# Sybil-Proof Incentive Mechanisms for Crowdsensing

Jian Lin, Ming Li, Dejun Yang, Guoliang Xue, Jian Tang

***Abstract*—The rapid growth of sensor-embedded smartphones has led to a new data sensing and collecting paradigm, known as crowdsensing. Many auction-based incentive mechanisms have been proposed to stimulate smartphone users to participate in crowdsensing. However, none of them have taken into consideration the Sybil attack where a user illegitimately pretends multiple identities to gain benefits. This attack may undermine existing inventive mechanisms. To deter the Sybil attack, we design Sybil-proof auction-based incentive mechanisms for crowdsensing in this paper. We investigate both the single-minded and multi-minded cases and propose *SPIM-S* and *SPIM-M*, respectively. *SPIM-S* achieves computational efficiency, individual rationality, truthfulness, and Sybil-proofness. *SPIM-M* achieves individual rationality, truthfulness, and Sybil-proofness. We evaluate the performance and validate the desired properties of *SPIM-S* and *SPIM-M* through extensive simulations.**

## I. Introduction

Nowadays, the advance of high-speed 3G/4G networks and the proliferation of powerful sensors-embedded smartphones has led to a new paradigm, known as crowdsensing, which senses and collects data efficiently. Examples include [1, 2]. A typical crowdsensing system consists of a cloud-based platform and a collection of smartphone users. The platform works as a sensing service buyer who launches a set of sensing tasks and selects a specific set of smartphone users to perform the sensing tasks. Once selected by the platform, a smartphone user starts to perform the assigned sensing tasks and sends sensing data back to the platform. With the low deploying cost and high sensing coverage, crowdsensing has enabled a wide rage of applications [16, 20]. However, most of them assume that the smartphone users contribute to the platform voluntarily. In reality, smartphone users may be reluctant to participate in a crowdsensing system because they consume their own resources, e.g., sensing time, battery, and cellular data traffic, while performing the sensing tasks. Furthermore, they might suffer from the potential privacy disclosure by sharing the sensing data with personal information, e.g., location tags. It is clear that the success of a crowdsensing system strongly relies on the number of participating users, and thus it is necessary to design incentive mechanisms to stimulate users to participate in crowdsensing.

Auction is an efficient method to design incentive mechanisms, and a number of auction-based incentive mechanisms have been proposed for crowdsensing [10, 13, 31, 36]. These incentive mechanisms model a crowdsensing system as a reverse auction in which the platform is the service buyer, and the smartphone users are the service sellers who bid to perform sensing tasks. In these mechanisms, the service buyer selects users according to their submitted bids. Some alternatives are based on all-pay auction [15] or double auction [4]. The objectives of these mechanisms focus on either maximizing the total value gained by the platform or minimizing the social cost. However, none of aforementioned mechanisms take the Sybil attack [5], also known as false-name attack, into consideration.

Recently, the effects of Sybil attack have been analyzed in social networks [24], incentive tree mechanism [35], cloud resource allocation [26], spectrum auction [27], and mobile apps [25]. The impact of Sybil attack in auctions has been analyzed in [32, 33].

In crowdsensing, a user may try to profit from submitting multiple bids under fictitious identities, e.g., creating multiple accounts. This attack is easy to conduct but difficult to detect. Existing auction-based incentive mechanisms are vulnerable to Sybil attack. Among them, all the VCG auction-based incentive mechanisms [6, 8, 21, 28] are not Sybil-proof, since the VCG auction has been proved not Sybil-proof in [33]. Mechanisms proposed in [4, 7, 11, 34, 36] are not Sybil-proof, since a user can exploit multiple fictitious identities to increase its critical value, and thus increase its payment. For mechanisms in [9, 29, 37, 38], a user can change from a loser to a winner with a positive utility through Sybil attack. A user can increase its utility by completing a subset of its sensing task set through mechanisms in [30, 31]. We will use two examples to demonstrate that existing mechanisms are not Sybil-proof in Section III. The vulnerability of a mechanism to Sybil attack may make the system fail to achieve the desired properties, e.g., social cost minimization [29], and jeopardize the fairness of the system, which discourages users from participating in crowdsensing. However, the problem of designing Sybil-proof auction-based incentive mechanisms for crowdsensing is still open. Moreover, the Sybil attack model in crowdsensing is yet to be formally defined.

In this paper, we focus on designing Sybil-proof auction-based incentive mechanisms for crowdsensing. A mechanism is Sybil-proof if, participating in crowdsensing using a single identity is a dominant strategy of each user. The main contributions of this paper are as follows:

- We are the first, to the best of our knowledge, to investigate Sybil attack in auction-based incentive mechanisms for crowdsensing. As an essential step, we formally define the Sybil attack model in crowdsensing.

This research was supported in part by NSF grants 1421685, 1444059, 1461886 and 1525920. The information reported here does not reflect the position or the policy of the federal government. J. Lin, M. Li and D. Yang are with Colorado School of Mines, Golden, CO 80401 USA (e-mail: {jilin, mili, djyang}@mines.edu). G. Xue is with Arizona State University, Tempe, AZ 85287 USA (e-mail: xue@asu.edu). J. Tang is with Syracuse University, NY 13210 (e-mail: jtang02@syr.edu).

- We analyze existing auction-based incentive mechanisms and demonstrate that all of them are vulnerable to Sybil attack.
- Depending on whether a user is willing to perform a subset of its submitted task set, we investigate both the single-minded and multi-minded cases. We design *SPIM-S* and *SPIM-M* for these two cases, respectively. In order to design *SPIM-S*, we provide a sufficient condition for a mechanism to be Sybil-proof. We prove that *SPIM-S* achieves computational efficiency, individual rationality, truthfulness, and Sybil-proofness, and that *SPIM-M* achieves individual rationality, truthfulness, and Sybil-proofness. Note that truthfulness is for both the task set and the cost.

The remainder of this paper is organized as follows. In Section II, we briefly review the related work. In Section III, we introduce the system model and the objectives. In Section IV and Section V, we present our mechanisms for single-minded case and multi-minded case in detail and prove their desired properties, respectively. We present performance evaluation in Section VI. We conclude this paper in Section VII.

## II. Related Work

Auction is an efficient method to capture and tackle the participants' strategic behaviors, and has been widely used to design incentive mechanisms. The objectives of most of the state-of-art auction-based incentive mechanisms are either maximizing the utility of the platform or the total value of the sensing tasks to the platform [4, 9, 23, 37] under a certain constraint e.g., budget, or minimizing the social cost [7, 29]. Yang *et al.* [30, 31] proposed two incentive mechanisms for both user-centric and platform-centric models using auction and Stackelberg game, respectively. Several works [9, 37, 38] have taken into consideration that smartphone users may come to a system in an on-line manner. Recently, many papers [11, 18, 28] considered the quality of the sensing data. Meanwhile, a number of works explored the privacy-preserving mechanisms in crowdsensing [12, 14] to protect users' privacy. However, none of aforementioned mechanisms take into consideration the Sybil attack.

The effects of Sybil attack on combinatorial auctions have been first analyzed in [33]. This work proved that the VCG auction is not Sybil-proof. Yokoo *et al.* [32] introduced the price-oriented rationing-free protocols, which characterize the Sybil-proof protocols for combinatorial auction. Different from [32], we consider the value of each task to the platform and characterize the cost of users to perform sensing tasks in our system model (to be elaborated in Section III-A).

The problem of designing Sybil-proof auction-based incentive mechanisms for crowdsensing is still open. Moreover, the Sybil attack model in crowdsensing is yet to be formally defined. All existing auction-based incentive mechanisms are vulnerable to Sybil attack, as explained in Section I.

## III. Model and Problem Formulation

In this section, we present an overview of our crowdsensing system, model it as a reverse auction, describe the threat models, and give our desired properties.

### A. System Model

Similar to most crowdsensing systems [7, 31], we consider a crowdsensing system consisting of a platform and a set $\mathcal{U} = \{1, 2, \ldots, n\}$ of $n \geqslant 2$ smartphone users, who are interested in performing sensing tasks. The platform first publicizes a set $\mathcal{T} = \{t_1, t_2, \ldots, t_m\}$ of $m$ sensing tasks. Each task $t_i \in \mathcal{T}$ has a value $v_i$ to the platform. We use *bundle* to refer to any subset of $\mathcal{T}$. There is a function $V(\mathcal{B})$ to calculate the value of bundle $\mathcal{B}$ to the platform, i.e., $V(\mathcal{B}) = \sum_{t_i \in \mathcal{B}} v_i, \mathcal{B} \subseteq \mathcal{T}$. Each user $i$ has a task set $\Gamma_i \subseteq \mathcal{T}$, which $i$ can perform according to its preference. Generalizing existing works, we assume that each user $i$ has a cost function $c_i(\mathcal{B})$, which determines the cost for $i$ to perform all tasks in bundle $\mathcal{B}$. The cost function $c_i(\cdot)$ satisfies the following properties:

- $c_i(\emptyset) = 0$;
- $c_i(\{t_j\}) = \infty, \forall t_j \in \mathcal{T} \setminus \Gamma_i$;
- $c_i(\mathcal{B}') \leq c_i(\mathcal{B}''), \forall \mathcal{B}', \mathcal{B}'' \subseteq \mathcal{T}$ with $\mathcal{B}' \subseteq \mathcal{B}''$;
- $c_i(\mathcal{B}) \leq c_i(\mathcal{B}') + c_i(\mathcal{B}''), \forall \mathcal{B}', \mathcal{B}'' \subseteq \mathcal{T}$ and $\mathcal{B} = \mathcal{B}' \cup \mathcal{B}''$.

