodular, monotone and non-negative, it is known that a greedy algorithm provides a $(1-1/e)$-approximation [32]. Without monotonicity, Feige *et al.* [12] have also developed constant-factor approximation algorithms. Unfortunately, $\tilde{u}_0$ can be negative.

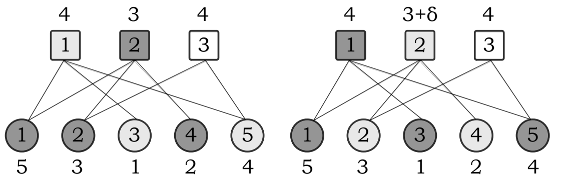
To circumvent this issue, let $f(\mathcal{S}) = \tilde{u}_0(\mathcal{S}) + \sum_{i\in\mathcal{U}} b_i$. It is clear that $f(\mathcal{S}) \geq 0$ for any $\mathcal{S}\subseteq\mathcal{U}$. Since $\sum_{i\in\mathcal{U}} b_i$ is a constant, $f(\mathcal{S})$ is also submodular. In addition, maximizing $\tilde{u}_0$ is equivalent to maximizing f. Therefore we design an auction mechanism based on the algorithm of [12], called Local Search-Based (LSB) auction, as illustrated in Algorithm 2. The mechanism relies on the local-search technique, which greedily searches for a better solution by adding a new user or deleting an existing user whenever possible. It was proved that, for any given constant $\epsilon > 0$, the algorithm can find a set of users $\mathcal{S}$ such that $f(\mathcal{S}) \geq (\frac{1}{3} - \frac{\epsilon}{n})f(\mathcal{S}^*)$, where $\mathcal{S}^*$ is the optimal solution [12].

Algorithm 2: LSB Auction

```
1  $\mathcal{S} \leftarrow \{i\}$, where $i \leftarrow \arg\max_{i\in\mathcal{U}} f(\{i\})$;
2  while there exists a user $i \in \mathcal{U}\setminus\mathcal{S}$ such that
     $f(\mathcal{S}\cup\{i\}) > (1+\frac{\epsilon}{n^2})f(\mathcal{S})$ do
3      $\mathcal{S} \leftarrow \mathcal{S}\cup\{i\}$;
4  end
5  if there exists a user $i \in \mathcal{S}$ such that
     $f(\mathcal{S}\setminus\{i\}) > (1+\frac{\epsilon}{n^2})f(\mathcal{S})$ then
6      $\mathcal{S} \leftarrow \mathcal{S}\setminus\{i\}$; go to Line 2;
7  end
8  if $f(\mathcal{U}\setminus\mathcal{S}) > f(\mathcal{S})$ then $\mathcal{S} \leftarrow \mathcal{U}\setminus\mathcal{S}$ ;
9  foreach $i \in \mathcal{U}$ do
10     if $i \in \mathcal{S}$ then $p_i \leftarrow b_i$ ;
11     else $p_i \leftarrow 0$ ;
12 end
13 return $(\mathcal{S}, p)$
```

Now we analyze the LSB auction using the four desirable properties described in Section 3-B as performance metrics.

- **Computational Efficiency**: The running time of the Local Search Algorithm is $O(\frac{1}{\epsilon}n^3 m\log m)$ [12], where evaluating the value of f takes $O(m)$ time and $|\mathcal{S}| \leq m$. Hence LSB is computationally efficient.
- **Individual Rationality**: The crowdsourcer pays what the winners bid. Hence LSB is individually rational.
- **Profitability**: Due to the assumption that there exists at least one user i such that $\tilde{u}_0(\{i\}) > 0$ and the fact that $f(\mathcal{S})$ strictly increases in each iteration, we guarantee that $\tilde{u}_0(\mathcal{S}) > 0$, which implies that LSB is profitable.
- **Truthfulness**: We use an example in Fig. 2 to show that the LSB auction is *not* truthful. In this example, $\mathcal{U} = \{1,2,3\}$, $\Gamma = \{\tau_1,\tau_2,\tau_3,\tau_4,\tau_5\}$, $\Gamma_1 = \{\tau_1,\tau_3,\tau_5\}$, $\Gamma_2 = \{\tau_1,\tau_2,\tau_4\}$, $\Gamma_3 = \{\tau_2,\tau_5\}$, $c_1 = 4$, $c_2 = 3$, $c_3 = 4$. Squares represent users, and disks represent tasks. The number above user i denotes its bid b_i. The number below task τ_j denotes its value ν_j. For example, $b_1 = 4$ and $\nu_3 = 1$. We also assume that $\epsilon = 0.1$.
  We first consider the case where users bid truthfully. Since $f(\{1\}) = v(\Gamma_1) - b_1 + \sum_{i=1}^3 b_i = (5+1+4) - 4 + (4+3+4) = 17$, $f(\{2\}) = 18$, and $f(\{3\}) = 14$, user 2 is first selected. Since $f(\{2,1\}) = v(\Gamma_2\cup\Gamma_1) - (b_2+b_1) + \sum_{i=1}^3 b_i = 19 > \left(1+\frac{0.1}{3^2}\right)f(\{2\}) = 18.2$, user 1 is then selected. The auction terminates here because the current value of f cannot be increased by a factor of $(1+\frac{0.1}{9})$ via either adding a user (that has not been selected) or removing a user (that has been selected). In addition, we have $p_1 = b_1 = 4$ and $p_2 = b_2 = 3$.
  We now assume that user 2 lies by bidding $3+\delta$, where $1 \leq \delta < 1.77$. Since $f(\{1\}) = 17+\delta$, $f(\{2\}) = 18$, and $f(\{3\}) = 14+\delta$, user 1 is first selected. Since $f(\{1,2\}) = 19 > \left(1+\frac{0.1}{9}\right)f(\{1\})$, user 2 is then selected. The auction terminates here because the current value of f cannot be increased by a factor of $(1+\frac{0.1}{9})$ via either adding a user or removing a user. Note that user 2 increases its payment from 3 to $3+\delta$ by lying about its cost.



