

# Data-Centric Mobile Crowdsensing

Changkun Jiang, Lin Gao, *Senior Member, IEEE*, Lingjie Duan, *Senior Member, IEEE*, and Jianwei Huang, *Fellow, IEEE*

**Abstract**—Mobile crowdsensing (MCS) is a promising sensing paradigm that leverages the diverse embedded sensors in massive mobile devices. A key objective in MCS is to efficiently schedule mobile users to perform multiple sensing tasks. Prior work mainly focused on interactions between the task-layer and the user-layer, without considering tasks' similar data requirements and users' heterogeneous sensing capabilities. In this work, we propose a three-layer data-centric MCS model by introducing a new data-layer between tasks and users, enable different tasks to reuse the same data items, hence effectively leverage both task similarities and user heterogeneities. We formulate a joint task selection and user scheduling problem based on the new framework, aiming at maximizing social welfare. We first analyze the centralized optimization problem with the statistical information of tasks and users, and show the bound of the social welfare gain due to data reuse. Then we consider the two-sided information asymmetry of selfish task-owners and users, and propose a decentralized market mechanism for achieving the centralized social optimality. In particular, considering the NP-hardness of the optimization, we propose a truthful two-sided randomized auction mechanism that ensures computational efficiency and a close-to-optimal performance. Simulations verify the effectiveness of our proposed model and mechanism.

**Index Terms**—Mobile Crowdsensing, Data Reuse and Analysis, Incentive Mechanism Design, Randomized Auction.



## 1 INTRODUCTION

### 1.1 Background and Motivations

THE proliferation of hand-held mobile devices with rich embedded sensors has enabled a new sensing paradigm known as *Mobile CrowdSensing (MCS)* [2], where individual mobile users are involved in performing the sensing tasks by using their mobile devices. Due to the low deploying cost and the high sensing coverage, this new sensing paradigm has attracted a broad range of applications such as urban dynamic mining, public safety, and environment monitoring [3]–[5]. In a general multi-task MCS system (e.g., PRISM [6] and Medusa [7]), each sensing task is first initiated and announced by a task planner (task owner) via a web portal. Then the task is assigned to a pool of mobile users (registered in the system), who will perform the sensing task accordingly (e.g., sensing the required data and sending the collected data to the system). While performing a sensing task, mobile users consume their own device resources such as battery energy and CPU time, hence incur certain costs. Thus, users may not be willing to participate in MCS, unless they receive proper rewards to compensate their costs.

Many prior studies (e.g., [8]–[14]) have studied the problem of incentivizing users to participate in the MCS system. These works focused on the interactions of tasks and users (e.g., the assignment of tasks among users through a proper matching), without considering common data requirements

(hence the potential data reuse) among multiple tasks and heterogeneous sensing capabilities of different users. In a practical system, however, there is a high likelihood that multiple tasks require some common data [2]. For example, the road traffic data at a particular time and location may be useful for Waze, Uber, and Google Traffic simultaneously. Therefore, it is likely to cause duplicated data sensing and processing in a multi-task scenario, if multiple tasks are completed separately by the same user. Moreover, in a practical system, users may have different sensing capabilities due to factors such as locations and device types. For example, it is easier for a user to sense the data close to her current location. Thus, it is more flexible and efficient to schedule users on the data level than on the task level.

To complete multiple tasks more efficiently, it is critical to identify the common data requirements of these tasks and enable the reuse of sensory data across different tasks. Some practical MCS platforms (e.g., PRISM [6] and Medusa [7]) have allowed task developers to specify their data requirements in a high-level language. Then, they identify and reuse the common data across multiple tasks in order to reduce or avoid duplicated sensing and processing. There are several advantages enable data reuse in the MCS system. First, data is digital goods and can be reused without additional cost. Second, multiple tasks can share a large pool of mobile users collectively through the platform. Third, by reusing data across different tasks, the overall system efficiency can be improved. A similar MCS architecture has been discussed in [2] as a future vision. However, [2] did not provide any theoretical framework or analysis about the performance gain that can be achieved through data reuse.

### 1.2 Novelty and Contributions

In this work, we propose a novel three-layer *data-centric* MCS model, consisting of a data layer, a task layer, and a user layer, which is different from traditional two-layer *task-centric* models in [8]–[14] (with the task layer and the user

- Part of the results have appeared in IEEE GLOBECOM, Dec. 2016 [1].
- Changkun Jiang and Jianwei Huang are with the Network Communications and Economics Lab, Department of Information Engineering, The Chinese University of Hong Kong, Shatin, N.T., Hong Kong, China. E-mail: {jc012, jwhuang}@ie.cuhk.edu.hk
- Lin Gao is with the School of Electronic Information and Engineering, Harbin Institute of Technology (Shenzhen), Shenzhen, China. E-mail: gaolin@hitsz.edu.cn
- Lingjie Duan is with the Engineering Systems and Design Pillar, Singapore University of Technology and Design, 8 Somapah Road, Singapore. E-mail: lingjie\_duan@sutd.edu.sg
- This work is supported by the General Research Funds (CUHK 14206315 and CUHK 14219016) established by the Hong Kong UGC.

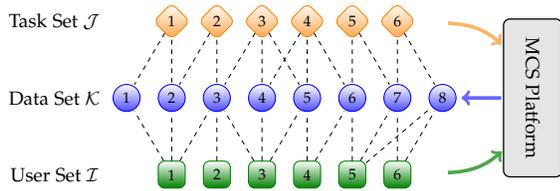


Fig. 1. Three-layer data-centric mobile crowdsensing model. The MCS platform acts as the social planner to maximize the total social welfare.

layer only). Specifically, in our data-centric model, tasks and users are connected through the data layer, that is, *each task is translated to a set of data items that it requires, and each user is connected to a set of data items that she can sense. Moreover, different tasks may require a common data item (hence can reuse the data item reported by users), and different users may be able to sense the same data item (hence compete with each other for the sensing opportunity)*. Thus, it is able to leverage both the *task similarity* (in terms of data requirements) and the *user heterogeneity* (in terms of sensing capabilities). Fig. 1 illustrates such a crowdsensing model with 6 tasks, 6 users, and 8 data items, where task 1 requires data items  $\{1, 2\}$ , and user 1 is able to sense data items  $\{1, 2, 3\}$  simultaneously.<sup>1</sup>

In this model, the MCS platform (social planner) collects the data requirements of tasks and the sensing capabilities of users, and then decides whether and how to complete these tasks by a proper set of users efficiently. Formally,

- Which tasks can be completed?
- Which users will be scheduled to sense which data?

We focus on the optimal task selection and user scheduling that maximize the social welfare, where the social welfare is the difference between the total values of completed tasks and the total costs of scheduled users. We are interested in understanding two key questions. The first question is what is the performance gain due to data reuse? Such a gain depends on the numbers of tasks and users as well as their data requirements. We want to analytically derive the performance gain for any given sets of tasks and users. Solving this problem is very challenging, as it is NP-hard due to the combinatorial nature. Moreover, it requires the complete information regarding the task values and users' sensing costs, which are often private information of task owners and users, respectively. Hence the second related question is how to achieve the optimal performance gain in a practical scenario with a limited computational capability and incomplete information? One approach is to design a *truthful* incentive mechanism to elicit such private information from both task owners and users. However, some well-known truthful incentive mechanisms such as the standard VCG auction [15] are not suitable for our problem due to the high computational complexity.

To answer the above two questions, we first conduct a statistical analysis for the bound of the performance gain due to data reuse, and show that such a gain can be quite significant. To reach the optimal performance bound, a social planner needs to make a centralized decision on behalf of all task owners and users. However, in practice,

1. In our three-layer model, a common data item required by different tasks only needs to be sensed once. This is the key difference between our model and the related literature (e.g., [10], [14]).

task owners and users are selfish and unwilling to report their private information about task values and sensing cost, which makes the centralized implementation infeasible. To address this issue, we will design an incentive mechanism that satisfies the individual rationality and incentive compatibility for all task owners and users. Such a mechanism also needs to have a low computational complexity and ensures a proper budget balance. To satisfy all the above requirements, we propose a two-sided randomized auction that is tractable for theoretical performance analysis.

Specifically, we resort to the *randomized auction* framework [16] for our mechanism design, with the MCS platform acting as the *auctioneer* and the participating task owners and users acting as the *bidders*. We propose a truthful randomized auction, consisting of (i) a randomized allocation rule, which picks up an "allocation" (i.e., a feasible solution to the task selection and user scheduling) randomly from a set of feasible solutions according to some probability distribution, and (ii) a payment rule, which assigns a payment for each task owner and user under the chosen allocation. Randomized auctions have been adopted for the resource allocation in wireless networking [17], covering problems [18], cloud computing [19], and electricity markets [20]. The key difference between our randomized auction and those in [17]–[20] is that our auction is *two-sided*, i.e., we need to decide both the task selection (task values) and the user scheduling (sensing costs) under *mutual information asymmetry*; while the auction models in [17]–[20] are *single-sided* (i.e., considering either values or costs), hence are not directly applied to our problem setting.

The proposed randomized auction is truthful (in expectation), in the sense that task owners and users have no incentives to misreport their private task values and sensing costs, respectively. We further show that the proposed truthful randomized auction is computationally efficient, as both the allocation rule and payment rule can be computed in polynomial time. In summary, we list the key results and the corresponding section numbers in Table 1. Our main results and key contributions are summarized as follows.

- *Novel Data-Centric Crowdsensing Model*: To our best knowledge, this is the first work that analytically exploits data reuse across multiple tasks and analyzes the performance bound due to data reuse in MCS. We propose a novel three-layer data-centric model, which leverages both the task similarity and the user sensing heterogeneity.
- *Performance Bound Analysis of Data Reuse*: Our analytical result shows that the lower bound of the social welfare gain due to data reuse is 2 for a single reusable data item, when the number of tasks and the number of users are comparable to each other. That is, the social welfare is at least doubled by exploiting data reuse across tasks for a single reusable data item. As the number of data items increases, the social welfare gain due to data reuse also increases.
- *Randomized Auction Mechanism for Incomplete Information and Limited Computation*: To address the complexity issue of the joint task selection and user scheduling and elicit the two-sided private information from both task owners and users, we propose a truthful

TABLE 1  
Key Results In This Paper

Mechanism	Truthfulness	Individual Rationality	Efficiency	Budget Balance	Complexity	Section no.
1: Two-sided VCG auction	Truthful	Yes	Optimal	No	NP-hard	4.2
2: Fractional VCG auction	Truthful	Yes	Fractional optimal	No	Polynomial	4.3
3: Randomized auction	Truthful in expectation	Yes	Close-to-optimal	No	Polynomial	4.3
4: Randomized auction (reserve price)	Truthful in expectation	Yes	Close-to-optimal	Yes	Polynomial	5.2

two-sided randomized auction mechanism, which is computationally efficient, individually rational, and incentive compatible (truthful) in expectation. We further design a randomized auction mechanism with a reserve price to achieve the budget balance, with a slightly reduced social welfare.

- *Observations and Insights:* Simulations show that (i) the social welfare gain due to data reuse increases with the task similarity and reaches up to 1300% in our simulations, and (ii) our proposed randomized auction achieves at least 90% of the maximum social welfare. Furthermore, the increase of the task similarity increases the social welfare with data reuse, as the required number of users performing the tasks can be reduced. However, the increase of the task similarity decreases the social welfare without data reuse, due to the increased user competition.

The rest of the paper is organized as follows. In Section 2, we present the system model. In Section 3, we theoretically analyze the performance bound. In Section 4, we propose the two-sided auction framework to address the incomplete information problem. In Section 5, we further propose the budget balanced auction design. We present the simulation results in Section 6, and conclude in Section 7.

## 2 SYSTEM MODEL

In this section, we first present the crowdsensing platform model, task model, and user model. Then we formulate the social welfare maximization problem.

### 2.1 Crowdsensing Platform Model

We consider a general multi-task MCS platform consisting of a set  $\mathcal{J} = \{1, 2, \dots, J\}$  of tasks, a set  $\mathcal{I} = \{1, 2, \dots, I\}$  of mobile users, and a set  $\mathcal{K} = \{1, 2, \dots, K\}$  of target data items. Each data item  $k \in \mathcal{K}$  is characterized by a set of fine-grained parameters such as the data type, location, and time.<sup>2</sup> Each task  $j \in \mathcal{J}$  is associated with a set of data requirements  $\mathcal{K}_j \subseteq \mathcal{K}$ , and each user  $i \in \mathcal{I}$  is able to sense a specific set  $\mathcal{S}_i \subseteq \mathcal{K}$  of data items.<sup>3</sup> As different tasks can reuse the same data item, there may exist two tasks  $j_1$  and  $j_2$  with overlapping data requirements, i.e.,  $\mathcal{K}_{j_1} \cap \mathcal{K}_{j_2} \neq \emptyset$ . Fig. 1 illustrates such a three-layer data-centric MCS model.

The crowdsensing model operates in a time-slotted manner. We divide the whole time period into multiple *time slots*,

2. A data item can be the humidity of a room at 11 am, the traffic of a highway at 6 pm, or a raw sensor reading such as GPS and light sensor.

