8-2012

# Crowdsourcing to Smartphones: Incentive Mechanism Design for Mobile Phone Sensing

Dejun Yang
*Arizona State University*

Guoliang Xue
*Arizona State University*

Xi Fang
*Arizona State University*

Jian Tang
*Syracuse University*, jtang02@syr.edu



## Recommended Citation

D. Yang, G. Xue, X. Fang, and J. Tang, "Crowdsourcing to smartphones: Incentive mechanism design for mobile phone sensing," in 18th Annual International Conference on Mobile Computing and Networking, MobiCom 2012, August 22, 2012 - August 26, 2012, Istanbul, Turkey, 2012, pp. 173-184.

# Crowdsourcing to Smartphones: Incentive Mechanism Design for Mobile Phone Sensing *

Dejun Yang
Arizona State University
Tempe, AZ 85287
dejun.yang@asu.edu

Guoliang Xue
Arizona State University
Tempe, AZ 85287
xue@asu.edu

Xi Fang
Arizona State University
Tempe, AZ 85287
xi.fang@asu.edu

Jian Tang
Syracuse University
Syracuse, NY 13244
jtang02@syr.edu

## ABSTRACT

Mobile phone sensing is a new paradigm which takes advantage of the pervasive smartphones to collect and analyze data beyond the scale of what was previously possible. In a mobile phone sensing system, the *platform* recruits smartphone *users* to provide sensing service. Existing mobile phone sensing applications and systems lack good incentive mechanisms that can attract more user participation. To address this issue, we design incentive mechanisms for mobile phone sensing. We consider two system models: the platform-centric model where the platform 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 platform-centric model, we design an incentive mechanism using a Stackelberg game, where the platform is the leader while the users are the followers. We show how to compute the *unique Stackelberg Equilibrium*, at which the utility of the platform 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.

## Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design

## General Terms

Algorithm, Design, Economics

## Keywords

Crowdsourcing, Mobile Phone Sensing, Incentive Mechanism Design

*This research was supported in part by NSF grants 1217611, 1218203, and 0901451, and ARO grant W911NF-09-1-0467. The information reported here does not reflect the position or the policy of the federal government.

## 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 [3]. This inflection point occurred much quicker than predicted, which was supposed to be 2012 [15]. According to the International Data Corporation (IDC) Worldwide Quarterly Mobile Phone Tracker, it is estimated that 982 million smartphones will be shipped worldwide in 2015 [11].

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 surrounding environment. Smartphones are undoubtedly revolutionizing many sectors of our life, including social networks, environmental monitoring, business, healthcare, and transportation [13].

If all the smartphones on the planet together constitute a mobile phone sensing network, it would be the largest sensing network in the history. One can leverage millions of personal smartphones and a near-pervasive wireless network infrastructure to collect and analyze sensed data far beyond the scale of what was possible before, without the need to deploy thousands of static sensors.

Realizing the great potential of the mobile phone sensing, many researchers have developed numerous applications and systems, such as Sensorly [23] for making cellular/WiFi network coverage maps, Nericell [16] and VTrack [25] for providing traffic information, PIER [17] for calculating personalized environmental impact and exposure, and Ear-Phone [20] for creating noise maps. For more details on mobile phone sensing, we refer interested readers to the survey paper [13].

As shown in Figure 1, a mobile phone sensing system consists of a mobile phone sensing *platform*, which resides in the cloud and consists of multiple sensing servers, and many smartphone *users*, which are connected with the platform via the cloud. These smartphone users can act as sensing service providers. The platform recruits smartphone users to provide sensing services.

Although there are many applications and systems on mo-

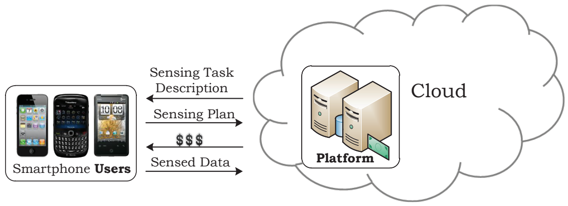



**Figure 1: A mobile phone sensing system**

bile phone sensing [16, 17, 20, 23, 25], most of them are based on voluntary participation. While participating in a mobile phone sensing 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 mobile phone sensing, unless it receives a satisfying reward to compensate its resource consumption and potential privacy breach*. Without adequate user participation, it is impossible for the mobile phone sensing applications to achieve good service quality, since sensing services are truly dependent on users' sensed data. While many researchers have developed different mobile phone sensing applications [5, 14], they either do not consider the design of incentive mechanisms or have neglected some critical properties of incentive mechanisms. To fill this void, we will design several incentive mechanisms to motivate users to participate in mobile phone sensing applications.

We consider two types of incentive mechanisms for a mobile phone sensing system: *platform-centric incentive mechanisms* and *user-centric incentive mechanisms*. In a platform-centric incentive mechanism, the platform has the absolute control over the total payment to users, and users can only tailor their actions to cater for the platform. Whereas in a user-centric incentive mechanism, the roles of the platform 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 platform then selects a subset of users and pay each of them an amount that is no lower than the user's reserve price.

## 1.1 Summary of Key Contributions

The following is a list of our main contributions.

- We design incentive mechanisms for mobile phone sensing, 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 platform-centric model where the platform 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 platform-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 platform 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, more importantly, truthful.

## 1.2 Paper Organization

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

# 2. SYSTEM MODEL AND PROBLEM FORMULATION

We use Figure 1 to aid our description of the mobile phone sensing system. The system consists of a mobile phone sensing *platform*, which resides in the cloud and consists of multiple sensing servers, and many smartphone *users*, which are connected to the platform via the cloud. The platform first publicizes the sensing tasks. Assume that there is a set $\mathcal{U} = \{1, 2, \ldots, n\}$ of smartphone users interested in participating in mobile phone sensing after reading the sensing task description, where $n \geq 2$. A user participating in mobile phone sensing will incur a *cost*, to be elaborated later. Therefore it expects a *payment* in return for its service. Taking cost and return 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 platform. After collecting the sensing plans from users, the platform 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 platform. This completes the whole mobile phone sensing process.