The physical meanings of the first two properties are obvious. The third property implies that performing addition tasks may incur more cost. The last property guarantees that the cost of performing a set of tasks is not greater than the cost of performing these tasks separately. These four properties together closely depict the cost of performing sensing tasks in practice and serve as a base of our attack model to be defined in Section III-B.

The platform will assign each user $i \in \mathcal{U}$ a bundle $\mathcal{A}_i \subseteq \Gamma_i$ to complete. Note that $\mathcal{A}_i = \emptyset$ means user $i$ is not assigned any task to perform. Let $\overrightarrow{\mathcal{A}} = (\mathcal{A}_1, \mathcal{A}_2, \ldots, \mathcal{A}_n)$ denote the assignment profile. At last, the platform calculates the payment $p_i$ for each user $i$. Note that $p_i = 0$, if $\mathcal{A}_i = \emptyset$. Let $\overrightarrow{p} = (p_1, p_2, \ldots, p_n)$ denote the payment profile. Depending on whether a user is willing to perform its whole task set, we consider two cases in this paper. For the *single-minded* (SM) case, each user $i \in \mathcal{U}$ is willing to perform only $\Gamma_i$, and the utility of $i$ is

$$u_i = \begin{cases} p_i - c_i(\mathcal{A}_i), & \text{if } \mathcal{A}_i = \Gamma_i; \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

Since $c_i(\mathcal{A}_i)$ can only be equal to $c_i(\Gamma_i)$, we use $c_i$ instead of $c_i(\mathcal{A}_i)$ for notational simplicity. For the *multi-minded* (MM) case, each user $i \in \mathcal{U}$ is willing to perform any subset of $\Gamma_i$, and the utility of $i$ when performing bundle $\mathcal{A}_i \subseteq \mathcal{T}$ is

$$u_i = \begin{cases} p_i - c_i(\mathcal{A}_i), & \text{if } \mathcal{A}_i \subseteq \Gamma_i; \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

The utility of the platform is

$$u_0 = V(\bigcup_{i \in \mathcal{U}} \mathcal{A}_i) - \sum_{i \in \mathcal{U}} p_i. \tag{3}$$

In this paper, we model the interaction between the platform and the smartphone users as a sealed-bid reverse auction, where the platform is a buyer who buys sensing service and the smartphone users are sellers who bid to perform sensing tasks. We call user $i$ a winner if $\mathcal{A}_i \neq \emptyset$, and a loser otherwise.

Let $\beta_i = (\tilde{\Gamma}_i, b_i)$ denote the *task-cost pair* of user $i$. In the SM case, $b_i$ is a value, while $b_i$ is a cost function in the MM case. A task-cost pair is true if $\tilde{\Gamma}_i = \Gamma_i$ and $b_i = c_i(\Gamma_i)$ in the SM case; $\tilde{\Gamma}_i = \Gamma_i$ and $b_i = c_i(\cdot)$ in the MM case. At the beginning of the auction, each user $i \in \mathcal{U}$ submits its task-cost pair as its bid to the platform, which is not necessarily its true task-cost pair. Let $\overrightarrow{\beta} = (\beta_1, \beta_2, \ldots, \beta_n)$ denote the task-cost profile. Given the task-cost profile $\overrightarrow{\beta}$, the platform determines the outcome of the auction, which consists of the assignment profile $\overrightarrow{\mathcal{A}}$ and the payment profile $\overrightarrow{p}$.

### B. Threat Models

**Threats to Incentive:** We assume that users are selfish but rational. Hence it is possible that user $i$ maximizes its utility by reporting a false cost value $\tilde{b}_i$, which differs from its true cost $c_i(\Gamma_i)$ in the SM case; or reporting a false cost function $\tilde{c}_i(\cdot) \neq c_i(\cdot)$ in the MM case. Besides, user $i$ could also misreport the task set by submitting $\tilde{\Gamma}_i \neq \Gamma_i$. Other threats to incentive, e.g., collusion, are out the scope of this paper.

**Sybil Attack:** Based on our system model, a user could conduct Sybil attack by submitting multiple task-cost pairs under fictitious identities. As a simple case, user $i$ could submit two task-cost pairs $\beta_{i'} = (\tilde{\Gamma}_{i'}, \tilde{b}_{i'})$ and $\beta_{i''} = (\tilde{\Gamma}_{i''}, \tilde{b}_{i''})$ under two identities $i'$ and $i''$, respectively. This case is sufficient to represent the general Sybil attack. Depending on whether a user is interested in only performing its whole task set, we consider the following two cases.

Single-Minded Case: Each user is interested in only performing its whole task set. User $i$ submits $\beta_{i'} = (\tilde{\Gamma}_{i'}, \tilde{b}_{i'})$ and $\beta_{i''} = (\tilde{\Gamma}_{i''}, \tilde{b}_{i''})$ using identities $i'$ and $i''$, where $\tilde{\Gamma}_{i'} \cup \tilde{\Gamma}_{i''} = \Gamma_i$. User $i$'s utility $\tilde{u}_i$ through Sybil attack is

$$\tilde{u}_i = \begin{cases} p_{i'} + p_{i''} - c_i(\Gamma_i), & \text{if } \mathcal{A}_{i'} \cup \mathcal{A}_{i''} = \Gamma_i; \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

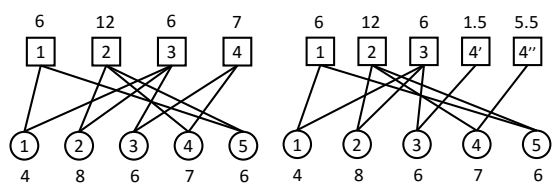
Multi-Minded Case: Each user is willing to perform any subset of its task set. User $i$ submits $\beta_{i'} = (\tilde{\Gamma}_{i'}, \tilde{b}_{i'})$ and $\beta_{i''} = (\tilde{\Gamma}_{i''}, \tilde{b}_{i''})$ using identities $i'$ and $i''$, where $\tilde{\Gamma}_{i'} \subseteq \Gamma_i$ and $\tilde{\Gamma}_{i''} \subseteq \Gamma_i$. User $i$'s utility $\tilde{u}_i$ through Sybil attack is

$$\tilde{u}_i = \begin{cases} p_{i'} + p_{i''} - c_i(\mathcal{A}_{i'} \cup \mathcal{A}_{i''}), & \text{if } \mathcal{A}_{i'} \cup \mathcal{A}_{i''} \subseteq \Gamma_i; \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

If $\tilde{u}_i > u_i$, user $i$ has an incentive to conduct Sybil attack in either case.

Note that because SM case is a special case of MM case, we have: 1) if a mechanism is not Sybil-proof in SM case, it is not Sybil-proof in MM case; 2) if a mechanism is Sybil-proof in MM case, it is Sybil-proof in SM case. Next, we show that existing mechanisms are not Sybil-proof in either SM case or MM case. We classify them into four categories according to their vulnerabilities to Sybil attack. The first category is composed of the VCG-based mechanisms [6, 8, 21, 28]. They are vulnerable to Sybil attack, since VCG auction is not Sybil-proof, as proved in [33]. In the second category, each winner is paid its critical value, which is based on a certain user's bid. All the mechanisms [4, 7, 11, 34, 36] are not Sybil-proof, since a user can exploit multiple fictitious identities to increase its critical value, and thus increase its payment. The third category consists of mechanisms [9, 29, 37, 38] where users are selected iteratively according to a ratio criterion, and a loser can become a winner by rigging the criterion value through Sybil attack. The fourth category consists of mechanisms [30, 31] where users are selected iteratively according to a linear criterion, and a user can increase its utility by completing a subset of its sensing task set. Due to space limitation, we use examples to show the vulnerabilities to Sybil attack only for the last two categories.

First, we use *MMT* [29] as an example from the third category and show that it is not Sybil-proof in SM case. *MMT* selects users iteratively. In each iteration, *MMT* selects user with the lowest $b_i/v_i(S)$ value, where $v_i(S) = V(\bigcup_{j \in S \cup \{j\}} \Gamma_j) - V(\bigcup_{j \in S} \Gamma_j)$ is user $i$'s marginal value to the platform given the selected users in $S$. Besides, *MMT* is also an $H_k$-approximation algorithm in terms of the social cost, where $H_k$ is the $k$-th harmonic number, and $k$ is the largest user task set size. Social cost is the summation of the true cost of all the selected users. We use the example in Fig. 1 to show that *MMT* is not Sybil-proof. In this example, squares represent users, and disks represent tasks. A link between a user and a task represents that the task is in that user's task set. The number above user $i$ denotes its bids for $\Gamma_i$. Since *MMT* is truthful, we assume $b_i = c_i(\Gamma_i)$. The number below task $t_j$ denotes its value to the platform. In Fig. 1 (a), we have $\mathcal{U} = \{1, 2, 3, 4\}$, $\mathcal{T} = \{t_1, t_2, t_3, t_4, t_5\}$, $\Gamma_1 = \{t_1, t_5\}$, $\Gamma_2 = \{t_2, t_4, t_5\}$, $\Gamma_3 = \{t_1, t_2, t_3\}$, $\Gamma_4 = \{t_3, t_4\}$, $b_1 = 6$, $b_2 = 12$, $b_3 = 6$, $b_4 = 7$. According to *MMT*, users 2 and 3 will be selected with the social cost 18. Note that user 4 is a loser in this case, and thus its utility is 0. Now, assume user 4 conducts Sybil attack by submitting two bids $\beta_{4'} = (\{t_3\}, 1.5))$ and $\beta_{4''} = (\{t_4\}, 5.5)$ under identities $4'$ and $4''$, respectively, as shown in Fig. 1 (b). In this case, *MMT* selects users 1, 3, $4'$, and $4''$ with the social cost 19. From this example, we see that *MMT* is not Sybil-proof in SM case, since user 4 could increase its utility through Sybil attack. Therefore, other mechanisms [7, 34, 38] similar to *MMT* in terms of user selection criterion are not Sybil-proof, either.