3. For example, Waze [3] requires accelerometers to measure traffic speeds and WiFi/cellular readings for localizations, and OpenSignal [4] requires WiFi/cellular readings to measure their coverages. A smartphone can provide the accelerometer data and WiFi/cellular readings.

where each time slot can be an hour or a day, depending on the data precisions of tasks or users. At the beginning of each time slot, (i) each task owner registers her task on the platform, indicating the data requirements of the task and the potential value that she can achieve when the task is completed; and (ii) each user reports her information on the platform, indicating the sensing capability of the user (i.e., the set of data items that she can sense) and the potential cost for sensing any subset of data items within her capability. After collecting the reported information from all task owners and users, the platform decides the task selection (i.e., selecting a set of tasks to be completed) and the user scheduling (i.e., scheduling a set of users to sense the associated data items of the selected tasks).<sup>4</sup>

### 2.2 Task Model

Recall that each task  $j \in \mathcal{J}$  is associated with a set of data requirements  $\mathcal{K}_j \subseteq \mathcal{K}$  in the time slot that we focus on, and a task value  $v_j > 0$  when it is completed. The task value  $v_j$  is the *private information* of task  $j$ , and cannot be observed by the platform, users, or other tasks. This is one of the two key challenges for optimizing a crowdsensing system with data reuse. We assume that a task  $j$  is completed if and only if each of its required data items in  $\mathcal{K}_j$  has been sensed by at least one user. Let  $z_j \in \{0, 1\}$  denote whether a task  $j \in \mathcal{J}$  is completed, and  $y_k \in \{0, 1\}$  denote whether a data item  $k \in \mathcal{K}$  is sensed by at least one user. Then, for each task  $j \in \mathcal{J}$ , we have the following constraint:

$$z_j \leq y_k, \quad \forall k \in \mathcal{K}_j. \quad (1)$$

Given a feasible task selection  $\mathbf{z} = (z_j, j \in \mathcal{J})$ , the total achieved value (of all completed tasks) is:

$$V(\mathbf{z}) = \sum_{j \in \mathcal{J}} v_j \cdot z_j. \quad (2)$$

### 2.3 User Model

Recall that each user  $i \in \mathcal{I}$  is able to sense a set  $\mathcal{S}_i$  of data items in the time slot that we focus on. The platform can schedule user  $i$  to sense a subset  $\mathcal{S} \subseteq \mathcal{S}_i$  of data items within her sensing capability, associated with a sensing cost  $c_i(\mathcal{S})$ . Let  $x_i(\mathcal{S}) \in \{0, 1\}$  denote whether a user  $i$  is scheduled to sense a data set  $\mathcal{S} \subseteq \mathcal{S}_i$ . When  $\mathcal{S} = \emptyset$ , then  $x_i(\emptyset) = 1$  denotes that user  $i$  is not scheduled to sense any data set, hence has a zero sensing cost, i.e.,  $c_i(\emptyset) = 0$ .

We assume that a user can only be scheduled to sense one data set within her capability in the target time slot. That is, for each user  $i \in \mathcal{I}$ , we have the following *user scheduling constraint*:

$$\sum_{\mathcal{S} \subseteq \mathcal{S}_i} x_i(\mathcal{S}) = 1. \quad (3)$$

4. In this work, we focus on investigating the problem in one time slot, and assume no correlations among different time slots.

If a user is scheduled to sense multiple data sets, say  $\mathcal{S}_i^1$  and  $\mathcal{S}_i^2$ , it is equivalent to scheduling the user to sense the data set  $\mathcal{S}_i^1 \cup \mathcal{S}_i^2$ . Let  $\mathbf{x}_i = (x_i(\mathcal{S}), \mathcal{S} \subseteq \mathcal{S}_i)$  denote the scheduling vector of user  $i$ . Then, given a feasible user scheduling  $\mathbf{x} = (\mathbf{x}_i, i \in \mathcal{I})$ , the total incurred cost (of all scheduled users) is:

$$C(\mathbf{x}) = \sum_{i \in \mathcal{I}} \sum_{\mathcal{S} \subseteq \mathcal{S}_i} c_i(\mathcal{S}) \cdot x_i(\mathcal{S}). \quad (4)$$

Let  $y_{ki} \in \{0, 1\}$  denote whether a data item  $k$  is sensed by a user  $i$ , that is,  $y_{ki} = \sum_{\mathcal{S} \subseteq \mathcal{S}_i: k \in \mathcal{S}} x_i(\mathcal{S})$ . Recall that  $y_k \in \{0, 1\}$  denotes whether a data item  $k \in \mathcal{K}$  is sensed by at least one user. Then, for each data item  $k \in \mathcal{K}$ , we have

$$y_k \leq \sum_{i \in \mathcal{I}} y_{ki}. \quad (5)$$

Moreover, we denote  $\mathbf{c}_i = (c_i(\mathcal{S}), \mathcal{S} \subseteq \mathcal{S}_i)$  as the sensing cost vector of user  $i$  for all possible subsets of data items that she can sense. In practice, the sensing cost vector  $\mathbf{c}_i$  is the *private information* of user  $i$ , and cannot be observed by the platform, task owners, or other users. This is the second key challenge for optimizing a crowdsensing system with data reuse. Besides the task values and the user sensing costs, all the other information (i.e., the data requirement  $\mathcal{K}_j$  of task  $j$  and the sensing capability  $\mathcal{S}_i$  of user  $i$ ) are *public information* to the MCS platform. This is because both task owners and users need to first register with the MCS platform, and have no incentives to misreport the information.<sup>5</sup>

## 2.4 Social Welfare Maximization

The social welfare  $W(\mathbf{x}, \mathbf{z})$  is defined as the difference between the total value  $V(\mathbf{z})$  of all completed tasks and the total sensing cost  $C(\mathbf{x})$  of all scheduled users, i.e.,

$$W(\mathbf{x}, \mathbf{z}) = V(\mathbf{z}) - C(\mathbf{x}). \quad (6)$$

The objective of the platform is to decide the best task selection  $\mathbf{z}$  and user scheduling  $\mathbf{x}$  to maximize the social welfare  $W(\mathbf{x}, \mathbf{z})$ . Formally, we can formulate such a joint task selection and user scheduling problem (P1) as follows.

$$\begin{aligned} \text{P1:} \quad & \max_{\mathbf{x}, \mathbf{y}, \mathbf{z}} V(\mathbf{z}) - C(\mathbf{x}) \\ & \text{s.t. (1)-(5), } \quad \forall i \in \mathcal{I}, j \in \mathcal{J}, k \in \mathcal{K}; \\ & \text{var. } x_i(\mathcal{S}) \in \{0, 1\}, \quad \forall \mathcal{S} \in \mathcal{S}_i, i \in \mathcal{I}; \\ & \quad z_j \in \{0, 1\}, \quad \forall j \in \mathcal{J}; \\ & \quad y_k \in \{0, 1\}, \quad \forall k \in \mathcal{K}. \end{aligned}$$

Here  $\mathbf{y} \triangleq (y_k, k \in \mathcal{K})$  is an intermediate variable denoting whether each data item is sensed (by at least one user), which bridges the relationship between the task selection and the user scheduling. It is easy to see that Problem P1 is a binary integer linear programming problem. Let  $\{\mathbf{x}^o, \mathbf{y}^o, \mathbf{z}^o\}$  denote the optimal solution to P1. For presentation clarity, we will also write  $\{\mathbf{x}^o, \mathbf{y}^o, \mathbf{z}^o\}$  as  $\{\mathbf{x}^o(\mathbf{c}, \mathbf{v}), \mathbf{y}^o(\mathbf{c}, \mathbf{v}), \mathbf{z}^o(\mathbf{c}, \mathbf{v})\}$ , as all of them are functions of the user sensing costs  $\mathbf{c} = (c_i, i \in \mathcal{I})$  and the task values  $\mathbf{v} = (v_j, j \in \mathcal{J})$ .

There are two main issues that we are interested in investigating. First, what is the performance gain via data reuse?<sup>6</sup>

5. For task owner  $j \in \mathcal{J}$ , under-reporting the data requirement  $\mathcal{K}_j$  means that her data will never be completed (which leads to a task value of 0), and over-reporting  $\mathcal{K}_j$  causes additional cost for achieving the same task value. For user  $i \in \mathcal{I}$ , under-reporting the sensing capability  $\mathcal{S}_i$  weakens her own competitiveness, and over-reporting  $\mathcal{S}_i$  can be easily detected by the MCS platform.

6. Due to the page limit, we put the problem formulation for the social welfare maximization *without* data reuse to Appendix A [28].

Second, how to achieve the above performance gain in a practical scenario with a limited computational capability and incomplete information? Solving Problem P1 is very challenging. Problem P1 is NP-hard (as the special case of P1 can be reduced to the set cover problem), and hence it is important to design a low-complexity approximate algorithm to find an approximate solution. Meanwhile, solving Problem P1 requires the complete system information including the data requirements and values of all tasks as well as the sensing capabilities and costs of all users. However, as we have mentioned earlier, users' sensing costs and tasks' values are their private information, and cannot be observed by the MCS platform. Thus, we need to design a truthful incentive mechanism to elicit such private information.

To this end, we will first study the performance gain of data reuse and analyze the performance bound in Section 3, where the social planner makes decisions for all users and task owners. Then we will focus on incentive mechanisms design to address the complexity and incomplete information issues in Section 4.

## 3 PERFORMANCE BOUND ANALYSIS OF DATA REUSE

In this section, we analyze the performance bound with data reuse. We start with the simplest case of one data item, the analysis of which provides us insights into the more general case. We will consider multiple tasks and multiple users, with explicitly closed results derived for the case of two tasks and two users. Then we will consider the more general case of multiple data items, multiple users, and multiple tasks through numerical studies.

### 3.1 Order Statistics Basics

The analysis in Section 3 will rely on the tools from Order Statistics [21], the basics of which will be reviewed in this subsection.

Let  $X_1, X_2, \dots, X_n$  be  $n$  random variables sampled from a continuous distribution with the p.d.f.  $f(x)$  and the c.d.f.  $F(x)$ . The corresponding order statistics are the sequence arranged in the nondecreasing order. The smallest of the sample is denoted by  $X_{1:n}$ , i.e.,  $X_{1:n} = \min(X_1, X_2, \dots, X_n)$ , the  $m$ -th smallest of the sample is denoted by  $X_{m:n}$ , and finally the largest of the sample is denoted by  $X_{n:n}$ , i.e.,  $X_{n:n} = \max(X_1, X_2, \dots, X_n)$ . Then we have  $X_{1:n} \leq \dots \leq X_{m:n} \leq \dots \leq X_{n:n}$ . The p.d.f. of  $X_{m:n}$  for  $1 \leq m \leq n$  is

$$f_{m:n}(x) = n \binom{n-1}{m-1} (F(x))^{m-1} (1 - F(x))^{n-m} f(x). \quad (7)$$

Now we derive the joint distribution of all  $n$  order statistics and the joint distribution of the first  $s$  ( $1 \leq s \leq n$ ) order statistics, respectively. First notice that if  $F(x)$  is continuous, then with probability 1 the order statistics of the samples take *distinct* values. Hence it is reasonable to assume the realizations of the  $n$  order statistics  $X_{1:n} \leq X_{2:n} \leq \dots \leq X_{n:n}$  to be  $x_{1:n} < x_{2:n} < \dots < x_{n:n}$ . This means that the original random variables  $X_1, X_2, \dots, X_n$  are restrained to take on the values  $x_{m:n}$  ( $m = 1, 2, \dots, n$ ), which by symmetry assigns the equal probability for each of the  $n!$  permutations

of  $(1, 2, \dots, n)$ . Hence, we have the joint density function of all  $n$  order statistics to be

$$f_{1,2,\dots,n:n}(x_1, x_2, \dots, x_n) = n! \prod_{m=1}^n f(x_m), x_1 < \dots < x_n. \quad (8)$$

Furthermore, for the first  $s$  ( $1 \leq s \leq n$ ) order statistics  $X_{1:n} \leq \dots \leq X_{s:n}$ , by symmetry we need to assign the equal probability for each of the  $n!$  permutations of  $(1, 2, \dots, n)$ , and then take into account the first  $s$ , i.e.,  $(1, 2, \dots, s)$ . Hence, we have the joint density function of the first  $s$  ( $1 \leq s \leq n$ ) order statistics to be

$$f_{1,2,\dots,s:n}(x_1, x_2, \dots, x_s) = n! \prod_{m=1}^s f(x_m), x_1 < \dots < x_s. \quad (9)$$

Similarly, if we define the nonincreasing order statistics as  $X_{1:n} \geq X_{2:n} \geq \dots \geq X_{n:n}$ , then the joint distribution of the first  $s$  ( $1 \leq s \leq n$ ) nonincreasing order statistics is

$$f_{1,2,\dots,s:n}(x_1, x_2, \dots, x_s) = n! \prod_{m=1}^s f(x_m), x_1 > \dots > x_s. \quad (10)$$

Given the above preliminaries, we will conduct the performance bound analysis in the following subsections.

### 3.2 Analysis for a Single Reusable Data Item

In the case with a single data item, each task requires the data item to be completed, and each user can sense the same data item. We assume that the task values  $(v_j, j \in \mathcal{J})$  follow the i.i.d. distribution with the same p.d.f.  $f(v)$ , and the user costs  $(c_i, i \in \mathcal{I})$  follow the i.i.d. distribution with the same p.d.f.  $g(c)$ .

