# Price Learning Based Incentive Mechanism for Mobile Crowd Sensing

YIFAN ZHANG, School of Computer Science and Engineering, South China University of Technology
XINGLIN ZHANG*, School of Computer Science and Engineering, South China University of Technology

Mobile crowd sensing (MCS) is an emerging sensing paradigm that can be applied to build various smart city and IoT applications. In an MCS application, the participation level of mobile users plays an essential role. Thus a great many incentive mechanisms have been proposed to motivate users. However, most of these works focus on the bidding behavior of users and overlook the feature of task requesters. Specifically, there exists a disparity between the low payment a requester would like to make and the high reward a user would like to receive. In this work, we address this issue by designing a group-buying-based online incentive mechanism, which contains two stages: In Stage I, a price learning algorithm is designed to select winning tasks for each group of sensing tasks, and obtain a competitive total budget for recruiting users. In Stage II, an online auction is conducted between group agents and online users before a given recruitment deadline. Through theoretical analysis and extensive evaluations, we show that the proposed mechanisms possess computational efficiency, individual rationality, budget balance, truthfulness, and good performance.

CCS Concepts: • **Human-centered computing → Collaborative and social computing; Mobile computing**; • **Networks → Location based services**.

Additional Key Words and Phrases: Mobile crowd sensing, incentive mechanism, group buying, price learning.

**ACM Reference Format:**
Yifan Zhang and Xinglin Zhang. 2020. Price Learning Based Incentive Mechanism for Mobile Crowd Sensing. *ACM Trans. Sensor Netw.* 1, 1 (November 2020), 24 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

## 1 INTRODUCTION

Mobile crowd sensing (MCS) is a prevalent sensing paradigm that harnesses the proliferation of smart mobile devices and mobile internet [9, 11]. It has been widely applied to build smart city and IoT applications, covering almost every aspect of our lives, such as urban traffic information mapping [16, 21, 24], visual summarization of objects [12, 28], object tracking [15], and environment monitoring [5, 17].

Sufficient participation is one of the critical factors for the aforementioned MCS systems, laying the foundation for efficient task allocation [3, 18, 22, 23] and determining whether the systems can ensure good quality of service. During the time of performing sensing tasks, mobile users consume various resources, such as battery power and data transmission cost, and endure the risk of privacy leakage. Hence it is essential to provide them with satisfiable rewards by delicate incentive design. Noticing this key issue, a great many incentive mechanisms have been proposed

*Corresponding Author.

Authors' addresses: Yifan Zhang, School of Computer Science and Engineering, South China University of Technology, yifanzhang.scut@foxmail.com; Xinglin Zhang, School of Computer Science and Engineering, South China University of Technology, zhxlinse@gmail.com.


1550-4859/2020/11-ART $15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

in the literature [6, 13, 26, 30, 34, 35, 41]. As these works mostly assume that the budget provided by the MCS platform or task requester is sufficient to attract users to perform sensing tasks, they focus on the characteristics of users when designing mechanisms.

However, in many MCS scenarios, a requester's budget for user recruitment may be insufficient to attract any user to perform a sensing task. For example, a requester may need a real-time photo of a nearby restaurant before he gets off work to determine whether it is open or not and then decide whether to eat at that restaurant. Naturally, the requester is unwilling to pay a lot for this photo (it is unacceptable if it costs more than his lunch). From the perspective of a user, he may be far away from this restaurant and needs to walk a long distance to complete the task before a certain deadline. Hence, the user expects to receive a relatively high reward for completing such tasks. Due to this disparity between the requester's expected budget and the user's expected reward, no user is likely to perform the small sensing task and the requester cannot obtain the expected sensing data.

To address the above issue, in this work, we propose a new group-buying-based online incentive framework where both requesters and users are effectively motivated to participate in MCS. Inspired by group buying services on the radio spectrum sharing [20, 31], we assume that tasks with the similar location requirement can be automatically grouped to obtain and share sensing capabilities of users. For example, some requesters are interested in crowdedness information of multiple nearby restaurants at lunch time and thus publish corresponding sensing tasks. These tasks can be grouped as a big task and be completed by a suitable user. In this way, the budgets of tasks in one group can be aggregated to form a total budget that is high enough to attract mobile users, although the budget of each single task is too low to recruit a user.

In our online incentive framework, we consider that there is an agent in each group as a representative. The agent runs a mechanism to decide which tasks should be incorporated (these tasks are termed as winning tasks) and collect their budgets as the group's budget. Whenever one user arrives, the MCS platform should make an immediate decision on whether the user should be recruited. If the platform decides to recruit this user, it also needs to decide which agent he is assigned to and how much money he is paid. At the same time, each winning agent will charge the winning tasks in his group for the payment.

Although group-buying-based mechanisms have been studied in spectrum auctions [20, 31] and a similar concept "task bundling scheme" has been proposed and used by some existing works in MCS [25, 29], no existing mechanism can be directly applied to our scenario and the following characteristics make our work challenging.

First, most works related to group buying do not take into account the online arrival of users, while in our scenario, we assume users arrive one by one online in a random order. Once a user arrives, we need to make an irrevocable decision on whether to accept the user's bid. It is a challenging task without knowing future information.

Second, a well-designed incentive mechanism needs to guarantee the crucial property of truthfulness, with which we can prevent users from disrupting the market through strategic untruthful bidding. Most existing works assume fixed budgets for sensing tasks, and only consider the bidding behaviors of users when designing truthful mechanisms. On the contrary, in the MCS system studied in this work, task requesters and online users can both bid freely. It is more complicated to design a double truthful mechanism, which ensures both users and requesters are truthful at the same time.

To accommodate the above challenges, we propose a two-stage incentive mechanism termed GMZ where the user's arrival time equals his departure time. The main idea of GMZ is as follows. In Stage I, nearby tasks form different groups. To increase the budget for each group's agent, we design a price learning algorithm PLnG. Given a set of candidate prices, PLnG chooses one price with a certain probability to evaluate each task and makes decisions accordingly. Each evaluated task will then serve as a sample to update the probability for selecting each candidate price. Through continuous learning, the algorithm will soon find a near-optimal evaluation price. In Stage II, users submit their bids upon arrival. To improve the platform's utility and ensure the truthfulness of users, we design a price learning algorithm PLnZ for determining winning user-agent pairs, which adopts a similar framework as PLnG. We then revise the GMZ mechanism and propose a more general GMNZ mechanism where users' have non-zero arrival-departure time intervals. GMNZ ensures that the user not only reports his bidding price truthfully, but also reports his true arrival time and departure time. Through theoretical analysis and extensive evaluations, we show that the proposed mechanisms possess computational efficiency, individual rationality, budget balance, truthfulness, and good performance.

The remainder of this paper is organized as follows: Section 2 presents the related work. In Section 3, we describe the system model and formulate the problem as a two-stage auction. We then present the group-buying-based online mechanisms GMZ and GMNZ in Section 4 and Section 5, respectively. In Section 6, we present the evaluation results. Finally, we conclude this paper in Section 7.

## 2 RELATED WORK

Research efforts have been made to develop various incentive mechanisms for MCS systems [37]. Among the different forms of incentive mechanisms, reverse auction-based mechanisms are studied comprehensively [6, 13, 26, 30, 34, 35, 39, 41]. These mechanisms focus on the user's strategic behavior in the MCS system and mostly assume that the budget for recruitment is sufficient. Our work differs from these works in that we observe the necessity of cooperation of task requesters with small budgets and thus propose a new incentive mechanism framework for MCS, which jointly considers the behaviors of requesters and users.

Considering double auction, Deshmukh *et al.* [7] proposed a technique to convert basic auctions into double auctions, ensuring the truthfulness of both sellers and buyers at the same time. Feng *et al.* [8] designed TAHES based on double auction considering the spatial heterogeneity of spectrum. Zhai e*t al.* [32] proposed a mechanism based on double auction to improve networks' benefit with high energy efficiency. Zhang *et al.* [33] proposed a double auction mechanism to ensure fair service trading considering applications of proximity-based mobile crowd services.

Furthermore, online auction is also considered in this paper. Blum *et al.* [2] designed competitive algorithms for the online market to efficiently match buying and selling bids without future information. Then they used an online learning algorithm in the auction and obtained a more efficient result [1]. Zhao *et al.* [41] proposed two mechanisms based on the online auction model, aiming to maximize the utility of the platform. Wei *et al.* [27] considered that both users and providers are dynamic and proposed truthful mechanisms based on online auction. Compared with these works, our work addresses the incentive mechanism design problem with three parties (i.e., requesters, agents, and mobile users) in MCS and considers the online arrival of users. The above-mentioned works cannot be directly applied to our scenario.

Group buying scheme is recently considered when designing spectrum auctions in cognitive radio networks and incentive mechanisms in MCS. To efficiently tackle the issue that an individual user cannot afford an integral spectrum channel, Lin *et al.* [20] proposed a novel three-stage mechanism TASG based on group buying. Huang *et al.* [14] extended TASG for MCS and proposed TGBA to tackle the mismatch of requesters' small budgets and workers' high prices. In addition, a similar concept "task bundling" is also used in MCS. To address the imbalance of user participation, Wang *et al.* [25] proposed a truthful incentive mechanism based on task bundling. Xie *et al.* [29] proposed a mechanism combining "task bundling scheme" and "rating system", which optimizes the system efficiency considering a service delay. Although the aforementioned works used the concept of "task bundling scheme" in different scenarios and tackled different problems, these mechanisms cannot adapt to our scenario where we not only need requesters to share the payment for users, but also aim to guarantee all participants' truthfulness and make good decisions whenever the users arrive.

## 3 SYSTEM MODEL

In this section, we first describe the group-buying scenario in MCS and formulate the problem of recruiting online mobile users as a two-stage auction. Then, we introduce the desired properties for incentive mechanisms in MCS.

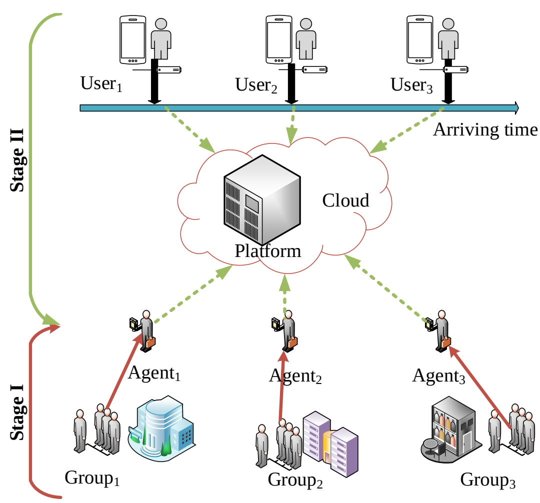



Fig. 1. A two-stage auction framework.

### 3.1 Problem Formulation

Figure 1 illustrates a group-buying scenario consisting of task requesters, agents, mobile users, and a platform:

- Each requester can submit a sensing task at a time. As each requester's budget is insufficient to recruit a user with a relatively high price, tasks form different groups in order to use an accumulated budget to recruit users.
- An agent is a virtual object generated by the platform (such as programs run by the platform), which acts as a representative for a group of tasks. The agents determine the winning tasks in each group, collect the budget of the tasks and report this information to the platform truthfully.
- Mobile users act as workers who are interested in performing sensing tasks and their information is unknown to the platform until they arrive at the area of interest and contact with the platform online. Upon arrival, the user negotiates with the platform for selling his sensing capability.