(a) Users bid truthfully.

(b) User 2 lies by bidding $3+\delta$, where $1 \leq \delta < 1.77$.

Fig. 2. Example showing the untruthfulness of the Local Search-Based Auction mechanism, where $\mathcal{U} = \{1,2,3\}$, $\Gamma = \{\tau_1,\tau_2,\tau_3,\tau_4,\tau_5\}$, $\Gamma_1 = \{\tau_1,\tau_3,\tau_5\}$, $\Gamma_2 = \{\tau_1,\tau_2,\tau_4\}$, $\Gamma_3 = \{\tau_2,\tau_5\}$. Squares represent users. Disks represent tasks. The number above user i denotes its bid b_i. The number below task τ_j denotes its value ν_j. We also assume that $\epsilon = 0.1$.

B. IMCU Auction

Although the LSB auction mechanism is designed to approximately maximize the crowdsourcer's utility, the failure of guaranteeing truthfulness makes it less attractive. Since our ultimate goal is to design an incentive mechanism that motivates smartphone users to participate in crowdsensing while preventing any user from rigging its bid to manipulate the market, we need to settle for a trade-off between utility maximization and truthfulness. Our focus is to design an incentive mechanism that satisfies all of the four desirable properties, even at the cost of sacrificing the crowdsourcer utility. One possible direction is to make use of the off-the-shelf results on the *budgeted mechanism* design [4, 39]. In the budgeted mechanism, the payment paid to the winners is a constraint instead of a factor in the objective function. One intuitive idea is to plug different values of the budget into the budgeted mechanism and select the one giving the largest utility. However, this can potentially destroy the truthfulness of the incentive mechanism.

In this section, we present a novel auction mechanism that satisfies all four desirable properties. The design rationale relies on Myerson's well-known characterization [30].

Theorem 3: ([39, Theorem 2.1]) An auction mechanism is truthful if and only if:

- The selection rule is monotone: If user i wins the auction by bidding b_i, it also wins by bidding $b'_i \leq b_i$;
- Each winner is paid the critical value: User i would not win the auction if it bids higher than this value.

1) Auction Design: Based on Theorem 3, we design our auction mechanism in this section, which is called IMCU auction. Illustrated in Algorithm 3, the IMCU auction mechanism consists of two phases: the *winner selection* phase and the *payment determination* phase.

Algorithm 3: IMCU Auction

```
1  // Phase 1: Winner selection
2  $\mathcal{S} \leftarrow \emptyset$, $i \leftarrow \arg\max_{j \in \mathcal{U}} (v_j(\mathcal{S}) - b_j)$;
3  while $b_i < v_i$ and $\mathcal{S} \neq \mathcal{U}$ do
4      $\mathcal{S} \leftarrow \mathcal{S} \cup \{i\}$;
5      $i \leftarrow \arg\max_{j \in \mathcal{U} \setminus \mathcal{S}} (v_j(\mathcal{S}) - b_j)$;
6  end
7  // Phase 2: Payment determination
8  foreach $i \in \mathcal{U}$ do $p_i \leftarrow 0$;
9  foreach $i \in \mathcal{S}$ do
10     $\mathcal{U}_{-i} \leftarrow \mathcal{U} \setminus \{i\}$, $\mathcal{T} \leftarrow \emptyset$;
11     repeat
12         $i_j \leftarrow \arg\max_{j \in \mathcal{U}_{-i} \setminus \mathcal{T}} (v_j(\mathcal{T}) - b_j)$;
13         $p_i \leftarrow \max\{p_i, \min\{v_i(\mathcal{T}) - (v_{i_j}(\mathcal{T}) - b_{i_j}), v_i(\mathcal{T})\}\}$;
14         $\mathcal{T} \leftarrow \mathcal{T} \cup \{i_j\}$;
15     until $b_{i_j} \geq v_{i_j}$ or $\mathcal{T} = \mathcal{U}_{-i}$;
16     if $\mathcal{T} = \mathcal{U}_{-i}$ then $p_i \leftarrow \max\{p_i, v_i(\mathcal{T})\}$;
17 end
18 return $(\mathcal{S}, p)$
```

The winner selection phase follows a greedy approach: Users are essentially sorted according to the difference of their marginal values and bids. Given the selected users $\mathcal{S}$, the marginal value of user i is $v_i(\mathcal{S}) = v(\mathcal{S} \cup \{i\}) - v(\mathcal{S})$. In this sorting the $(i+1)$th user is the user j such that $v_j(\mathcal{S}_i) - b_j$ is maximized over $\mathcal{U} \setminus \mathcal{S}_i$, where $\mathcal{S}_i = \{1, 2, \ldots, i\}$ and $\mathcal{S}_0 = \emptyset$. We use v_i instead of $v_i(\mathcal{S}_{i-1})$ to simplify the notation. Considering the submodularity of v, this sorting implies that

$$v_1 - b_1 \geq v_2 - b_2 \geq \cdots \geq v_n - b_n. \tag{5.5}$$

The set of winners are $\mathcal{S}_L = \{1, 2, \ldots, L\}$, where $L \leq n$ is the largest index such that $v_L - b_L > 0$.