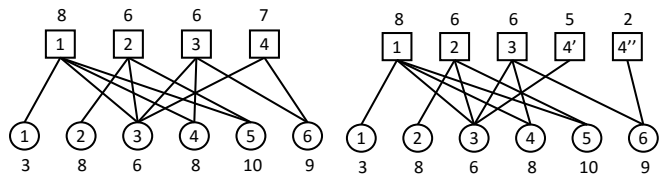


(a) No Sybil attack (b) With Sybil attack

Fig. 1. Example showing *MMT* is not Sybil-proof in SM case

Next, we use *MSensing* [31] as an example from the fourth category. *MSensing* is intentionally designed for SM case and can be proved Sybil-proof in SM case (see Appendix). However, it is not Sybil-proof in MM case. *MSensing* selects users iteratively. In each iteration, *MSensing* selects user with the highest $v_i(S) - b_i$ value, where $v_i(S)$ is the same as that defined in *MMT*. Since *MSensing* is truthful, we assume $b_i = c_i(\Gamma_i)$. In the example shown in Fig. 2 (a), *MSensing* will select users 1, 2, and 3 as winners. In this case, user 4 is

a loser with utility 0. Assume that user 4 conducts Sybil attack by submitting two bids $\beta_{4'} = (\{t_3\}, 5)$ and $\beta_{4''} = (\{t_6\}, 2)$ under identities $4'$ and $4''$, respectively, as shown in Fig. 2 (b). In this case, *MSensing* selects users 1, 2 and $4''$. Compared with the former case, user 4 changes from a loser to a winner, and thus its utility increases. This example shows that a user can increase its utility through Sybil attack in *MSensing*, and thus *MSensing* is not Sybil-proof in MM case.



(a) No Sybil attack (b) With Sybil attack

Fig. 2. Example showing *MSensing* is not Sybil-proof in MM case

### C. Desired Properties and Objective

We consider the following important properties:

- **Computational Efficiency:** A mechanism is computaionally efficient if it terminates in polynomial time.
- **Individual Rationality:** A mechanism is individually rational if each user has a non-negative utility when bidding its true task-cost pair.
- **Truthfulness:** A mechanism is truthful if any user's utility is maximized when bidding its true task-cost pair.
- **Sybil-Proofness:** A mechanism is Sybil-proof if any user's utility is maximized when bidding its true task-cost pair using a single identity.

In this paper, we aim to design Sybil-proof incentive mechanisms (SPIM), which also achieve computational efficiency, individual rationality and truthfulness.

## IV. *SPIM-S*: Sybil-Proof Incentive Mechanism for Single-Minded Case

In this section, we design and analyze *SPIM-S*, a Sybil-proof auction-based incentive mechanism for SM case.

### A. Design Rationale

In SM case, a user could maximize its utility by splitting its task set into multiple subsets and submitting these subsets using multiple identities in the hope that all the identities will be selected as winners. In order to design Sybil-proof mechanisms, we provide a sufficient condition for a mechanism to be Sybil-proof in the following lemma.

*Lemma 1:* A mechanism is Sybil-proof if it satisfies the following two conditions:

1) If any user $i$ pretends two identities $i'$ and $i''$, and both $i'$ and $i''$ are selected as winners, then $i$ should be selected as a winner while using only one identity;
2) If any user $i$ pretends two identities $i'$ and $i''$, the payment to $i$ should not be less than the summation of the payments to $i'$ and $i''$.

*Proof:* According to the first condition, if any user $i$ pretends two identities $i'$ and $i''$, and both $i'$ and $i''$ are selected as winners, then $i$ is a winner with assigned tasks $\mathcal{A}_i$. According to (1), user $i$'s utility is $u_i = p_i - c_i(\mathcal{A}_i)$. According to (4), the utility of $i$ through Sybil attack is $\tilde{u}_i = p_{i'} + p_{i''} - c_i(\mathcal{A}_i)$. Because of the second condition, we have $p_{i'} + p_{i''} \leq p_i$, and thus $\tilde{u}_i \leq u_i$. Therefore user $i$ cannot increase its utility through Sybil attack. ■

The truthfulness of *SPIM-S* relies on Myerson's well-known characterization [17, 22].

**Theorem 1:** [17, 22] An auction mechanism is truthful iff:

- 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, which is the smallest value such that user $i$ would lose the auction if it bids higher than this value.

In order to satisfy the first condition in Lemma 1, *SPIM-S* should select $i$ before $i'$ and $i''$ under the same selection criterion. To guarantee this, *SPIM-S* groups users by the size of their task sets and starts from the group with the largest task set size. Because the task set size of neither $i'$ nor $i''$ is greater than that of $i$, $i$ will be selected before both $i'$ and $i''$. Following Theorem 1, in each group, *SPIM-S* selects users iteratively according to the value of a criterion function, which is non-decreasing in terms of users' bids. Besides, the payment to each winner is its critical value, which also guarantees the second condition of Lemma 1. The details of the proofs will be shown in Section IV-C.

### B. Design of SPIM-S

In this section, we describe the details of *SPIM-S*, which is illustrated in Algorithm 1. At first, *SPIM-S* groups all users by the task set size and sorts all the groups in decreasing order. *SPIM-S* starts from the group with the largest task set size. Within each group, *SPIM-S* calls WPG, as shown in Algorithm 2, to select winners and calculate their payments. *SPIM-S* repeatedly calls WPG until all the tasks are assigned or all the groups are processed.

As a fundamental part of *SPIM-S*, WPG consists of two phases: *winner selection* and *payment determination*. The inputs to WPG are a set $\mathcal{R}$ of sensing tasks to be assigned, a group $\mathcal{G}_k$ of users, and a submitted task-bid profile $\overrightarrow{\beta_k}$ by users in $\mathcal{G}_k$. The output is a tuple consisting of an assignment profile $\overrightarrow{\mathcal{A}_k}$, a payment profile $\overrightarrow{p_k}$, and a set $\mathcal{T}_k$ of assigned tasks of users in $\mathcal{G}_k$. In the winner selection phase, WPG selects winners iteratively. Given the set $\mathcal{R}$ of unassigned tasks, let $v_i(\mathcal{R}) = V(\mathcal{R} \cap \Gamma_i)$ denote the marginal value of user $i$ to the platform. Let $[i]$ denote the winner selected in the $i$-th iteration such that the value of the criterion function $Q_S(v_{[i]}(\mathcal{R}_{[i]}), b_{[i]})$ is the minimum over $\mathcal{G}_k \setminus \mathcal{S}_{[i]}$, where

$$\mathcal{S}_{[i]} = \begin{cases} \{[1], [2], \ldots, [i-1]\}, & i \geq 2; \\ \emptyset, & i = 1, \end{cases}$$

and $\mathcal{R}_{[i]} = \mathcal{R} \setminus \bigcup_{j \in \mathcal{S}_{[i]}} \Gamma_j$. Note that we prefer to select users with higher marginal values but lower bids, and thus the criterion function $Q_S : \mathbb{R}_{\geq 0} \times \mathbb{R}_{\geq 0} \to \mathbb{R}$ could be any function that satisfies the following properties:

- $Q_S(x, y)$ is non-increasing with respect to $x$;
- $Q_S(x, y)$ is non-decreasing with respect to $y$.

This implies that

$$Q_S(v_{[1]}, b_{[1]}) \leq Q_S(v_{[2]}, b_{[2]}) \leq Q_S(v_{[3]}, b_{[3]}) \leq \cdots \quad (6)$$

For notational simplicity, we use $v_{[i]}$ instead of $v_{[i]}(R_{[i]})$.

In the payment determination phase, WPG computes the payment $p_i$ to each winner $i$, i.e., $\mathcal{A}_i \neq \emptyset$. It processes the users in $\mathcal{G}_k \setminus \{i\}$ similarly to how it selects users in the winner selection phase. In the $j$-th iteration, let $i_j$ denote the selected user. WPG uses a payment function $P_S : \mathbb{R}_{\geq 0} \times \mathbb{R} \to \mathbb{R}_{\geq 0}$ to compute the maximum bid, with which user $i$ can be selected as a winner instead of $i_j$. The payment function $P_S$ could be any function that satisfies the following properties:

- $P_S(x, y)$ is non-decreasing with respect to $y$;
- $\forall x, y \in \mathbb{R}_{\geq 0}, \forall z \in \mathbb{R}$, if $Q_S(x, y) = z, P_S(x, z) = y$.