#### 3.2.1 Analysis without Data Reuse

In the scenario without data reuse, since all tasks require the same data, user  $i$  has the same cost  $c_i$  to complete any of the tasks. We propose the following method to analyze this scenario. We sort the task values by the descending order, i.e.,  $v_{1:J} \geq v_{2:J} \geq \dots \geq v_{J:J}$ , and sort sensing costs by the ascending order, i.e.,  $c_{1:I} \leq c_{2:I} \leq \dots \leq c_{I:I}$ . Then, there is a threshold  $m$  such that the  $m$ -th task value is no greater than the  $m$ -th user cost. The social welfare maximization selection selects tasks with values  $v_{1:J}, \dots, v_{m:J}$  and users with sensing costs  $c_{1:I}, \dots, c_{m:I}$ . Hence, we have  $\min\{I, J\} + 1$  cases in terms of the threshold  $m$  as follows.

- Case 0:  $v_{1:J} < c_{1:I}$ , then the task and user selection set is  $\mathcal{T}_0 = \{(v, c) : v_{1:J} < c_{1:I}\}$ ;
- Case  $m, 1 \leq m \leq \min\{I, J\} - 1$ :  $v_{m:J} \geq c_{m:I}$  and  $v_{m+1:J} < c_{m+1:I}$ , then the task and user selection set is  $\mathcal{T}_m = \{(v, c) : v_{m:J} \geq c_{m:I}, v_{m+1:J} < c_{m+1:I}\}$ ;
- Case  $\min\{I, J\}$ :  $v_{\min\{I, J\}:J} \geq c_{\min\{I, J\}:I}$ , then the task and user selection set is  $\mathcal{T}_{\min\{I, J\}} = \{(v, c) : v_{\min\{I, J\}:J} \geq c_{\min\{I, J\}:I}\}$ .

For case 0, no tasks or users will be selected, and the social welfare is 0.

For case  $m, 1 \leq m \leq \min\{I, J\} - 1$ , tasks with values  $v_{1:J}, \dots, v_{m:J}$  and users with costs  $c_{1:I}, \dots, c_{m:I}$  will be selected. Hence, the social welfare  $SW_n[m]$  is

$$SW_n[m] = \int_{\mathcal{T}_m} \sum_{k=1}^m (v_{k:J} - c_{k:I}) f_{1,\dots,m+1:J} g_{1,\dots,m+1:I} dv dc.$$

For case  $\min\{I, J\}$ , tasks with values  $v_{1:J}, \dots, v_{\min\{I, J\}:J}$  and users with costs  $c_{1:I}, \dots, c_{\min\{I, J\}:I}$  will be selected. Hence, the social welfare  $SW_n[\min\{I, J\}]$  is

$$SW_n[\min\{I, J\}] = \int_{\mathcal{T}_{\min\{I, J\}}} \sum_{k=1}^{\min\{I, J\}} (v_{k:J} - c_{k:I}) \times f_{1,\dots,\min\{I, J\}:J} g_{1,\dots,\min\{I, J\}:I} dv dc.$$

Hence, the total social welfare without data reuse is the sum of the social welfare in the  $\min\{I, J\} + 1$  cases, i.e.,

$$SW_n = \sum_{m=1}^{\min\{I, J\}-1} SW_n[m] + SW_n[\min\{I, J\}]. \quad (11)$$

By transforming the domains of integration  $\mathcal{T}_m (m = 1, \dots, \min\{I, J\})$ , it turns out that we can derive the social welfare without data reuse in the following explicit form. That is,

$$SW_n = \sum_{m=1}^{\min\{I, J\}} \int_0^1 \int_0^{v_{1:J}} \dots \int_0^{v_{m-1:J}} \dots \int_0^{v_{\min\{I, J\}-1:J}} \int_0^{v_{m:J}} \int_{c_{1:I}}^{v_{2:J}} \dots \int_{c_{m-1:I}}^{v_{m:I}} \int_{c_{m:I}}^1 \int_{c_{m+1:I}}^1 \dots \int_{c_{\min\{I, J\}-1:I}}^1 (v_{m:J} - c_{m:I}) [(\min\{I, J\})!]^2 \prod_{k=1}^{\min\{I, J\}} f(v_k) g(c_k) dc_{\min\{I, J\}:I} \dots dc_{2:I} dc_{1:I} dv_{\min\{I, J\}:J} \dots dv_{2:J} dv_{1:J}. \quad (12)$$

In particular, if  $(v_j, j \in \mathcal{J})$  and  $(c_i, i \in \mathcal{I})$  follow i.i.d. uniform distributions, the integral can be further simplified since  $\prod_{k=1}^{\min\{I, J\}} f(v_k) g(c_k) = 1$ . Under the assumption of uniform distributions, we have the following results.

**Proposition 1** (Social Welfare without Data Reuse). *Under the i.i.d. uniform distributions of  $(v_j, j \in \mathcal{J})$  and  $(c_i, i \in \mathcal{I})$ , the social welfare without data reuse is  $J/4$  when the number of users and the number of tasks are identically and sufficiently large, i.e.,  $I = J \rightarrow \infty$ . The social welfare without data reuse is  $J/2$  when  $J$  is finite and  $I \rightarrow \infty$ , and is  $I/2$  when  $I$  is finite and  $J \rightarrow \infty$ .*

#### 3.2.2 Analysis with Data Reuse

In the scenario with data reuse, we have two possible cases:

- Case I: If  $\min\{c_i, i \in \mathcal{I}\} \leq \sum_{j \in \mathcal{J}} v_j$ , then the minimum cost user will sense the data of all tasks, and all tasks will be selected. Let  $c = \min\{c_i, i \in \mathcal{I}\}$  and  $v = \sum_{j \in \mathcal{J}} v_j$ , then the task and user selection set is  $\mathcal{R} = \{(v, c) : \min\{c_i\} \leq \sum_{j \in \mathcal{J}} v_j\}$ . Let the p.d.f. of  $\min\{c_i, i \in \mathcal{I}\}$  be  $g_{\min\{c_i, i \in \mathcal{I}\}}(c)$  and the p.d.f. of  $\sum_{j \in \mathcal{J}} v_j$  be  $f_{\sum_{j \in \mathcal{J}} v_j}(v)$ , then the social welfare is  $\int_{\mathcal{R}} (v - c) f_{\sum_{j \in \mathcal{J}} v_j}(v) g_{\min\{c_i, i \in \mathcal{I}\}}(c) dv dc$ .
- Case II: If  $\min\{c_i, i \in \mathcal{I}\} > \sum_{j \in \mathcal{J}} v_j$ , then no task or user will be selected. The social welfare is 0.

Hence, the total social welfare with data reuse is the sum of the social welfare in the two cases, which is given by

$$SW_r = \int_{\mathcal{R}} (v - c) f_{\sum_{j \in \mathcal{J}} v_j}(v) g_{\min\{c_i, i \in \mathcal{I}\}}(c) dv dc. \quad (13)$$

By transforming the domain of integration  $\mathcal{R}$ , it turns out that we can derive the social welfare with data reuse in the following explicit form. That is,

$$SW_r = \int_0^1 \int_0^v (v - c) f_{\sum_{j \in \mathcal{J}} v_j}(v) g_{\min\{c_i, i \in \mathcal{I}\}}(c) dc dv + \sum_{j=1}^{J-1} \int_j^{j+1} \int_0^1 (v - c) f_{\sum_{j \in \mathcal{J}} v_j}(v) g_{\min\{c_i, i \in \mathcal{I}\}}(c) dc dv. \quad (14)$$

In particular, if  $(v_j, j \in \mathcal{J})$  and  $(c_i, i \in \mathcal{I})$  follow i.i.d. uniform distributions, then the term  $\sum_{j \in \mathcal{J}} v_j$  follows the Irwin-Hall distribution with the p.d.f.

$f_{\sum_{j \in \mathcal{J}} v_j}(v) = \frac{1}{2^{(J-1)!}} \sum_{j=0}^J (-1)^j \binom{J}{j} (v-j)^{n-1} \text{sgn}(v-j)$ , where  $\text{sgn}(\cdot)$  is the sign function, i.e.,  $\text{sgn}(x) = -1$  if  $x < 0$ ,  $\text{sgn}(x) = 0$  if  $x = 0$ , and  $\text{sgn}(x) = 1$  if  $x > 0$ . The term  $\min\{c_i, i \in \mathcal{I}\}$  follows a distribution with the p.d.f.

$$g_{\min\{c_i, i \in \mathcal{I}\}}(c) = I(1-c)^{I-1}.$$

We have the following result with uniform distribution.

**Proposition 2** (Social Welfare with Data Reuse). *Under the i.i.d. uniform distributions of  $(v_j, j \in \mathcal{J})$  and  $(c_i, i \in \mathcal{I})$ , the social welfare with data reuse is given by*

$$SW_r = J/2 - 1 + I/(I+1). \quad (15)$$

That is, the social welfare with data reuse is  $J/2$  when  $I \rightarrow \infty$ .

### 3.2.3 Performance Bound

We show the performance bound by comparing the social welfare with and without data reuse. In particular, we define the (relative) performance gain due to data reuse as

$$\gamma = SW_r/SW_n. \quad (16)$$

Based on Propositions 1 and 2, we have the following results on the relative performance gain  $\gamma$  defined in (16).

**Proposition 3** (Performance Bound). *Under the i.i.d. uniform distributions of  $(v_j, j \in \mathcal{J})$  and  $(c_i, i \in \mathcal{I})$ ,*

- when the numbers of users and tasks are identical and sufficiently large, e.g.,  $I = J \rightarrow \infty$ , the lower bound of the relative performance gain is  $\gamma_{\text{lower bound}} = (J/2)/(J/4) = 2$ . That is, the social welfare is at least doubled by exploiting data reuse across tasks;
- when the number of users is sufficiently large, e.g.,  $I \rightarrow \infty$ , with a limited  $J$ , the lower bound of the relative performance gain is  $\gamma_{\text{lower bound}} = (J/2)/(J/2) = 1$ . That is, the social welfare due to data reuse is at least the same as that without data reuse;
- when the number of tasks is sufficiently large, e.g.,  $J \rightarrow \infty$ , with a limited  $I$ , the lower bound of the relative performance gain is  $\gamma_{\text{lower bound}} = (J/2)/(I/2) = J/I$ . That is, the social welfare due to data reuse is much larger than that without data reuse.

### 3.3 A Special Case of Two Tasks and Two Users

The previous results are derived for large values of  $I$  and  $J$ . Next, we will consider finite values of  $I$  and  $J$ . As a special case, we will consider two tasks and two users, and derive some additional insights regarding data reuse.

We assume two tasks and two users, and the task values and sensing costs follow i.i.d. uniform distributions on  $[0, 1]$ . We will derive the *explicit* expression for the gain  $\gamma$ .

We first consider the case without data reuse. The joint distribution of  $g_{1,2,2}(c_1, c_2)$  is

$$g_{1,2,2}(c_1, c_2) = 2g(c_1)g(c_2) = 2, 0 \leq c_1 < c_2 \leq 1.$$

The joint distribution of  $f_{1,2,2}(v_1, v_2)$  is

$$f_{1,2,2}(v_1, v_2) = 2f(v_1)f(v_2) = 2, 1 \geq v_1 > v_2 \geq 0.$$

Hence, the social welfare without data reuse can be computed by (12) as  $SW_n = 2/5$ .

Now we consider the case with data reuse. We have  $f_{v_1+v_2}(v) = \min\{1, v\} - \max\{0, v-1\}$ ,  $0 \leq v \leq 2$ , and  $f_{\min\{c_1, c_2\}}(c) = 2(1-c)$ ,  $0 \leq c \leq 1$ . The social welfare with data reuse can be computed by (14) as  $SW_r = 41/60$ .

Hence, the relative performance gain due to data reuse is  $\gamma = SW_r/SW_n = \frac{41}{60}/\frac{2}{5} \approx 1.7$ .

For the general case of the finite number of tasks and users, we will use the *Monte Carlo* method [22] to compute  $\gamma$  numerically. Figs. 2 and 3 show the impact of the task number and the user number on the relative performance gain, respectively. We can see when the number of tasks equals the number of users (both are larger than 10), the relative performance gain is 2, which means that the social welfare with data reuse is doubled of that without data reuse. Furthermore, the relative performance gain is decreasing with the number of users, and increasing with the number of tasks. Increasing the number of users has little impact on the social welfare with data reuse  $SW_r$ , since the user with the minimum sensing cost completes all tasks; while it can increase the social welfare without data reuse  $SW_n$  due to the increasing user competition. Increasing the number of tasks can increase  $SW_r$  due to data reuse; while it can decrease  $SW_n$  due to the increasing task competition.<sup>7</sup>

### 3.4 Analysis for Multiple Data Items

So far, we have considered the simplified scenario with one data item. The analysis for the scenario with multiple data items is quite challenging, due to the complicated coupling of tasks' data requirements and users' sensing capability across different data items. To show the key insights, we numerically study the impact of the number of data items on the performance gain with data reuse.

In the numerical studies, we fix both the number of tasks  $J$  and the number users  $I$  as 50. The number of data items increases from 1 to 7. To show the impact of the number of data items on the relative performance gain  $\gamma$ , we assume that each task requires each data item with a fixed probability, and each user can sense each data item with the same probability. This captures the average data supply (users' sensing capabilities) and data demand (tasks' data requirements) for each data item.