In the payment determination phase, we compute the payment p_i for each winner $i \in \mathcal{S}$. To compute the payment for user i, we sort the users in $\mathcal{U}_{-i}$ similarly,

$$v'_{i_1} - b_{i_1} \geq v'_{i_2} - b_{i_2} \geq \cdots \geq v'_{i_{n-1}} - b_{i_{n-1}}, \tag{5.6}$$

where $v'_{i_j} = v(\mathcal{T}_{j-1} \cup \{i_j\}) - v(\mathcal{T}_{j-1})$ denotes the marginal value of the jth user and $\mathcal{T}_j$ denotes the first j users according to this sorting over $\mathcal{U}_{-i}$ and $\mathcal{T}_0 = \emptyset$. The marginal value of user i at position j is $v_{i(j)} = v(\mathcal{T}_{j-1} \cup \{i\}) - v(\mathcal{T}_{j-1})$. Let K denote the position of the last user $i_j \in \mathcal{U}_{-i}$, such that $b_{i_j} < v'_{i_j}$. For each position j in the sorting, we compute the maximum price that user i can bid such that i can be selected instead of user at jth place. We repeat this until the position after the last winner in $\mathcal{U}_{-i}$. In the end we set the value of p_i to the maximum of these $K+1$ prices.

2) A Walk-Through Example: We use the example in Fig. 3 to illustrate how the IMCU auction works.

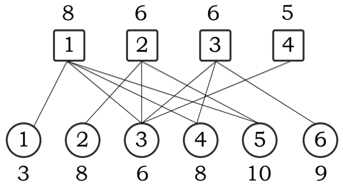


Fig. 3. Illustration for IMCU

Winner Selection:

◇ $\mathcal{S} = \emptyset$: $v_1(\emptyset) - b_1 = (v(\emptyset \cup \{1\}) - v(\emptyset)) - b_1$ $= ((\nu_1 + \nu_3 + \nu_4 + \nu_5) - 0) - 8 = ((3 + 6 + 8 + 10) - 0) - 8 = 19$, $v_2(\emptyset) - b_2 = (v(\emptyset \cup \{2\}) - v(\emptyset)) - b_2 = 18$, $v_3(\emptyset) - b_3 = 17$, and $v_4(\emptyset) - b_4 = 1$.

◇ $\mathcal{S} = \{1\}$: $v_2(\{1\}) - b_2 = (v(\{1\} \cup \{2\}) - v(\{1\})) - b_2 = (35 - 27) - 6 = 2$, $v_3(\{1\}) - b_3 = (v(\{1\} \cup \{3\}) - v(\{1\})) - b_3 = 3$, and $v_4(\{1\}) - b_4 = -5$.

◇ $\mathcal{S} = \{1, 3\}$: $v_2(\{1, 3\}) - b_2 = (v(\{1, 3\} \cup \{2\}) - v(\{1, 3\})) - b_2 = 2$ and $v_4(\{1, 3\}) - b_4 = -5$.

◇ $\mathcal{S} = \{1, 3, 2\}$: $v_4(\{1, 3, 2\}) - b_4 = -5$.

During the payment determination phase, we directly give winners when user i is excluded from the consideration, due to the space limitations. Also recall that $v'_{i_j} > b_{i_j}$ for $j \leq K$ and $v'_{i_j} \leq b_{i_j}$ for $j \geq K + 1$.

Payment Determination:

◇ p_1: Winners are {2, 3}.
$v_1(\emptyset) - (v_2(\emptyset) - b_2) = 9$, $v_1(\{2\}) - (v_3(\{2\}) - b_3)) = 0$, $v_1(\{2, 3\}) = 3$. Thus $p_1 = 9$.

◇ p_2: Winners are {1, 3}.
$v_2(\emptyset) - (v_1(\emptyset) - b_1) = 5$, $v_2(\{1\}) - (v_3(\{1\}) - b_3)) = 5$, $v_2(\{1, 3\}) = 8$. Thus $p_2 = 8$.

◇ p_3: Winners are {1, 2}.
$v_3(\emptyset) - (v_1(\emptyset) - b_1) = 4$, $v_3(\{1\}) - (v_2(\{1\}) - b_2)) = 7$, $v_3(\{1, 2\}) = 9$. Thus $p_3 = 9$.

3) Properties of IMCU: We will prove the computational efficiency (Lemma 2), the individual rationality (Lemma 3), the profitability (Lemma 4), and the truthfulness (Lemma 5) of the IMCU auction in the following.

Lemma 2: IMCU is computationally efficient.

Proof: Finding the user with maximum marginal value takes $O(nm)$ time, where computing the value of v_i takes $O(m)$ time. Since there are m tasks and each winner should contribute at least one new task to be selected, the number of winners is at most m. Hence, the while-loop (Lines 3–6) thus takes $O(nm^2)$ time. In each iteration of the for-loop (Lines 9–17), a process similar to Lines 3–6 is executed. Hence the running time of the whole auction is dominated by this for-loop, which is bounded by $O(nm^3)$. ■

Note that the running time of the IMCU Auction, $O(nm^3)$, is very conservative. In addition, m is much less than n in practice, which makes the running time of the IMCU Auction dominated by n.

Before turning our attention to the proofs of the other three properties, we would like to make some critical observations: 1) $v_{i(j)} \geq v_{i(j+1)}$ for any j due to the submodularity of v; 2) $\mathcal{T}_j = \mathcal{S}_j$ for any $j < i$; 3) $v_{i(i)} = v_i$; and 4) $v'_{i_j} > b_{i_j}$ for $j \leq K$ and $v'_{i_j} \leq b_{i_j}$ for $K + 1 \leq j \leq n - 1$.

Lemma 3: IMCU is individually rational.