The platform 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 mobile phone sensing 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 mobile phone sensing system is out of the scope of this paper. Please refer to MAUI [4] for energy saving issues, PRISM [6] for application developing issues, and PEPSI [7] and TP [22] for privacy issues.

We study two models: *platform-centric* and *user-centric*. In the platform-centric model, the sensing plan of an interested user is in the form of its sensing time. A user participating in mobile phone sensing 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 platform-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.

Table 1 lists frequently used notations.

**Table 1: Frequently used notations**

| Notation | Description |
|---|---|
| $\mathcal{U}, i, j$ | set of users and user |
| $n$ | number of users |
| $R$ | reward of the platform |
| $t_i, t, t_{-i}$ | sensing time/strategy of user $i$, strategy profile of all users, strategy profile excluding user $i$'s strategy |
| $\kappa_i$ | 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 platform in the platform-centric model |
| $\lambda$ | system parameter in $\bar{u}_0$ |
| $\Gamma, \Gamma_i, \tau_j$ | set of tasks, set of user $i$'s tasks and task |
| $m$ | number of tasks |
| $\nu_j$ | value of task $\tau_j$ |
| $c_i, b_i$ | 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 platform in the user-centric model |

## 2.1 Platform-Centric Model

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

The *sensing plan of user $i$* is represented by $t_i$, the number of time units it is willing to provide the sensing service. Hence $t_i \geq 0$. By setting $t_i = 0$, user $i$ indicates that it will not participate in mobile phone sensing. The *sensing cost of user $i$* is $\kappa_i \times t_i$, where $\kappa_i > 0$ is its *unit cost.* Assume that the reward received by user $i$ is proportional to $t_i$. Then the *utility of user $i$* is

$$\bar{u}_i = \frac{t_i}{\sum_{j\in\mathcal{U}} t_j} R - t_i\kappa_i, \tag{2.1}$$

i.e., reward minus cost. The *utility of the platform* is

$$\bar{u}_0 = \lambda \log\left(1 + \sum_{i\in\mathcal{U}} \log(1+t_i)\right) - R, \tag{2.2}$$

where $\lambda > 1$ is a system parameter, the $\log(1+t_i)$ term reflects the platform's diminishing return on the work of user $i$, and the outer log term reflects the platform's diminishing return on participating users.

Under this model, the objective of the platform is to decide the optimal value of $R$ so as to maximize (2.2), while each user $i \in \mathcal{U}$ selfishly decides its sensing time $t_i$ to maximize (2.1) 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 $R \leq \kappa_i \sum_{j\neq i\in\mathcal{U}} t_j$.

## 2.2 User-Centric Model

In this model, the platform 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 platform.* Each user $i$ selects a subset of tasks $\Gamma_i \subseteq \Gamma$ according to its preference. Based on the selected task set, 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 $(\Gamma_i, b_i)$ to the platform, where $b_i$, called user $i$'s *bid*, is the *reserve price* user $i$ wants to sell the service for. Upon receiving the task-bid pairs from all the users, the platform 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

$$\tilde{u}_i = \begin{cases} p_i - c_i, & \text{if } i \in \mathcal{S}, \\ 0, & \text{otherwise.} \end{cases} \tag{2.3}$$

The *utility of the platform* is

$$\tilde{u}_0 = v(\mathcal{S}) - \sum_{i\in\mathcal{S}} p_i, \tag{2.4}$$

where $v(\mathcal{S}) = \sum_{\tau_j \in \cup_{i\in\mathcal{S}}\Gamma_i} \nu_j$.

Our objective for the user-centric model is to design an incentive mechanism satisfying the following four desirable properties:

- **Computational Efficiency**: A mechanism is computationally efficient if the outcome can be computed in polynomial time.
- **Individual Rationality**: Each participating user will have a non-negative utility.
- **Profitability**: The platform should not incur a deficit. In other words, the value brought by the winners should be at least as large as the total payment paid to the winners.
- **Truthfulness**: A mechanism is truthful if 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.

# 3. INCENTIVE MECHANISM FOR THE PLATFORM-CENTRIC MODEL

We model the platform-centric incentive mechanism as a *Stackelberg game* [9], which we call the MSensing game. There are two stages in this mechanism: In the first stage, the platform announces its reward $R$; in the second stage, each user strategizes its sensing time to maximize its own utility. Therefore the platform is the *leader* and the users are the *followers* in this Stackelberg game. Meanwhile, both the platform and the users are *players*. The *strategy of the platform* 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 the second stage of the MSensing game itself can be considered a non-cooperative game, which we call the Sensing Time Determination (STD) game. Given the MSensing game formulation, we are interested in answering 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 yes, 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 platform select the value of $R$ to maximize its utility in (2.2)?

The stable strategy set in Q1 corresponds to the concept of *Nash Equilibrium* (NE) in game theory [9].

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

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

*for any $t_i \geq 0$, where $\bar{u}_i$ is defined (2.1).*

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 allows the platform to predict the behaviors of the users and thus enables the platform 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 MSensing game, called *Stackelberg Equilibrium*.

In Section 3.1, 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 3.2, we prove that **the MSensing game has a unique Stackelberg Equilibrium**, and present an efficient algorithm for computing it.