Note that given a value of criterion function $Q_S$, the higher the maximum bid is, the higher the payment should be. This process repeats until no user's submitted cost is less than its marginal value or no user is left in $\mathcal{G}_k \setminus \{i\}$. Let $v_{i_j} = V(\mathcal{R}_{i_j} \cap \Gamma_{i_j})$ denote the marginal value of the $j$-th winner to the platform in the payment determination phase, where $\mathcal{R}_{i_j}$ is a set of unassigned tasks before $i_j$ is selected. Let $K$ denote the number of iterations of the while-loop in the payment determination phase of WPG. Therefore, we have $K$ values, with which user $i$ could have been selected as a winner in one of the $K$ iterations if $i$ was considered. We set the payment $p_i$ to the maximum of these $K$ values and the marginal value of $i$ after these $K$ iterations.

**Algorithm 1:** *SPIM-S*

**Input**: Sensing task set $\mathcal{T}$, user set $\mathcal{U}$, bid profile $\overrightarrow{\beta}$, criterion function $Q_S$ and payment function $P_S$.
**Output**: Assignment profile $\overrightarrow{\mathcal{A}}$ and payment profile $\overrightarrow{p}$.

1 $\mathcal{R} \leftarrow \mathcal{T}$, $p_i \leftarrow 0$, $\mathcal{A}_i \leftarrow \emptyset$, $\forall i \in \mathcal{U}$;
2 Group users by the task set size, and sort these groups in decreasing order $\mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_l$;
3 $k \leftarrow 1$;
4 **while** $\mathcal{R} \neq \emptyset$ **and** $k \leq l$ **do**
5 $\quad (\overrightarrow{\mathcal{A}_k}, \overrightarrow{p_k}, \mathcal{T}_k) \leftarrow$ WPG$(\mathcal{R}, \mathcal{G}_k, \overrightarrow{\beta_k})$;
6 $\quad \mathcal{R} \leftarrow \mathcal{R} \setminus \mathcal{T}_k$;
7 $\quad k \leftarrow k + 1$;
8 **end**
9 **return** $\overrightarrow{\mathcal{A}}$ *and* $\overrightarrow{p}$.

### C. Analysis of SPIM-S

In this section, we analyze the properties of *SPIM-S*.

**Theorem 2:** *SPIM-S* is computationally efficient, individually rational, truthful and Sybil-proof in SM case.

We prove this theorem with the following lemmas.

*Lemma 2: SPIM-S* is computationally efficient.

*Proof:* The running time of *SPIM-S* is dominated by the while-loop (Lines 4-8). Because there are $m$ tasks and each winner should contribute at least one new task to be selected, the number of winners is at most $m$. Thus the while-loop will run at most $m$ iterations. The running time of WPG is dominated by the for-loop (Lines 8-18), which is bounded by bounded by $O(nm^3)$. Because finding the user with minimum criterion value takes $O(nm^2)$ time and the number of winners is at most $m$. Therefore, the total computational complexity of *SPIM-S* is bounded by $O(nm^4)$, since Algorithm 1 will call WPG at most $m$ times. ■

**Algorithm 2:** Winner Selection and Payment Determination in a Group (WPG)

**Input**: Sensing task set $\mathcal{R}$, user group $\mathcal{G}_k$, bid profile $\overrightarrow{\beta_k}$, criterion function $Q_S$ and payment function $P_S$.
**Output**: Assignment profile $\overrightarrow{\mathcal{A}_k}$, payment profile $\overrightarrow{p_k}$ and a set of assigned tasks $\mathcal{T}_k$.

// **Winner Selection Phase**
1 $\mathcal{R}' \leftarrow \mathcal{R}$, $\mathcal{T}_k \leftarrow \emptyset$, $\mathcal{G}'_k \leftarrow \mathcal{G}_k$;
2 $i \leftarrow \arg\min_{j \in \mathcal{G}'_k} Q_S(v_j(\mathcal{R}'), b_j)$;
3 **while** $b_i \leq v_i$ **and** $\mathcal{G}'_k \neq \emptyset$ **do**
4 $\quad \mathcal{A}_i \leftarrow \Gamma_i$, $\mathcal{T}_k \leftarrow \mathcal{T}_k \cup \Gamma_i$, $\mathcal{R}' \leftarrow \mathcal{R}' \setminus \Gamma_i$;
5 $\quad \mathcal{G}'_k \leftarrow \mathcal{G}'_k \setminus \{i\}$;
6 $\quad i \leftarrow \arg\min_{j \in \mathcal{G}'_k} Q_S(v_j(\mathcal{R}'), b_j)$;
7 **end**
// **Payment Determination Phase**
8 **foreach** $i \in \mathcal{G}_k$ *s.t.* $\mathcal{A}_i \neq \emptyset$ **do**
9 $\quad \mathcal{G}'_k \leftarrow \mathcal{G}_k \setminus \{i\}$, $\mathcal{R}' \leftarrow \mathcal{R}$;
10 $\quad i_j \leftarrow \arg\min_{j \in \mathcal{G}'_k} Q_S(v_j(\mathcal{R}'), b_j)$;
11 $\quad$ **while** $\mathcal{G}'_k \neq \emptyset$ **and** $\mathcal{R}' \neq \emptyset$ **do**
12 $\quad\quad$ **if** $b_{i_j} > v_{i_j}$ **then break**;
13 $\quad\quad p_i \leftarrow \max\{p_i, \min\{P_S\big(v_i(\mathcal{R}'), Q_S(v_{i_j}(\mathcal{R}'), b_{i_j})\big), v_i(\mathcal{R}')\}\}$;
14 $\quad\quad \mathcal{R}' \leftarrow \mathcal{R}' \setminus \Gamma_{i_j}$, $\mathcal{G}'_k \leftarrow \mathcal{G}'_k \setminus \{i_j\}$;
15 $\quad\quad i_j \leftarrow \arg\min_{j \in \mathcal{G}'_k} Q_S(v_j(\mathcal{R}'), b_j)$;
16 $\quad$ **end**
17 $\quad$ **if** $b_i \leq v_i(\mathcal{R}')$ **then** $p_i \leftarrow \max\{p_i, v_i(\mathcal{R}')\}$;
18 **end**
19 **return** $(\overrightarrow{\mathcal{A}_k}, \overrightarrow{p_k}, \mathcal{T}_k)$.

*Remarks:* Note that, the running time of *SPIM-S* is only linear in the number of users $n$. In crowdsensing systems, $n$ is usually very large, whereas the number of sensing tasks $m$ is much less than $n$. Thus *SPIM-S* is very efficient.

*Lemma 3: SPIM-S* is individually rational.

*Proof:* Let $[i]$ denote the winner selected in the $i$-th iteration in the winner determination phase of WPG. If $[i]$ is the last winner in $\mathcal{G}_k$, there is no winner selected in the $i$-th iteration in the payment determination phase. According to Line 17 in Algorithm 2, we have $b_{[i]} \leq v_{[i]} \leq p_{[i]}$. Otherwise, let $[i]_i$ denote the winner selected in the $i$-th iteration when processing users in $\mathcal{G}_k \setminus \{i\}$. Since user $[i]_i$ would not be selected in the $i$-th iteration if $[i]$ is considered, according to (6), we have $Q_S(v^{[i]}_{[i]_i}, b_{[i]}) \leq Q_S(v_{[i]_i}, b_{[i]_i})$. Thus we have $b_{[i]} = P_S(v^{[i]}_{[i]_i}, Q_S(v^{[i]}_{[i]_i}, b_{[i]})) \leq P_S(v^{[i]}_{[i]_i}, Q_S(v_{[i]_i}, b_{[i]_i}))$, where the equality is due to the second property of function $P_S$, and the inequality is due to the first property of function $P_S$. We also have $b_{[i]} \leq v_{[i]} = v^{[i]}_{[i]_i}$, since user $[i]$ is the

winner selected in the $i$-th iteration. It follows that $b_{[i]} \leq \min\{P_S(v^{[i]}_{[i]_i}, Q_S(v_{[i]_i}, b_{[i]_i})), v^{[i]}_{[i]_i}\} \leq p_{[i]}$, where the second inequality is because of Line 13 in Algorithm 2. Therefore, $u_{[i]} = p_{[i]} - b_{[i]} \geq 0$, and *SPIM-S* is individually rational. ■

*Lemma 4: SPIM-S* is truthful.

*Proof:* We first prove that user $i$ cannot increase its utility by submitting a false task set. Then, we prove that user $i$ cannot increase its utility by submitting a false cost. We assume that user $i$ submits a false bid $\tilde{\beta}_i = (\tilde{\Gamma}_i, \tilde{b}_i)$. If user $i$ submits a false task set $\tilde{\Gamma}_i \subset \Gamma_i$, the utility of $i$ is 0 according to (1). On the contrary, if $\Gamma_i \subset \tilde{\Gamma}_i$, user $i$ will not be paid because it cannot finish all the tasks in $\tilde{\Gamma}_i$. Therefore, there is no incentive for $i$ to submit a false task set $\tilde{\Gamma}_i$.