Proof: Let i_i be user i's replacement which appears in the ith place in the sorting over $\mathcal{U}_{-i}$. Since user i_i would not be at ith place if i is considered, we have $v_{i(i)} - b_i \geq v'_{i_i} - b_{i_i}$. Hence we have $b_i \leq v_{i(i)} - (v'_{i_i} - b_{i_i})$. Since user i is a winner, we have $b_i \leq v_i = v_{i(i)}$. It follows that $b_i \leq \min\left\{v_{i(i)} - (v'_{i_i} - b_{i_i}), v_{i(i)}\right\} \leq p_i$. If i_i does not exist, it means i is the last winner in $\mathcal{U}$. We then have $b_i \leq v_i(\mathcal{U}_{-i}) \leq p_i$, according to Line 16. ■

Lemma 4: IMCU is profitable.

Proof: Let L be the last user $j \in \mathcal{U}$ in the sorting (5.5), such that $b_j < v_j$. We then have $\tilde{u}_0 = \sum_{1 \leq i \leq L} v_i - \sum_{1 \leq i \leq L} p_i$. Hence it suffices to prove that $p_i \leq v_i$ for each $1 \leq i \leq L$. Recall that K is the position of the last user $i_j \in \mathcal{U}_{-i}$ in the sorting (5.6), such that $b_{i_j} < v'_{i_j}$. When $K < n - 1$, let r be the position such that

$$r = \arg \max_{1 \leq j \leq K+1} \min\left\{v_{i(j)} - (v'_{i_j} - b_{i_j}), v_{i(j)}\right\}.$$

If $r \leq K$, we have

$$\begin{aligned} p_i &= \min\left\{v_{i(r)} - (v'_{i_r} - b_{i_r}), v_{i(r)}\right\} \\ &= v_{i(r)} - (v'_{i_r} - b_{i_r}) < v_{i(r)} \leq v_i, \end{aligned}$$

where the penultimate inequality is due to the fact that $b_{i_r} < v'_{i_r}$ for $r \leq K$, and the last inequality relies on the fact that $\mathcal{T}_{j-1} = \mathcal{S}_{j-1}$ for $j \leq i$ and the decreasing marginal value property of v. If $r = K + 1$, we have

$$p_i = \min\left\{v_{i(r)} - (v'_{i_r} - b_{i_r}), v_{i(r)}\right\} = v_{i(r)} \leq v_i.$$

Similarly, when $K = n - 1$, we have

$$p_i \leq v_i(r) \leq v_i,$$

for some $1 \leq r \leq K$. Thus we proved that $p_i \leq v_i$ for each $1 \leq i \leq K$. ■

Lemma 5: IMCU is truthful.

Proof: Based on Theorem 3, it suffices to prove that the selection rule of IMCU is monotone, and the payment p_i for each i is the critical value. The monotonicity of the selection rule is obvious as bidding a smaller value can not push user i backwards in the sorting.

We next show that p_i is the critical value for i in the sense that bidding higher p_i could prevent i from winning the auction. Note that

$$p_i = \max\left\{\max_{1 \leq j \leq K}\left(v_{i(j)} - (v'_{i_j} - b_{i_j})\right), v_{i(K+1)}\right\}.$$

If user i bids $b_i > p_i$, it will be placed after K since $b_i > v_{i(j)} - (v'_{i_j} - b_{i_j})$ implies $v'_{i_j} - b_{i_j} > v_{i(j)} - b_i$. At the $(K+1)$th iteration, user i will not be selected because $b_i > v_{i(K+1)}$. As $K + 1$ is the position of the first loser over $\mathcal{U}_{-i}$ when $K < n - 1$ or the last user to check when $K = n - 1$, the selection procedure terminates. ■

Lemmas 2 to 5 together prove the following theorem.

Theorem 4: IMCU is computationally efficient, individually rational, profitable and truthful.

Remark: IMCU still works when the valuation function is changed to any other efficiently computable submodular function. The four desirable properties still hold.

6. Performance Evaluation

To evaluate the performance of our incentive mechanisms, we implemented the incentive mechanism for the crowdsourcer-centric model, the Local Search-Based auction, denoted by LSB, and the IMCU auction, denoted by $IMCU$.

Performance Metrics: The performance metrics include crowdsourcer utility and user utility in general. For the crowdsourcer-centric incentive mechanism, we also study the number of participating users.

We have also evaluated the running time of both models. We observe that the running time of IMCC is almost linear in the number of users, and the running time of IMCU is linear in the number of users, as we have proved in Lemma 2. More details can be found in [45].

A. Simulation Setup

For the crowdsourcer-centric model, we assumed that the cost of each user was uniformly distributed over $[1, \kappa_{max}]$, where κ_{max} was varied from 1 to 10 with the increment of 1. For the crowdsourcer's utility function, we set $g(\tilde{t}_1, \tilde{t}_2, \ldots, \tilde{t}_l; n_1, n_2, \ldots, n_l)$ to $\lambda \log\left(1 + \sum_{\theta_j \in \Theta} n_j \log(1 + \tilde{t}_j)\right)$, where $\lambda = 10$ is a system parameter, the $\log(1 + \tilde{t}_j)$ term reflects the crowdsourcer's diminishing return on the work of a user with unit cost θ_j, and the outer $\log$ term reflects the crowdsourcer's diminishing return on participating users. We varied the number of users (n) from 100 to 1000 with the increment of 100. For the user-centric model, we considered a Cellular Signal Coverage application, where the tasks are to measure cellular signals around specific locations. We used two data sets. For the first set, tasks and users are randomly distributed in a 1000m × 1000m region, as shown in Fig. 4. This data set is code-named *Random*. Each user's task set includes all the tasks within a distance of 30m from the user. We varied the number of users (n) from 1000 to 10000 with the increment of 1000, and the number of tasks (m) from 100 to 500 with the increment of 100. For the second data set, we adopted a similar setting as in [37], code-named *Manhattan*. As shown in Fig. 5, the crowdsensing region is located in Manhattan, NY, which spans 3 blocks from west to east with a total length of 859m, and 3 blocks from north to south with a total length of 239m. For crowdsensing tasks, we called the Google Map API to collect all the points of interest (POIs) with types *food, bar, museum, cafe, gym, library, university,* *store, and school*. In total, there are 389 POIs within this region. We used a mobility model similar to the well-known Manhattan model to generate users moving trajectories [1]. To decide a user's task set, we assumed that each user will sense the cellular signal along its trajectory every 6 meters. If any POI is within 30m from the sensing point, it will be included in the user's task set. We varied the number of users (n) from 200 to 1000 with the increment of 200.