- The MCS platform, which resides in the cloud, is responsible for determining the winning user-agent pairs such that the sensing tasks can be assigned to suitable users and at the same time it achieves a high utility.

Mathematically, we consider that there are $n$ groups of sensing tasks $\mathcal{G} = \{\mathcal{G}_1, \mathcal{G}_2, ..., \mathcal{G}_n\}$, which should be assigned before a specified deadline $D$ [19, 41, 42].

Each group has a virtual agent (Hereinafter referred to as agent) and thus there are $n$ agents $\mathcal{A} = \{a_1, a_2, ..., a_n\}$. Note that these agents are not selected from requesters. They are virtual objects generated by the platform and are responsible for task selection and information collection. Therefore, they do not possess the property of utility and do not need to involve the property of truthfulness. In the group $\mathcal{G}_j$, there are $n_j$ tasks $\mathcal{G}_j = \{t_j^1, t_j^2, ..., t_j^{n_j}\}$. A task $t_j^k \in \mathcal{G}_j$ is identified by the tuple $(b_j^k, v_j^k, p_j^k, \mu_j^k)$, where the budget $b_j^k$ is the task's maximum payment for the sensing data, the valuation $v_j^k$ is the value of the sensing data that is only known to the task's requester himself, the payment $p_j^k$ is the reward given to the platform for completing the task, and $\mu_j^k$ reflects the requester's utility for participating in the auction. A crowd of mobile users $\mathcal{U} = \{u_1, u_2, ..., u_m\}$ would like to perform sensing tasks. The user $u_i \in \mathcal{U}$ is associated with the tuple $(v_i, r_i, \mu_i, \tau_i, d_i)$, where the reserved price $v_i$ reflects the cost for performing sensing tasks and is only known to the user. The user can set different reserved prices based on his current condition, such as the traveling cost and battery consumption. The reward $r_i$ is the payment received from the platform for completing sensing tasks, $\mu_i$ is the user's utility for participating in the auction, $\tau_i \in \{1, 2, ..., D\}$ and $d_i \in \{1, 2, ..., D\}$ are the user's true arrival time and true departure time, respectively. Each user is assumed to be game-theoretic. To maximize the utility, the user may strategically report a bidding price $s_i$ deviating from his reserved price or report an untruthful arrival time $\widehat{\tau_i}$ and a fake departure time $\widehat{d_i}$ subject to $\tau_i \leq \widehat{\tau_i} \leq \widehat{d_i} \leq d_i$ [36, 38, 41]. We assume that a user serves for at most one group of tasks here, since the tasks in each group are usually spatially clustered and thus are convenient for the user to complete.

We model the interaction between different characters as a two-stage auction. In Stage I, we assume that tasks are independent of each other, and we use the K-means algorithm to divide tasks into different groups based on their geographic locations. The number of tasks in each group cannot exceed the maximum group capacity $M$, which depends on the task's timeliness requirement. For example, in [4], the photo-taking task needs sufficient space-time complexity and hence $M$ can be set to a large value. Conversely, if a task requires a quick response, then $M$ should be set to a small value. After grouping tasks, a virtual agent is generated by the platform to represent the group for participating in Stage II. Then, the requester of the task $t_j^k$ submits his budget to the agent $a_j$. The agent $a_j$ will decide the set of winning tasks $\mathcal{W}_j$ and obtain the budget $b_j$ accordingly.

In Stage II, we recruit users before a recruitment deadline $D$. At first, a set of budgets $\{b_1, b_2, ..., b_n\}$ are submitted by agents and each arriving user submits his bid $s_i$ to the platform. The platform then makes an irrevocable decision on whether to accept the user's bid and which agent should be matched to this accepted user. Finally, the matching result $\Lambda = \{(u_i, a_j) | i \in \{1, 2, ..., m\}, j \in \{1, 2, ..., n\}\}$ is generated. Let $\Lambda_u = \{u_i | (u_i, a_j) \in \Lambda\}$ denote the matched users and $\Lambda_a = \{a_j | (u_i, a_j) \in \Lambda\}$ denote the matched agents. If the agent $a_j$ is successfully matched with a user (i.e., $a_j \in \Lambda_a$), the corresponding winning tasks in $\mathcal{W}_j$ will receive the sensing data and each task $t_j^k \in \mathcal{W}_j$ will be charged an amount of $p_j^k$. Thus the utility of a task $t_j^k \in \mathcal{G}_j$ is defined as:

$$\mu_j^k = \begin{cases} v_j^k - p_j^k, & t_j^k \in \mathcal{W}_j \text{ and } a_j \in \Lambda_a, \\ 0, & \text{otherwise}. \end{cases} \tag{1}$$

Table 1. Main Notations

| Symbol | Description |
|---|---|
| $\mathcal{U}, u_i, m$ | set of users, $i$th user and number of users |
| $\mathcal{G}, \mathcal{G}_j, n, n_j$ | set of groups, $j$th group, number of group, number of tasks in the $j$th group |
| $\mathcal{A}, a_j, b_j$ | set of agents, $j$th agent, agent $a_j$'s budget |
| $t_j^k, \mathcal{W}_j$ | $k$th task and set of winners in $j$th group |
| $v_j^k, b_j^k, p_j^k, \mu_j^k$ | task $t_j^k$'s valuation, budget, payment and utility |
| $v_i, s_i, r_i, \mu_i$ | user $u_i$'s reserved price, bid, reward and utility |
| $\tau_i, \widehat{\tau_i}, d_i, \widehat{d_i}$ | user $u_i$'s true arrival time, reported arrive time, true departure time, and reported departure time |
| $D, d$ | deadline in Stage II and one time step |
| $\alpha, \beta$ | learning rate and interval span |
| $l, h$ | lowest and highest budgets for tasks or bids for users |
| $\Lambda$ | matching between users and agents |
| $M$ | maximum group capacity |

For the user $u_i \in \Lambda_u$, he will be given a reward $r_i$ which is not smaller than his bid. The utility is the difference of his reward and his reserved price. If $u_i \notin \Lambda_u$, his utility will be 0. Therefore, the utility of $u_i$ can be defined as:

$$\mu_i = \begin{cases} r_i - v_i, & u_i \in \Lambda_u, \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

For a matching pair $(u_i, a_j) \in \Lambda$, the agent $a_j$ will hire the user $u_i$ to collect sensing data and the platform's utility is defined as the difference of agents' budgets and users' rewards:

$$\mu = \sum_{a_j \in \Lambda_a} b_j - \sum_{u_i \in \Lambda_u} r_i. \tag{3}$$

Note that for each $(u_i, a_j) \in \Lambda$, the value of $b_j$ must be not smaller than the value of $r_i$, which ensures that the utility of the platform is nonnegative.

Table 1 lists frequently used notations.

### 3.2 Desirable Properties

In this work, our purpose is to design group-buying-based two-stage incentive mechanisms, which are expected to satisfy the following properties.

- **Computational Efficiency:** An incentive mechanism is computationally efficient if it returns the result in polynomial time.
- **Individual Rationality:** A mechanism is individually rational if the utility of each participant is nonnegative. The proposed mechanisms should ensure that users and task requesters are all rational. In other words, Eq. (1) and Eq. (2) should have nonnegative results.
- **Budget Balance:** A mechanism is budget balanced if the utility of the auctioneer (i.e., the platform here) is nonnegative.
- **Truthfulness:** A mechanism is truthful if a bidder cannot improve his utility by submitting a bidding information deviating from his true information. Here, the bidders include

requesters and online-arriving users. If the requester cannot improve his utility by misreporting his budget and the user's dominant strategy is reporting his true reserved price, true arrival/departure time, then the two-stage mechanism is truthful.

## 4 ONLINE MECHANISM UNDER ZERO ARRIVAL-DEPARTURE INTERVAL SCENARIO

In this section, we propose a two-stage **G**roup-buying-based online incentive **M**echanism under the **Z**ero arrival-departure interval scenario (GMZ), where the arrival time of each user is equal to his departure time.

### 4.1 Stage I

In Stage I, the agent of each group runs an auction to select a subset of tasks in the group and determines the group's budget for recruiting users in Stage II. The key challenge here is how to maximize the agent's obtained budget, as a higher budget indicates a higher probability of successfully recruiting a user in Stage II. Some existing works consider that the idea of maximizing the agent's budget could result in untruthful bids of task requesters. Therefore they usually use some relatively random algorithms, such as the SAMU algorithm in [20] and the SUCP algorithm in [14]. Differently, we design a learning-based algorithm that tries to maximize the budget for each agent strategically and, at the same time, maintain the truthfulness of requesters.

Algorithm 1 sketches the designed PLnG algorithm in Stage I. We first initialize the learning rate $\alpha$, the interval span $\beta$, and the clearing price $c$, respectively. Let $l$ and $h$ be the lowest and highest possible budgets of all tasks, respectively. For any task $t_j^k$, we have $b_j^k \in [l, h]$. Let $\hat{\sigma}$ denote the largest index such that $l(1+\beta)^{\hat{\sigma}} < h$ and we then generate a set of candidate fixed prices $X = \{l(1+\beta)^{\sigma} | \sigma = 1, 2, ..., \hat{\sigma}\}$ according to the value of $\beta$. In other words, $X$ consists of all powers of $(1+\beta)$ between $l$ to $h$ (Lines 2-3).

To clearly explain how our algorithm works, we first need to introduce the optimal single-price auction in [10]. Assume that $\mathbf{b}$ is a sorted array of budgets in nonincreasing order, and $|\mathbf{b}|$ denotes the length of $\mathbf{b}$. Let $\Phi(\mathbf{b})$ denote the profit of the optimal single-price auction, which is defined by:

$$\Phi(\mathbf{b}) = \max_{1 \le j \le |\mathbf{b}|} j b_j, \tag{4}$$

where $b_j$ denotes the $j$th budget in $\mathbf{b}$.

In PLnG, the optimal price $b_{j^*}$ ($j^* = \arg\max_{1 \le j \le |\mathbf{b}|} j b_j$) is rounded to the nearest $l(1+\beta)^{\sigma}$ for the integer $\sigma$ and we want to learn the approximate price in the candidate set $X$. The value of $\beta$ thus determines how close the price we learn can be to the optimal price. With a smaller $\beta$, there will be more candidate prices, and the optimal price can be better approximated. But it also increases the cost of learning.

Each candidate price $x_\sigma \in X$ maintains two parameters: given the evaluated set of tasks $\{t_j^1, ..., t_j^k\}$, $r_\sigma(k)$ represents the total budget that the agent can obtain by choosing $x_\sigma$ as the evaluation price; $w_\sigma(k)$ is the weight of $x_\sigma$ after evaluating the first $k$ tasks, which will affect the agent's probability of choosing $x_\sigma$ for evaluation (Lines 5-7). For any task $t_j^k$ in the group $\mathcal{G}_j$, we choose a price $c = x_\sigma$ with the probability $\frac{w_\sigma(k-1)}{\sum_{\sigma'=1}^{\hat{\sigma}} w_{\sigma'}(k-1)}$ to evaluate this task. If $t_j^k$'s budget $b_j^k$ is larger than $c$, he will be a winner and make a payment $p_j^k = c$; Otherwise, he loses and pays nothing (Lines 13-14). After evaluating one task, we use it as a sample for parameter update (Lines 15-23). For each candidate price $x_\sigma \in X$, let $g_\sigma(k)$ be the agent's budget obtained from the $k$th task. If the task's budget is larger than $x_\sigma$, $g_\sigma(k)$ will be set to $x_\sigma$; Otherwise, $g_\sigma(k)$ will be set to 0. Then we use the cumulative sum to update $r_\sigma(k)$ and update $w_\sigma(k)$ according to $r_\sigma(k)$. This update method guarantees the

**Algorithm 1** PLnG: **P**rice **L**earni**n**g algorithm for determining the budget and winners for all **G**roups

**Input:** Each group's budget vector $\mathbf{b}_j = \{b_j^k \in [l, h] \mid k = 1, ..., |\mathcal{G}_j|\}$, the learning rate $\alpha \geq e - 1$, and the interval span $\beta \in (0, 1]$.