For the truthfulness of the submitted cost, by Theorem 1, it suffices to prove that the selection rule of *SPIM-S* is monotone and the payment to each winner is its critical value. It is obvious that the selection rule is monotone, since according to the criterion function $Q_S$, the criterion value of a user will not increase if it bids a smaller value. Next, we prove that the payment $p_i$ to winner $i$ is its critical value. Note that

$$p_i = \max\left\{\max_{1\leq j\leq K}\left(P_S\left(v^i_{i_j}, Q_S(v_{i_j}, b_{i_j})\right)\right), v_i(\mathcal{R}_{i_{K+1}})\right\}, \quad (7)$$

where $\mathcal{R}_{i_{K+1}}$ is a set of unassigned tasks after $K$ iterations. If user $i$ bids $b_i > p_i$, we have $b_i > P_S\left(v^i_{i_j}, Q_S(v_{i_j}, b_{i_j})\right)$, which implies $Q_S(v^i_{i_j}, b_{i_j}) < Q_S(v^i_{i_j}, b_i)$. Thus $i$ will not be selected within $K$ iterations. We also have $b_i > v_i(\mathcal{R}_{i_{K+1}})$, thus $i$ will still not be selected after $K$ iterations. Therefore, $p_i$ is the critical value for user $i$.

Since no user can increase its utility by submitting a false task-cost pair, *SPIM-S* is truthful. ■

*Lemma 5: SPIM-S* is Sybil-proof.

*Proof:* To prove *SPIM-S* is Sybil-proof, we show that *SPIM-S* satisfies the sufficient conditions in Lemma 1. Suppose user $i$ submits $(\Gamma_{i'}, b_{i'})$ and $(\Gamma_{i''}, b_{i''})$ using two fictitious identities $i'$ and $i''$, respectively.

We first prove that *SPIM-S* satisfies the fist condition in Lemma 1. The following discussion is focused on the winner selection phase of WPG. We assume that both $i'$ and $i''$ are winners. This implies that $\Gamma_{i'} \subset \Gamma_i$ and $\Gamma_{i''} \subset \Gamma_i$, since one will make the other lose otherwise. Thus, both $i'$ and $i''$ will be in the group(s) after $i$'s group. Let $\tilde{\mathcal{R}}'$ and $\tilde{\mathcal{R}}''$ denote the set of unassigned tasks before $i'$ and $i''$ are selected, respectively. According to *SPIM-S*, we have $v_{i'}(\tilde{\mathcal{R}}') \geq b_{i'}$ and $v_{i''}(\tilde{\mathcal{R}}'') \geq b_{i''}$, since both $i'$ and $i''$ are winners. In addition, due to the truthfulness of *SPIM-S* and the fourth property of the cost function, we have $b_i = c_i \leq c_{i'} + c_{i''} = b_{i'} + b_{i''}$. Because $\Gamma_i = \Gamma_{i'} \cup \Gamma_{i''}$, $i$ will be considered before the group(s), which $i'$ and $i''$ should belong to. Let $\tilde{\mathcal{R}}$ denote the set of unassigned tasks when $i$ is considered. Therefore we have $\tilde{\mathcal{R}}' \cup \tilde{\mathcal{R}}'' \subseteq \tilde{\mathcal{R}}$. It follows that $v_{i'}(\tilde{\mathcal{R}}') \leq v_{i'}(\tilde{\mathcal{R}})$ and $v_{i''}(\tilde{\mathcal{R}}'') \leq v_{i''}(\tilde{\mathcal{R}})$, due to the decreasing property of $v_i(\mathcal{R})$. Meanwhile, we have $v_{i'}(\tilde{\mathcal{R}}) + v_{i''}(\tilde{\mathcal{R}}) \leq v_i(\tilde{\mathcal{R}})$, since $\Gamma_i = \Gamma_{i'} \cup \Gamma_{i''}$. Therefore, $v_i(\tilde{\mathcal{R}}) \geq v_{i'}(\tilde{\mathcal{R}}') + v_{i''}(\tilde{\mathcal{R}}'') \geq b_{i'} + b_{i''} \geq b_i$. This implies that $i$ is still a winner while using a single identity. Thus the first condition in Lemma 1 is satisfied.

We next prove that *SPIM-S* satisfies the second condition in Lemma 1. The following discussion is focused on the payment determination phase of WPG. Since $i$ is still a winner while using one identity, let $\tilde{\mathcal{R}}_i$ denote the set of unassigned tasks before $i$ is selected. Let $\tilde{\mathcal{R}}_{i'}$ and $\tilde{\mathcal{R}}_{i''}$ denote the set of unassigned tasks before $i'$ and $i''$ are selected, respectively. In addition, there is no user in $i$'s group can make $i$ lose, i.e., $b_i > v_i(\tilde{\mathcal{R}}_i)$, otherwise neither $i'$ nor $i''$ can be a winner. Therefore, the payment to $i$ is at least $v_i(\mathcal{R}_{i_{K+1}})$ according to (7). Recall that $K$ is the number of iterations of the while-loop in the payment determination phase of WPG. In any iteration $r \leq K$, we have $b_{i_r} \leq v_{i_r}$. Due to the properties of functions $P_S$ and $Q_S$ and the decreasing property of $v_i(\mathcal{R})$, we have $\min\{P_S(v^i_{i_r}, Q_S(v_{i_r}, b_{i_r})), v^i_{i_r}\} \leq v^i_{i_r} \leq v_i(\tilde{\mathcal{R}}_i)$. After $K$ iterations, $v_i(\mathcal{R}_{i_{K+1}}) \leq v_i(\tilde{\mathcal{R}}_i)$, due to the decreasing property of $v_i(\mathcal{R})$. Thus it follows that $p_i \leq v_i(\tilde{\mathcal{R}}_i)$. Similarly, we have $p_{i'} \leq v_{i'}(\tilde{\mathcal{R}}_{i'})$ and $p_{i''} \leq v_{i''}(\tilde{\mathcal{R}}_{i''})$. Let $\mathcal{R}_{i_{K+1}}$ denote the set of unassigned tasks after $K$ iterations. Since $\Gamma_i = \Gamma_{i'} \cup \Gamma_{i''}$, $\tilde{\mathcal{R}}_{i'} \subseteq \mathcal{R}_{i_{K+1}}$ and $\tilde{\mathcal{R}}_{i''} \subseteq \mathcal{R}_{i_{K+1}}$, we have $v_{i'}(\tilde{\mathcal{R}}_{i'}) + v_{i''}(\tilde{\mathcal{R}}_{i''}) \leq v_{i'}(\mathcal{R}_{i_{K+1}}) + v_{i''}(\mathcal{R}_{i_{K+1}}) \leq v_i(\mathcal{R}_{i_{K+1}})$. Thus, we have $p_i \geq v_i(\mathcal{R}_{i_{K+1}}) \geq v_{i'}(\tilde{\mathcal{R}}_{i'}) + v_{i''}(\tilde{\mathcal{R}}_{i''}) \geq p_{i'} + p_{i''}$. Hence, the second condition in Lemma 1 is satisfied.

Therefore, *SPIM-S* is Sybil-proof according to Lemma 1. We can use a similar proof for the case where a user pretends more than two identities. ■

## V. *SPIM-M*: Sybil-Proof Incentive Mechanism for Multi-Minded Case

In this section, we design and analyze *SPIM-M*, a Sybil-proof auction-based incentive mechanism for MM case.

### A. Design Rationale

In MM case, a user is willing to perform any subset of its task set, and tries to maximize its utility by submitting multiple task-cost pairs using fictitious identities. We design *SPIM-M* based on the characterization of Sybil-proof mechanisms in [32]. *SPIM-M* guarantees that the utility of a user is not less than its utility while using multiple identities. To achieve this, *SPIM-M* first calculates the payments to each user for any subset of its task set. The payment is determined independently of its own cost function. At last, *SPIM-M* assigns each user a subset of its task set that maximizes its utility independently of the assignments to other users.

### B. Design of SPIM-M

In this section, we elaborate *SPIM-M*, as illustrated in Algorithm 3.

At first, *SPIM-M* uses a payment function $P_M(x, y) = x - \max\{0, y\}$ to calculate the payment $p_{i,\mathcal{B}}$ to user $i$ for any bundle $\mathcal{B} \subseteq \Gamma_i$. $P_M$ is based on the value of the criterion function $Q_M(V(\mathcal{B}), c_j(\mathcal{B}))$, where $Q_M(x, y) = x - y$, $V(\mathcal{B})$ is the value of bundle $\mathcal{B}$ to the platform, and $c_j(\mathcal{B})$ is the cost of bundle $\mathcal{B}$ to any user $j \in \mathcal{U} \setminus \{i\}$. Note that, user $i$ will only be assigned a bundle $\mathcal{B} \subseteq \Gamma_i$, since $c_i(\{t_j\}) = \infty$ for

**Algorithm 3:** *SPIM-M*

**Input**: Sensing task set $\mathcal{T}$, user set $\mathcal{U}$, bid profile $\overrightarrow{\beta}$, criterion function $Q_M$ and payment function $P_M$.
**Output**: Assignment profile $\overrightarrow{\mathcal{A}}$ and payment profile $\overrightarrow{p}$.