We set ϵ to 0.01 for LSB. We also made the following assumptions. The value of each task is uniformly distributed over $[1, 5]$. The cost c_i is $\rho|\Gamma_i|$, where ρ is uniformly distributed over $[1, 10]$. Here ρ does not need to be restricted in uniform distribution. For other distribution functions, for example Gaussian distribution, we have similar results.

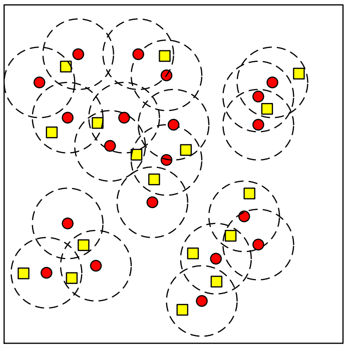


Fig. 4. Simulation setup of *Random*, where squares represent tasks and circles represent users.

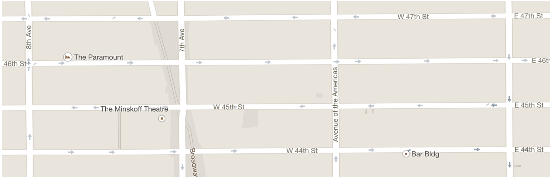


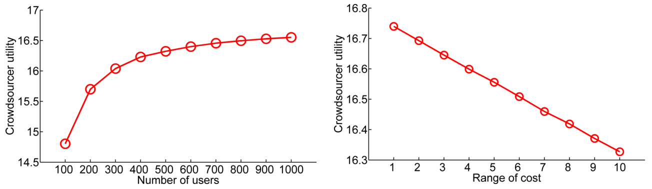
Fig. 5. Crowdsensing region of *Manhattan*

B. Evaluation of IMCC

Number of Participating Users: We observed that the number of participating users decreases as the unit costs of users become diverse. The reason is that according to the while-loop condition, if all users have the same unit cost, then all of them would satisfy this condition and thus participate. When the unit costs become diverse, users with larger unit costs would have higher chances to violate the condition.

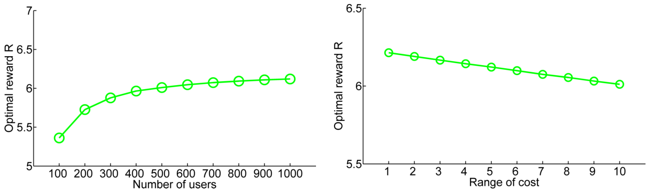
Crowdsourcer Utility: Fig. 6 shows the impact of n and κ_{max} on the crowdsourcer utility. In Fig. 6(a), we fixed $\kappa_{max} = 5$. We observe that the crowdsourcer utility indeed demonstrates diminishing returns as n increases. In Fig. 6(b), we fixed $n = 1000$. We note that the crowdsourcer utility decreases as the unit costs of users become more diverse.

Optimal Reward R^*: Fig. 7 shows the impact of n and κ_{max} on the crowdsourcer utility. In Fig. 7(a), we observe that the value of R^* increases with n and gradually becomes steady as n becomes larger. In Fig. 7(b), as the unit costs of users become more diverse, the optimal reward for incentivizing users decreases.



(a) Impact of n on $\bar{u}_0$ (b) Impact of κ_{max} on $\bar{u}_0$

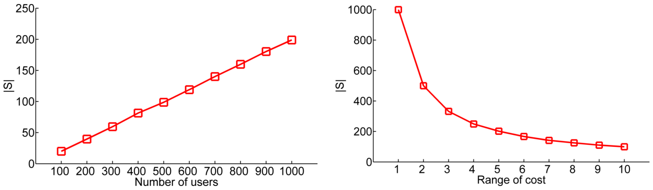
Fig. 6. Crowdsourcer utility



(a) Impact of n on R^* (b) Impact of κ_{max} on R^*

Fig. 7. Optimal reward R^* of the crowdsourcer

We observe that the optimal value of R^* and the crowdsourcer's utility $\bar{u}_0$ show similar dependency on n (as shown in Fig. 6(a) and Fig. 7(a)) and on κ_{max} (as shown in Fig. 6(a) and Fig. 7(b)). To study this dependency, we also show the number of participating users $|\mathcal{S}|$ as a function of n and κ_{max} in Fig. 8(a) and Fig. 8(b), respectively. We observe that when n increases, the number of participating users increases almost linearly. This is because the uniform distribution of users' unit cost results in the almost linear increase in the number of users with the same unit cost. As $|\mathcal{S}|$ increases, R^* is expected to increase as well, because each participating user shares R^* with more users and should have a non-negative utility. The dependency of $\bar{u}_0$ and R^* on κ_{max} can be explained similarly.