Fig. 4 shows the impacts of the number of data items and the data demand probability on the relative performance gain. We have two observations. First, we can see that the relative performance gain increases with the number of data items. On one hand, increasing the number of data items will decrease the social welfare with data reuse and that without data reuse, due to the decreased task value per data and the increased sensing cost per data. On the other hand, allowing data reuse across tasks weakens the above effect, so that the reduction of the social welfare with data reuse is less than that without data reuse. Hence, the relative gain increases with the number of data items. Second, the relative gain first increases and then decreases with the data demand probability. On one hand, the social welfare without data reuse first decreases and then increases with the data demand probability, due to the different impacts

<sup>7</sup> For other non-uniform distributions such as the truncated normal distribution, we provide more results and insights in Appendix D of the online technical report [28].

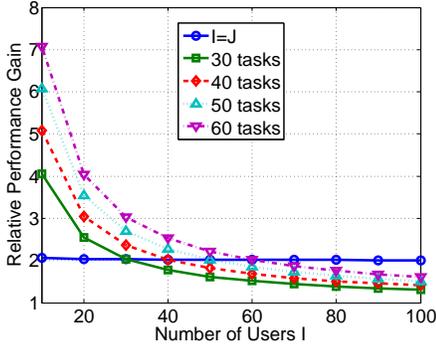


Fig. 2. Impact of the number of users on the relative performance gain

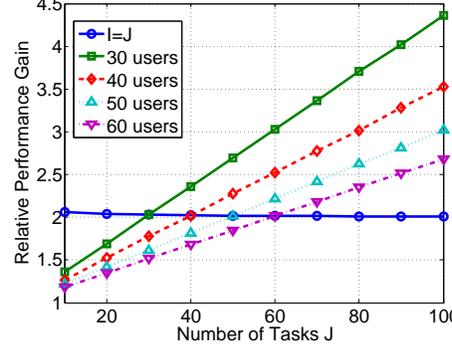


Fig. 3. Impact of the number of tasks on the relative performance gain

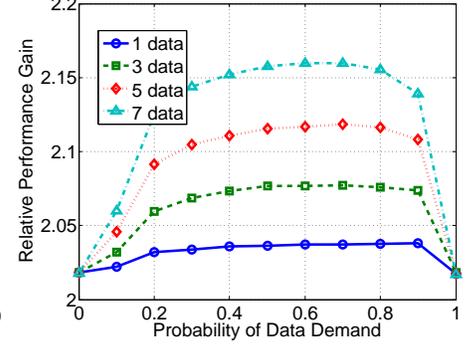


Fig. 4. Impact of multiple data items on the relative performance gain

of task competition and user competition. When the probability is small, each task's data requirement is small and can be easily completed, leading to a larger social welfare. When the probability is large, each user's sensing capability is large, and the user competition leads to a larger social welfare.<sup>8</sup> On the other hand, the social welfare with data reuse is approximately concave increasing with the data demand probability, due to the increasing reuse of data items. Hence, a larger relative reuse gain can be achieved when the data demand probability is medium.

However, theoretically understanding the benefit of data reuse is only the first step towards realizing the benefit of data reuse. In practice, tasks owners and users are selfish, and maybe unwilling to report their private information about task values and sensing cost. Hence we need to design an incentive mechanism to induce task owners and users to truthfully report their private information, while satisfying other properties such as achieving the maximum social welfare and computational efficiency.

#### 4 AUCTION-BASED INCENTIVE MECHANISMS

In this section, we study the problem of achieving the above performance gain in the practical scenario with limited computational capability and incomplete information. We propose a two-sided auction-based incentive mechanism framework for solving Problem P1. First, we propose a two-sided VCG auction mechanism (as the benchmark) for solving Problem P1 exactly, which is feasible, socially optimal, but computationally difficult. Then we further propose a feasible, close-to-optimal, and low-complexity randomized auction mechanism for solving Problem P1 approximately in polynomial time. We aim to design an incentive mechanism satisfying the following five desirable properties:

- **Incentive Compatibility (IC, Truthfulness):** Reporting the true task value (and the true sensing cost, respectively) is the dominant strategy for each task owner (and each user, respectively), no matter what others report.
- **Individual Rationality (IR):** Each participating task owner and user will have a non-negative utility

8. When the probability is 0, all tasks (no data needs) can be completed without decreasing the social welfare. When the probability is 1, each task requires all data, and each user can sense all data. The above two cases are both equivalent to the case with one data item, and hence the gain is approximately 2, as is shown in Fig. 2.

by reporting the true task value and sensing cost, respectively.

- **Feasibility and Economic Efficiency:** The outcome of the mechanism can be implemented in practice (i.e., through an integer allocation) and maximizes the social welfare.
- **Computational Efficiency:** The outcome of the mechanism can be computed in polynomial time.
- **Budget Balance:** The total payment obtained from the selected task owners should be no less than the total payment paid to the scheduled users.

#### 4.1 Two-sided Auction Framework

To solve Problem P1 with two-sided private information, we propose a two-sided auction-based incentive mechanism, where the platform acts as an *auctioneer* purchasing data from mobile users (*bidders* on one side) and selling data to task owners (*bidders* on the other side). In this auction framework, the platform first announces an *allocation rule* (for task selection and user scheduling) and a *payment rule* (for payments to the scheduled users and prices charged to the selected task owners). Then, each task owner submits a bid (indicating her task value) and each user submits a bid (indicating her sensing cost) to the platform, which can be different from the true task value and the true user sensing cost, respectively. Finally, the platform computes the allocations and payments, based on the reported bids of all task owners and users, together with other public information (e.g., tasks' data requirements and users' sensing capabilities). In this work, we are interested in designing the *truthful* auction, where task owners and users submit their private information truthfully.

Next, we provide the key notations. Let  $u_j$  denote the reported value (bid) of task  $j$ . Let  $\mathbf{b}_i = (b_i(\mathcal{S}), \mathcal{S} \subseteq \mathcal{S}_i)$  denote the reported sensing cost vector (bid) of user  $i$ , where  $b_i(\mathcal{S})$  denotes the user reported sensing cost for a data set  $\mathcal{S} \subseteq \mathcal{S}_i$ . Let  $\mathbf{u} \triangleq (u_j, j \in \mathcal{J})$  denote the bids of all tasks and  $\mathbf{b} \triangleq (\mathbf{b}_i, i \in \mathcal{I})$  denote the bids of all users. If an auction is truthful, we will have  $\mathbf{b} = \mathbf{c}$  and  $\mathbf{u} = \mathbf{v}$  at the equilibrium. With a little abuse of notation, we denote  $\{\mathbf{x}(\cdot), \mathbf{z}(\cdot)\}$  as the allocation rule, where  $\mathbf{x}(\cdot) \triangleq (x_i(\cdot), i \in \mathcal{I})$  is the user scheduling vector and  $\mathbf{z}(\cdot) \triangleq (z_j(\cdot), j \in \mathcal{J})$  is the task selection vector. We further denote  $\{\mathbf{p}(\cdot), \mathbf{q}(\cdot)\}$  as the payment rule, where  $\mathbf{p}(\cdot) \triangleq (p_i(\cdot), i \in \mathcal{I})$  is the user payment vector and  $\mathbf{q}(\cdot) \triangleq (q_j(\cdot), j \in \mathcal{J})$  is the

task charge vector. Note that  $x(\cdot)$ ,  $z(\cdot)$ ,  $p(\cdot)$ , and  $q(\cdot)$  can also be written as  $x(\mathbf{b}, \mathbf{u})$ ,  $z(\mathbf{b}, \mathbf{u})$ ,  $p(\mathbf{b}, \mathbf{u})$ , and  $q(\mathbf{b}, \mathbf{u})$ , as they are all functions of the user bid vector  $\mathbf{b}$  and the task bid vector  $\mathbf{u}$ . For convenience, we write such an auction mechanism as  $\Omega \triangleq \{x(\cdot), z(\cdot); p(\cdot), q(\cdot)\}$  or  $\Omega \triangleq \{x(\mathbf{b}, \mathbf{u}), z(\mathbf{b}, \mathbf{u}); p(\mathbf{b}, \mathbf{u}), q(\mathbf{b}, \mathbf{u})\}$ .

## 4.2 Two-sided VCG Auction (Benchmark)

We first propose a two-sided VCG auction, which is a nontrivial extension of the classic VCG auction [15], due to the two-sided information asymmetry. In our two-sided VCG auction, the allocation rule aims to maximize the social welfare *defined on the reported sensing costs and task values*, and the payment rule aims to pay each scheduled user the social benefit that she generates *and* to charge each selected task owner the social damage that she imposes. Formally,

**Mechanism 1** (Two-sided VCG Auction Mechanism –  $\Omega^o$ ).

- *Allocation Rule*  $\{x(\mathbf{b}, \mathbf{u}), z(\mathbf{b}, \mathbf{u})\}$ :  

$$x(\mathbf{b}, \mathbf{u}) = x^o(\mathbf{b}, \mathbf{u}) \text{ and } z(\mathbf{b}, \mathbf{u}) = z^o(\mathbf{b}, \mathbf{u}),$$
*where*  $\{x^o(\mathbf{b}, \mathbf{u}), z^o(\mathbf{b}, \mathbf{u})\}$  *is the optimal solution to Problem P1, by replacing*  $\mathbf{c}$  *with the reported cost*  $\mathbf{b}$  *and*  $\mathbf{v}$  *with the reported value*  $\mathbf{u}$  *in Problem P1;*
- *Payment Rule*  $\{p(\mathbf{b}, \mathbf{u}), q(\mathbf{b}, \mathbf{u})\}$ :  

$$p(\mathbf{b}, \mathbf{u}) = p^o(\mathbf{b}, \mathbf{u}) \triangleq (p_i^o(\mathbf{b}, \mathbf{u}))_{i \in \mathcal{I}},$$

$$q(\mathbf{b}, \mathbf{u}) = q^o(\mathbf{b}, \mathbf{u}) \triangleq (q_j^o(\mathbf{b}, \mathbf{u}))_{j \in \mathcal{J}},$$

where

$$p_i^o(\mathbf{b}, \mathbf{u}) \triangleq \sum_{j \in \mathcal{J}} u_j z_j^o(\mathbf{b}, \mathbf{u}) - \sum_{n \in \mathcal{I} \setminus \{i\}} \sum_{S \subseteq \mathcal{S}_n} b_n(S) x_n^o(S) - W_{-i}^o,$$

$$q_j^o(\mathbf{b}, \mathbf{u}) \triangleq W_{-j}^o - \sum_{i \in \mathcal{I} \setminus \{j\}} u_i z_i^o(\mathbf{b}, \mathbf{u}) + \sum_{n \in \mathcal{I}} \sum_{S \subseteq \mathcal{S}_n} b_n(S) x_n^o(S),$$

where  $W_{-i}^o$  is the maximum social welfare (defined on bids  $\mathbf{b}, \mathbf{u}$ ) excluding user  $i$ 's bid;  $W_{-j}^o$  is the maximum social welfare (defined on bids  $\mathbf{b}, \mathbf{u}$ ) excluding task  $j$ 's bid.<sup>9</sup>

In Mechanism 1, task owner  $j \in \mathcal{J}$  chooses the bid  $u_j^o$  such that  $u_j^o = \arg \max_{u_j} (v_j - q_j^o(\mathbf{b}, \mathbf{u}))$ ; user  $i \in \mathcal{I}$  chooses the bid  $\mathbf{b}_i^o$  such that  $\mathbf{b}_i^o = \arg \max_{\mathbf{b}_i} (p_i^o(\mathbf{b}, \mathbf{u}) - \sum_{S \subseteq \mathcal{S}_i} c_i(S))$ . The bid  $(\mathbf{b}^o, \mathbf{u}^o)$  is a Nash equilibrium, if each user and each task owner have no incentives to unilaterally change her bid, respectively. For convenience, we write Mechanism 1 as  $\Omega^o = \{x^o(\cdot), z^o(\cdot); p^o(\cdot), q^o(\cdot)\}$  or  $\Omega^o = \{x^o(\mathbf{b}, \mathbf{u}), z^o(\mathbf{b}, \mathbf{u}); p^o(\mathbf{b}, \mathbf{u}), q^o(\mathbf{b}, \mathbf{u})\}$ . By extending the analysis of the standard VCG auction [15] to our *two-sided scenario*, we can show that truthful reporting is a dominant strategy for both users and task owners, i.e.,  $\mathbf{b}^o = \mathbf{c}$  and  $\mathbf{u}^o = \mathbf{v}$  constitute the unique Nash equilibrium. This further implies that Mechanism 1 is efficient, as its allocation maximizes the social welfare defined in (6).

**Proposition 4** (Truthfulness and Efficiency). *Mechanism 1 is individually rational, incentive compatible (truthful), and maximizes the social welfare (efficient).*

Although Mechanism 1 exhibits several desirable properties, computing the two-sided VCG auction outcome needs to solve the NP-hard Problem P1, which is computationally intractable. To this end, we will propose a low-complexity auction mechanisms next.

9. Specifically,  $W_{-i}^o$  is the maximizer of Problem P1, by replacing  $\mathbf{c}$  with  $\mathbf{b}$  and  $\mathbf{v}$  with  $\mathbf{u}$ , and excluding user  $i$ 's bid  $\mathbf{b}_i$ , before solving Problem P1. Similarly,  $W_{-j}^o$  is obtained by excluding task  $j$ 's bid  $u_j$ .