**Output:** Each agent's budget $b_j$ and each group's set of winning tasks $\mathcal{W}_j$.

$c \leftarrow 0$;
Let $\hat{\sigma}$ denote the largest index such that $l(1+\beta)^{\hat{\sigma}} < h$;
$X \leftarrow \{l(1+\beta)^{\sigma} | \sigma = 1, 2, ..., \hat{\sigma}\}$;
**for** $j = 1$ to $n$ **do**
  **for** $\sigma = 1$ to $\hat{\sigma}$ **do**
    $r_\sigma(0) \leftarrow 0; w_\sigma(0) \leftarrow 1$;
  **end for**
  **for** $k = 1$ to $n_j$ **do**:
    Set $c \leftarrow x_\sigma$ with probability $\frac{w_\sigma(k-1)}{\sum_{\sigma'=1}^{\hat{\sigma}} w_{\sigma'}(k-1)}$;
    **if** $b_j^k \geq c$ **then**
      $p_j^k \leftarrow c; \mathcal{W}_j \leftarrow \mathcal{W}_j \cup \{t_j^k\}$;
    **else**
      $p_j^k \leftarrow 0$;
    **end if**
    **for** $\sigma = 1$ to $\hat{\sigma}$ **do**
      **if** $b_j^k \geq x_\sigma$ **then**
        $g_\sigma(k) \leftarrow x_\sigma$;
      **else**
        $g_\sigma(k) \leftarrow 0$;
      **end if**
      $r_\sigma(k) \leftarrow r_\sigma(k-1) + g_\sigma(k)$;
      $w_\sigma(k) \leftarrow (1+\alpha)^{r_\sigma(k)/h}$;
    **end for**
  **end for**
**end for**

good competitiveness of our algorithm. Here we give a simple example to further illustrate the idea of PLnG.

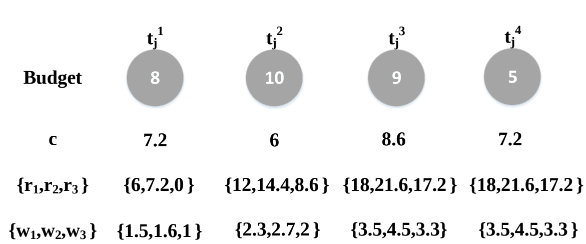



Fig. 2. An illustrative example of running PLnG.

**Example 1.** *As shown in Figure 2, there are 4 tasks in the group $\mathcal{G}_j$. We first get the lowest and highest budgets $l = 5, h = 10$. Assume $\alpha = 1, \beta = 0.2$ and we can obtain the largest index $\hat{\sigma} = 3$ such that $5 * 1.2^3 < 10$ and $5 * 1.2^4 > 10$. Then we generate the candidate price set $X = \{5 * 1.2^\sigma | 1 \leq \sigma \leq \hat{\sigma}\} = \{6, 7.2, 8.6\}$. At the beginning of the outer for-loop of PLnG, we have three candidate prices $x_1 = 6, x_2 = 7.2, x_3 = 8.6$ and we initialize their corresponding parameters $r_1(0) = r_2(0) = r_3(0) = 0$ and $w_1(0) = w_2(0) = w_3(0) = 1$. First, $t_j^1$ will be evaluated and one of the candidate price will be chosen as the clearing price $c$. The probability of selecting these three candidate prices at this time are all $\frac{1}{3}$. Assume that PLnG sets $c = 7.2$. As $b_j^1 = 8 > 7.2$, $t_j^1$ will be a winner. From Lines 15-23, as $b_j^1 = 8 > x_1 = 6$, we have $g_1(1) = 6$. And we can update $r_1(1) = r_1(0) + 6 = 6, w_1(1) = (1+1)^{r_1(1)/h} = 2^{0.6} = 1.5$. Similarly, we update $r_2(1) = 0 + 7.2 = 7.2, r_3(1) = 0 + 0 = 0$ and $w_2(1) = 2^{0.72} = 1.6, w_3(1) = 1$. Then we evaluate $t_j^2$, whose budget is 10. Now we have probability $\frac{1.5}{1.5+1.6+1} = 0.36$ to set $c = x_1$. Accordingly, we set $c = x_2$ with probability 0.39 and set $c = x_3$ with probability 0.24. Assume that we set $c = x_1 = 6$ which is smaller than $b_j^2$, then $t_j^2$ is a winner. Similarly, we update each candidate price's parameters $r_\sigma(2) = r_\sigma(1) + g_\sigma(2)$ and $w_\sigma(2) = 2^{r_\sigma(2)/10}$. The subsequent calculation results are shown in Figure 2.*

## 4.2 Stage II

After determining the winners and the total budget of each group, in Stage II, a double auction is performed between users and agents, aiming at maximizing the platform's utility. The challenge here is how to make a decision upon each user's arrival and ensure his truthfulness.

Algorithm 2 depicts the PLnZ algorithm for Stage II. We first sort the budget vector in descending order and find the index $n^*$ such that $n^* b_{n^*}$ is maximized (Line 1). Then we initialize parameters $\alpha$, $\beta$, $c$ and $m^*$ (Line 2). We generate the set of candidate prices $X = \{l(1+\beta)^\sigma | \sigma = 1, 2, ..., \hat{\sigma}\}$ similar to PLnG, where $l/h$ is the lowest/highest possible bid of users (Lines 3-4). For each $x_\sigma \in X$, we maintain two parameters $r_\sigma(i)$ and $w_\sigma(i)$, where $r_\sigma(i)$ denotes the total profit by using $x_\sigma$ to evaluate the sequence $\{u_1, u_2, ..., u_i\}$ and $w_\sigma(i)$ is $x_\sigma$'s weight (Lines 5-7). Note that the evaluation sequence here is a set of users. At each time step $d$, if we have selected $n^*$ users, the algorithm halts and returns the matching result (Lines 10-12). Otherwise, if there arrives one user, we set $c = x_\sigma$ with the probability $\frac{w_\sigma(i-1)}{\sum_{\sigma'=1}^{\hat{\sigma}} w_{\sigma'}(i-1)}$ to be the price for evaluating this user. If the user's bid $s_i \leq c$, he will be selected and given a reward $c$. In other cases, his reward will always be 0 (Lines 14-17). After that, he will be matched with a group by the function $Match(i, \Lambda)$, which returns the matched agent's index according to the historical matching result $\Lambda$ and the user $u_i$'s information. In this paper, we calculate each group's center point and match the user with the group that is closest to him and is not assigned. Finally, we update the parameters according to this user's bid (Lines 18-25). This process is similar to PLnG. The difference is that, for each $x_\sigma \in X$, if the arriving user can be selected when we use $x_\sigma$ for evaluation, the profit $g_\sigma(i)$ brought by $u_i$ will be $h - x_\sigma$. Setting $g_\sigma(i) = h - x_\sigma$ instead of $g_\sigma(i) = x_\sigma$ tends to give less reward.

## 4.3 Theoretical Analysis

Here, we analyze the properties of GMZ.

### *4.3.1 The Analysis of Stage I.*

**Lemma 1.** *PLnG is computationally efficient.*

Proof. The innermost for-loop (Lines 15-23) of PLnG is bounded by $O(\hat{\sigma})$. The middle for-loop (Lines 8-24) is bounded by $O(n_j)$. Since $n_j \leq M$, it is also bounded by $O(M)$. Finally, the outermost for-loop (Lines 4-25) is bounded by $O(n)$. In summary, we can conclude that PLnG is bounded by $O(\hat{\sigma} nM)$. □

**Algorithm 2** PLnZ: **P**rice **L**earn**n**ing algorithm for matching agents and users under **Z**ero arrival-departure interval scenario.

**Input:** Agents' budget vector $\mathbf{b} = \{b_j \in [l, h] | j = 1, 2, ..., n\}$ in descending order, the learning rate $\alpha \geq e - 1$, the interval span $\beta \in (0, 1]$, each arriving user's arrival time $\tau_i$ and bidding price $s_i$.
**Output:** A matching between online users and agents $\Lambda$ and the corresponding reward.

$n^* \leftarrow \arg\max_{1 \leq j \leq n} jb_j; h \leftarrow b_{n^*};$
$c \leftarrow 0; m^* \leftarrow 0;$
Let $\hat{\sigma}$ denote the largest index such that $l(1+\beta)^{\hat{\sigma}} < h$;
$X \leftarrow \{l(1+\beta)^{\sigma} | \sigma = 1, 2, ..., \hat{\sigma}\}$;
**for** $\sigma = 1$ to $\hat{\sigma}$ **do**
    $r_\sigma(0) \leftarrow 0; w_\sigma(0) \leftarrow 1;$
**end for**
**while** $d \leq D$ **do**
    **while** there is a user $u_i$ arriving at time step $d$ **do**
        **if** $m^* = n^*$ **then**
            halt and return the result;
        **end if**
        Set $c \leftarrow x_\sigma$ with probability $\frac{w_\sigma(i-1)}{\sum_{\sigma'=1}^{\hat{\sigma}} w_{\sigma'}(i-1)}$;
        **if** $s_i \leq c$ **then**
            $r_i \leftarrow c; m^* \leftarrow m^* + 1; \Lambda \leftarrow \Lambda \cup \{(i, Match(i, \Lambda))\};$
        **else** $r_i \leftarrow 0$;
        **end if**
        **for** $\sigma = 1$ to $\hat{\sigma}$ **do**
            **if** $s_i \leq x_\sigma$ **then**
                $g_\sigma(i) \leftarrow h - x_\sigma;$
            **else** $g_\sigma(i) \leftarrow 0$;
            **end if**
            $r_\sigma(i) \leftarrow r_\sigma(i-1) + g_\sigma(i);$
            $w_\sigma(i) \leftarrow (1+\alpha)^{r_\sigma(i)/h};$
        **end for**
    **end while**
    $d \leftarrow d + 1;$
**end while**

**Lemma 2.** *PLnG is individually rational.*

Proof. Consider that $t_j^k \notin \mathcal{W}_j$ or $a_j \notin \Lambda_a$, from Eq. (1), the utility of $t_j^k$ will be 0. Otherwise, the requester's payment will be $p_j^k = c$ (Line 11 in Algorithm 1). Note that $b_j^k \geq c = p_j^k$ if the requester reports his true valuation, i.e., $b_j^k = v_j^k$. Hence we have $b_j^k - p_j^k \geq 0$, which completes our proof. □

**Lemma 3.** *PLnG is truthful for task requesters.*

Proof. Given a task $t_j^k$, if its agent $a_j$ loses in Stage II, then $\mu_j^k$ will always be 0 regardless of the submitted budget. If $a_j$ wins in Stage II, we consider two cases regarding $t_j^k$'s valuation.