## 3.1 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 $t_i \geq 0$.*

Based on the definition of NE, every user is playing its best response strategy in an NE. From (2.1), we know that $t_i \leq \frac{R}{\kappa_i}$ 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 $t_i$:

$$\frac{\partial \bar{u}_i}{\partial t_i} = \frac{-Rt_i}{(\sum_{j\in\mathcal{U}} t_j)^2} + \frac{R}{\sum_{j\in\mathcal{U}} t_j} - \kappa_i, \tag{3.1}$$

$$\frac{\partial^2 \bar{u}_i}{\partial t_i^2} = -\frac{2R\sum_{j\in\mathcal{U}\setminus\{i\}} t_j}{(\sum_{j\in\mathcal{U}} t_j)^3} < 0. \tag{3.2}$$

Since the second-order derivative of $\bar{u}_i$ is negative, the utility $\bar{u}_i$ is a *strictly concave function* in $t_i$. Therefore 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 $j \neq i$ 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. Therefore we are only interested in the best response for user $i$ when $\sum_{j\in\mathcal{U}\setminus\{i\}} t_j > 0$. Setting the first derivative of $\bar{u}_i$ to 0, we have

$$\frac{-Rt_i}{(\sum_{j\in\mathcal{U}} t_j)^2} + \frac{R}{\sum_{j\in\mathcal{U}} t_j} - \kappa_i = 0. \tag{3.3}$$

Solving for $t_i$ in (3.3), we obtain

$$t_i = \sqrt{\frac{R\sum_{j\in\mathcal{U}\setminus\{i\}} t_j}{\kappa_i}} - \sum_{j\in\mathcal{U}\setminus\{i\}} t_j. \tag{3.4}$$

If the RHS (right hand side) of (3.4) is positive, is also the best response strategy of user $i$, due to the concavity of $\bar{u}_i$. If the RHS of (3.4) is less than or equal to 0, then user $i$ does not participate in the mobile sensing by setting $t_i = 0$ (to avoid a deficit). Hence we have

$$\beta_i(t_{-i}) = \begin{cases} 0, & \text{if } R \leq \kappa_i \sum_{j\in\mathcal{U}\setminus\{i\}} t_j; \\ \sqrt{\frac{R\sum_{j\in\mathcal{U}\setminus\{i\}} t_j}{\kappa_i}} - \sum\limits_{j\in\mathcal{U}\setminus\{i\}} t_j, & \text{otherwise.} \end{cases} \tag{3.5}$$

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

**Algorithm 1:** Computation of the NE

Sort users according to their unit costs, $\kappa_1 \leq \kappa_2 \leq \cdots \leq \kappa_n$;

$\mathcal{S} \leftarrow \{1, 2\}$, $i \leftarrow 3$;

**while** $i \leq n$ *and* $\kappa_i < \frac{\kappa_i + \sum_{j\in\mathcal{S}} \kappa_j}{|\mathcal{S}|}$ **do**

  $\mathcal{S} \leftarrow \mathcal{S} \cup \{i\}$, $i \leftarrow i + 1$;

**end**

**foreach** $i \in \mathcal{U}$ **do**

  **if** $i \in \mathcal{S}$ **then** $t_i^{ne} = \frac{(|\mathcal{S}|-1)R}{\sum_{j\in\mathcal{S}} \kappa_j}\left(1 - \frac{(|\mathcal{S}|-1)\kappa_i}{\sum_{j\in\mathcal{S}} \kappa_j}\right)$;

  **else** $t_i^{ne} = 0$;

**end**

**return** $t^{ne} = (t_1^{ne}, t_2^{ne}, \ldots, t_n^{ne})$

**Theorem 1.** *The strategy profile $t^{ne} = (t_1^{ne}, \ldots, t_n^{ne})$ computed by Algorithm 1 is an NE of the STD game. The time complexity of Algorithm 1 is $O(n \log n)$.*

Proof. We first prove that the strategy profile $t^{ne}$ is an NE. Let $n_0 = |\mathcal{S}|$. We have the following observations based on the algorithm: 1)$\kappa_i \geq \frac{\sum_{j\in\mathcal{S}} \kappa_j}{n_0 - 1}$, for any $i \notin \mathcal{S}$; 2) $\sum_{j\in\mathcal{S}} t_j^{ne} = \frac{(n_0-1)R}{\sum_{j\in\mathcal{S}} \kappa_j}$; and 3) $\sum_{j\in\mathcal{S}\setminus\{i\}} t_j^{ne} = \frac{(n_0-1)^2 R\kappa_i}{(\sum_{j\in\mathcal{S}} \kappa_j)^2}$ for any $i \in \mathcal{S}$. We next prove that for any $i \notin \mathcal{S}$, $t_i^{ne} = 0$ is its best response strategy given $t_{-i}^{ne}$. Since $i \notin \mathcal{S}$, we have $\kappa_i \sum_{j\in\mathcal{U}\setminus\{i\}} t_j^{ne} = \kappa_i \sum_{j\in\mathcal{S}} t_j^{ne}$. Using 1) and 2), we have $\kappa_i \sum_{j\in\mathcal{S}} t_j^{ne} \geq R$. According to (3.5), we know that $\beta_i(t_{-i}^{ne}) = 0$.

We then prove that for any $i \in \mathcal{S}$, $t_i^{ne}$ is its best response strategy given $t_{-i}^{ne}$. Note that $\kappa_i < \frac{\sum_{j=1}^{i} \kappa_j}{i-1}$ according to Algorithm 1. We then have

$$(n_0 - 1)\kappa_i = (i-1)\kappa_i + (n_0 - i)\kappa_i < \sum_{j=1}^{i} \kappa_j + \sum_{j=i+1}^{n_0} \kappa_j,$$

where $\kappa_i \leq \kappa_j$ for $i + 1 \leq j \leq n_0$. Hence we have $\kappa_i < \frac{\sum_{i\in\mathcal{S}} \kappa_i}{n_0 - 1}$. Furthermore, we have

$$\kappa_i \sum_{j\in\mathcal{U}\setminus\{i\}} t_j^{ne} = \kappa_i \sum_{j\in\mathcal{S}\setminus\{i\}} t_j^{ne} = \kappa_i \frac{(n_0-1)^2 R\kappa_i}{\left(\sum_{j\in\mathcal{S}} \kappa_j\right)^2} < R.$$

According to (3.5),

$$\beta_i(t_{-i}^{ne}) = \sqrt{\frac{R\sum_{j\in\mathcal{U}\setminus\{i\}} t_j^{ne}}{\kappa_i}} - \sum_{j\in\mathcal{U}\setminus\{i\}} t_j^{ne}$$
$$= \frac{(n_0-1)R}{\sum_{j\in\mathcal{S}}\kappa_j} - \frac{(n_0-1)^2 R\kappa_i}{\left(\sum_{j\in\mathcal{S}}\kappa_j\right)^2} = t_i^{ne}.$$

Therefore $t^{ne}$ is an NE of the STD game.