## 4.3 Two-sided Randomized Auction

Inspired by the randomized auction in [16], [17], we now propose a low-complexity two-sided randomized auction mechanism, which operates in polynomial time. Due to the two-sided structure of mutual information asymmetry, our auction is different from the traditional single-sided auctions [16], [17].

In the following, we start from the linear programming relaxation of Problem P1, obtain an associated linear programming Problem P2 *in the fractional domain*, from which we further derive the fractional VCG auction (which may not be implementable in practice). Then, through proper decompositions, we transform the fractional VCG auction to a two-sided randomized auction (which is implementable).

### 4.3.1 Linear Programming Relaxation

We first relax the joint task selection and user scheduling Problem P1 to the fractional domain (i.e., relax every binary variable in  $\{0, 1\}$  to the domain  $[0, 1]$ ), and denote the associated linear programming problem as Problem P2. Note that Problem P2 can be solved to its optimality in polynomial time, as it is a standard linear programming problem [27]. We refer to the optimal solution of Problem P2 as the *fractional optimal solution*, denoted by  $\{x^*, y^*, z^*\}$  or  $\{x^*(\mathbf{c}, \mathbf{v}), y^*(\mathbf{c}, \mathbf{v}), z^*(\mathbf{c}, \mathbf{v})\}$ . It is notable that the maximum objective value of Problem P2 provides an upper-bound for the optimal objective function value of Problem P1, and the gap is usually called the *integrality gap* [27]. Intuitively, a fractional solution can be viewed as *the fraction of the time* that users are scheduled or tasks are selected.

Next, we present the fractional VCG auction  $\Omega^*$ , where the allocation rule aims to maximize the social welfare (based on user bids  $\mathbf{b}$  and task bids  $\mathbf{u}$ ) *in the fractional domain*, and the payment rule aims to pay each scheduled user her social benefit and charge each selected task her social damage. The detailed mechanism is similar to  $\Omega^o$ , except that we replace the integer solution  $\{x^o(\mathbf{b}, \mathbf{u}), z^o(\mathbf{b}, \mathbf{u})\}$  by the fractional optimal solution  $\{x^*(\mathbf{b}, \mathbf{u}), z^*(\mathbf{b}, \mathbf{u})\}$ , and solving Problem P2 rather than P1 when deciding the payments. We formally show the mechanism as follows.

**Mechanism 2** (Fractional VCG Auction Mechanism –  $\Omega^*$ ).

- *Allocation Rule*  $\{x(\mathbf{b}, \mathbf{u}), z(\mathbf{b}, \mathbf{u})\}$ :  

$$x(\mathbf{b}, \mathbf{u}) = x^*(\mathbf{b}, \mathbf{u}) \text{ and } z(\mathbf{b}, \mathbf{u}) = z^*(\mathbf{b}, \mathbf{u}),$$
*where*  $\{x^*(\mathbf{b}, \mathbf{u}), z^*(\mathbf{b}, \mathbf{u})\}$  *is the optimal solution to Problem P2, by replacing*  $\mathbf{c}$  *with the reported cost*  $\mathbf{b}$  *and*  $\mathbf{v}$  *with the reported value*  $\mathbf{u}$  *in Problem P2;*
- *Payment Rule*  $\{p(\mathbf{b}, \mathbf{u}), q(\mathbf{b}, \mathbf{u})\}$ :  

$$p(\mathbf{b}, \mathbf{u}) = p^*(\mathbf{b}, \mathbf{u}) \triangleq (p_i^*(\mathbf{b}, \mathbf{u}))_{i \in \mathcal{I}},$$

$$q(\mathbf{b}, \mathbf{u}) = q^*(\mathbf{b}, \mathbf{u}) \triangleq (q_j^*(\mathbf{b}, \mathbf{u}))_{j \in \mathcal{J}},$$

where

$$p_i^*(\mathbf{b}, \mathbf{u}) \triangleq \sum_{j \in \mathcal{J}} u_j z_j^*(\mathbf{b}, \mathbf{u}) - \sum_{n \in \mathcal{I} \setminus \{i\}} \sum_{S \subseteq \mathcal{S}_n} b_n(S) x_n^*(S) - W_{-i}^*,$$

$$q_j^*(\mathbf{b}, \mathbf{u}) \triangleq W_{-j}^* - \sum_{i \in \mathcal{I} \setminus \{j\}} u_i z_i^*(\mathbf{b}, \mathbf{u}) + \sum_{n \in \mathcal{I}} \sum_{S \subseteq \mathcal{S}_n} b_n(S) x_n^*(S),$$

where  $W_{-i}^*$  is the maximum social welfare (defined on bids  $\mathbf{b}, \mathbf{u}$ ) excluding user  $i$ 's bid in the fractional domain, and  $W_{-j}^o$  is the maximum social welfare (defined on bids  $\mathbf{b}, \mathbf{u}$ ) excluding task  $j$ 's bid in the fractional domain.

We summarize the properties of Mechanism 2 as follows.

**Proposition 5** (Truthfulness and Efficiency). *Mechanism 2 is individually rational, incentive compatible (truthful), and maximizes the social welfare (efficient) in the fractional domain.*

Note that the optimal solution to Problem P2 (i.e., the outcome of Mechanism 2) may not be feasible to Problem P1. This implies that Mechanism 2 may not be implementable. To see this, consider an example with 3 data items  $\mathcal{D} = \{1, 2, 3\}$ , 3 users  $\mathcal{I} = \{1, 2, 3\}$  with sensing capabilities  $\mathcal{S}_1 = \{1, 2\}$ ,  $\mathcal{S}_2 = \{1, 3\}$ , and  $\mathcal{S}_3 = \{2, 3\}$ , and 1 task requiring all of 3 data items. Suppose that the user's sensing capability is single-minded, i.e., each user  $i$  either senses all the data items in  $\mathcal{S}_i$  or does not sense any data item. Then, the fractional optimal solution is to schedule each user half of the time, i.e.,  $x_1^*(\mathcal{S}_1) = x_2^*(\mathcal{S}_2) = x_3^*(\mathcal{S}_3) = 0.5$ , and to complete the task all the time, i.e.,  $z_1^* = 1$ . However, such a fractional solution cannot be implemented in practice, since each user should be either selected or not selected. In the following, we will transform Mechanism 2, i.e., the fractional VCG auction  $\Omega^*$ , to a randomized auction, which always generates a feasible solution to Problem P1 randomly according to certain probability, hence is implementable.

#### 4.3.2 Randomized Mechanism Definition

We first define a randomized mechanism and the associated concept of truthfulness in expectation [16].

Recall that a two-sided *deterministic* mechanism  $\Omega = \{\mathbf{x}(\cdot), \mathbf{z}(\cdot); \mathbf{p}(\cdot), \mathbf{q}(\cdot)\}$  consists of a deterministic allocation rule  $\{\mathbf{x}(\cdot), \mathbf{z}(\cdot)\}$  and a payment rule  $\{\mathbf{p}(\cdot), \mathbf{q}(\cdot)\}$ , and returns a deterministic outcome  $\{\mathbf{x}(\mathbf{b}, \mathbf{u}), \mathbf{z}(\mathbf{b}, \mathbf{u}); \mathbf{p}(\mathbf{b}, \mathbf{u}), \mathbf{q}(\mathbf{b}, \mathbf{u})\}$  given any bids  $\mathbf{b}$  and  $\mathbf{u}$ . Note that both Mechanisms 1 and 2 introduced before are deterministic mechanisms.

A mechanism  $\tilde{\Omega} \triangleq \{\tilde{\mathbf{x}}(\cdot), \tilde{\mathbf{z}}(\cdot); \tilde{\mathbf{p}}(\cdot), \tilde{\mathbf{q}}(\cdot)\}$  can also be randomized, in which the allocation and payment determinations involve randomizations. In other words, given any bids  $\mathbf{b}$  and  $\mathbf{u}$ , the outcomes  $\tilde{x}_i(\mathbf{b}, \mathbf{u})$ ,  $\tilde{z}_j(\mathbf{b}, \mathbf{u})$ ,  $\tilde{p}_i(\mathbf{b}, \mathbf{u})$  and  $\tilde{q}_j(\mathbf{b}, \mathbf{u})$  are all random variables. As the result, each task owner's utility (i.e., value minus charge) and each user's utility (i.e., payment minus sensing cost) are also random variables. Intuitively, such a randomized mechanism can be viewed as a set of randomizations over the deterministic mechanism. For randomized mechanisms, the concept of truthfulness is defined in the expected sense. That is, if a randomized mechanism  $\tilde{\Omega}$  is truthful in expectation, then the *expected* utilities of each user and each task owner are maximized when reporting the true sensing cost and value.

#### 4.3.3 Randomized Mechanism Design Criterion

We now provide our design criterion of a *truthful* randomized mechanism. The key idea is to find a randomized mechanism that generates the equivalent outcome of a truthful deterministic mechanism.

We first introduce an  $(\alpha, \beta)$ -scaled fractional mechanism for the deterministic mechanism  $\Omega = \{\mathbf{x}(\cdot), \mathbf{z}(\cdot); \mathbf{p}(\cdot), \mathbf{q}(\cdot)\}$ , inspired by the  $\alpha$ -scaled fractional mechanism defined in [16], [17]. Comparing with the one-sided mechanisms in [16], [17], our mechanism considers the scaling of both sides.

**Definition 1** (Scaled Fractional Mechanism). *An  $(\alpha, \beta)$ -scaled fractional mechanism of  $\Omega = \{\mathbf{x}(\cdot), \mathbf{z}(\cdot); \mathbf{p}(\cdot), \mathbf{q}(\cdot)\}$ , denoted by  $\Omega_{(\alpha, \beta)} = \{\mathbf{x}_\alpha(\cdot), \mathbf{z}_\beta(\cdot); \mathbf{p}_\alpha(\cdot), \mathbf{q}_\beta(\cdot)\}$ , is defined as:*

$$\mathbf{x}_\alpha(\cdot) = \alpha \cdot \mathbf{x}(\cdot), \quad \mathbf{p}_\alpha(\cdot) = \alpha \cdot \mathbf{p}(\cdot), \quad (17)$$

$$\mathbf{z}_\beta(\cdot) = \beta \cdot \mathbf{z}(\cdot), \quad \mathbf{q}_\beta(\cdot) = \beta \cdot \mathbf{q}(\cdot), \quad (18)$$

where  $\alpha, \beta > 0$  are the scaling factors such that every element of  $\alpha \cdot \mathbf{x}(\cdot)$  belongs to  $[0, 1]$  and every element of  $\beta \cdot \mathbf{z}(\cdot)$  belongs to  $[0, 1]$ , respectively.

Intuitively, in an  $(\alpha, \beta)$ -scaled fractional mechanism, the incurred cost and payment of each user are scaled with  $\alpha$ , and the achieved value and charge of each task owner are scaled with  $\beta$ , compared with those in the original mechanism  $\Omega$ . This implies that both the users' and the task owners' optimal bidding strategies in these two mechanisms are equivalent, which leads to the equivalence of the truthfulness property of both mechanisms.

**Proposition 6.** *If a mechanism  $\Omega$  is truthful, then its  $(\alpha, \beta)$ -scaled fractional mechanism  $\Omega_{(\alpha, \beta)}$  is also truthful.*

Based on Proposition 6, we propose the following two-sided randomized mechanism design criterion: design a two-sided randomized mechanism  $\tilde{\Omega}$  that provides the equivalent outcome (in terms of task selection, user scheduling, and payment) as an  $(\alpha, \beta)$ -scaled fractional mechanism  $\Omega_{(\alpha, \beta)}^*$  of the fractional VCG auction  $\Omega^*$  in Mechanism 2.

As the fractional VCG auction mechanism  $\Omega^*$  in Mechanism 2 is truthful, we can obtain the truthfulness of its  $(\alpha, \beta)$ -scaled fractional mechanism by Proposition 6. Moreover, as the randomized mechanism  $\tilde{\Omega}$  generates the same task selection, user scheduling, and payment as  $\Omega_{(\alpha, \beta)}^*$ , we can further obtain the truthfulness (in expectation) of  $\tilde{\Omega}$ .

#### 4.3.4 Two-sided Randomized Mechanism Design

Now we provide the details about our two-sided randomized mechanism design.

For convenience, we express a randomized mechanism  $\tilde{\Omega} = \{\tilde{\mathbf{x}}(\cdot), \tilde{\mathbf{z}}(\cdot); \tilde{\mathbf{p}}(\cdot), \tilde{\mathbf{q}}(\cdot)\}$  as a set of allocation probabilities  $\boldsymbol{\lambda} = (\lambda^l)_{l \in \mathcal{A}}$  and a set of payment rules  $\{\mathbf{p}^l(\cdot), \mathbf{q}^l(\cdot)\}_{l \in \mathcal{A}}$  under all possible allocations, where  $\mathcal{A}$  is the set of all feasible *integer* allocations (regarding  $\mathbf{x}$  and  $\mathbf{z}$ ) and  $\lambda^l \geq 0$  is the probability of picking up a particular allocation  $\{\mathbf{x}^l, \mathbf{z}^l\}$  and the corresponding payment  $\{\mathbf{p}^l, \mathbf{q}^l\}$ . Then, designing a randomized mechanism  $\tilde{\Omega}$  is equivalent to finding a set of allocation probabilities  $\boldsymbol{\lambda} = (\lambda^l)_{l \in \mathcal{A}}$  and a set of payment rules  $\{\mathbf{p}^l(\cdot), \mathbf{q}^l(\cdot)\}_{l \in \mathcal{A}}$ .