**foreach** $i \in \mathcal{U}$ **do**
  Calculate the payment to $i$ for any bundle $\mathcal{B} \subseteq \Gamma_i$, $p_{i,\mathcal{B}} \leftarrow P_M\left(V(\mathcal{B}), \max_{j\neq i, \mathcal{B}'\subseteq\Gamma_j, \mathcal{B}'\cap\mathcal{B}\neq\emptyset} Q_M(V(\mathcal{B}'), c_j(\mathcal{B}'))\right)$;
**end**
**foreach** $i \in \mathcal{U}$ **do**
  $\mathcal{A}_i \leftarrow \arg\max_{\mathcal{B}\subseteq\Gamma_i}(p_{i,\mathcal{B}} - c_i(\mathcal{B}))$;
  $p_i \leftarrow p_{i,\mathcal{A}_i}$;
**end**
**return** $\overrightarrow{\mathcal{A}}$ and $\overrightarrow{p}$.

any task $t_j \in \mathcal{T} \setminus \Gamma_i$. Note that the payment to user $i$ for any bundle $\mathcal{B} \subseteq \Gamma_i$ is independent of its cost function $c_i(\cdot)$, i.e.,

$$p_{i,\mathcal{B}} = V(\mathcal{B}) - \max\{0, \max_{j\neq i, \mathcal{B}'\subseteq\Gamma_j, \mathcal{B}'\cap\mathcal{B}\neq\emptyset}(V(\mathcal{B}') - c_j(\mathcal{B}'))\}. \quad (8)$$

At last, *SPIM-M* will assign each user $i$ a set $\mathcal{A}_i$ of tasks, which is a bundle $\mathcal{B} \subseteq \Gamma_i$ maximizing its utility based on the calculated payment, i.e.,

$$\mathcal{A}_i = \arg\max_{\mathcal{B}\subseteq\Gamma_i}(p_{i,\mathcal{B}} - c_i(\mathcal{B})). \quad (9)$$

*SPIM-M* gives each user $i$ a payment $p_i = p_{i,\mathcal{A}_i}$. Note that $p_i = p_{i,\mathcal{A}_i} = 0$, if $\mathcal{A}_i = \emptyset$.

### C. Analysis of SPIM-M

In this section, we analyze the properties of *SPIM-M*.

**Theorem 3:** *SPIM-M* is individually rational, truthful and Sybil-proof in MM case.

We prove this theorem with the following lemmas.

*Lemma 6: SPIM-M* is individually rational.

*Proof:* According to (9), any user $i$ is assigned a bundle $\mathcal{A}_i$, which is a subset of $\Gamma_i$ maximizing $i$'s utility. Since $p_{i,\emptyset} = 0$ and $c_i(\emptyset) = 0$, the utility of any user $i$ is non-negative according to (2). Thus *SPIM-M* is individually rational. ■

*Lemma 7: SPIM-M* is truthful.

*Proof:* We first prove that user $i$ cannot increase its utility by submitting a false task set. Then, we prove that user $i$ cannot increase its utility by submitting a false cost function. We assume that user $i$ submits a false bid $\tilde{\beta}_i = (\tilde{\Gamma}_i, \tilde{c}_i(\cdot))$. According to (8), the payment to $i$ for any bundle $\mathcal{B} \subseteq \Gamma_i$ is calculated independently of $i$'s cost function. If $\tilde{\Gamma}_i \subset \Gamma_i$, the payment to $i$ for any subset of $\tilde{\Gamma}_i$ is the same as that when $i$ submits $\Gamma_i$. Meanwhile, according to (9), *SPIM-M* will assign $i$ a bundle $\mathcal{A}_i$, which maximizes $i$'s utility. Therefore, user $i$ cannot increase its utility by submitting $\tilde{\Gamma}_i \subset \Gamma_i$. On the contrary, if $\Gamma_i \subset \tilde{\Gamma}_i$, user $i$ cannot finish the assigned tasks in $\mathcal{A}_i \setminus \Gamma_i \neq \emptyset$, because of the second property of the cost function. It follows that $p_i = 0$ in this case. Hence there is no incentive for $i$ to submit a false task set $\tilde{\Gamma}_i$.

Furthermore, the false cost function $\tilde{c}_i(\cdot)$ can only affect the result of (9). Let $\tilde{\mathcal{A}}_i$ and $\mathcal{A}_i$ denote the bundles assigned to $i$ when $i$ submits the false cost function $\tilde{c}_i(\cdot)$ and the true cost function $c_i(\cdot)$, respectively. It is obvious that the utility of $i$ will not change if $\tilde{\mathcal{A}}_i = \mathcal{A}_i$. On the contrary, if $\tilde{\mathcal{A}}_i \neq \mathcal{A}_i$, according to $i$'s true cost function, we have $p_{i,\tilde{\mathcal{A}}_i} - c_i(\tilde{\mathcal{A}}_i) \leq p_{i,\mathcal{A}_i} - c_i(\mathcal{A}_i)$. This is due to the fact that both $\mathcal{A}_i$ and $\tilde{\mathcal{A}}_i$ are the subset of $\Gamma_i$, and $\mathcal{A}_i$ is the bundle maximizing $i$'s utility. Therefore, user $i$ cannot increase its utility by submitting a false cost function $\tilde{c}_i(\cdot)$. Thus *SPIM-M* is truthful. ■

*Lemma 8: SPIM-M* is Sybil-proof.

*Proof:* We assume that user $i$ pretends two identities $i'$ and $i''$, and both $i'$ and $i''$ are assigned $\mathcal{A}_{i'}$ and $\mathcal{A}_{i''}$, respectively. Let $m_{i'}$ denote $\max_{j\neq i', \mathcal{B}'\subseteq\Gamma_j, \mathcal{B}'\cap\mathcal{A}_{i'}\neq\emptyset}(V(\mathcal{B}') - c_j(\mathcal{B}'))$, and $m_{i''}$ denote $\max_{j\neq i'', \mathcal{B}''\subseteq\Gamma_j, \mathcal{B}''\cap\mathcal{A}_{i''}\neq\emptyset}(V(\mathcal{B}'') - c_j(\mathcal{B}''))$. According to (8), the payments to $i'$ and $i''$ are

$$p_{i',\mathcal{A}_{i'}} = V(\mathcal{A}_{i'}) - \max\{0, m_{i'}\},$$
$$p_{i'',\mathcal{A}_{i''}} = V(\mathcal{A}_{i''}) - \max\{0, m_{i''}\}.$$

If $i$ is assigned a set $\mathcal{A}_i = \mathcal{A}_{i'} \cup \mathcal{A}_{i''}$ of tasks while using a single identity, the payment to $i$ is

$$p_{i,\mathcal{A}_i} = V(\mathcal{A}_i) - \max\left\{0, \max_{j\neq i, \mathcal{B}\subseteq\Gamma_j, \mathcal{B}\cap\mathcal{A}_i\neq\emptyset}(V(\mathcal{B}) - c_j(\mathcal{B}))\right\}.$$

Since $\mathcal{A}_i = \mathcal{A}_{i'} \cup \mathcal{A}_{i''}$, we know that

$$\begin{aligned}&\{\mathcal{B} | j \neq i, \mathcal{B} \subseteq \Gamma_j, \mathcal{B} \cap \mathcal{A}_i \neq \emptyset\}\\ =&\{\mathcal{B}' | j \neq i', \mathcal{B}' \subseteq \Gamma_j, \mathcal{B}' \cap \mathcal{A}_{i'} \neq \emptyset\}\\ &\cup \{\mathcal{B}'' | j \neq i'', \mathcal{B}'' \subseteq \Gamma_j, \mathcal{B}'' \cap \mathcal{A}_{i''} \neq \emptyset\}.\end{aligned}$$

Let $m_i$ denote $\max_{j\neq i, \mathcal{B}\subseteq\Gamma_j, \mathcal{B}\cap\mathcal{A}_i\neq\emptyset}(V(\mathcal{B}) - c_j(\mathcal{B}))$. Thus we have $m_i = \max\{m_{i'}, m_{i''}\}$ and $m_i \leq m_{i'} + m_{i''}$. In addition, we can prove that $\mathcal{A}_{i'} \cap \mathcal{A}_{i''} = \emptyset$ by contradiction. Assume $\mathcal{A}_{i'} \cap \mathcal{A}_{i''} \neq \emptyset$, the payments to $i'$ is

$$p_{i',\mathcal{A}_{i'}} = V(\mathcal{A}_{i'}) - \max\{0, m_{i'}\} \leq V(\mathcal{A}_{i'}) - (V(\mathcal{A}_{i''}) - c_i(\mathcal{A}_{i''})),$$

where the inequality is due to the fact $V(\mathcal{A}_{i''}) - c_i(\mathcal{A}_{i''}) \leq m_{i'}$, since $\mathcal{A}_{i''}$ is in $\{\mathcal{B}' | j \neq i', \mathcal{B}' \subseteq \Gamma_j, \mathcal{B}' \cap \mathcal{A}_{i'} \neq \emptyset\}$. Similarly, we have $p_{i'',\mathcal{A}_{i''}} \leq V(\mathcal{A}_{i''}) - (V(\mathcal{A}_{i'}) - c_i(\mathcal{A}_{i'}))$. Therefore the summation of the utilities of $i'$ and $i''$ is

$$\begin{aligned}u_{i'} + u_{i''} &= p_{i',\mathcal{A}_{i'}} - c_i(\mathcal{A}_{i'}) + p_{i'',\mathcal{A}_{i''}} - c_i(\mathcal{A}_{i''})\\ &\leq V(\mathcal{A}_{i'}) - (V(\mathcal{A}_{i''}) - c_i(\mathcal{A}_{i''})) - c_i(\mathcal{A}_{i'})\\ &\quad + V(\mathcal{A}_{i''}) - (V(\mathcal{A}_{i'}) - c_i(\mathcal{A}_{i'})) - c_i(\mathcal{A}_{i''}) = 0.\end{aligned}$$