(a) Impact of n on $|\mathcal{S}|$ (b) Impact of κ_{max} on $|\mathcal{S}|$

Fig. 8. Number of participating users

User Utility: We randomly picked a user (ID = 31) and plot its utility in Fig. 9. We observe that as more and more users are interested in crowdsensing, the utility of the user decreases since more competitions are involved.

C. Evaluation of IMCU

Crowdsourcer Utility: Now we show how much crowdsourcer utility is affected in order to achieve the truthfulness compared to LSB. As shown in Fig. 10 and Fig. 11, we can observe the crowdsourcer utility achieved by $IMCU$ is larger

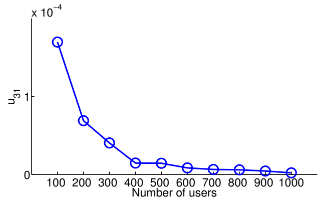
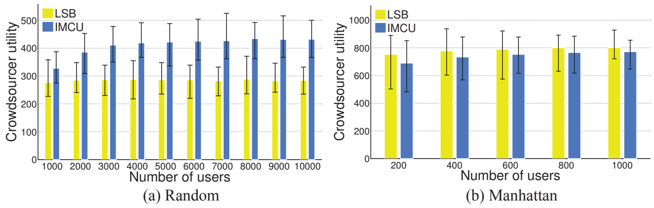


Fig. 9. Impact of n on $\bar{u}_i$

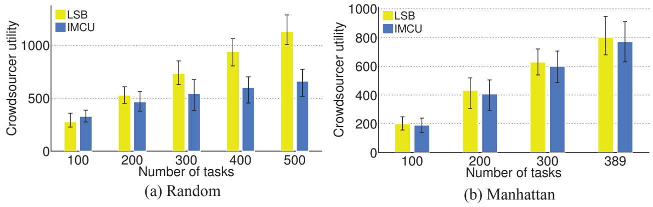
than that by LSB when m is relatively small, compared to n. This relation is reversed when m is large, and the sacrifice becomes more severe when m increases. However, note that in practice m is usually relatively small compared to n. We also observe that, similar to the crowdsourcer-centric model, the crowdsourcer utility demonstrates the diminishing returns as well when the number of users becomes larger.



(a) Random

(b) Manhattan

Fig. 10. Impact of n on crowdsourcer utility



(a) Random

(b) Manhattan

Fig. 11. Impact of m on crowdsourcer utility

7. Conclusion

In this paper, we designed incentive mechanisms that can be used to motivate smartphone users to participate in crowdsensing, which is a new sensing paradigm allowing us to collect and analyze sensed data far beyond the scale of what was previously possible. We considered two different models from different perspectives: the crowdsourcer-centric model where the crowdsourcer provides a reward shared by participating users, and the user-centric model where each user can ask for a reserve price for its sensing service.

For the crowdsourcer-centric model, we modeled the incentive mechanism as a Stackelberg game in which the crowdsourcer is the leader and the users are the followers. We proved that this Stackelberg game has a unique equilibrium, and designed an efficient mechanism for computing it. This enables the crowdsourcer to maximize its utility while each user is playing its best response strategy.

For the user-centric model, we designed an auction mechanism, called IMCU. We proved that IMCU is 1) computationally efficient, meaning that the winners and the payments can be computed in polynomial time; 2) individually rational, meaning that each user will have a non-negative utility if bidding its true cost; 3) profitable, meaning that the crowdsourcer will not incur a deficit; and more importantly, 4) truthful, meaning that no user can improve its utility by asking for a price different from its true cost. Our mechanism is scalable because its running time is linear in the number of users.

References

[1] F. Bai, N. Sadagopan, and A. Helmy. The important framework for analyzing the impact of mobility on performance of routing protocols for adhoc networks. *Ad Hoc Networks*, pages 383–403, 2003.

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

[3] L. Chen, L. Libman, and J. Leneutre. Conflicts and incentives in wireless cooperative relaying: A distributed market pricing framework. *IEEE Trans. Parallel Distrib. Syst.*, pages 758–772, 2011.

[4] N. Chen, N. Gravin, and P. Lu. On the approximability of budget feasible mechanisms. In *Proc. ACM-SIAM SODA*, pages 685–699, 2011.

[5] C. T. Chou, N. Bulusu, and S. Kanhere. Sensing data market. In *Proc. IEEE DCOSS*, pages 1–4, 2007.

[6] B. Chun, P. Buonadonna, A. AuYoung, C. Ng, D. Parkes, J. Shneidman, A. Snoeren, and A. Vahdat. Mirage: A microeconomic resource allocation system for sensornet testbeds. In *Proc. IEEE EmNetS-II*, pages 19–28, 2005.

[7] E. Cuervo, A. Balasubramanian, D.-k. Cho, A. Wolman, S. Saroiu, R. Chandra, and P. Bahl. MAUI: making smartphones last longer with code offload. In *Proc. MobiSys*, pages 49–62, 2010.

[8] G. Danezis, S. Lewis, and R. Anderson. How much is location privacy worth? In *Proc. WEIS*, 2005.

[9] T. Das, P. Mohan, V. N. Padmanabhan, R. Ramjee, and A. Sharma. PRISM: platform for remote sensing using smartphones. In *Proc. ACM MobiSys*, pages 63–76, 2010.

[10] E. De Cristofaro and C. Soriente. Short paper: Pepsi—privacy-enhanced participatory sensing infrastructure. In *Proc. WiSec*, pages 23–28, 2011.

[11] L. Duan, T. Kubo, K. Sugiyama, J. Huang, T. Hasegawa, and J. Walrand. Incentive mechanisms for smartphone collaboration in data acquisition and distributed computing. In *Proc. IEEE INFOCOM*, pages 1701–1709, 2012.