**Case (a)**: $v_j^k \geq c$, where $c$ is a fixed price chosen with a certain probability (Line 11 in Algorithm 1). If the task requester reports his true valuation, namely $b_j^k = v_j^k$, his utility will be $v_j^k - p_j^k = v_j^k - c \geq 0$.

His utility will not change if he reports any budget above $c$. However, reporting any budget $b_j^k < c$ will make him lose the auction and his utility will then be 0.

**Case (b)**: $v_j^k < c$. In this case, reporting the true budget will make the requester's utility be 0. However, if he submits a budget $b_j^k \geq c$, he will be a winner while his utility becomes $v_j^k - c < 0$.

In summary, reporting the true budget is a dominant strategy for the task requester. □

In Stage I, the metric of the algorithm performance is the amount of budget the agent receives. Here, a mechanism $\Psi$ is said to be $\alpha$-competitive if, for any group of tasks $\mathcal{G}_j$ with a budget vector $\mathbf{b} = [b_j^1, b_j^2, ..., b_j^{n_j}]$, the expected budget obtained by $\Psi$ satisfies $\mathbb{E}[\Psi(\mathbf{b})] \geq \Phi(\mathbf{b})/\alpha$, where $\Phi$ is the optimal single price mechanism. Then we can conclude the theoretical performance of PLnG by the following lemma.

**Lemma 4.** *Given $\epsilon \in (0, 1/(e-1)]$, PLnG is $\frac{3(1+\epsilon)}{2\epsilon}$-competitive.*

Proof. The proof of Lemma 4 is given in Appendix A. □

With Lemma 4, We can show the advantage of PLnG compared with SAMU [20] and SUCP [14]. Specifically, assume that the budget of a task in the group $\mathcal{G}_j$ obeys an uneven distribution:

$$b_j^k = \begin{cases} n_j, & i = 1, 2, \\ 1, & 3 \leq i \leq n_j, \end{cases}$$

where $n_j \gg 1$. Let $\mathbf{b} = \{b_j^k | 1 \leq k \leq n_j\}$, we have $\Phi(\mathbf{b}) = 2n_j$. Given any $\epsilon \in (0, e-1]$, PLnG is at least $\frac{3(1+\epsilon)}{2\epsilon}$-competitive and thus PLnG($\mathbf{b}$) $\geq \frac{4n_j\epsilon}{3(1+\epsilon)}$. According to SUCP, $a_j$ randomly chooses $m \in [1, n_j]$ and uses $b_j^m$ as the clearing price, selecting tasks whose budgets are larger than $b_j^m$. If there are $\omega$ selected tasks, SUCP's budget will be $b_j^m\omega$. Therefore, the expected budget of $a_j$ is $P[m = 1 | m = 2] * 0 + P[m > 2] * 2 = \frac{2}{n_j} * 0 + \frac{n_j - 2}{n_j} * 2 = (2 - \frac{4}{n_j})$. Thus, the ratio of the expected budget achieved by PLnG to that by SUCP is $\frac{\frac{4n_j\epsilon}{3(1+\epsilon)}}{2 - \frac{4}{n_j}} > \frac{2\epsilon}{3(1+\epsilon)} n_j$. Similarly, the ratio between PLnG and SAMU is also $\Omega(n_j)$, which shows the superiority of PLnG.

**Theorem 1.** *PLnG is computationally efficient, individually rational, truthful and $\frac{3(1+\epsilon)}{2\epsilon}$-competitive, given $\epsilon \in (0, 1/(e-1)]$.*

### 4.3.2 *The Analysis of Stage II.*

**Lemma 5.** *PLnZ is computationally efficient.*

Proof. The while-loop executed at each time step (Lines 9-26) is the most time-consuming part of PLnZ. There are no more than $m$ users arriving at one time step. For each user, the for-loop in Lines 18-25 is bounded by $O(\hat{\sigma})$. Therefore, PLnZ is bounded by $O(\hat{\sigma}m)$. □

**Lemma 6.** *PLnZ is individually rational and budget balanced.*

Proof. From Line 15 we can see that each winning user has $r_i = c \geq s_i$ and thus $r_i - s_i \geq 0$. If he is not selected, his payment will be 0 and thus the utility will be 0 too.

From Lines 3-4, for each $x_\sigma \in X$, we have $x_\sigma \leq h = b_{n^*}$. For each winning user, his reward must be a price $x_\sigma \in X$, i.e., each user's reward is less than the minimum budget, which completes our proof. □

**Lemma 7.** *PLnZ is truthful for users.*

Proof. The proof of PLnZ's truthfulness follows the same procedure of the proof of PLnG's truthfulness, and thus is omitted here. □

**Theorem 2.** *PLnZ is computationally efficient, individually rational, budget balanced, and truthful.*

*4.3.3 The Analysis of GMZ.* As GMZ is the sequential combination of PLnG and PLnZ, it can be directly inferred that GMZ satisfies the properties of computational efficiency, individual rationality, budget balance, and double truthfulness (i.e. truthfulness for both task requesters and users).

**Theorem 3.** *GMZ is computationally efficient, individually rational, budget balanced, and double truthful.*

## 5 ONLINE MECHANISM UNDER NON-ZERO ARRIVAL-DEPARTURE INTERVAL SCENARIO

In this section, we consider a more general scenario where each user has non-zero arrival-departure interval. In this case, the user's bidding information contains not only the bidding price but also the arrival/departure time. We first use an example to show that GMZ is not truthful in this scenario.

**Example 2.** *Suppose there is a user $u_i$ with $v_i = s_i = 5$, $\tau_i = 1$, $d_i = 2$. At the time step $d = 1$, the user is evaluated. Assume the clearing price $c$ in PLnZ is 6. As the user's bid is smaller than $c$, he will be rewarded $r_i = 6$ and $\mu_i = 1$. However, if he reports an untruthful $\widehat{\tau_i} = 2$. At the time step $d = 2$, assuming that the clearing price $c = 10$, then $r_i$ will be 10. In other words, by announcing an untruthful arrival time, the user improves his utility from 1 to 5 according to GMZ.*

To accommodate this issue, we revise Stage II of GMZ and propose a **G**roup-buying-based online incentive **M**echanism under the **N**on-**Z**ero arrival-departure interval scenario (GMNZ).

### 5.1 Revised Stage II

In GMNZ, the user can not only declare an untruthful bidding price but also announce a later arrival time or an earlier departure time to improve his utility. The challenge here becomes how to ensure the truthfulness of these strategic users.

Since only the nature of the users has changed, the process of Stage I is not affected. Therefore, we only need to modify the process of Stage II, which is sketched in the PLnNZ algorithm (Algorithm 3). Similar to Algorithm 2, we first initialize the required parameters (Lines 1-7). The difference here is that we initialize an active set $\mathcal{S}$, which consists of all users who have been selected by the platform and have not yet departed at each time step. At the time step $d$, we conduct the user evaluation and parameter update using the same strategy as Algorithm 2. Meanwhile, whenever a user is selected, we add this user to the active set $\mathcal{S}$ (Line 16). To ensure the truthfulness in this case, we update the payments of users in $\mathcal{S}$ at the end of the current time step (Lines 28-32). Specifically, the reward of each user $u_i \in \mathcal{S}$ is set to $\max\{r_i, c\}$, namely the maximal candidate price chosen during $[\widehat{\tau_i}, \widehat{d_i}]$. Here, $\widehat{\tau_i}$ and $\widehat{d_i}$ are the user $u_i$'s reported arrival and departure time, respectively. Finally, we remove users who depart at the current time step from $\mathcal{S}$.

With the design of PLnNZ, let us reconsider Example 2. If the user $u_i$ reports his information truthfully, PLnNZ works as follows: when the user reports his true arrival and departure time $\tau_i = 1$ and $d_i = 2$, he will be evaluated with a clearing price $c = 6$ at $d = 1$. He then will be selected with a reward $r_i = 6$ and added to $\mathcal{S}$. At $d = 2$ with the clearing price being 10, the user $u_i$'s reward will become $r_i = \max\{6, 10\} = 10$. As $d_i = 2$, the user will be removed from $\mathcal{S}$ and his final utility is $\mu_i = 5$. Thus the user $u_i$ can obtain the utility of 5 according to the GMNZ mechanism by using PLnNZ. Even if he reports a later arrival time or an earlier departure time, he still cannot improve his utility. In other words, the truthfulness of users can be guaranteed in this example.

**Algorithm 3** PLnNZ: **P**rice **L**ear**n**ing algorithm for matching agents and online users under **N**on-**Z**ero arrival-departure interval scenario.

**Input:** Agents' budget vector $\mathbf{b} = \{b_i \in [l, h]\}_{i=1}^{n}$ in descending order, the learning rate $\alpha \geq e - 1$, the interval span $\beta \in (0, 1]$, each arriving user's arrival/departure time $\tau_i/d_i$ and his bidding price $s_i$.

**Output:** A matching between online users and agents $\Lambda$ and the corresponding reward.

$n^* \leftarrow \arg\max_{j \geq 1} jb_j; h \leftarrow b_{n^*}; \mathcal{S} \leftarrow \emptyset$;
$c \leftarrow 0, m^* \leftarrow 0$;
Let $\hat{\sigma}$ denote the largest index such that $l(1+\beta)^{\hat{\sigma}} < h$;
$X \leftarrow \{l(1+\beta)^{\sigma} | \sigma = 1, 2, ..., \hat{\sigma}\}$;
**for** $\sigma = 1$ to $\hat{\sigma}$ **do**
    $r_\sigma(0) \leftarrow 0; w_\sigma(0) \leftarrow 1$;
**end for**
**while** $d \leq D$ **do**
    Set $c \leftarrow x_\sigma$ with probability $\frac{w_\sigma(i-1)}{\sum_{\sigma'=1}^{\hat{\sigma}} w_{\sigma'}(i-1)}$;
    **while** there is a user $u_i$ arriving at time step $d$ **do**
        **if** $m^* = n^*$ **then**
            halt and return the result;
        **end if**
        **if** $s_i \leq c$ **then**
            $r_i \leftarrow c; m^* \leftarrow m^* + 1$;
            $\Lambda \cup \{(i, Match(i, \Lambda))\}; \mathcal{S} \leftarrow \mathcal{S} \cup \{u_i\}$;
        **else** $r_i \leftarrow 0$;
        **end if**
        **for** $\sigma = 1$ to $\hat{\sigma}$ **do**
            **if** $s_i \leq x_\sigma$ **then**
                $g_\sigma(i) \leftarrow h - x_\sigma$;
            **else** $g_\sigma(i) \leftarrow 0$;
            **end if**
            $r_\sigma(i) \leftarrow r_\sigma(i-1) + g_\sigma(i)$;
            $w_\sigma(i) \leftarrow (1+\alpha)^{r_\sigma(i)/h}$;
        **end for**
    **end while**
    **for** each user $u_i \in \mathcal{S}$ **do**
        **if** $c > r_i$ **then**
            $r_i \leftarrow c$;
        **end if**
    **end for**
    $d \leftarrow d + 1$;
    Remove users departing at time step $d$ from $\mathcal{S}$.
**end while**

## 5.2 Mechanism Analysis

Here, we analyze the desirable properties of GMNZ.

**Lemma 8.** *GMNZ is computationally efficient, individually rational and budget balanced.*

Proof. Because the frame of PLnNZ is similar to that of PLnZ, the proofs of computational efficiency, individual rationality, and budget balance are almost identical and hence the details are omitted here. □

**Lemma 9.** *GMNZ is double truthful.*

Proof. The proof of Lemma 9 is given in Appendix B. □

The following theorem summarizes the properties of GMNZ.