Next, we propose the two-sided randomized auction  $\tilde{\Omega}^\dagger$ , which aims to maximize the two-sided scaled social welfare subject to the exact decomposition of the fractional optimal solution into the weighted sum of integer solutions. Due to the two-sided social welfare maximization,  $\tilde{\Omega}^\dagger$  nontrivially extends those with one-sided utility maximization or cost minimization in [16], [17].

**Mechanism 3** (Randomized Auction Mechanism –  $\tilde{\Omega}^\dagger$ ).

*Starting from the fractional VCG auction  $\Omega^* = \{\mathbf{x}^*(\mathbf{b}, \mathbf{u}), \mathbf{z}^*(\mathbf{b}, \mathbf{u}); \mathbf{p}^*(\mathbf{b}, \mathbf{u}), \mathbf{q}^*(\mathbf{b}, \mathbf{u})\}$  in Mechanism 2, we define the randomized auction mechanism  $\tilde{\Omega}^\dagger$  as:*

- *Allocation Rule*  $\tilde{\lambda} = (\lambda^l)_{l \in \mathcal{A}}$ :  

$$\tilde{\lambda} = \arg \max_{\lambda, 0 < \alpha, \beta \leq 1} \beta \cdot V^* - \alpha \cdot C^* \quad (19)$$

$$\text{s.t. } \sum_{l \in \mathcal{A}} \lambda^l \cdot \mathbf{x}_i^l = \alpha \cdot \mathbf{x}_i^*(\mathbf{b}, \mathbf{u}), \quad \forall i \in \mathcal{I}, \quad (20)$$

$$\sum_{l \in \mathcal{A}} \lambda^l \cdot \mathbf{z}_j^l = \beta \cdot \mathbf{z}_j^*(\mathbf{b}, \mathbf{u}), \quad \forall j \in \mathcal{J}, \quad (21)$$

where  $V^*$  and  $C^*$  are the optimal total task values and user costs w.r.t.  $\mathbf{z}^*(\mathbf{b}, \mathbf{u})$  and  $\mathbf{x}^*(\mathbf{b}, \mathbf{u})$ , respectively.

- *Payment Rule*  $\{\mathbf{p}^l(\mathbf{b}, \mathbf{u}), \mathbf{q}^l(\mathbf{b}, \mathbf{u})\}_{l \in \mathcal{A}}$ :  

$$\mathbf{p}_i^l(\mathbf{b}, \mathbf{u}) = \alpha \cdot \mathbf{p}_i^*(\mathbf{b}, \mathbf{u}) \cdot \frac{C_i(\mathbf{x}_i^l)}{\sum_{l' \in \mathcal{A}} \lambda_{l'} \cdot C_i(\mathbf{x}_i^{l'})}, \quad \forall i \in \mathcal{I},$$

$$\mathbf{q}_j^l(\mathbf{b}, \mathbf{u}) = \beta \cdot \mathbf{q}_j^*(\mathbf{b}, \mathbf{u}) \cdot \frac{V_j(\mathbf{z}_j^l)}{\sum_{l' \in \mathcal{A}} \lambda_{l'} \cdot V_j(\mathbf{z}_j^{l'})}, \quad \forall j \in \mathcal{J},$$

where  $C_i(\mathbf{x}_i^l)$  is user  $i$ 's cost under the allocation  $\mathbf{x}_i^l$ , and  $V_j(\mathbf{z}_j^l)$  is task  $j$ 's value under the allocation  $\mathbf{z}_j^l$ .

We can see that in Mechanism 3, both the expected payment and sensing cost of each user and the expected charge and value of each task are equivalent to those in the fractional VCG auction  $\Omega^*$  in Mechanism 2, which implies that Mechanism 3 is truthful in expectation.

**Proposition 7** (Incentive Compatibility in Expectation). *Mechanism 3 is incentive compatible in expectation, in the sense that each user and task owner can maximize her expected utility when reporting the true sensing cost and value, respectively.*

We can further check that under Mechanism 3, each user and task owner can always achieve a non-negative utility under any possible realization of allocations. This implies that Mechanism 3 is individually rational in the strict sense.

**Proposition 8** (Individual Rationality). *Mechanism 3 is individually rational in the strict sense, as each user and task owner can achieve a non-negative expected utility.*

Furthermore, we can see that in Mechanism 3, each user's sensing cost equals  $\alpha^*$  times the sensing cost incurred in Mechanism 2, while each task's value equals  $\beta^*$  times the value achieved in Mechanism 2 (where  $\alpha^*$  and  $\beta^*$  are the optimal solutions to the allocation problem in Mechanism 3). The efficiency of Mechanism 3 is guaranteed in this sense.

**Proposition 9** (Efficiency of  $\tilde{\Omega}^l$ ). *Mechanism 3 guarantees to achieve a  $\beta^*$ -fraction of the total task value in Mechanism 2 with an  $\alpha^*$ -fraction of the total sensing cost in Mechanism 2.*

So far we have proposed the randomized auction mechanism and proved several desirable economic properties. There are many possible ways to implement the randomized auction, depending on how we obtain the set of probability distribution for the allocation problem and the parameter  $\alpha$  and  $\beta$  for the payment in Mechanism 3. Next, we will propose one easy-to-implement solution method.

#### 4.3.5 Implementation of the Randomized Auction

As we have mentioned, one key step of designing Mechanism 3 is the two exact decompositions of the scaled fractional solutions into the weighted sum of integer solutions in (20) and (21), to obtain the two scaling factors  $\alpha$  and  $\beta$ . Next, we will show that it may sacrifice some social welfare in order to achieve the exact decompositions efficiently, which is a key difference of our approach here and the approach proposed in [17]. In [17], the authors proposed a

decomposition method to ensure system efficiency, but with a very complicated procedure that may not be practical.<sup>10</sup> Next, we exploit our two-sided problem structure to obtain a tailored easy-to-implement decomposition.<sup>11</sup>

The key idea of our solution approach is a two-step DEcomposition-MOdification (DEMO) procedure:

- *Step I: Decomposition.* We start from the fractional optimal solution  $(\mathbf{x}^*, \mathbf{z}^*)$ . Given the fractional user scheduling solution  $\mathbf{x}^*$ , we treat the fractional  $\mathbf{x}^*$  as the probability of scheduling the corresponding user. More specifically, we propose the following approach to compute  $\lambda_l$ . First recall that a feasible integer user scheduling is  $\mathbf{x}^l = \{x_i^l(\mathcal{S}), \forall \mathcal{S} \subseteq \mathcal{S}_i, \forall i \in \mathcal{I}\}$  and the fractional optimal user scheduling is  $\mathbf{x}^* = \{x_i^*(\mathcal{S}), \forall \mathcal{S} \subseteq \mathcal{S}_i, \forall i \in \mathcal{I}\}$ . Then we define the probability distribution  $\lambda^l$  as

$$\lambda^l = \prod_{i \in \mathcal{I}} \prod_{\mathcal{S} \subseteq \mathcal{S}_i} \phi(x_i^*(\mathcal{S}), x_i^l(\mathcal{S})), \quad \forall l \in \mathcal{A}, \quad (22)$$

where  $\phi(x_i^*(\mathcal{S}), x_i^l(\mathcal{S})) = x_i^*(\mathcal{S})$  if  $x_i^l(\mathcal{S}) = 1$ , and  $\phi(x_i^*(\mathcal{S}), x_i^l(\mathcal{S})) = 1 - x_i^*(\mathcal{S})$  if  $x_i^l(\mathcal{S}) = 0$ . The function  $\phi(x_i^*(\mathcal{S}), x_i^l(\mathcal{S}))$  characterizes the probability of scheduling user  $i$  with the set  $\mathcal{S}$ , and the probability is given by the corresponding fractional solution.

- *Step II: Modification.* Each integer solution  $\mathbf{x}^l$  corresponds to a maximum set of task selection  $\mathbf{z}^l$ . Given the probability distribution  $\lambda^l, \forall l \in \mathcal{A}$ , we compute

$$\beta_j = \frac{(\sum_{l \in \mathcal{A}} \lambda^l \cdot \mathbf{z}_j^l)}{\mathbf{z}_j^*}, \quad \forall j \in \mathcal{J}. \quad (23)$$

In order to ensure that all  $\mathbf{z}_j^*$  are scaled by the same factor  $\beta$ , we choose the smallest value of  $(\beta_j, j \in \mathcal{J})$  as the target  $\beta$ . Then we modify  $(\mathbf{z}_j^l, j \in \mathcal{J})$  such that  $\beta_j = \beta, \forall j \in \mathcal{J}$ . The detailed modification approach is as follows. First notice that each user scheduling  $\mathbf{x}^l$  corresponds to the maximum set of the task selection  $\mathbf{z}^l = (\mathbf{z}_j^l, j \in \mathcal{J})$ . Then for a particular  $\mathbf{z}_j^l = 1$ , we can modify  $\mathbf{z}_j^l = 1$  as  $\mathbf{z}_j^l = 0$  without violating the feasibility of the task selection. We iteratively conduct the above modification procedure until we have  $\beta_j = \beta, \forall j \in \mathcal{J}$ .

We can show the probability distribution  $\lambda^l$  in (22) satisfies  $\mathbf{x}^* = \sum_{l \in \mathcal{A}} \lambda^l \mathbf{x}^l$  with  $\alpha = 1$ . Then, we can choose  $\beta$  according to Step II such that  $\sum_{l \in \mathcal{A}} \lambda^l \cdot \mathbf{z}_j^l = \beta \cdot \mathbf{z}_j^*, \forall j \in \mathcal{J}$ .

Through the above proposed DEMO scheme, we derive the target  $\lambda^l, \alpha^*$ , and  $\beta^*$ . That is, we obtain the exact decompositions of the scaled fractional solutions into the weighted sum of integer solutions as in (20) and (21). According to Propositions 6 and 7, we have ensured the truthfulness of the mechanism at the cost of a reduced system efficiency.

**Proposition 10** (Truthfulness and Efficiency Bound). *The DEMO procedure implements Mechanism 3, and guarantees to achieve the same total sensing cost (i.e.,  $\alpha^* = 1$ ) in Mechanism 2 with a  $\beta^*$ -fraction of the total task value in Mechanism 2,<sup>12</sup> where*

$$\beta^* = \min_{j \in \mathcal{J}} \left( \frac{\sum_{l \in \mathcal{A}} \lambda^l \cdot \mathbf{z}_j^l}{\mathbf{z}_j^*} \right). \quad (24)$$

10. For example, the ellipsoid method used in [17] is quite complicated, and incurs a high time complexity in practical systems.

11. This is just one of the many possible solutions, which may differ in computational complexity and system efficiency loss.

12. The good performance is due to the inherent data correlations among multiple tasks and users. That is, the number of modifications in Step II becomes less if there are more data correlations among tasks.

We have proved several desirable properties of our designed mechanisms. Due to the two-sided structure of the mechanisms, the platform may lose money if the total payment obtained from task owners is less than the total payment paid to users. That is, Mechanism 3 may not be budget-balanced, which can be a practical concern. In fact, it is a well-known result in the literature that truthful efficient mechanisms may not be budget-balanced [23], [24]. Next, we further focus on the budget-balanced mechanism design.

## 5 BUDGET-BALANCED RANDOMIZED AUCTION

In this section, we focus on the budget-balance property of Mechanism 3, i.e., the expected total payment paid to users should be no large than the expected total payment obtained from task owners. This means that the MCS platform will not lose money, which is important for its realistic operation. Since the expected payments in Mechanism 3 for all users and task owners are scaled from Mechanism 2 by the same factors  $\alpha$  and  $\beta$ , we first focus on the budget balance of Mechanism 2, and then extend the results to Mechanism 3.

### 5.1 Budget Balance

In our model, we say that a mechanism is budget-balanced, if the MCS platform can achieve a non-negative profit, where the MCS platform's profit is defined as the difference between the total payments obtained from task owners and the total payment paid to users. It turns out that the budget balance of our two-sided auction cannot be guaranteed in general. In particular, with the increase of the task similarity, the positive network effect among tasks also increases, and the total payments from task owners to the platform become smaller and smaller (even zero). In such cases, the budget-balance property is not satisfied.

We use an illustrative example to show the budget imbalance. Suppose we have four tasks (tasks 1-4) with values 0.5, 0.6, 0.7, and 0.8, respectively. Each task only requires the same one data item. Two users (users 1-2) can sense the data item, with costs 0.1 and 0.2, respectively. The social welfare maximizer would require user 1 to sense the data item with the cost 0.1, and allow all four tasks to reuse the data. Hence, the maximum social welfare is  $0.5+0.6+0.7+0.8-0.1=2.5$ . Now we consider Mechanism 2. The VCG auction would schedule user 1 to sense the data with a payment 0.2, and all tasks would be selected. The payment of task 1 would be  $(0.5+0.6+0.7+0.8-0.1-0.2+0.2-0.5)-(0.6+0.7+0.8-0.1-0.2+0.2)=0$ , where the first term is the total social welfare except task 1 (when task 1 is considered in the auction), and the second term is the total social welfare when excluding task 1 from the auction. Similarly, we can show that the payments of tasks 2-4 are all 0. Hence, the total payment from task owners to the platform is 0, while the total payment from the platform to the sensing user is 0.2. This shows that Mechanism 2 may not be budget-balanced. Due to the scaled payments from Mechanism 2, Mechanism 3 is not guaranteed to be budget-balanced either.