We next analyze the running time of the algorithm. Sorting can be done in $O(n\log n)$ time. The while-loop (Lines 3-5) requires a total time of $O(n)$. The for-loop (Lines 6-9) requires a total time of $O(n)$. Hence the time complexity of Algorithm 1 is $O(n\log n)$. □

The next theorem shows the uniqueness of the NE for the STD game.

**Theorem** 2. *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) $|\bar{\mathcal{S}}| \geq 2$.
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 $\kappa_q \leq \max_{j\in\bar{\mathcal{S}}}\{\kappa_j\}$, then $q \in \bar{\mathcal{S}}$.*
4) *Assume that the users are ordered such that $\kappa_1 \leq \kappa_2 \leq \cdots \leq \kappa_n$. 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 the STD game has a unique NE, which is the one computed by Algorithm 1.*

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 $|\bar{\mathcal{S}}| \geq 1$. 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 $j \in \mathcal{U}\setminus\{k\}$. According to (2.1) 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 $|\bar{\mathcal{S}}| \geq 2$.

We next prove 2). Let $n_0 = |\bar{\mathcal{S}}|$. Since we already proved that $n_0 \geq 2$, we can use the analysis at the beginning of this section (3.3), 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

$$\frac{-R\bar{t}_i}{(\sum_{j\in\bar{\mathcal{S}}}\bar{t}_j)^2} + \frac{R}{\sum_{j\in\bar{\mathcal{S}}}\bar{t}_j} - \kappa_i = 0, \quad i \in \bar{\mathcal{S}}. \tag{3.6}$$

Summing up (3.6) 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

$$\sum_{j\in\bar{\mathcal{S}}}\bar{t}_j = \frac{(n_0-1)R}{\sum_{j\in\bar{\mathcal{S}}}\kappa_j}. \tag{3.7}$$

Substituting (3.7) into (3.6) and considering $\bar{t}_j = 0$ for any $j \in \mathcal{U}\setminus\bar{\mathcal{S}}$, we obtain the following:

$$\bar{t}_i = \frac{(n_0-1)R}{\sum_{j\in\bar{\mathcal{S}}}\kappa_j}\left(1 - \frac{(n_0-1)\kappa_i}{\sum_{j\in\bar{\mathcal{S}}}\kappa_j}\right) \tag{3.8}$$

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 (3.8), $\bar{t}_i > 0$ implies $\frac{(n_0-1)\kappa_i}{\sum_{j\in\bar{\mathcal{S}}}\kappa_j} < 1$. Therefore we have

$$\kappa_i < \frac{\sum_{j\in\bar{\mathcal{S}}}\kappa_j}{|\bar{\mathcal{S}}|-1}, \forall i \in \bar{\mathcal{S}}. \tag{3.9}$$

(3.9) implies that

$$\max_{i\in\bar{\mathcal{S}}}\kappa_i < \frac{\sum_{j\in\bar{\mathcal{S}}}\kappa_j}{|\bar{\mathcal{S}}|-1}. \tag{3.10}$$

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

$$\frac{R}{\sum_{j\in\bar{\mathcal{S}}}\bar{t}_j} - \kappa_q = \frac{\sum_{j\in\bar{\mathcal{S}}}\kappa_j}{n_0-1} - \kappa_q > \max_{i\in\bar{\mathcal{S}}}\{\kappa_i\} - \kappa_q \geq 0. \tag{3.11}$$

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 (3.9), 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. □

## 3.2 Platform Utility Maximization

According to the above analysis, the platform, which is the leader in the Stackelberg game, knows that there exists a unique NE for the users for any given value of $R$. Hence the platform can maximize its utility by choosing the optimal $R$. Substituting (3.8) into (2.2) and considering $t_i = 0$ if $i \notin \mathcal{S}$, we have

$$\bar{u}_0 = \lambda\log\left(1 + \sum_{i\in\mathcal{S}}\log(1 + X_iR)\right) - R, \tag{3.12}$$

where

$$X_i = \frac{(n_0-1)}{\sum_{j\in\mathcal{S}}\kappa_j}\left(1 - \frac{(n_0-1)\kappa_i}{\sum_{j\in\mathcal{S}}\kappa_j}\right).$$

**Theorem** 3. *There exists a unique Stackelberg Equilibrium $(R^*, t^{ne})$ in the MSensing game, where $R^*$ is the unique maximizer of the platform utility in (3.12) over $R \in [0, \infty)$, $\mathcal{S}$ and $t^{ne}$ are given by Algorithm 1 with the total reward set to $R^*$.*

Proof. The second order derivative of $\bar{u}_0$ is

$$\frac{\partial^2\bar{u}_0}{\partial R^2} = -\lambda\frac{\sum_{i\in\mathcal{S}}\frac{X_i^2}{(1+X_iR)^2}Y + \left(\sum_{i\in\mathcal{S}}\frac{X_i}{(1+X_iR)}\right)^2}{Y^2} < 0, \tag{3.13}$$

where $Y = 1 + \sum_{i\in\mathcal{S}}\log(1 + X_iR)$. Therefore the utility $\bar{u}_0$ defined in (3.12) is a strictly concave function of $R$ for $R \in [0, \infty)$. Since the value of $\bar{u}_0$ in (3.12) is 0 for $R = 0$ and goes to $-\infty$ when $R$ goes to $\infty$, it has a unique maximizer $R^*$ that can be efficiently computed using either bisection or Newton's method [1]. □

# 4. INCENTIVE MECHANISM FOR THE USER-CENTRIC MODEL

Auction theory [12] 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.