Since *SPIM-M* is individually rational, it follows that $\mathcal{A}_{i'} = \mathcal{A}_{i''} = \emptyset$, which contradicts the assumption. Thus $\mathcal{A}_{i'} \cap \mathcal{A}_{i''} = \emptyset$. We also have $V(\mathcal{A}_i) = V(\mathcal{A}_{i'} \cup \mathcal{A}_{i''}) = V(\mathcal{A}_{i'}) + V(\mathcal{A}_{i''})$, since $\mathcal{A}_i = \mathcal{A}_{i'} \cup \mathcal{A}_{i''}$. According to (8), the payment to $i$ is

$$\begin{aligned}p_{i,\mathcal{A}_i} &= V(\mathcal{A}_i) - \max\{0, m_i\}\\ &\geq V(\mathcal{A}_i) - (m_{i'} + m_{i''})\\ &= V(\mathcal{A}_{i'} \cup \mathcal{A}_{i''}) - (m_{i'} + m_{i''})\\ &= V(\mathcal{A}_{i'}) - m_{i'} + V(\mathcal{A}_{i''}) - m_{i''}\\ &\geq p_{i',\mathcal{A}_{i'}} + p_{i'',\mathcal{A}_{i''}}.\end{aligned}$$

This implies that if a user pretends two identities and is assigned tasks separately, its payment will not increase. Due to the fourth property of the cost function, we have $c_i(\mathcal{A}_i) \leq c_i(\mathcal{A}_{i'}) + c_i(\mathcal{A}_{i''})$, since $\mathcal{A}_i = \mathcal{A}_{i'} \cup \mathcal{A}_{i''}$. Therefore, according to (2) and (5), the utility of $i$ when using two identities is not greater than that obtained by using a single identity.

Thus *SPIM-M* is Sybil-proof. ■

*Remarks:* From the proof in Lemma 8, we can also prove $\mathcal{A}_i \cap \mathcal{A}_j = \emptyset$ for any $i, j \in \mathcal{U}, i \neq j$. In addition, the time complexity is $O(n \cdot 2^{|\Gamma_{max}|})$, where $n$ is the number of users, and $|\Gamma_{max}|$ is the largest user task set size, since *SPIM-M* will calculate the payments to each user for every subset of its task set. In practice, the execution time of *SPIM-M* is still reasonable, as shown in Section VI, since the number of tasks each user can perform is limited by some constraints, e.g., travel budget [19], and thus very small.

## VI. Performance Evaluation

In this section, we compare the performances of *SPIM-S* and *SPIM-M* with *MMT* and *MSensing*. Specifically, we implement *SPIM-S* with the same criterion and payment function as in *MMT* [29], i.e., $Q_S(x, y) = y/x$ and $P_S(x, y) = xy$. Note that the criterion of *SPIM-M* is the same as in *MSensing* [31]. The performance metrics are *running time*, *total payment*, and *platform utility*.

### A. Evaluation Setup

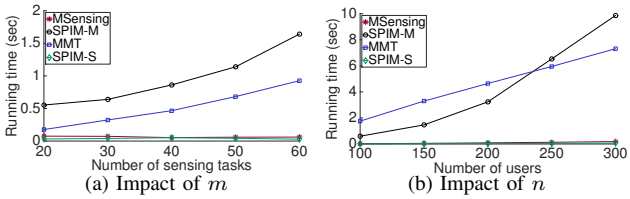
We use a real data set for evaluation. It consists of the traces of 320 taxi drivers, who work in the center of Rome [3]. Each trace is represented by a sequence of locations. Each taxi driver has a tablet, which periodically retrieves the GPS locations and sends them with the corresponding driver ID to a server. The mobility pattern of taxi traces can be used to depict the mobility of smartphone users as in [29].

We consider a crowdsensing system where the tasks are to measure the Wi-Fi signal strength at specific locations. Each user can sense the Wi-Fi signal strength within 30 meters from its location. Tasks are represented by GPS locations reported by taxis. We assume that all drivers are willing to participate in the crowdsensing system. We preprocess the tasks such that each task can be sensed by at least two users to prevent the monopoly and guarantee the quality of sensing task.

In our evaluation, we randomly select locations on taxi drivers' traces as the sensing tasks. We assume the value of each task is uniformly distributed over $[1, 5]$, and users' cost for each task is uniformly distributed over $[1, 10]$. To evaluate the impact of the number of sensing tasks ($m$) on the performance metrics, we fix the number of users ($n$) at 200 and vary $m$ from 20 to 60 with a step of 10. To evaluate the impact of the number of users on the performance metrics, we fix $m$ at 150 and vary $n$ from 100 to 300 with a step of 50. All the results are averaged over 1000 independent runs.

### B. Evaluation of Running Time

Fig. 3 shows the impacts of $m$ and $n$ on the running time. We see that the running time of *SPIM-S*, *SPIM-M*, *MMT*, and *MSensing* all increase with the increase of $m$ and $n$. In both Fig. 3 (a) and Fig. 3 (b), the running time of *SPIM-M* is more than that of *SPIM-S*, *MMT*, and *MSensing*. This is because *SPIM-M* calculates the payment to each user for every subset of its task set. In addition, the running time of *SPIM-S* is less than that of *MMT*, though they use the same criterion. This is because *SPIM-S* starts from the group with the largest task set size, and thus may finish assigning tasks earlier than *MMT*.

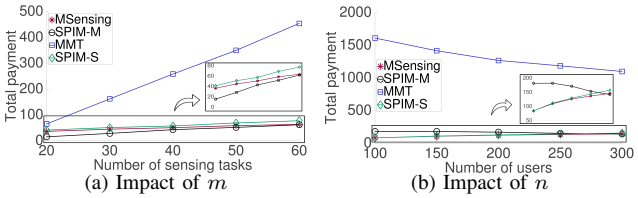


(a) Impact of $m$ (b) Impact of $n$

Fig. 3. Running time

### C. Evaluation of Total Payment

Fig. 4 plots the impacts of $m$ and $n$ on the total payment to users. In Fig. 4 (a), we see that the total payment of *SPIM-S*, *SPIM-M*, *MMT*, and *MSensing* both increase with the increase of $m$. This is because, with more tasks, the platform may select more users to perform the tasks, which incurs a higher payment. In Fig. 4 (b), we observe that the total payments of *SPIM-M* and *MMT* decrease with the increase of $n$. This is because, with more users, the platform may find more low-cost users to perform the tasks. Note that the total payment of *MMT* is larger than those of others. The reason is that *MMT* selects a user as long as its marginal value is nonzero, and thus may select more users incurring a higher payment. In addition, the total payments of *SPIM-S* and *MSensing* all increase slightly with the increase of $n$. This is because, with more users, *SPIM-S* and *MSensing* can assign more tasks, which incurs a higher payment.

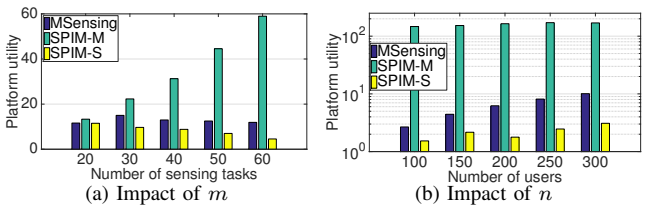


(a) Impact of $m$ (b) Impact of $n$

Fig. 4. Total payment

### D. Evaluation of Platform Utility

Fig. 5 shows the impacts of $m$ and $n$ on the platform utility. Note that the y-axis in Fig. 5 (b) is log-scaled. The platform utility of *MMT* is negative, and thus omitted from Fig. 5. The reason is *MMT* will select a user as long as its marginal value to the platform is not zero, which may make the total payment to users higher than the total value to the platform. In both Fig. 5 (a) and Fig. 5 (b), we see that the platform utility achieved by *SPIM-M* is larger than those achieved by *SPIM-S* and *MSensing*. This is because *SPIM-M* assigns each task to at most one user, and thus avoids paying users to perform duplicated tasks. This advantage is amplified with large $m$.



(a) Impact of $m$ (b) Impact of $n$

Fig. 5. Platform utility

## VII. Conclusion

In this paper, we proposed two Sybil-proof auction-based incentive mechanisms for crowdsensing. We designed *SPIM-S* and *SPIM-M* for single-minded case and multi-minded case, respectively. Specifically, *SPIM-S* achieves computational efficiency, individual rationality, truthfulness and Sybil-proveness. *SPIM-M* achieves individual rationality, truthfulness and Sybil-proveness. We rigorously proved the desired properties of the mechanisms and validated them through simulations.