**Theorem 4.** *GMNZ is computationally efficient, individually rational, budget balanced, and double truthful.*

## 6 EVALUATION

In this section, we evaluate the performance of GMZ and GMNZ. We first compare PLnG with $\Phi$, SUCP [14], and SAMU [20] to show the performance of the designed pricing learning auction. $\Phi$ is the optimal single price mechanism as introduced in Section 4.1. Given a group of tasks $\mathcal{G}_j$, SUCP randomly chooses one task $t_j^k$ and uses $b_j^k$ as the clearing price, selecting all tasks with budgets larger than $b_j^k$. If there are $\kappa$ winning tasks, the budget for $a_j$ will be $\kappa b_j^k$. SAMU sorts $\mathcal{G}_j$ in descending order according to each task's budget. Then it chooses $m \in [1, |\mathcal{G}| - 1]$ randomly and the clearing price will be $b_j^{|\mathcal{G}|-m+1}$. Finally, the first $|\mathcal{G}_j| - m$ tasks will be winners and $a_j$'s budget will be $(|\mathcal{G}_j| - m)b_j^{|\mathcal{G}|-m+1}$.

We then compare the overall performance of GMZ and GMNZ with the offline incentive mechanism (OIM) and the Vickery mechanism adapted to the double auction with a random selection (VIC). OIM uses $\Phi(\mathbf{b})$ to maximize the agent's budget in Stage I. In Stage II, OIM applies $\Phi(\mathbf{b}, \mathbf{s})$ to maximize the utility of the platform. $\Phi(\mathbf{b}, \mathbf{s})$ is defined as

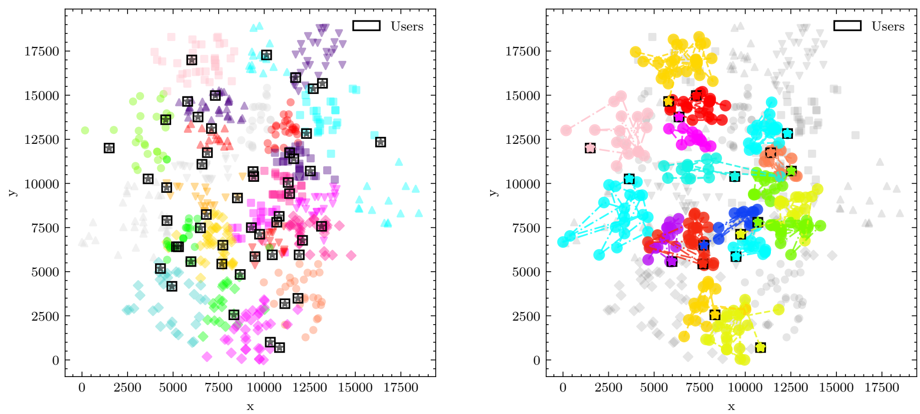
$$\Phi(\mathbf{b}, \mathbf{s}) = \max_{1 \le k \le \varepsilon} k(b_k - s_k), \tag{5}$$

where $\varepsilon = \min\{|\mathbf{b}|, |\mathbf{s}|\}$. VIC uses SUCP in Stage I. In Stage II, given the nonincreasing budget vector $\mathbf{b}$ and the nondecreasing bid vector $\mathbf{s}$, VIC finds the largest $\hat{\sigma}$ such that $b_{\hat{\sigma}} \ge s_{\hat{\sigma}}$ and then chooses $m \in [1, \hat{\sigma} - 1]$ randomly. The first $m$ agents and $m$ users win with the payment $b_{m+1}$ and the reward $s_{m+1}$, respectively. Finally, we show the truthfulness of GMZ and GMNZ, respectively.

### 6.1 Experimental Setup

We conduct experiments with both synthetic and real-world datasets. In the synthetic dataset, the locations of users and tasks are generated in a 2D data space $[0, 500]^2$. We generate 30 users ($m = 30$) and 200 tasks ($\sum_j n_j = 200$) in total, varying the maximum group capacity $M$ from 10 to 40 with an increment of 10. The number of agents $n$ is determined by the result of task grouping. In Stage I, each task $t_j^k$'s valuation $v_j^k$ and budget $b_j^k$ are uniformly distributed in $[5, 10]$. In Stage II, each user's reserved price $s_i$ and bid $v_i$ are uniformly distributed in $[20, 40]$, which are larger than the budget of any individual task. We set the deadline $T = 500$, the learning rate $\alpha = 5$ and the interval span $\beta = 0.2$ as default parameters. Each user's arrival time satisfies the uniform distribution and the arrival-departure interval is uniformly distributed over $[10, 30]$. Whenever one user arrives, he will be placed at a random location. We also vary the learning rate $\alpha$ from 1 to 100 with an increment of 1 and vary the interval span $\beta$ from 0 to 1 with an increment of 0.01 to investigate their influence on the mechanisms. Each measurement is averaged over 500 instances.

We also use the real-world dataset T-Drive [4, 40] in the experiment. T-Drive contains the GPS trajectories of 10,357 taxis during the period from Feb. 2 to Feb. 8, 2008, within Beijing, China. We randomly select 50 trajectories and sample one point in each trajectory as a user. Similarly, we




(a) Distribution map of tasks and users sampled from taxi trajectories. Different colors represent different task groups.

(b) Matched result. The matched user and group are drawn in the same color and connected.

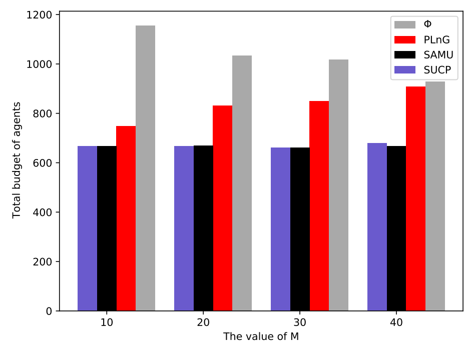
Fig. 3. Visualization of the T-Drive dataset.

randomly select 650 trajectories and sample one point in each of these trajectory as a task. The position coordinates of the sampled points are used to initialize the positions of tasks and users. Other parameters are the same as in the synthetic dataset.

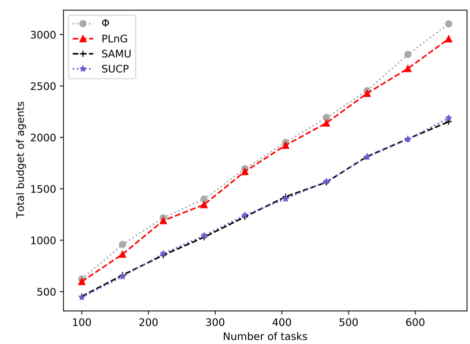
## 6.2 Experiment result

*6.2.1 Visualization of the T-Drive dataset.* We first approximately convert all latitude and longitude coordinates of the sampled points into plane coordinates. After eliminating outliers, we fix all points in a two-dimensional plane of 18000m*18000m. As shown in Figure 3(a), the 650 task points are divided into 25 groups in different colors. The 50 user points are depicted by black bounding boxes. Figure 3(b) shows the matching result by using the proposed mechanism GMZ. The winning tasks in most groups (colorful points) are successfully matched with nearby users who have relatively low bids, while some groups (grey points) cannot be matched with suitable users. This phenomenon is reasonable as (1) there is a strategy in Stage II that matches a user with the closest available agent and (2) few users locate near the unmatched groups and their bids are likely to be dominated by the majority of users.

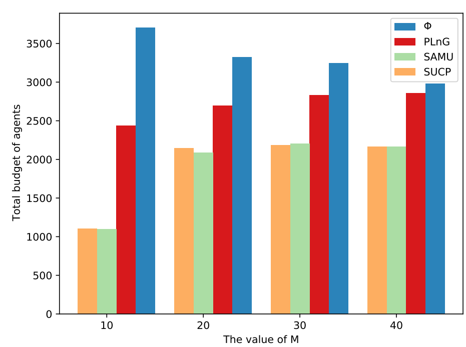
*6.2.2 Comparison of different algorithms used in Stage I.* Figure 4 compares the budgets given to agents calculated by PLnG, $\Phi(\mathbf{b})$, SUCP, and SAMU on two datasets respectively. Generally, the optimal mechanism $\Phi(\mathbf{b})$ obtains the highest budget compared with the other algorithms. As SUCP and SAMU randomly choose one clearing price to evaluate tasks, their performances are very close. The proposed PLnG performs better than SUCP and SAMU, which verifies its price learning capability. Considering the synthetic dataset, in Figure 4(a), it can be seen that the gap between PLnG and $\Phi(\mathbf{b})$ is the biggest when $M = 10$. This is because the strength of the learning strategy is not fully utilized as there are only a few tasks in each group. With the increment of $M$, the performance advantage of PLnG compared to SUCP and SAMU becomes larger and the gap between PLnG and $\Phi(\mathbf{b})$ becomes smaller. In Figure 4(b), we set $M = 40$. The performance of PLnG is very close to $\Phi(\mathbf{b})$ and is much better than SUCP and SAMU with different numbers of tasks. Figure 4(c) and Figure 4(d) show the performance on the T-Drive dataset when we vary the number



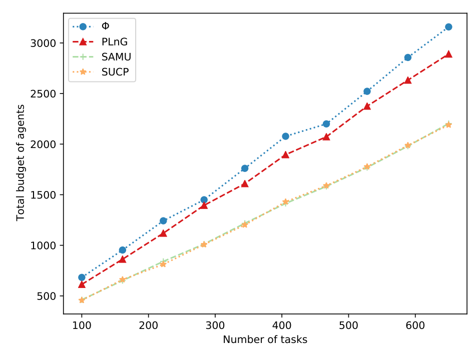

(a) Collected budgets by varying $M$.




(b) Collected budgets by varying #tasks.




(c) Collected budgets by varying $M$.




(d) Collected budgets by varying #tasks.

Fig. 4. Comparison of different algorithms used in Stage I. (a)(b): Synthetic Dataset. (c)(d): T-Drive Dataset.

of tasks and the value of $M$. Again, PLnG performs better than SUCP and SAMU and has a close performance compared with $\Phi(\mathbf{b})$.