## 4.1 Auctions Maximizing Platform Utility

Our first attempt is to design an incentive mechanism maximizing the utility of the platform. 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

$$\tilde{u}_0(\mathcal{S}) = v(\mathcal{S}) - \sum_{i\in\mathcal{S}} b_i. \tag{4.1}$$

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

Unfortunately, as the following theorem shows, it is NP-hard to find the optimal solution to the User Selection problem.

**Theorem** 4. *The User Selection problem is NP-hard.*

PROOF. We will prove this theorem in the appendix for a better flow of the paper. □

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 the set of reals.*

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

$$\begin{aligned} v(\mathcal{S} \cup \{i\}) - v(\mathcal{S}) &= \sum_{\tau_j \in \Gamma_i \setminus \cup_{j\in\mathcal{S}}\Gamma_j} \nu_j & (4.2)\\ &\geq \sum_{\tau_j \in \Gamma_i \setminus \cup_{j\in\mathcal{T}}\Gamma_j} \nu_j & (4.3)\\ &= v(\mathcal{T} \cup \{i\}) - v(\mathcal{T}). & (4.4) \end{aligned}$$

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 [19]. Without monotonicity, Feige *et al.* [8] 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 [8], 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 [8].

**Algorithm 2:** LSB Auction

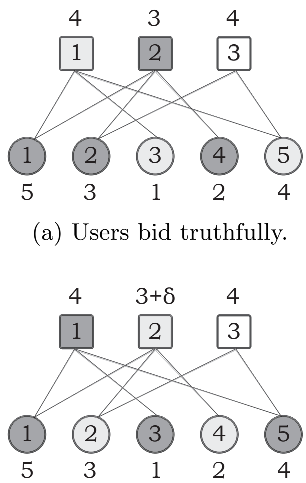
```
1  S ← {i}, where i ← arg max_{i∈U} f({i});
2  while there exists a user i ∈ U \ S such that
     f(S ∪ {i}) > (1 + ε/n²) f(S) do
3  |  S ← S ∪ {i};
4  end
5  if there exists a user i ∈ S such that
     f(S \ {i}) > (1 + ε/n²) f(S) then
6  |  S ← S \ {i}; go to Line 2;
7  end
8  if f(U \ S) > f(S) then  S ← U \ S;
9  foreach i ∈ U do
10 |  if i ∈ S then  p_i ← b_i;
11 |  else  p_i ← 0;
12 end
13 return (S, p)
```

How good is the LSB auction? In the following we analyze this mechanism using the four desirable properties described in Section 2.2 as performance metrics.

- **Computational Efficiency**: The running time of the Local Search Algorithm is $O(\frac{1}{\epsilon}n^3 m \log m)$ [8], where evaluating the value of $f$ takes $O(m)$ time and $|\mathcal{S}| \leq m$. Hence our mechanism is computationally efficient.
- **Individual Rationality**: The platform pays what the winners bid. Hence our mechanism 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 $\tilde{u}_0(\mathcal{S}) > 0$ at the end of the auction. Hence our mechanism is profitable.
- **Truthfulness**: We use an example in Figure 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 $\tau_j$ denotes its value $\nu_j$. 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 consider the case where 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 $1 \leq \delta < 1.77$.

**Figure 2: An 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 $\tau_j$ denotes its value $\nu_j$. We also assume that $\epsilon = 0.1$.**

## 4.2 MSensing Auction

Although the LSB auction mechanism is designed to approximately maximize the platform 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 mobile phone sensing 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 highest priority is to design an incentive mechanism that satisfies all of the four desirable properties, even at the cost of sacrificing the platform utility. One possible direction is to make use of the off-the-shelf results on the *budgeted mechanism* design [2, 24]. The budgeted mechanism design problem is very similar with ours, with the difference that the payment paid to the winners is a constraint instead of a factor in the objective function. To address this issue, it is intuitive that we can 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 [18].

**Theorem** 5. *([24, 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.*

### 4.2.1 Auction Design

Based on Theorem 5, we design our auction mechanism in this section, which is called MSensing auction. Illustrated in Algorithm 3, the MSensing auction mechanism consists of two phases: the *winner selection* phase and the *payment determination* phase.

**Algorithm 3:** MSensing Auction

// Phase 1: Winner selection
$\mathcal{S} \leftarrow \emptyset$, $i \leftarrow \arg\max_{j \in \mathcal{U}} (v_j(\mathcal{S}) - b_j)$;
**while** $b_i < v_i$ *and* $\mathcal{S} \neq \mathcal{U}$ **do**
  $\mathcal{S} \leftarrow \mathcal{S} \cup \{i\}$;
  $i \leftarrow \arg\max_{j \in \mathcal{U} \setminus \mathcal{S}} (v_j(\mathcal{S}) - b_j)$;
**end**
// Phase 2: Payment determination
**foreach** $i \in \mathcal{U}$ **do** $p_i \leftarrow 0$;
**foreach** $i \in \mathcal{S}$ **do**
  $\mathcal{U}' \leftarrow \mathcal{U} \setminus \{i\}$, $\mathcal{T} \leftarrow \emptyset$;
  **repeat**
    $i_j \leftarrow \arg\max_{j \in \mathcal{U}' \setminus \mathcal{T}} (v_j(\mathcal{T}) - b_j)$;
    $p_i \leftarrow \max\{p_i, \min\{v_i(\mathcal{T}) - (v_{i_j}(\mathcal{T}) - b_{i_j}), v_i(\mathcal{T})\}\}$;
    $\mathcal{T} \leftarrow \mathcal{T} \cup \{i_j\}$;
  **until** $b_{i_j} \geq v_{i_j}$ *or* $\mathcal{T} = \mathcal{U}'$;
  **if** $b_{i_j} < v_{i_j}$ **then** $p_i \leftarrow \max\{p_i, v_i(\mathcal{T})\}$;
**end**
**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