### 5.2 Reserve Price based Randomized Auction

In the following, we will first focus on making Mechanism 2 budget-balanced, and then extend the results to the

budget-balanced randomized auction design by scaling the payments in Mechanism 2 according to the scaling rule proposed in Mechanism 3.

To this end, we introduce a reserve price for each data item in the proposed Mechanism 2, which denotes the minimal payment that a task owner has to pay for each data item. Let  $\sigma_k \geq 0$  denote the reserve price for each data item  $k \in \mathcal{K}$ . Then, for each task owner  $j \in \mathcal{J}$ , the minimum payment (if task  $j$  is completed) is

$$\underline{q}_j^\sigma = \sum_{k \in \mathcal{K}_j} \sigma_k. \quad (25)$$

Given the above minimum payments ( $\underline{q}_j^\sigma, j \in \mathcal{J}$ ) due to the reserve price  $\sigma_k$ , to ensure truthfulness, we propose the following bids reduction and payment rule for task owners.

**Definition 2** (Bids Reduction and Payment Rule). *Given users' bids  $\mathbf{b}$ , task owners' bids  $\mathbf{u}$ , and the minimum payments ( $\underline{q}_j^\sigma, j \in \mathcal{J}$ ), the reduced bid of task owner  $j$  is  $u'_j = u_j - \underline{q}_j^\sigma$ . With the new reduced bids  $\mathbf{u}' = (u'_j, j \in \mathcal{J})$ , Mechanism 2 leads to the allocation  $(\mathbf{x}^\sigma(\mathbf{b}, \mathbf{u}'), \mathbf{z}^\sigma(\mathbf{b}, \mathbf{u}'))$  and the payment  $(\mathbf{p}^\sigma(\mathbf{b}, \mathbf{u}'), \mathbf{q}^\sigma(\mathbf{b}, \mathbf{u}'))$ . Then the payment of task owner  $j \in \mathcal{J}$  is*

$$q_j^\sigma(\mathbf{b}, \mathbf{u}) = q_j^\sigma(\mathbf{b}, \mathbf{u}') + \underline{q}_j^\sigma. \quad (26)$$

The key idea of proposing the above bids reduction and payment rule is to reduce the mechanism design problem to a setting with no minimum payments. In particular, we first subtract the minimum payment of each task owner from her bid, run Mechanism 2, and then add the minimum payment of each task owner to her resulting payment.

Next, we propose the two-sided randomized auction mechanism  $\tilde{\Omega}^\sigma$  in Mechanism 4, i.e., the two-sided randomized auction mechanism with the reserve price.

**Mechanism 4** (Randomized Auction Mechanism with the Reserve Price  $-\tilde{\Omega}^\sigma$ ).

- *Allocation Rule  $\tilde{\lambda} = (\lambda^l)_{l \in \mathcal{A}}$ :*

$$\tilde{\lambda} = \arg \max_{\lambda, 0 < \alpha, \beta \leq 1} \beta \cdot V^\sigma - \alpha \cdot C^\sigma$$

$$\text{s.t. } \sum_{l \in \mathcal{A}} \lambda^l \cdot \mathbf{x}_i^l = \alpha \cdot \mathbf{x}_i^\sigma(\mathbf{b}, \mathbf{u}), \quad \forall i \in \mathcal{I},$$

$$\sum_{l \in \mathcal{A}} \lambda^l \cdot \mathbf{z}_j^l = \beta \cdot \mathbf{z}_j^\sigma(\mathbf{b}, \mathbf{u}), \quad \forall j \in \mathcal{J},$$
- *Payment Rule  $\{\mathbf{p}^l(\mathbf{b}, \mathbf{u}), \mathbf{q}^l(\mathbf{b}, \mathbf{u})\}$ :*

$$p_i^l(\mathbf{b}, \mathbf{u}) = \alpha \cdot p_i^\sigma(\mathbf{b}, \mathbf{u}) \cdot \frac{C_i(\mathbf{x}_i^l)}{\sum_{l' \in \mathcal{A}} \lambda_{l'} \cdot C_i(\mathbf{x}_i^{l'})}, \quad \forall i \in \mathcal{I},$$

$$q_j^l(\mathbf{b}, \mathbf{u}) = \beta \cdot q_j^\sigma(\mathbf{b}, \mathbf{u}) \cdot \frac{V_j(z_j^l)}{\sum_{l' \in \mathcal{A}} \lambda_{l'} \cdot V_j(z_j^{l'})}, \quad \forall j \in \mathcal{J},$$

where  $q_j^\sigma(\mathbf{b}, \mathbf{u})$  is given in (26). The derivations of  $\mathbf{x}^\sigma(\mathbf{b}, \mathbf{u})$ ,  $\mathbf{z}^\sigma(\mathbf{b}, \mathbf{u})$ , and  $p_i^\sigma(\mathbf{b}, \mathbf{u})$  are the same as those in Mechanism 2, and  $V^\sigma$ ,  $C^\sigma$ ,  $C_i(\mathbf{x}_i^l)$ , and  $V_j(z_j^l)$  are the same as those in Mechanism 3.

Next, we show that Mechanism 4 with the reserve price is truthful in expectation, but may be not optimal in terms of maximizing the total social welfare.

**Proposition 11** (Truthfulness and Efficiency Loss). *Mechanism 4 is truthful in expectation, but is not optimal in terms of maximizing the total social welfare.*

We show the impact of the reserve price on the social efficiency as follows. Given the reserve price, some task owners, i.e., those with task values lower than the minimum

payments given in (25), will decide not to join the auction. Hence, the maximum social welfare may be reduced. Therefore, there is a tradeoff between the social efficiency and the budget balance. A larger reserve price may lead to a better budget balance and a worse social efficiency. We will show the impact of the reserve price on the budget balance and the social efficiency via simulations in Section 6.

## 6 SIMULATION RESULTS

In this section, we provide simulation results to evaluate the performances of our proposed mechanisms. In particular, we first illustrate the performance of our proposed Mechanism 3. Then, we evaluate the performance gain due to data reuse. Finally, we show the impact of the reserve price on the achieved social welfare and the budget balance.

### 6.1 Simulation Setup

In the simulations, we fix the number of tasks to  $J = 50$  and the number of data items to  $K = 30$ , while varying the number of users from  $I = 10$  to 100 with an increment of 10. Each data item is location-based (such as the temperature at a particular location), and randomly and uniformly distributed in an area of  $1000\text{m} \times 1000\text{m}$ . Each user randomly moves to a particular location in a time slot, and can sense all the data items within a distance of 100m to her location. The unit cost  $\rho_c$  of each user for sensing *one* data item is chosen randomly from  $[1, 5]$ , hence the cost for sensing a set  $S$  of data items is  $\rho_c \cdot |S|$ . The unit value  $\rho_v$  of each task for *one* data item is also chosen randomly from  $[1, 5]$ , hence the value of a task requiring a set  $S$  of data items is  $\rho_v \cdot |S|$ .

We characterize the *task similarity* (in terms of data requirements) as follows. First, we define the *popularity* of a data item as the probability that a task requires this particular data item, and denote  $P_w$  as the  $w$ -th highest popularity of all data. As demonstrated in [25], the popularity of data, i.e.,  $\{P_w, w \in \mathcal{K}\}$ , follows a Zipf distribution [26] with p.m.f.

$$P_w = \frac{(1/w)^\mu}{\sum_{t=1}^K (1/t)^\mu}, \quad \forall w \in \mathcal{K}, \quad (27)$$

where  $\mu \geq 0$  is the parameter of Zipf distribution. Obviously, with a larger  $\mu$ , tasks are more likely to require a small set of high popularity data items (hence with a higher task similarity). In our simulations, we vary  $\mu$  from 0 to 3 with an increment of 0.3.

In each simulation, we choose a particular user number  $I$  and parameter  $\mu$ , and randomly generate 1000 systems (in terms of tasks' data requirements and unit values as well as users' sensing capabilities and unit costs) and compute the average outcome of all systems as the simulation result.<sup>13,14</sup>

### 6.2 Social Welfare Gap

We first compare the social welfare achieved in Mechanisms 1, 2, and 3. This can help us understand the performance gap of our proposed Mechanism 3 to the maximum social

welfare (achieved in Mechanism 1) or the fractional maximum social welfare (achieved in Mechanism 2).

Fig. 5 illustrates the average social welfare achieved in different auction mechanisms, under different numbers of users, where the parameter of Zipf distribution is fixed at  $\mu = 1$ . The red curve (with marker  $\circ$ ) denotes the social welfare achieved in Mechanism 1, which is equivalent to the maximum social welfare benchmark. The blue curve (with marker  $*$ ) denotes the social welfare achieved in Mechanism 2, which is equivalent to the fractional maximum social welfare. The green curve (with marker  $\square$ ) denotes the social welfare achieved in Mechanism 3.

From Fig. 5, we can see that the difference between the maximum social welfare and the fractional maximum social welfare is negligible. Moreover, the achieved social welfare in all three auction mechanisms increase with the number of users. The performance gap of the randomized auction in Mechanism 3 to the maximum social welfare (benchmark) increases with the number of users, and the maximal gap in our simulations (when there are 100 users) is less than 10%.

### 6.3 Performance Gain of Data Reuse

We now evaluate the performance gain achieved by the data reuse across tasks, by comparing the social welfare achieved in the systems with data reuse and without data reuse.

Next, we will implement the randomized auction (Mechanism 3) and the fractional VCG auction (Mechanism 2), by solving Problem P1 and the problem without data reuse (see Appendix A [28]), respectively. We will compare the performance gain due to data reuse in the two mechanisms.

#### 6.3.1 Impact of the Number of Users

We first show the impact of the number of users on the performance gain. Fig. 6 illustrates the achieved social welfare with and without data reuse, under different numbers of users, where the parameter of Zipf distribution is fixed at  $\mu = 1$ . The blue curve (with marker  $*$ ) denotes the social welfare achieved in Mechanism 2 with data reuse, representing the maximum social welfare with data reuse based on the result in Fig. 5. The green curve (with marker  $\square$ ) denotes the social welfare achieved in Mechanism 3 with data reuse. The red curve (with marker  $\circ$ ) and the cyan curve (with marker  $\diamond$ ) denote the results without data reuse.

From Fig. 6, we can see that the achieved social welfare with and without data reuse both increase with the number of users; and the increase rate is higher with data reuse, especially when the number of users is small. Furthermore, with data reuse, the maximum social welfare (benchmark) can increase up to 350%, and the social welfare achieved by the randomized auction (Mechanism 3) can increase up to 300%, comparing with those without data reuse.

#### 6.3.2 Impact of the Task Similarity

We next show the impact of the task similarity on the performance gain. Recall that  $\mu$  of Zipf reflects the task similarity: a larger  $\mu$  implies a higher task similarity. Fig. 7 illustrates the achieved social welfare with and without data reuse, under different values of  $\mu$ , where the user number is fixed at  $I = 60$ . The blue curve (with marker  $*$ ) denotes the social welfare achieved in Mechanism 2 with data reuse,

13. In this work, we generate the simulation datasets by using the Zipf distribution. We will consider more extensive evaluations based on real-world datasets in the future work.

14. Our simulations rely on Matlab and CVX (cvxr.com/cvx/) to solve the (integer) linear programming and the DEMO problems.

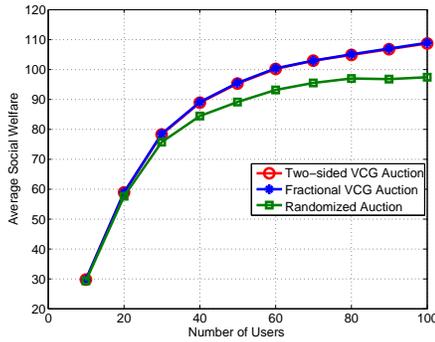


Fig. 5. Social Welfare vs. Number of Users in Different Auctions. Tasks: 50, data: 30, and the parameter  $\mu = 1$  in Zipf distribution.

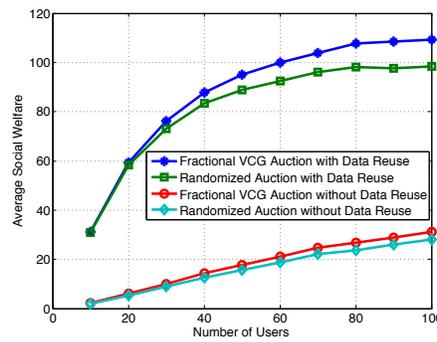


Fig. 6. Social Welfare vs. Number of Users With/Without Data Reuse. Tasks: 50, data: 30, and the parameter  $\mu = 1$  in Zipf distribution.