[12] U. Feige, V. S. Mirrokni, and J. Vondrak. Maximizing non-monotone submodular functions. *SIAM J. on Computing*, pages 1133–1153, 2011.

[13] Fortune. Industry first: Smartphones pass PCs in sales. http://tech.fortune.cnn.com/2011/02/07/idc-smartphone-shipment-numbers-passed-pc-in-q4-2010/.

[14] D. Fudenberg and J. Tirole. *Game theory*. MIT Press, 1991.

[15] L. Gao, Y. Xu, and X. Wang. Map: Multiauctioneer progressive auction for dynamic spectrum access. *IEEE Trans. Mobile Comput.*, pages 1144 –1161, 2011.

[16] G. Goel, A. Nikzad, and A. Singla. Allocating tasks to workers with matching constraints: truthful mechanisms for crowdsourcing markets. In *Proc. WWW*, pages 279–280, 2014.

[17] IDC. Smartphone vendor market share, Q4 2014. http://www.idc.com/prodserv/smartphone-market-share.jsp.

[18] E. Kamar and E. Horvitz. Incentives for truthful reporting in crowdsourcing. In *Proc. AAMAS*, pages 1329–1330, 2012.

[19] I. Koutsopoulos. Optimal incentive-driven design of participatory sensing systems. In *Proc. IEEE INFOCOM*, pages 1402–1410, 2013.

[20] V. Krishna. *Auction Theory*. Academic Press, 2009.

[21] N. Lane, E. Miluzzo, H. Lu, D. Peebles, T. Choudhury, and A. Campbell. A survey of mobile phone sensing. *IEEE Commun. Mag.*, pages 140–150, 2010.

[22] J. Lee and B. Hoh. Sell your experiences: A market mechanism based incentive for participatory sensing. In *Proc. IEEE PerCom*, pages 60–68, 2010.

[23] Q. Li and G. Cao. Providing privacy-aware incentives for mobile sensing. In *Proc. IEEE PerCom*, pages 76–84, 2013.

[24] P. Loiseau, G. Schwartz, J. Musacchio, S. Amin, and S. Sastry. Congestion pricing using a raffle-based scheme. In *Proc. NetGCooP*, pages 1–8, 2011.

[25] P. Loiseau, G. Schwartz, J. Musacchio, S. Amin, and S. Sastry. Incentive mechanisms for internet congestion management: Fixed-budget rebate versus time-of-day pricing. *IEEE/ACM Trans. Netw.*, pages 647–661, 2014.

[26] T. Luo, H.-P. Tan, and L. Xia. Profit-maximizing incentive for participatory sensing. In *Proc. IEEE INFOCOM*, pages 127–135, 2014.

[27] P. Mohan, V. N. Padmanabhan, and R. Ramjee. Nericell: rich monitoring of road and traffic conditions using mobile smartphones. In *Proc. ACM SenSys*, pages 323–336, 2008.

[28] M. Mun, S. Reddy, K. Shilton, N. Yau, J. Burke, D. Estrin, M. Hansen, E. Howard, R. West, and P. Boda. PIER, the personal environmental impact report, as a platform for participatory sensing systems research. In *Proc. ACM MobiSys*, pages 55–68, 2009.

[29] M. Musthag, A. Raij, D. Ganesan, S. Kumar, and S. Shiffman. Exploring micro-incentive strategies for participant compensation in high-burden studies. In *Proc. ACM UbiComp*, pages 435–444, 2011.

[30] R. Myerson. Optimal auction design. *Math. of Operations Research*, 6:58–73, 1981.

[31] S. Nath, P. Dayama, D. Garg, Y. Narahari, and J. Zou. Mechanism design for time critical and cost critical task execution via crowdsourcing. In *Internet and Network Economics*, pages 212–226. Springer, 2012.

[32] G. Nemhauser, L. Wolsey, and M. Fisher. An analysis of the approximations for maximizing submodular set functions. *Mathematical Programming*, 14:265–294, 1978.

[33] R. Rana, C. Chou, S. Kanhere, N. Bulusu, and W. Hu. Earphone: An end-to-end participatory urban noise mapping. In *Proc. ACM/IEEE IPSN*, pages 105–116, 2010.

[34] S. Reddy, D. Estrin, and M. B. Srivastava. Recruitment framework for participatory sensing data collections. In *Proc. Pervasive*, pages 138–155, 2010.

[35] C. Riederer, V. Erramilli, A. Chaintreau, B. Krishnamurthy, and P. Rodriguez. For sale: your data by: you. In *Proc. ACM HotNets*, pages 13:1–13:6, 2011.

[36] Sensorly. Sensorly. http://www.sensorly.com.

[37] X. Sheng, J. Tang, and W. Zhang. Energy-efficient collaborative sensing with mobile phones. In *Proc. IEEE INFOCOM*, pages 1916–1924, 2012.

[38] M. Silberman, L. Irani, and J. Ross. Ethics and tactics of professional crowdwork. *XRDS*, pages 39–43, 2010.

[39] Y. Singer. Budget feasible mechanisms. In *Proc. IEEE FOCS*, pages 765–774, 2010.

[40] A. Singla and A. Krause. Truthful incentives in crowdsourcing tasks using regret minimization mechanisms. In *Proc. WWW*, pages 1167–1178, 2013.

[41] Strategy Analytics. Two billion people worldwide now own a smartphone. http://blogs.strategyanalytics.com/WSS/post/2015/02/26/Two-Billion-People-Worldwide-Now-Own-a-Smartphone.aspx.