$$v_1 - b_1 \geq v_2 - b_2 \geq \cdots \geq v_n - b_n. \tag{4.5}$$

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 $\mathcal{U} \setminus \{i\}$ similarly,

$$v'_{i_1} - b_{i_1} \geq v'_{i_2} - b_{i_2} \geq \cdots \geq v'_{i_{n-1}} - b_{i_{n-1}}, \tag{4.6}$$

where $v'_{i_j} = v(\mathcal{T}_{j-1} \cup \{i_j\}) - v(\mathcal{T}_{j-1})$ denotes the marginal value of the $j$th user and $\mathcal{T}_j$ denotes the first $j$ users according to this sorting over $\mathcal{U} \setminus \{i\}$ 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 $i_j \in \mathcal{U} \setminus \{i\}$, 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 $j$th place. We repeat this until the position after the last winner in $\mathcal{U} \setminus \{i\}$. In the end we set the value of $p_i$ to the maximum of these $K + 1$ prices.

### 4.2.2 A Walk-Through Example

We use the example in Figure 3 to illustrate how the MSensing auction works.

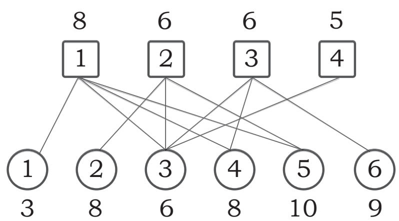



**Figure 3: Illustration for MSensing**

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 $j \le K$ and $v'_{i_j} \le b_{i_j}$ for $j \ge K + 1$.

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 $p_3 = 9$.

### 4.2.3 Properties of MSensing

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

**Lemma** 2. *MSensing 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 $O(nm^3)$. $\square$

Note that the running time of the MSensing Auction, $O(nm^3)$, is very conservative. In addition, $m$ is much less than $n$ in practice, which makes the running time of the MSensing 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) $v_{i(j)} \ge v_{i(j+1)}$ 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 $j \le K$ and $v'_{i_j} \le b_{i_j}$ for $K + 1 \le j \le n - 1$.

**Lemma** 3. *MSensing is individually rational.*

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

**Lemma** 4. *MSensing is profitable.*

Proof. Let $L$ be the last user $j \in \mathcal{U}$ in the sorting (4.5), such that $b_j < v_j$. We then have $\tilde{u}_0 = \sum_{1 \le i \le L} v_i - \sum_{1 \le i \le L} p_i$. Hence it suffices to prove that $p_i \le v_i$ for each $1 \le i \le L$. Recall that $K$ is the position of the last user $i_j \in \mathcal{U} \setminus \{i\}$ in the sorting (4.6), such that $b_{i_j} < v'_{i_j}$. When $K < n - 1$, let $r$ be the position such that

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

If $r \le K$, we have

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

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

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

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

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

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

**Lemma** 5. *MSensing is truthful.*

Proof. Based on Theorem 5, it suffices to prove that the selection rule of MSensing 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

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

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 $\mathcal{U}\setminus\{i\}$ when $K < n-1$ or the last user to check when $K = n-1$, the selection procedure terminates. □

The above four lemmas together prove the following theorem.

**Theorem** 6. *MSensing is computationally efficient, individually rational, profitable and truthful.*

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

# 5. PERFORMANCE EVALUATION

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

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

## 5.1 Simulation Setup

We varied the number of users ($n$) from 100 to 1000 with the increment of 100. For the platform-centric model, we assumed that the cost of each user was uniformly distributed over $[1, \kappa_{max}]$, where $\kappa_{max}$ was varied from 1 to 10 with the increment of 1. We set $\lambda$ to 10. For the user-centric model, tasks and users are randomly distributed in a 1000m $\times$ 1000m region, as shown in Figure 4. Each user's task set includes all the tasks within a distance of 30m from the user. We varied the number of tasks ($m$) from 100 to 500 with the increment of 100. We set $\epsilon$ 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 $\rho$ is uniformly distributed over $[1, 10]$.

All the simulations were run on a Linux machine with 3.2 GHz CPU and 16 GB memory. Each measurement is averaged over 100 instances.

## 5.2 Evaluation of the Platform-Centric Incentive Mechanism

Running Time: We first evaluate the running time of the incentive mechanism and show the results in Figure 5. We observe that the running time is almost linear in the number of users and less than $5 \times 10^{-4}$ seconds for the largest instance of 1000 users. As soon as the users are sorted and $\mathcal{S}$ is computed, all the values can be computed using closed-form expressions, which makes the incentive mechanism very efficient.

Number of Participating Users: Figure 6 shows the impact of $\kappa_{max}$ on the number of participating users, i.e., $|\mathcal{S}|$,

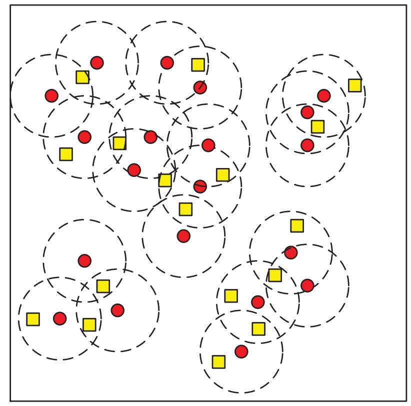


**Figure 4: Simulation setup for the user-centric model, where squares represent tasks and circles represent users.**

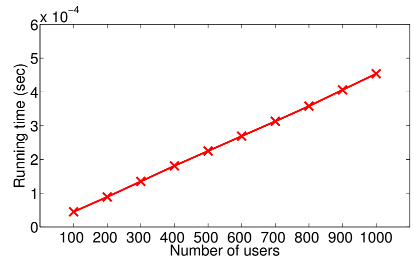



**Figure 5: Running time**

when $n$ is fixed at 1000. We can see that $|\mathcal{S}|$ decreases as the costs of users become diverse. The reason is that according to the while-loop condition, if all users have the same cost, then all of them would satisfy this condition and thus participate. When the costs become diverse, users with larger costs would have higher chances to violate the condition.