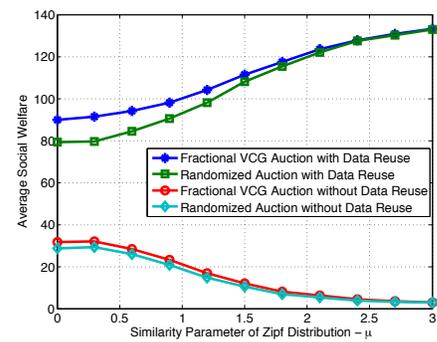


Fig. 7. Social Welfare vs. Task Similarity With/Without Data Reuse. Tasks: 50, data: 30, and users: 60.

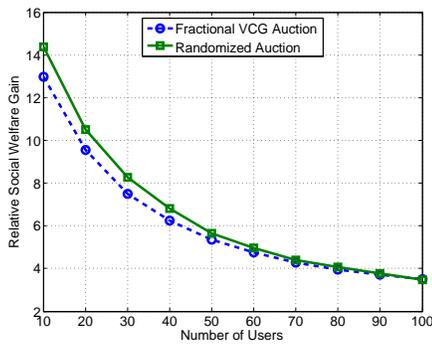


Fig. 8. Relative Social Welfare Gain vs. Number of Users. Tasks: 50, data: 30, and the parameter  $\mu = 1$  in Zipf distribution.

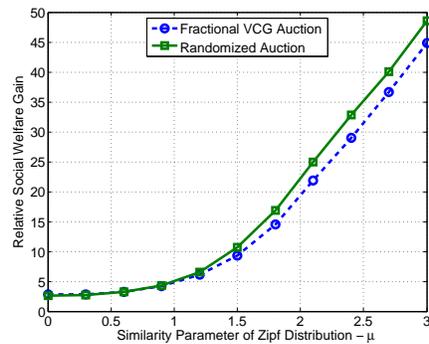


Fig. 9. Relative Social Welfare Gain vs. Task Similarity. Tasks: 50, data: 30, and users: 60.

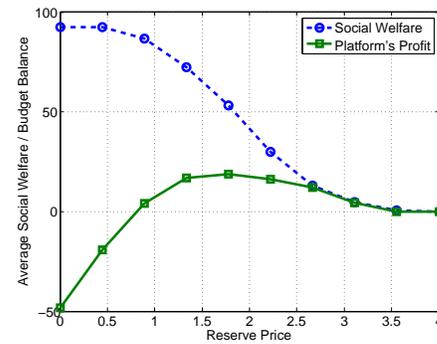


Fig. 10. Social Welfare and Budget Balance vs. Reserve Price. Tasks: 50, data: 30, users: 60, and the parameter  $\mu = 1$  in Zipf distribution.

representing the maximum social welfare with data reuse based on the result in Fig. 5. The green curve (with marker  $\square$ ) denotes the social welfare achieved in Mechanism 3 with data reuse. The red curve (with marker  $\circ$ ) and the cyan curve (with marker  $\diamond$ ) denote the results without data reuse.

From Fig. 7, we can see that the achieved social welfare increases with the task similarity parameter  $\mu$  with data reuse, while decreases with the parameter  $\mu$  without data reuse. The reason is as follows. With a higher task similarity  $\mu$ , most of the tasks' data requirements will concentrate on a smaller set of high popularity data. Hence, with data reuse, a smaller set of users (covering the high popularity data) are needed to cover all the required data requirements, leading to a higher social welfare; while without data reuse, a larger set of users are needed to cover all the required data multiple times, leading to a lower social welfare. Intuitively, without data reuse, the number of "effective" users in the high task similarity (i.e., those can sense high popularity data only) is fewer than that in the low task similarity (i.e., those who can sense any data item), hence the social welfare becomes smaller with a high task similarity.

From Fig. 7, we can also see that with data reuse, the social welfare (both the maximum social welfare and the social welfare achieved in Mechanism 3) can increase from 300% to 1300% when the task similarity increases from  $\mu = 0$  to  $\mu = 3$ , comparing with those without data reuse.

Furthermore, Fig. 8 shows the relative social welfare gain vs. the number of users for Mechanisms 2 and 3, and Fig. 9 shows the relative social welfare gain vs. the task similarity for Mechanisms 2 and 3. We can see that Fig. 8 is similar to

the green dash line in Fig. 2. The relative social welfare gain decreases with the number of users, and increases with the task similarity. In both Figs. 8 and 9, the randomized auction in Mechanism 3 leads to a relative performance gain very close to that of the social optimality (i.e., the fractional VCG auction in Mechanism 2). This verifies the effectiveness of the proposed randomized auction in Mechanism 3.

#### 6.4 Impact of the Reserve Price on the Budget Balance

Fig. 10 illustrates the impact of the reserve price on the social welfare and the budget balance of Mechanism 4. We can see that the social welfare always decreases with the reserve price, as a larger reserve price will drive more task owners out of the auction. The MCS platform can achieve the budget balance and gain a positive profit by setting a proper medium value reserve price. Moreover, the platform's profit first increases with the reserve price, due to the increase of the payments from task owners. When the reserve price is high enough, the platform's profit decreases with the reserve price until reaching zero. This is because a high reserve price may drive many task owners out of the auction, leading to a small social welfare and a small platform's profit.

## 7 CONCLUSION

In this work, we proposed a novel three-layer data-centric mobile crowdsensing model, which enables data reuse and leverages both the task similarity and the user heterogeneity.

We focused on the joint task selection and user scheduling problem, aiming at maximizing the social welfare. This problem is NP-hard and is challenging to solve due to the two-sided information asymmetry of selfish task owners and users. To understand the performance gain due to data reuse, we theoretically analyzed the social welfare gain with known statistical information, and proved the bound of the relative performance gain. To address both the limited computation and incomplete information issues, we proposed a two-sided randomized auction mechanism, which is computationally efficient, individually rational, and incentive compatible (truthful) in expectation. We further proposed a budget-balanced randomized auction mechanism to ensure the profitability of the platform in realistic settings.

## REFERENCES

- [1] C. Jiang, L. Gao, L. Duan, and J. Huang, "Exploiting data reuse in mobile crowdsensing," *Proc. IEEE GLOBECOM*, Dec. 2016.
- [2] R. K. Ganti, F. Ye, and H. Lei, "Mobile crowdsensing: current state and future challenges," *IEEE Commun. Mag.*, vol. 49, no. 11, pp. 32-39, Nov. 2011.
- [3] Waze: GPS Navigation, Maps & Social Traffic, [www.waze.com](http://www.waze.com)
- [4] OpenSignal: 3G and 4G LTE Cell Coverage Map, [opensignal.com](http://opensignal.com)
- [5] Intel Urban Atmosphere, [www.urban-atmospheres.net](http://www.urban-atmospheres.net)
- [6] T. Das, *et al.*, "PRISM: platform for remote sensing using smartphones," *Proc. ACM MobiSys*, June 2010.
- [7] M.-R. Ra, *et al.*, "Medusa: a programming framework for crowdsensing applications," *Proc. ACM MobiSys*, June 2012.
- [8] L. Duan, *et al.*, "Motivating smartphone collaboration in data acquisition and distributed computing," *IEEE Trans. Mobile Comput.*, vol. 13, no. 10, pp. 2320-2333, Oct. 2014.
- [9] D. Yang, G. Xue, X. Fang, and J. Tang, "Incentive mechanisms for crowdsensing: crowdsourcing with smartphones," *IEEE/ACM Trans. Netw.*, vol. 24, no. 3, pp. 1732-1744, June 2016.
- [10] X. Zhang, G. Xue, R. Yu, D. Yang and J. Tang, "Truthful incentive mechanisms for crowdsourcing," *Proc. IEEE INFOCOM*, Apr. 2015
- [11] C-K. Tham and T. Luo, "Fairness and social welfare in service allocation schemes for participatory sensing," *Elsevier Comput. Netw.*, vol. 73, pp. 58-71, Nov. 2014.
- [12] H. Jin, *et al.*, "Quality of information aware incentive mechanisms for mobile crowd sensing systems," *Proc. ACM MobiHoc*, Jun. 2015.
- [13] H. Jin, *et al.*, "INCEPTION: incentivizing privacy-preserving data aggregation for mobile crowd sensing systems," *Proc. ACM MobiHoc*, Jul. 2016.
- [14] H. Jin, *et al.*, "CENTURION: incentivizing multi-requester mobile crowd sensing," *Proc. IEEE INFOCOM*, May 2017.
- [15] N. Nisan and A. Ronen, "Computationally feasible VCG mechanisms," *J. Artif. Intell. Res.*, vol. 29, pp.19-47, May 2007.
- [16] S. Dobzinski, *et al.*, "On the power of randomization in algorithmic mechanism design," *SIAM J. Comput.*, vol. 42, no. 6, Dec. 2013.
- [17] R. Lavi and C. Swamy, "Truthful and near-optimal mechanism design via linear programming" *Proc. IEEE FOCS*, Oct. 2005.
- [18] H. Minooei and C. Swamy, "Truthful mechanism design for multidimensional covering problems," *Proc. ACM WINE*, Dec. 2012
- [19] L. Zhang, *et al.*, "Dynamic resource provisioning in cloud computing: a randomized auction approach," *Proc. IEEE INFOCOM*, 2014.
- [20] L. Zhang, *et al.*, "Randomized auction design for electricity markets between grids and microgrids," *Proc. ACM SIGMETRICS*, 2014.
- [21] M. Ahsanullah, V. B. Nevzorov, M. Shakil, "An introduction to order statistics," *Atlantis Studies in Probability and Statistics*, 2013.
- [22] C. P. Robert, "Monte carlo methods," John Wiley & Sons, Ltd, 2004.
- [23] J. R. Green and J.-J. Laffont, "Incentives in public decision making," North-Holland, Amsterdam, 1979.
- [24] R. Myerson and M. Satterthwaite, "Efficient mechanisms for bilateral trading," *J. Econ. Theory*, vol. 29, no. 2, pp. 265-281, Apr. 1983.
- [25] L. Breslau, *et al.*, "Web caching and Zipf-like distributions: evidence and implications," *Proc. IEEE INFOCOM*, Mar. 1999.
- [26] L. A. Adamic, "Zipf, Power-laws, and Pareto - a ranking tutorial," Xerox Palo Alto Research Center, Palo Alto, CA, 2000.
- [27] A. Schrijver, *Theory of linear and integer programming*, Wiley, 1998.
- [28] C. Jiang, L. Gao, L. Duan, J. Huang, "Data-centric mobile crowdsensing," *arXiv:1705.06055*, <https://arxiv.org/abs/1705.06055>



**Changkun Jiang** (S'14-M'17) received the Ph.D. degree in information engineering with The Chinese University of Hong Kong in 2017. His research interests include dynamic pricing and revenue management in communication networks, game theory and incentive mechanism design in network economics, and network optimization.



**Lin Gao** (S'08-M'10-SM'16) received M.S. and Ph.D. degrees in electronic engineering from Shanghai Jiao Tong University in 2006 and 2010, respectively. He was a Post-Doctoral Fellow with the Network Communications and Economics Laboratory, the Chinese University of Hong Kong, from 2010 to 2015. He is currently an Associate Professor with the Harbin Institute of Technology, Shenzhen, China. His research interests are in the interdisciplinary area combining telecommunications and microeconomics, with a particular focus on the game-theoretic and economic analysis for various communication and network scenarios, including cognitive radio networks, TV white space networks, cooperative communications, 5G communications, mobile crowd sensing, and IoT. He received the IEEE ComSoc Asia-Pacific Outstanding Young Researcher Award in 2016.



**Lingjie Duan** (S'09-M'12-SM'17) is an Assistant Professor with Engineering Systems and Design Pillar, Singapore University of Technology and Design (SUTD), Singapore. He received the Ph.D. degree from the Chinese University of Hong Kong in 2012. In 2011, he was a Visiting Scholar with University of California at Berkeley. His research interests include network economics and game theory, cognitive and cooperative communications, energy harvesting wireless communications, and mobile crowdsourcing. He is an Editor of IEEE TWC and IEEE COMST. In 2016, he was a Guest Editor of IEEE JSAC and IEEE Wireless Communications. He also served as a TPC member of many leading conferences in communications and networking (e.g., MobiHoc, INFOCOM, SECON, and WiOPT). He was the recipient of 2016 SUTD Awards - Excellence in Research, the 10th IEEE ComSoc Asia-Pacific Outstanding Young Researcher Award in 2015, and the Hong Kong Young Scientist Award in 2014.



**Jianwei Huang** (F'16) is a Professor in the Department of Information Engineering at the Chinese University of Hong Kong. He received the Ph.D. degree from Northwestern University in 2005, and worked as a Postdoc Research Associate at Princeton University during 2005-2007. Dr. Huang is the co-recipient of 8 Best Paper Awards, including IEEE Marconi Prize Paper Award in Wireless Communications in 2011. He has co-authored six books, including the textbook on "Wireless Network Pricing." He received the CUHK Young Researcher Award in 2014 and IEEE ComSoc Asia-Pacific Outstanding Young Researcher Award in 2009. Dr. Huang has served as an Associate Editor of IEEE/ACM ToN, IEEE TCCN, IEEE TNSE, IEEE TWC, and IEEE JSAC - Cognitive Radio Series. He has served as the Chair of IEEE MMTCC and TCCN. He is a Distinguished Lecturer of IEEE Communications Society, and a Thomson Reuters Highly Cited Researcher in Computer Science.