## Appendix

**Theorem 4:** *MSensing* is Sybil-proof in SM case.

*Proof:* We prove that *MSensing* satisfies the conditions in Lemma 1. Suppose a user $i$ pretends two identities $i'$ and $i''$ and submits $(\Gamma_{i'}, b_{i'})$ and $(\Gamma_{i''}, b_{i''})$, respectively. We assume $i'$ and $i''$ are winners and get payments $p_{i'}$ and $p_{i''}$, respectively. Let $v_{i'}$ and $v_{i''}$ denote the marginal value of $i'$ and $i''$ to the platform when they are selected. Let $\mathcal{S}_{i'}$ denote the winners after $i'$ is selected. Since *MSensing* is truthful, we have $b_i = c_i$. In addition, we have $b_{i'} \leq v_{i'}$ and $b_{i''} \leq v_{i''}$, since both $i'$ and $i''$ are winners. Because $\Gamma_i = \Gamma_{i'} \cup \Gamma_{i''}$, we have $v^i_{i'} \geq v_{i'} + v_{i''}$, where $v^i_{i'}$ is user $i$'s marginal value in the iteration where $i'$ is selected. It implies that $v^i_{i'} \geq b_{i'} + b_{i''} = c_{i'} + c_{i''} \geq c_i = b_i$. Meanwhile, $v_{i'} - b_{i'}$ is maximum over $\mathcal{U} \setminus \mathcal{S}_{i'}$ since $i'$ is a winners. Therefore, $v^i_{i'} - b_i \geq (v_{i'} + v_{i''}) - (b_{i'} + b_{i''}) \geq v_{i'} - b_{i'}$. Thus, $i$ would have been selected as the winner in the iteration where $i'$ is selected, if $i$ used one identity. This satisfies the first condition in Lemma 1. We next prove that *MSensing* satisfies the second condition in Lemma 1. First, we know that $i$ will be selected as a winner no later than the iteration where $i'$ is selected, while using one identity. In addition, we can use the same method, as in Lemma 5, to prove that $p_i \geq p_{i'} + p_{i''}$ in *MSensing*. Thus *MSensing* is Sybil-proof in SM case. ■

## References


[1] "Gigwalk," http://www.gigwalk.com/.

[2] "Waze," https://www.waze.com/.

[3] L. Bracciale, M. Bonola, P. Loreti, G. Bianchi, R. Amici, and A. Rabuffi, "CRAWDAD data set roma/taxi (v. 2014-07-17)," Downloaded from http://crawdad.org/roma/taxi/, Jul. 2014.

[4] C. Chen and Y. Wang, "SPARC: Strategy-proof double auction for mobile participatory sensing," in *CloudCom-Asia*, 2013, pp. 133–140.

[5] J. R. Douceur, "The sybil attack," in *IPTPS*. Springer, 2002, pp. 251–260.

[6] Z. Feng, Y. Zhu, and L. M. Ni, "imac: Strategy-proof incentive mechanism for mobile crowdsourcing," in *WASA*, 2013, pp. 337–350.

[7] Z. Feng, Y. Zhu, Q. Zhang, L. M. Ni, and A. V. Vasilakos, "TRAC: Truthful auction for location-aware collaborative sensing in mobile crowdsourcing," in *INFOCOM*, 2014, pp. 1231–1239.

[8] L. Gao, F. Hou, and J. Huang, "Providing long-term participation incentive in participatory sensing," in *INFOCOM*, 2015, pp. 2803–2811.

[9] K. Han, C. Zhang, J. Luo, M. Hu, and B. Veeravalli, "Truthful scheduling mechanisms for powering mobile crowdsensing," *IEEE Trans. Comput.*, vol. 65, no. 1, pp. 294–307, 2016.

[10] L. G. Jaimes, I. Vergara-Laurens, and A. Raij, "A crowd sensing incentive algorithm for data collection for consecutive time slot problems," in *LATINCOM*, 2014, pp. 1–5.

[11] H. Jin, L. Su, D. Chen, K. Nahrstedt, and J. Xu, "Quality of information aware incentive mechanisms for mobile crowd sensing systems," in *MobiHoc*, 2015.

[12] H. Jin, L. Su, H. Xiao, and K. Nahrstedt, "INCEPTION: incentivizing privacy-preserving data aggregation for mobile crowd sensing systems," in *MobiHoc*, 2016, pp. 341–350.

[13] I. Koutsopoulos, "Optimal incentive-driven design of participatory sensing systems," in *INFOCOM*, 2013, pp. 1402–1410.

[14] J. Lin, D. Yang, M. Li, J. Xu, and G. Xue, "BidGuard: A framework for privacy-preserving crowdsensing incentive mechanisms," in *CNS*, 2016, pp. 439–447.

[15] T. Luo, H.-P. Tan, and L. Xia, "Profit-maximizing incentive for participatory sensing," in *INFOCOM*, 2014, pp. 127–135.

[16] P. Mohan, V. N. Padmanabhan, and R. Ramjee, "Nericell: rich monitoring of road and traffic conditions using mobile smartphones," in *SenSys*, 2008, pp. 323–336.

[17] R. B. Myerson, "Optimal auction design," *Mathematics of operations research*, vol. 6, no. 1, pp. 58–73, 1981.

[18] D. Peng, F. Wu, and G. Chen, "Pay as how well you do: A quality based incentive mechanism for crowdsensing," in *MobiHoc*, 2015.

[19] L. Pournajaf, L. Xiong, V. Sunderam, and S. Goryczka, "Spatial task assignment for crowd sensing with cloaked locations," in *MDM*, vol. 1, 2014, pp. 73–82.

[20] R. K. Rana, C. T. Chou, S. S. Kanhere, N. Bulusu, and W. Hu, "Ear-phone: an end-to-end participatory urban noise mapping system," in *IPSN*, 2010, pp. 105–116.

[21] H. Shah-Mansouri and V. W. Wong, "Profit maximization in mobile crowdsourcing: A truthful auction mechanism," in *ICC*, 2015, pp. 3216–3221.

[22] Y. Singer, "Budget feasible mechanisms," in *FOCS*, 2010, pp. 765–774.

[23] J. Sun and H. Ma, "A behavior-based incentive mechanism for crowd sensing with budget constraints," in *ICC*, 2014, pp. 1314–1319.

[24] B. Viswanath, A. Post, K. P. Gummadi, and A. Mislove, "An analysis of social network-based sybil defenses," in *SIGCOMM*, 2010, pp. 363–374.

[25] G. Wang, B. Wang, T. Wang, A. Nika, H. Zheng, and B. Y. Zhao, "Defending against sybil devices in crowdsourced mapping services," in *MobiSys*, 2016, pp. 179–191.

[26] Q. Wang, B. Ye, B. Tang, S. Guo, and S. Lu, "eBay in the clouds: False-name-proof auctions for cloud resource allocation," in *ICDCS*, 2015, pp. 153–162.

[27] Q. Wang, B. Ye, B. Tang, T. Xu, S. Guo, S. Lu, and W. Zhuang, "ALETHEIA: Robust large-scale spectrum auctions against false-name bids," in *MobiHoc*, 2015, pp. 27–36.

[28] Y. Wen, J. Shi, Q. Zhang, X. Tian, Z. Huang, H. Yu, Y. Cheng, and X. Shen, "Quality-driven auction based incentive mechanism for mobile crowd sensing," *IEEE Trans. Veh. Technol.*, vol. 64, no. 9, pp. 4203–4214, 2015.

[29] J. Xu, J. Xiang, and D. Yang, "Incentive mechanisms for time window dependent tasks in mobile crowdsensing," *IEEE Transactions on Wireless Communications*, vol. 14, no. 11, pp. 6353–6364, 2015.

[30] D. Yang, G. Xue, G. Fang, and J. Tang, "Incentive mechanisms for crowdsensing: Crowdsourcing with smartphones," *IEEE/ACM Trans. Netw.*, vol. 24, no. 3, pp. 1732–1744, 2016.

[31] D. Yang, G. Xue, X. Fang, and J. Tang, "Crowdsourcing to smartphones: incentive mechanism design for mobile phone sensing," in *MobiCom*, 2012, pp. 173–184.

[32] M. Yokoo, "Characterization of strategy/false-name proof combinatorial auction protocols: price-oriented, rationing-free protocol," in *IJCAI*, 2003, pp. 733–742.

[33] M. Yokoo, Y. Sakurai, and S. Matsubara, "The effect of false-name bids in combinatorial auctions: New fraud in internet auctions," *Games and Economic Behavior*, vol. 46, no. 1, pp. 174–188, 2004.

[34] Q. Zhang, Y. Wen, X. Tian, X. Gan, and X. Wang, "Incentivize crowd labeling under budget constraint," in *INFOCOM*, 2015, pp. 2812–2820.

[35] X. Zhang, G. Xue, D. Yang, and R. Yu, "A sybil-proof and time-sensitive incentive tree mechanism for crowdsourcing," in *GLOBECOM*, 2015, pp. 1–6.

[36] X. Zhang, G. Xue, R. Yu, D. Yang, and J. Tang, "Truthful incentive mechanisms for crowdsourcing," in *INFOCOM*, 2015, pp. 2830–2838.

[37] 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.*, vol. 25, no. 12, pp. 3190–3200, 2014.

[38] D. Zhao, X.-Y. Li, and H. Ma, "How to crowdsource tasks truthfully without sacrificing utility: Online incentive mechanisms with budget constraint," in *INFOCOM*, 2014, pp. 1213–1221.