*6.2.3 Convergence of the candidate price selection probability.* Here, we choose group $\mathcal{G}_1$ on the T-Drive dataset to study the convergence speed of PLnG. $\mathcal{G}_1$ has 31 tasks and the smallest and largest budgets are 5.005 and 9.892, respectively. Let $\mathbf{b}$ denote the nonincreasing budget vector of tasks in $\mathcal{G}_1$. We calculate the optimal price $b_{j^*} = 5.69$ of $\Phi(\mathbf{b})$ (Eq. (4)). Then we set $\beta = 0.12$ and $\alpha = 5$ here and the candidate price set is $X = \{5.6, 6.27, 7.02, 7.87, 8.81, 9.87\}$. Figure 5 shows the convergence speed of the candidate price selection probability obtained by PLnG. The x-axis is the serialized number of the task $t_1^k$. The x-axis value equal to $k$ means that we have evaluated the first $k$ tasks. The y-axis is the probability value of selecting a candidate price. In the beginning, the probability of selecting any candidate price is 1/6. Each time after we evaluate a task, the parameters of the algorithm are updated accordingly. As the number of evaluated tasks grows, the probability of choosing $x_1 = 5.6$ as the clearing price increases rapidly, the probability of choosing $x_2 = 6.27$ fluctuates around 0.2, and the probability of choosing other candidate prices quickly drops to near 0. In fact, with fewer than 10 samples, we have a dominant probability to select $x_1$ or its closest price $x_2$ as our evaluation price. Namely, we use $x_1$ or $x_2$ to approximate the optimal evaluation price with a high probability, and the selection probability of other prices that are significantly different from $x_1$ and $x_2$ is near 0. Note that, $x_1$ is indeed the candidate price closest to the optimal

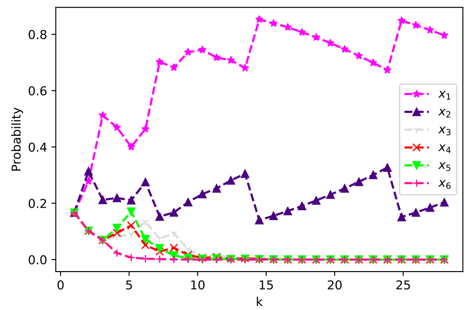
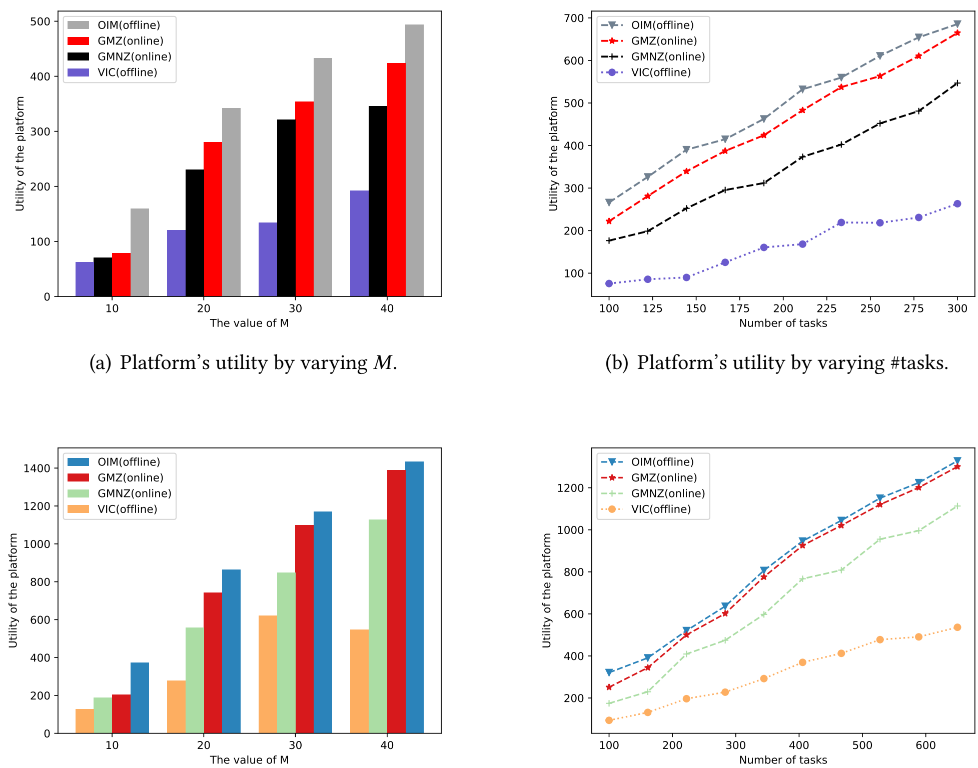



Fig. 5. (T-Drive) Convergence of the candidate price selection probability in PLnG.

price and $x_2$ is the second closest in our setting. It means that our algorithm can quickly converge to a near-optimal price with a high probability.

*6.2.4 Comparison of different mechanisms.* Figure 6(a) shows the experimental results by varying the value of $M$ on the synthetic dataset. The platform's utility of each compared mechanism increases with the increment of $M$. OIM uses the optimal single price omniscient mechanism $\Phi(\mathbf{b})$ to maximize the budget of each agent in Stage I and uses the optimal mechanism $\Phi(\mathbf{b}, \mathbf{s})$ with respect to budgets and bids in Stage II. Therefore, it always outperforms the other mechanisms. The gap between the proposed mechanisms (GMZ and GMNZ) and OIM becomes smaller with the increment of $M$, and the proposed mechanisms show advantages compared with VIC. When $M$ is fixed to 40, Figure 6(b) shows the results by varying the number of tasks on the synthetic dataset. All mechanisms obtain higher platform utility when the number of tasks increases. Again, the two proposed mechanisms GMZ and GMNZ perform much better than VIC. It can be noticed that in all cases, the performance of GMZ is better than GMNZ. This is because that GMNZ usually pays more than GMZ since users have non-zero arrival-departure intervals. We also conduct the experiments by varying the value of $M$ and the number of tasks on the T-Drive dataset. The results, as shown in Figure 6(c) and Figure 6(d), exhibit similar trends as those on the synthetic dataset. Again, GMZ and GMNZ have close performance with OIM and outperform VIC significantly.

*6.2.5 Impact of the parameters $\alpha$ and $\beta$.* Figure 7 shows the performance comparison of stage I when we vary the value of $\alpha$ and $\beta$ on the synthetic dataset. Here $M$ is fixed to 40 and there have 200 tasks in total. In Figure 7(a) and Figure 7(c), when the value of $\alpha$ increases, the agents' budgets will all increase. The reason is that, with the larger learning rate, the weight of the price $x_\sigma \in X$ with the higher profit will increase more rapidly. In other words, we have a greater probability to choose a fixed price close to the optimal price. Nevertheless, the trends in these figures also show that, the marginal increment of the algorithm becomes smaller when the learning rate is large enough. Figure 7(b) and Figure 7(d) show the impact of $\beta$ on PLnG. In Figure 7(b), we can see that a smaller interval generates a selected price closer to the optimal price, so better results can be achieved. However, in Figure 7(d) with $l = 1$, when we set $\beta$ to a small value like 0.1, the candidate price set $X$ has many unavailable prices. At this time, the task budgets are all in the range [5,10], while $X = \{1.1, 1.21, 1.331, ...\}$. When we choose candidate prices smaller than 5 or larger than 10 to evaluate these tasks, no task will be selected. Hence a smaller value of $\beta$ leads to a low budget of the agent. Similarly, if we set $\beta = 1$ and then $X = \{1, 2, 4, 8\}$, only when we select $c = 8$ can the agent get a budget. Thus in this case, the performance of PLnG is also poor. Similar results can be




(a) Platform's utility by varying $M$.

(b) Platform's utility by varying #tasks.

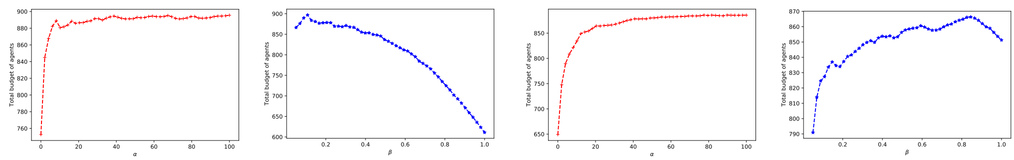
(c) Platform's utility by varying $M$.

(d) Platform's utility by varying #tasks.

Fig. 6. Comparison of different mechanisms. (a)(b): Synthetic Dataset. (c)(d): T-Drive Dataset.

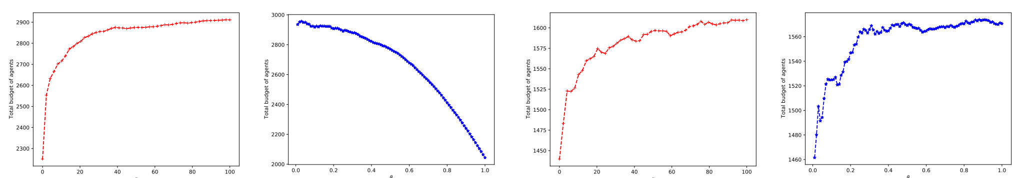
observed on the T-Drive dataset (Figure 8). Influenced by the range of task budgets $[l, h]$, different $\alpha$ and $\beta$ bring different performances. Thus, we need to determine the value of $\beta$ according to the values of $l$ and $h$ to achieve the excellent performance of the algorithm.

*6.2.6 Truthfulness of GMZ.* We first verify the truthfulness of users by randomly picking two mobile users (ID = 2 and ID = 25) and allowing them to bid prices that are different from their reserved prices on the synthetic dataset. We illustrate the results in Figure 9(a) and Figure 9(b). As can be seen, the user $u_2$ in Figure 9(a) is a winner (i.e., selected by the platform). He achieves his optimal utility if he bids truthfully ($s_i = v_i = 20.65$) and his utility will be 0 if he reports a bid larger than 24. $u_{25}$ in Figure 9(b) fails the auction and his utility is 0 originally. If he reports a bid such that $s_i \leq 11$, he will be selected while his utility will be negative. In addition, any other bid will leave his utility to 0. Figure 9(c) and Figure 9(d) show the truthfulness of task requesters. The requester of task $t_1^2$ in Figure 9(c) is a winner and he obtains the optimal utility when he reports $b_1^2 = v_1^2$. The requester of $t_3^4$ in Figure 9(d) fails the auction and his utility is 0 when he reports his valuation $v_3^4 = 5.12$. He will receive a negative utility if he reports any budget $b_3^4 \geq 12.5$. We also verify the truthfulness of users and requesters on the T-Drive dataset in Figure 10. Similarly, we have a winner $u_{45}$ with reversed price 24.8 and a loser $u_{32}$ with reversed price 51 in Figure 10(a)(b),



(a) Impact of $\alpha$. (b) Impact of $\beta$. (c) Impact of $\alpha$. (d) Impact of $\beta$.

Fig. 7. Impact of the parameters $\alpha$ and $\beta$ (Synthetic Dataset). (a)(b) $l = 5$. (c)(d) $l = 1$.



(a) Impact of $\alpha$. (b) Impact of $\beta$. (c) Impact of $\alpha$. (d) Impact of $\beta$.

Fig. 8. Impact of the parameters $\alpha$ and $\beta$ (T-Drive Dataset). (a)(b) $l = 5$. (c)(d) $l = 1$.

the winner cannot improve his utility by manipulating his bid and the loser cannot get a positive utility by changing his bid. In Figure 10(c)(d) we have two requesters $t_1^{12}$ and $t_6^{19}$. It can be seen that they both satisfy the property of truthfulness. In brief, these results verify the truthfulness of GMZ for both users and requesters.

*6.2.7 Double Truthfulness of GMNZ.* As users can strategically submit prices as well as arrival/departure time in GMNZ, we randomly pick one user $u_{11}$ for examination on the synthetic dataset. We allow him to report his bidding price, arrival time, and departure time freely. The results are shown in Figure 11. As can be seen, the user $u_{11}$ achieves his optimal utility if he reports his true information $b_i = v_i = 25.1, \tau_i = 368, d_i = 382$. Reporting any arrival time later than $\tau_i$ or departure time earlier than $d_i$ cannot improve his utility. Figure 12 shows the result on the T-Drive dataset, where the user $u_{53}$ has $v_i = 40, \tau_i = 370, d_i = 385$. Again, reporting any bidding information different from his true information cannot improve his utility.