[42] A. Thiagarajan, L. Ravindranath, K. LaCurts, S. Madden, H. Balakrishnan, S. Toledo, and J. Eriksson. Vtrack: accurate, energy-aware road traffic delay estimation using mobile phones. In *Proc. ACM SenSys*, pages 85–98, 2009.

[43] W. Wang, B. Li, and B. Liang. District: Embracing local markets in truthful spectrum double auctions. In *Proc. IEEE SECON*, pages 521–529, 2011.

[44] D. Yang, X. Fang, and G. Xue. Truthful auction for cooperative communications. In *Proc. ACM MobiHoc*, pages 89–98, 2011.

[45] D. Yang, G. Xue, X. Fang, and J. Tang. Crowdsourcing to smartphones: Incentive mechanism design for mobile phone sensing. In *Proc. ACM MobiCom*, pages 173–184, 2012.

[46] D. Yang, X. Zhang, and G. Xue. PROMISE: A framework for truthful and profit maximizing spectrum double auctions. In *Proc. IEEE INFOCOM*, pages 109–117, 2014.

[47] X. Zhang, G. Xue, R. Yu, D. Yang, and J. Tang. Truthful incentive mechanisms for crowdsourcing. In *Proc. IEEE INFOCOM*, 2015. to appear.

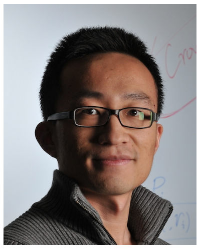
[48] X. Zhang, Z. Yang, Z. Zhou, H. Cai, L. Chen, and X. Li. Free market of crowdsourcing: Incentive mechanism design for mobile sensing. *IEEE Trans. Parallel Distrib. Syst.*, pages 3190–3200, 2014.

[49] Y. Zhang and M. van der Schaar. Reputation-based incentive protocols in crowdsourcing applications. In *Proc. IEEE INFOCOM*, pages 2140–2148, 2012.

[50] D. Zhao, X.-Y. Li, and H. Ma. How to crowdsource tasks truthfully without sacrificing utility: Online incentive mechanisms with budget constraint. In *Proc. IEEE INFOCOM*, pages 1213–1221, 2014.

[51] S. Zhong, L. Li, Y. G. Liu, and Y. Yang. On designing incentive-compatible routing and forwarding protocols in wireless ad-hoc networks: an integrated approach using game theoretical and cryptographic techniques. In *Proc. ACM MobiCom*, pages 117–131, 2005.

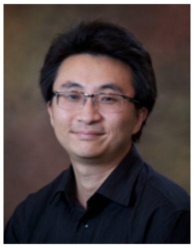
[52] X. Zhou, S. Gandhi, S. Suri, and H. Zheng. eBay in the sky: strategy-proof wireless spectrum auctions. In *Proc. ACM MobiCom*, pages 2–13, 2008.



Dejun Yang (M'13) is the Ben L. Fryrear Assistant Professor of Computer Science at Colorado School of Mines. He received the PhD degree in computer science from Arizona State University, Tempe, AZ, USA, in 2013 and the BS degree in computer science from Peking University, Beijing, China, in 2007. His research interests include economic and optimization approaches to networks, crowdsourcing, smart grid, cloud computing, and security and privacy. He has received Best Paper Awards at IEEE MASS'2011, IEEE ICC'2011, 2012, and a Best Paper Award Runner-up at IEEE ICNP'2010. He also served as a TPC member for many conferences including IEEE INFOCOM'2014, ICC'2013 and GLOBECOM'2013-2014.

Guoliang Xue (M'96–SM'99–F'11) is a Professor of Computer Science and Engineering at Arizona State University. He received the BS degree (1981) in mathematics and the MS degree (1984) in operations research from Qufu Normal University, Qufu, China, and the PhD degree (1991) in computer science from the University of Minnesota, Minneapolis, USA. His research interests include survivability, security, and resource allocation issues in networks. He received the Best Paper Award Runner-up at IEEE ICNP'2010 and the Best Paper Award at IEEE MASS'2011. He is an Editor of IEEE Network, and the Area Editor of IEEE Transactions on Wireless Communications for the Wireless Networking area. He was an Editor of IEEE/ACM Transactions on Networking from 2010 to 2014. He was a TPC co-Chair of IEEE INFOCOM'2010 and is a member of the IEEE INFOCOM Steering Committee. He is an Area TPC Chair for IEEE CNS'2015 and INFOCOM'2016. He is a TPC member of ACM CCS'2015 and CCS'2016. He was a Keynote Speaker at IEEE LCN'2011. He is a Fellow of the IEEE.

Xi Fang received the B.S. and M.S. degrees from Beijing University of Posts and Telecommunications, Beijing, China, and received the Ph.D. degree in computer science from Arizona State University, Tempe, AZ, USA in 2013. He is a software engineer at Google Inc. He has received Best Paper Awards at IEEE ICC'2012, IEEE MASS'2011, and IEEE ICC'2011.



Jian Tang (M'06–SM'13) is an associate professor in the Department of Electrical Engineering and Computer Science at Syracuse University. He earned the Ph.D degree in Computer Science from Arizona State University in 2006. His research interests lie in the areas of Cloud Computing, Big Data and Wireless Networking. Dr. Tang has published over 70 papers in premier journals and conferences. He received an NSF CAREER award in 2009 and a Best Paper Award in the 2014 IEEE International Conference on Communications. He has been an editor for IEEE Transactions on Vehicular Technology since 2010 and for IEEE Internet of Things Journal since 2013. He served as a TPC member for many conferences including IEEE Infocom 2010–2015, ICC 2006–2015 and Globecom 2006–2015.