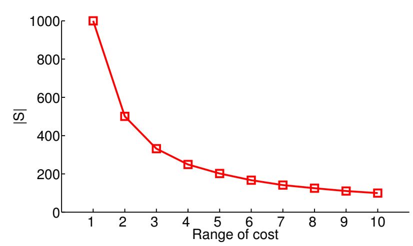
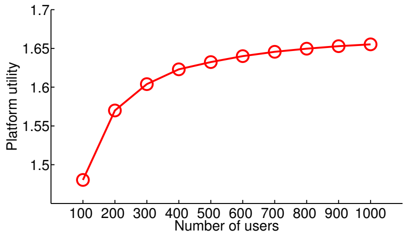


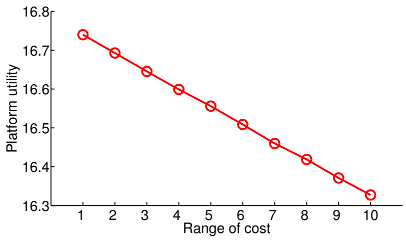

**Figure 6: Impact of $\kappa_{max}$ on $|\mathcal{S}|$**

Platform Utility: Figure 7 shows the impact of $n$ and $\kappa_{max}$ on the platform utility. In Figure 7(a), we fixed $\kappa_{max} = 5$. We observe that the platform utility indeed demonstrates diminishing returns when $n$ increases. In Figure 7(b), we fixed $n = 1000$. With the results in Figure 6, it is expected that the platform utility decreases as the costs of users become more diverse.

User Utility: We randomly picked a user (ID = 31) and plot its utility in Figure 8. We observe that as more and more users are interested in mobile phone sensing, the utility of the user decreases since more competitions are involved.




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




(b) Impact of $\kappa_{max}$ on $\bar{u}_0$

**Figure 7: Platform utility**

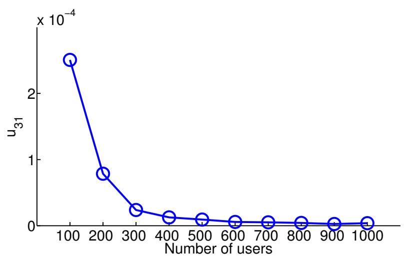


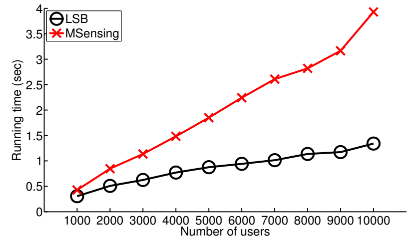

**Figure 8: Impact of $n$ on $\bar{u}_i$**

## 5.3 Evaluation of the User-Centric Incentive Mechanism

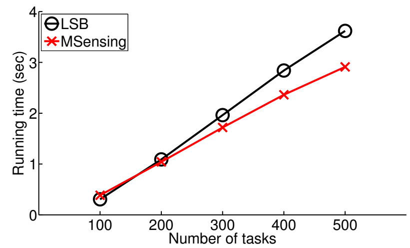
Running Time: Figure 9 shows the running time of different auction mechanisms proposed in Section 4. More specifically, Figure 9(a) plots the running time as a function of $n$ while $m = 100$. We can see that $LSB$ has better efficiency than $MSensing$. Note that $MSensing$ is linear in $n$, as we proved in Lemma 2. Figure 9(b) plots the running time as a function of $m$ while $n = 1000$. Both $LSB$ and $MSensing$ have similar performance while $MSensing$ outperforms $LSB$ slightly.

Platform Utility: Now we show how much platform utility we need to sacrifice to achieve the truthfulness compared to $LSB$. As shown in Figure 10, we can observe the platform utility achieved by $MSensing$ is larger than that by $LSB$ when the number of tasks is small ($m = 100$). 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 platform-centric model, the platform utility demonstrates the diminishing returns as well when the number of users becomes larger.

Truthfulness: We also verified the truthfulness of $MSensing$ by randomly picking two users (ID = 333 and ID = 851) and allowing them to bid prices that are different from their true costs. We illustrate the results in Figure 11. As we can see, user 333 achieves its optimal utility if it




(a) Impact of $n$




(b) Impact of $m$

**Figure 9: Running time**

bids truthfully ($b_{333} = c_{333} = 3$) in Figure 11(a) and user 851 achieves its optimal utility if it bids truthfully ($b_{851} = c_{851} = 18$) in Figure 11(b).

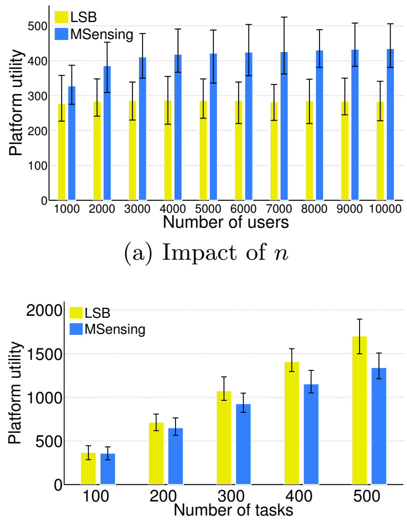
# 6. RELATED WORK

In [21], Reddy *et al.* developed recruitment frameworks to enable the platform to identify well-suited participants for sensing services. However, they focused only on the user selection, not the incentive mechanism design. To the best of our knowledge, there are few research studies on the incentive mechanism design for mobile phone sensing [5, 14]. In [5], Danezis *et al.* developed a sealed-bid second-price auction to motivate user participation. However, the utility of the platform was neglected in the design of the auction. In [14], 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.