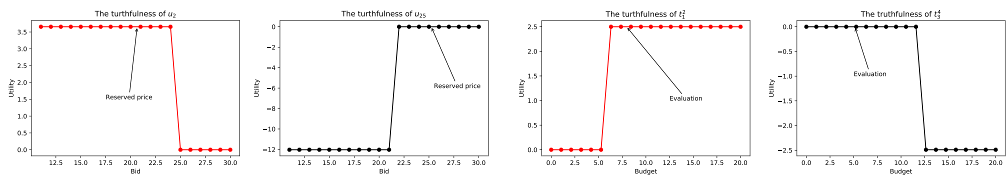
## 7 CONCLUSION

In this paper, we have proposed two group-buying-based two-stage incentive mechanisms for MCS systems. The proposed mechanisms can effectively bridge the gap between task requesters with low recruitment budgets and mobile users with relatively high working prices, and thus improve the applicability of MCS systems. Through theoretical analysis and extensive evaluation, the proposed mechanisms have been proved to have the desirable properties of computational efficiency, individual rationality, budget balance, truthfulness, and good performance.

## REFERENCES

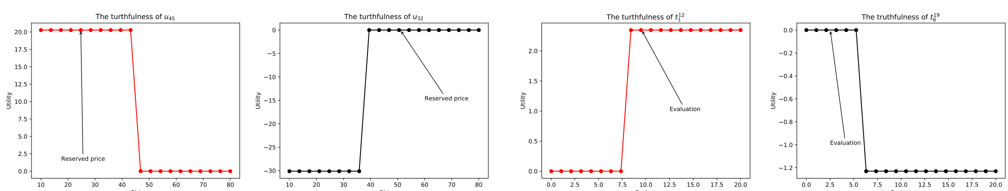

[1] Avrim Blum, Vijay Kumar, Atri Rudra, and Felix Wu. 2004. Online learning in online auctions. *Theoretical Computer Science* 324, 2-3 (2004), 137–146.

[2] Avrim Blum, Tuomas Sandholm, and Martin Zinkevich. 2006. Online algorithms for market clearing. *J. ACM* 53, 5 (2006), 845–879.



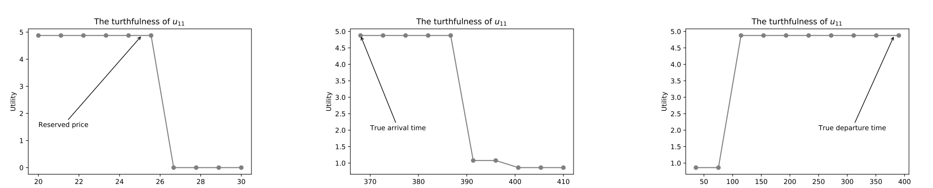

(a) Winning user. (b) Failed user. (c) Winning requester. (d) Failed requester.

Fig. 9. Truthfulness of GMZ (Synthetic Dataset). (a)(b) Truthfulness for users. (c)(d) Truthfulness for requesters.



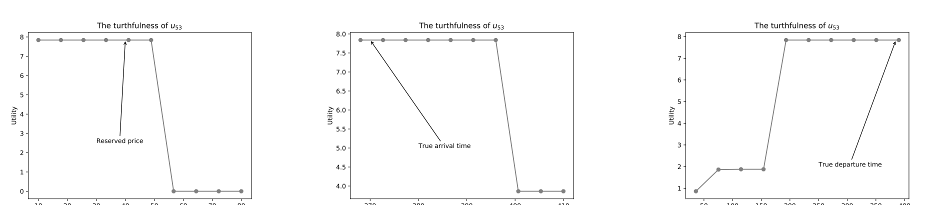

(a) Winning user. (b) Failed user. (c) Winning requester. (d) Failed requester.

Fig. 10. Truthfulness of GMZ (T-Drive Dataset). (a)(b) Truthfulness for users. (c)(d) Truthfulness for requesters.




(a) Truthfulness of bid. (b) Truthfulness of arrival time. (c) Truthfulness of departure time.

Fig. 11. Truthfulness of GMNZ (Synthetic Dataset).




(a) Truthfulness of bid. (b) Truthfulness of arrival time. (c) Truthfulness of departure time.

Fig. 12. Truthfulness of GMNZ (T-Drive Dataset).

[3] Yueyue Chen, Deke Guo, MD Zakirul Alam Bhuiyan, Ming Xu, Guojun Wang, and Pin Lv. 2019. Towards profit optimization during online participant selection in compressive mobile crowdsensing. *ACM Transactions on Sensor Networks* 15, 4 (2019), 38:1–29.

[4] Peng Cheng, Xiang Lian, Zhao Chen, Rui Fu, Lei Chen, Jinsong Han, and Jizhong Zhao. 2015. Reliable diversity-based spatial crowdsourcing by moving workers. *Proceedings of the VLDB Endowment* 8, 10 (2015), 1022–1033.

[5] Fulvio Corno, Teodoro Montanaro, Carmelo Migliore, and Pino Castrogiovanni. 2017. Smartbike: an IoT crowd sensing platform for monitoring city air pollution. *International Journal of Electrical and Computer Engineering* 7, 6 (2017), 3602–3612.

[6] Jingmei Cui, Yu-E Sun, He Huang, Hansong Guo, Yang Du, Wenjian Yang, and Meixuan Li. 2018. TCAM: a truthful combinatorial auction mechanism for crowdsourcing systems. In *Proc. of IEEE WCNC*.

[7] Kaustubh Deshmukh, Andrew V Goldberg, Jason D Hartline, and Anna R Karlin. 2002. Truthful and competitive double auctions. In *Proc. of Springer ESA*.

[8] Xiaojun Feng, Yanjiao Chen, Jin Zhang, Qian Zhang, and Bo Li. 2012. TAHES: a truthful double auction mechanism for heterogeneous spectrums. *IEEE Transactions on Wireless Communications* 11, 11 (2012), 4038–4047.

[9] Raghu K Ganti, Fan Ye, and Hui Lei. 2011. Mobile crowdsensing: current state and future challenges. *IEEE communications Magazine* 49, 11 (2011), 32–39.

[10] Andrew V Goldberg, Jason D Hartline, Anna R Karlin, Michael Saks, and Andrew Wright. 2006. Competitive auctions. *Games and Economic Behavior* 55, 2 (2006), 242–269.

[11] Bin Guo, Zhu Wang, Zhiwen Yu, Yu Wang, Neil Y Yen, Runhe Huang, and Xingshe Zhou. 2015. Mobile crowd sensing and computing: the review of an emerging human-powered sensing paradigm. *Comput. Surveys* 48, 1 (2015), 7:1–31.

[12] Tong Guo, Bin Guo, Yi Ouyang, Zhiwen Yu, Jacqueline CK Lam, and Victor OK Li. 2018. CrowdTravel: scenic spot profiling by using heterogeneous crowdsourced data. *Journal of Ambient Intelligence and Humanized Computing* 9, 6 (2018), 2051–2060.

[13] Yidan Hu and Rui Zhang. 2019. Differentially-private incentive mechanism for crowdsourced radio environment map construction. In *Proc. of IEEE INFOCOM*.

[14] Liqun Huang, Yanmin Zhu, Jiadi Yu, and Min-You Wu. 2016. Group buying based incentive mechanism for mobile crowd sensing. In *Proc. of IEEE SECON*.

[15] Yao Jing, Bin Guo, Zhu Wang, Victor OK Li, Jacqueline CK Lam, and Zhiwen Yu. 2017. CrowdTracker: optimized urban moving object tracking using mobile crowd sensing. *IEEE Internet of Things Journal* 5, 5 (2017), 3452–3463.

[16] Leyla Kazemi and Cyrus Shahabi. 2012. Geocrowd: enabling query answering with spatial crowdsourcing. In *Proc. of ACM SIGSPATIAL*.

[17] Ioannis Koukoutsidis. 2017. Estimating spatial averages of environmental parameters based on mobile crowdsensing. *ACM Transactions on Sensor Networks* 14, 1 (2017), 2:1–26.

[18] Xin Li and Xinglin Zhang. 2019. Multi-task allocation under time constraints in mobile crowdsensing. *IEEE Transactions on Mobile Computing* (2019).

[19] Jian Lin, Ming Li, Dejun Yang, and Guoliang Xue. 2018. Sybil-proof online incentive mechanisms for crowdsensing. In *Proc. of IEEE INFOCOM*.

[20] Peng Lin, Xiaojun Feng, Qian Zhang, and Mounir Hamdi. 2013. Groupon in the air: a three-stage auction framework for spectrum group-buying. In *Proc. of IEEE INFOCOM*.

[21] Xiaoqiang Teng, Deke Guo, Yulan Guo, Xiaolei Zhou, Zeliu Ding, and Zhong Liu. 2017. IONavi: an indoor-outdoor navigation service via mobile crowdsensing. *ACM Transactions on Sensor Networks* 13, 2 (2017), 1–28.

[22] Jiangtao Wang, Leye Wang, Yasha Wang, Daqing Zhang, and Linghe Kong. 2018. Task allocation in mobile crowd sensing: state-of-the-art and future opportunities. *IEEE Internet of Things Journal* 5, 5 (2018), 3747–3757.

[23] Jiangtao Wang, Yasha Wang, Daqing Zhang, Feng Wang, Haoyi Xiong, Chao Chen, Qin Lv, and Zhaopeng Qiu. 2018. Multi-task allocation in mobile crowd sensing with individual task quality assurance. *IEEE Transactions on Mobile Computing* 17, 9 (2018), 2101–2113.

[24] Xiong Wang, Jinbei Zhang, Xiaohua Tian, Xiaoying Gan, Yunfeng Guan, and Xinbing Wang. 2017. Crowdsensing-based consensus incident report for road traffic acquisition. *IEEE Transactions on Intelligent Transportation Systems* 19, 8 (2017), 2536–2547.

[25] Zhibo Wang, Jiahui Hu, Qian Wang, Ruizhao Lv, Jian Wei, Honglong Chen, and Xiaoguang Niu. 2019. Task-bundling-based incentive for location-dependent mobile crowdsourcing. *IEEE Communications Magazine* 57, 2 (2019), 54–59.

[26] Zhibo Wang, Jingxin Li, Jiahui Hu, Ju Ren, Zhetao Li, and Yanjun Li. 2019. Towards privacy-preserving incentive for mobile crowdsensing under an untrusted platform. In *Proc. of IEEE INFOCOM*.

[27] Yueming Wei, Yanmin Zhu, Hongzi Zhu, Qian Zhang, and Guangtao Xue. 2015. Truthful online double auctions for dynamic mobile crowdsourcing. In *Proc. of IEEE INFOCOM*.

[28] Yibo Wu, Yi Wang, Wenjie Hu, and Guohong Cao. 2015. Smartphoto: a resource-aware crowdsourcing approach for image sensing with smartphones. *IEEE Transactions on Mobile Computing* 15, 5 (2015), 1249–1263.

[29] Hong Xie and John CS Lui. 2016. Incentive mechanism and rating system design for crowdsourcing systems: Analysis, tradeoffs and inference. *IEEE Transactions on Services Computing* 11, 1 (2016), 90–102.

[30] Jia Xu, Chengcheng Guan, Haobo Wu, Dejun Yang, Lijie Xu, and Tao Li. 2018. Online incentive mechanism for mobile crowdsourcing based on two-tiered social crowdsourcing architecture. In *Proc. of IEEE SECON.*

[31] Dejun Yang, Guoliang Xue, and Xiang Zhang. 2016. Group buying spectrum auctions in cognitive radio networks. *IEEE Transactions on Vehicular Technology* 66, 1 (2016), 810–817.

[32] Xiangping Zhai, Tianqi Zhou, Chunsheng Zhu, Bing Chen, Weidong Fang, and Kun Zhu. 2018. Truthful double auction for joint internet of energy and profit optimization in cognitive radio networks. *IEEE Access* 6 (2018), 23180–23190.

[33] Honggang Zhang, Benyuan Liu, Hengky Susanto, Guoliang Xue, and Tong Sun. 2016. Incentive mechanism for proximity-based mobile crowd service systems. In *Proc. of IEEE INFOCOM.*

[34] Xinglin Zhang, Le Jiang, and Xiumin Wang. 2019. Incentive mechanisms for mobile crowdsensing with heterogeneous sensing costs. *IEEE Transactions on Vehicular Technology* 68, 4 (2019), 3992–4002.

[35] Xiang Zhang, Guoliang Xue, Ruozhou Yu, Dejun Yang, and Jian Tang. 2017. Robust incentive tree design for mobile crowdsensing. In *Proc. of IEEE ICDCS.*

[36] Xinglin Zhang, Zheng Yang, Yunhao Liu, Jianqiang Li, and Zhong Ming. 2017. Toward efficient mechanisms for mobile crowdsensing. *IEEE Transactions on Vehicular Technology* 66, 2 (2017), 1760–1771.

[37] Xinglin Zhang, Zheng Yang, Wei Sun, Yunhao Liu, Shaohua Tang, Kai Xing, and Xufei Mao. 2016. Incentives for mobile crowd sensing: a survey. *IEEE Communications Surveys & Tutorials* 18, 1 (2016), 54–67.

[38] Xinglin Zhang, Zheng Yang, Zimu Zhou, Haibin Cai, Lei Chen, and Xiangyang Li. 2014. Free market of crowdsourcing: Incentive mechanism design for mobile sensing. *IEEE Transactions on Parallel and Distributed Systems* 25, 12 (2014), 3190–3200.

[39] Yifan Zhang and Xinglin Zhang. 2020. BundleSense: a task-bundling-based incentive mechanism for mobile crowd sensings. In *Proc. of IEEE ICCCN.*

[40] Yifan Zhang, Xinglin Zhang, and Feng Li. 2020. BiCrowd: online bi-objective incentive mechanism for mobile crowd sensing. *IEEE Internet of Things Journal* (2020).

[41] Dong Zhao, Xiang-Yang Li, and Huadong Ma. 2014. How to crowdsource tasks truthfully without sacrificing utility: Online incentive mechanisms with budget constraint. In *Proc. of IEEE INFOCOM.*

[42] Dong Zhao, Huadong Ma, and Liang Liu. 2016. Frugal online incentive mechanisms for mobile crowd sensing. *IEEE Transactions on Vehicular Technology* 66, 4 (2016), 3319–3330.


## A PROOF OF LEMMA 4

Here we prove the performance ratio of PLnG to the optimal single priced omniscient auction determined by Eq. (4). We first prove the following lemma.

**Lemma 10.** *For any input sequence* $\mathbf{b}$ *and learning rate* $\alpha \geq e-1$, *let* $\mathbb{E}[R(\mathbf{b})]$ *denote the expected total budget aggregated from PLnG and let* $\Phi_X(\mathbf{b}) = \max_{1\leq\sigma\leq\hat{\sigma}} r_\sigma(|\mathbf{b}|)$ *denote the maximum final budget by choosing the optimal candidate price to evaluate the input sequence* $\mathbf{b}$, *then we have:*

$$\mathbb{E}[R(\mathbf{b})] \geq \frac{\Phi_X(\mathbf{b})}{\alpha} - \frac{h}{\alpha}\hat{\sigma}.$$

Proof. Let $W(k) = \sum_{\sigma=1}^{\hat{\sigma}} w_\sigma(k)$. According to PLnG, the expected profit (i.e., paid budget) of the task $t_j^{k+1}$ is:

$$\mathbb{E}[V(k+1)] = \frac{\sum_{\sigma=1}^{\hat{\sigma}} w_\sigma(k) g_\sigma(k+1)}{W(k)}.$$

Since $w_\sigma(k+1) = (1+\alpha)^{r_\sigma(k+1)/h}$ and $r_\sigma(k+1) = r_\sigma(k) + g_\sigma(k+1)$, we have:

$$\begin{aligned} W(k+1) &= \sum_{\sigma=1}^{\hat{\sigma}} w_\sigma(k)(1+\alpha)^{g_\sigma(k+1)/h} \\ &\leq \sum_{\sigma=1}^{\hat{\sigma}} w_\sigma(k)(1+\alpha(g_\sigma(k+1)/h)) \\ &= W(k) + \alpha W(k)\frac{\sum_{\sigma=1}^{\hat{\sigma}} w_\sigma(k)(g_\sigma(k+1)/h)}{W(k)} \\ &= W(k)(1+\frac{\alpha}{h}\mathbb{E}[V(k+1)]), \end{aligned}$$

where the inequality is derived from the fact that $(1+\alpha)^x \leq 1+\alpha x$ for $x \in [0,1]$ and $\alpha \geq 0$. Since $W(0) = \sum_{\sigma=1}^{\hat{\sigma}} w_\sigma(0) = \hat{\sigma}$, we have:

$$\begin{aligned} W(|\mathbf{b}|) &\leq \hat{\sigma}\prod_{k=1}^{|\mathbf{b}|}(1+\frac{\alpha}{h}\mathbb{E}[V(k)]) \\ &\leq \exp(\hat{\sigma})\prod_{k=1}^{|\mathbf{b}|}\exp(\frac{\alpha}{h}\mathbb{E}[V(k)]) \\ &= \exp(\hat{\sigma} + \frac{\alpha}{h}\sum_{k=1}^{|\mathbf{b}|}\mathbb{E}[V(k)]), \end{aligned}$$

where the second inequality is derived from the fact that $x \leq e^x$ and $1+x \leq e^x$ for all $x \geq 0$.

As $\Phi_X(\mathbf{b}) = \max_{1\leq\sigma\leq\hat{\sigma}} r_\sigma(|\mathbf{b}|)$ denotes the maximum final profit, $(1+\alpha)^{\Phi_X(\mathbf{b})/h}$ is the maximum final weight. Since the sum of the final weights is at least the value of the maximum final weight, namely $W(|\mathbf{b}|) \geq (1+\alpha)^{\Phi_X(\mathbf{b})/h}$, we have:

$$(1+\alpha)^{\Phi_X(\mathbf{b})/h} \leq \exp(\hat{\sigma} + \frac{\alpha}{h}\sum_{k=1}^{|\mathbf{b}|}\mathbb{E}[V(k)]).$$

As $1+\alpha \geq e$, we can derive from the above inequality that:

$$\Phi_X(\mathbf{b})/h \leq \hat{\sigma} + \frac{\alpha}{h}\sum_{k=1}^{|\mathbf{b}|}\mathbb{E}[V(k)].$$

Let $\mathbb{E}[R(\mathbf{b})] = \sum_{k=1}^{|\mathbf{b}|}\mathbb{E}[V(k)]$ denote the expected total budget aggregated from PLnG, then we have

$$\mathbb{E}[R(\mathbf{b})] \geq \frac{\Phi_X(\mathbf{b})}{\alpha} - \frac{h}{\alpha}\hat{\sigma}.$$

□

Now we are ready to prove Lemma 4. Given the optimal auction $\Phi(\mathbf{b})$ and the maximum final budget $\Phi_X(\mathbf{b})$ obtained by PLnG, as rounding down a price to a power of $(1+\beta)$ loses at most a factor of $(1+\beta)$ in the result, hence $\Phi(\mathbf{b}) \leq (1+\beta)\Phi_X(\mathbf{b})$. Combining Lemma 10 and setting $\alpha = 1/\epsilon, \beta = \epsilon$ ($\epsilon \in (0, 1/(e-1)]$), we have:

$$\mathbb{E}[R(\mathbf{b})] \geq \frac{\Phi_X(\mathbf{b})}{\alpha} - \frac{h}{\alpha}\hat{\sigma} \geq \frac{\Phi(\mathbf{b})}{\alpha(1+\beta)} - \frac{h}{\alpha}\hat{\sigma} = \frac{\epsilon\Phi(\mathbf{b})}{1+\epsilon} - \epsilon h\hat{\sigma}.$$

Given that $\Phi(\mathbf{b}) \geq 3h(1+\epsilon)\frac{\ln(h/l)}{\ln(1+\beta)}$, since $\hat{\sigma} = \frac{\ln(h/l)}{\ln(1+\beta)}$, we have:

$$\epsilon h\hat{\sigma} \leq \frac{\epsilon}{3(1+\epsilon)}\Phi(\mathbf{b}).$$

Hence

$$\mathbb{E}[R(\mathbf{b})] \geq \frac{\epsilon}{1+\epsilon}\Phi(\mathbf{b}) - \frac{\epsilon}{3(1+\epsilon)}\Phi(\mathbf{b}) = \frac{2\epsilon}{3(1+\epsilon)}\Phi(\mathbf{b}),$$

which completes our proof.

## B PROOF OF LEMMA 9

Given the user $u_i$ with the true information tuple $(v_i, \tau_i, d_i)$ and his reported information tuple $(s_i, \widehat{\tau_i}, \widehat{d_i})$, we first give the following two lemmas.

**Lemma 11.** *At the time step $d = \widehat{\tau_i}$, given a fixed clearing price $c$, the user $u_i$'s dominant strategy is reporting his reserved price $v_i$.*

Proof. We consider two cases. (i) $v_i \leq c$. Reporting the user's reserved price brings him a reward $r_i = c$ and $\mu_i = c - v_i \geq 0$. Reporting any bid $s_i \leq c$ will not make any difference while reporting $s_i > c$ will make him lose the auction and his utility becomes 0. (ii) $v_i > c$. The user's utility will be 0 if he reports truthfully. Reporting $s_i \leq c$ will let him be a winner while his utility $\mu_i = c - v_i < 0$. In brief, reporting the price truthfully is the dominant strategy. □

**Lemma 12.** *Given $s_i$, the other users' strategies, and each time step's clearing price $c$, the user $u_i$'s dominant strategy is reporting the true arrival/departure time $\tau_i/d_i$.*

Proof. Note that $u_i$ always obtains the maximal clearing price during $[\widehat{\tau_i}, \widehat{d_i}]$ as his reward. Assume that at the time step $d \in [\tau_i, d_i]$, the user $u_i$ obtains the maximal reward $r^*$. Then reporting any $\widehat{\tau_i} \in [\tau_i, d]$ and $\widehat{d_i} \in [d, d_i]$ cannot improve the reward. However, if $u_i$ reports $\widehat{\tau_i} > d$ or $\widehat{d_i} < d$, his reward will be less than $r^*$. In other words, reporting any untruthful arrival/departure time cannot increase the user's reward. □

According to GMNZ, if $u_i$ is not selected at the time step $d = \widehat{\tau_i}$, he will be discarded and $\mu_i$ will be 0. If selected, he will get a reward $r_i = c$, where $c$ is the clearing price at this time step. Because $u_i$ can only bid once when $d = \widehat{\tau_i}$, from Lemma 11, we know that reporting his reserved price $s_i = v_i$ is a dominant strategy. From Lemma 12, we know that reporting truthful arrival/departure time is a dominant strategy for $u_i$. In summary, the user cannot improve his utility by misreporting his bidding information.