The design of the incentive mechanism was also studied for other networking problems, such as spectrum trading [10, 26, 28] and routing [27]. However none of them can be directly applied to mobile phone sensing applications, as they all considered properties specifically pertain to the studied problems.

# 7. CONCLUSION

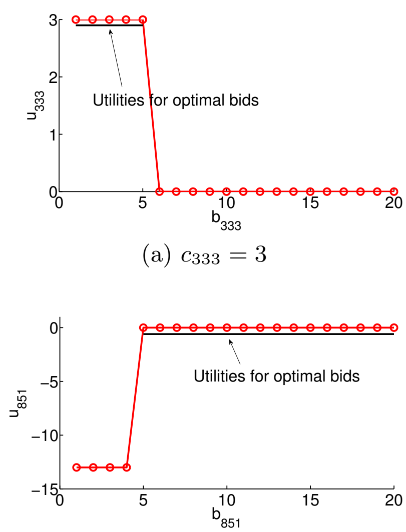
In this paper, we have designed incentive mechanisms that can be used to motivate smartphone users to participate in mobile phone sensing, which is a new sensing paradigm allowing us to collect and analyze sensed data far beyond the scale of what was previously possible. We have considered two different models from different perspectives: the platform-centric model where the platform 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.




(a) Impact of $n$

(b) Impact of $m$

**Figure 10: Platform utility**




(a) $c_{333} = 3$

(b) $c_{851} = 18$

**Figure 11: Truthfulness of MSensing**

For the platform-centric model, we have modeled the incentive mechanism as a Stackelberg game in which the platform is the leader and the users are the followers. We have proved that this Stackelberg game has a unique equilibrium, and designed an efficient mechanism for computing it. This enables the platform to maximize its utility while no user can improve its utility by deviating from the current strategy unilaterally.

For the user-centric model, we have designed an auction mechanism, called MSensing. We have proved that MSensing 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; 3) profitable, meaning that the platform 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.

## ACKNOWLEDGMENT

We thank the anonymous reviewers and the shepherd, whose comments and guidance have helped to significantly improve the paper.

## 8. REFERENCES

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

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

[3] CNN 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/`.

[4] 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 *Proceedings of MobiSys*, pages 49–62, 2010.

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

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

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

[8] U. Feige, V. S. Mirrokni, and J. Vondrak. Maximizing non-monotone submodular functions. *SIAM J. on Computing*, 40(4):1133–1153, 2011.

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

[10] L. Gao, Y. Xu, and X. Wang. Map: Multiauctioneer progressive auction for dynamic spectrum access. *IEEE Transactions on Mobile Computing*, 10(8):1144 –1161, August 2011.

[11] IDC. Worldwide smartphone market expected to grow 55% in 2011 and approach shipments of one billion in 2015, according to IDC. `http://www.idc.com/getdoc.jsp?containerId=prUS22871611`.

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

[13] N. Lane, E. Miluzzo, H. Lu, D. Peebles, T. Choudhury, and A. Campbell. A survey of mobile phone sensing. *IEEE Communications Magazine*, 48:140–150, 2010.

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

[15] M. Meeker. Mary Meeker: Smartphones will surpass PC shipments in two years. `http://techcrunch.com/2010/11/16/meeker-smartphones-pcs/`.

[16] P. Mohan, V. N. Padmanabhan, and R. Ramjee. Nericell: rich monitoring of road and traffic conditions

using mobile smartphones. In *Proceedings of SenSys*, pages 323–336, 2008.

[17] 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 *Proceedings of ACM MobiSys*, pages 55–68, 2009.

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

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

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

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

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

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

[24] Y. Singer. Budget feasible mechanisms. In *Proceedings of IEEE FOCS*, pages 765–774, 2010.

[25] 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 *Proceedings of ACM SenSys*, pages 85–98, 2009.

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

[27] 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 *Proceedings of ACM Mobicom*, pages 117–131, 2005.

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

# APPENDIX

## A. PROOF OF THEOREM 4

We prove the NP-hardness of the optimization problem by giving a polynomial time reduction from the NP-hard **Set Cover** problem:

INSTANCE: A universe $Z = \{z_1, z_2, \dots, z_m\}$, a family $\mathcal{C} = \{C_1, C_2, \dots, C_n\}$ of subsets of $Z$ and a positive integer $s$. QUESTION: Does there exist a subset $\mathcal{C}' \subseteq \mathcal{C}$ of size $s$, such that every element in $Z$ belongs to at least one member in $\mathcal{C}'$?

We construct a corresponding instance of the **User Selection** problem as follows: Let $\Gamma$ be the task set corresponding to $Z$, where there is a task $\tau_j \in \Gamma$ for each $z_j \in Z$. Corresponding to each subset $C_i \in \mathcal{C}$, there is a user $i \in \mathcal{U}$ with task set $\Gamma_i$, which consists of the tasks corresponding to the elements in $C_i$. We set $\nu_j$ to $n$ for each task $\tau_j$ and $c_i$ to 1 for each user $i \in \mathcal{U}$. We prove that *there exists a solution to the instance of the* Set Cover *problem if and only if there exists a subset $\mathcal{S}$ of users such that $\tilde{u}_0(\mathcal{S}) \geq nm - s$.*

We first prove the forward direction. Let $\mathcal{C}'$ be a solution to the *Set Cover* instance. We can select the corresponding set $\mathcal{S}$ of users as the solution to the mechanism design instance. Clearly, $\tilde{u}_0(\mathcal{S}) = nm - |\mathcal{S}| \geq nm - s$. Next we prove the backward direction. Let $\mathcal{S}$ be a solution to mechanism design instance. We then have $\tilde{u}_0(\mathcal{S}) \geq nm - s$. The only possibility that we have such a value is when the selected user set covers all the tasks, because $s \leq m$. Therefore the corresponding set $\mathcal{C}'$ is a solution to the *Set Cover* instance.