

Incentive Mechanism for Uncertain Tasks under Differential Privacy

Xikun Jiang, Chenhao Ying, Lei Li, Boris Düdder, Haiqin Wu, Haiming Jin, Yuan Luo

Abstract—Mobile crowd sensing (MCS) has emerged as an increasingly popular sensing paradigm due to its cost-effectiveness. This approach relies on platforms to outsource tasks to participating workers when prompted by task publishers. Although incentive mechanisms have been devised to foster widespread participation in MCS, most of them focus only on static tasks (i.e., tasks for which the timing and type are known in advance) and do not protect the privacy of worker bids. In a dynamic and resource-constrained environment, tasks are often uncertain (i.e., the platform lacks *a priori* knowledge about the tasks) and worker bids may be vulnerable to inference attacks. This paper presents an incentive mechanism HERALD*, that takes into account the uncertainty and hidden bids of tasks without real-time constraints. Theoretical analysis reveals that HERALD* satisfies a range of critical criteria, including truthfulness, individual rationality, differential privacy, low computational complexity, and low social cost. These properties are then corroborated through a series of evaluations.

Index Terms—Uncertain Tasks without Real-time Constraints, Differential Privacy, Mobile Crowd Sensing, Incentive Mechanism.

I. INTRODUCTION

THE proliferation of mobile devices equipped with advanced processors and numerous sensors (like GPS and microphones), has been a key driver behind the rise of mobile crowd sensing (MCS) as a prominent sensing paradigm. MCS relies on a pool of workers equipped with mobile devices to gather sensory data and has fueled the development of various applications, including smart transportation, traffic management, and IoT. Numerous MCS systems have been developed and implemented accordingly in these and other areas [1]–[8].

Fig. 1 provides an illustration of a typical MCS system, in which platforms enlist participating workers to execute

Xikun Jiang, Chenhao Ying, Haiming Jin and Yuan Luo are with the Department of Computer Science, Shanghai Jiao Tong University, Shanghai, 200240, China. Xikun Jiang is also with the Department of Computer Science, University of Copenhagen, Copenhagen, 2100, Denmark. Chenhao Ying and Yuan Luo are also with Shanghai Jiao Tong University (Wuxi) Blockchain Advanced Research Center, Jiangsu, 214101, China. (E-mails: {xikunjiang, yingchenhao, jinhaiming, yuanluo}@sjtu.edu.cn, xikun@di.ku.dk.)

Lei Li and Boris Düdder are with the Department of Computer Science, University of Copenhagen, Copenhagen, 2100, Denmark. (E-mail: {lilei, boris.d}@di.ku.dk.)

Haiqin Wu is with the Department of Software Engineering Institute, East China Normal University, Shanghai, 200062, China. (E-mail: hqwu@sei.ecnu.edu.cn.)

This work is supported in part by the Shanghai Science and Technology Innovation Action Plan 23511100400, in part by the 2023-2024 Open Project of Key Laboratory Ministry of Industry and Information Technology-Blockchain Technology and Data Security 20242216, in part by the National Natural Science Foundation of China (NSFC) under Grants 62372288, and U20A20181.

Corresponding author: Chenhao Ying.

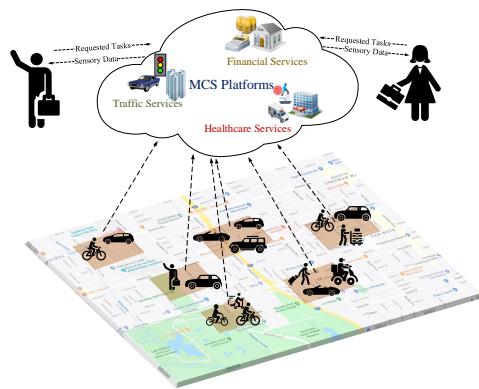


Fig. 1. A typical MCS system.

tasks upon request from demanders (also referred to as task publishers). The success of the majority of MCS applications hinges on the willingness of an adequate number of mobile workers to partake in the process, thereby ensuring the provision of high-quality services. However, workers may exhibit hesitance towards participating in MCS, since task execution may drain their battery power, storage, computation, and communication resources. Furthermore, participating in MCS may render workers' private information (including their location and bid details) vulnerable to exposure during data collection and exchange. To counterbalance these costs and safeguard their privacy, it is crucial to furnish workers with suitable incentives that do not jeopardize their confidentiality.

The importance of incentives has led to the development of numerous mechanisms [9]–[27] to encourage participation in MCS. However, a considerable number of these mechanisms are grounded on the presumption that tasks are static (i.e., the platform knows the timing and type of tasks in advance). In the real world, MCS tasks are often uncertain due to their unknown arrival time and incomplete information known to the platform. Additionally, these mechanisms do not protect the privacy of worker bids, which may be exposed to potential inference attacks [28] if published by the platform. The term “bid” represents the amount a worker (participant) is willing to accept as compensation for undertaking a task in the mobile crowd-sensing platform. Therefore, there is a desperate need for an incentive mechanism that can handle uncertain tasks and protect worker bid privacy in MCS systems, meanwhile satisfying a set of desired properties such as truthfulness, individual rationality, differential privacy, and low social cost.

In an MCS system, tasks such as collecting information on the number of bends, bifurcations, or roadside shops can be completed in advance, which reduces latency and improves

efficiency by allowing the platform to respond immediately when these tasks arrive. These tasks are referred to as non-real-time tasks. However, due to the complexity of the real-world environment, it is often difficult for the platform to predict which tasks will arrive in the future and when they will arrive. These unpredictable tasks are referred to as *uncertain tasks*.

Designing a suitable incentive mechanism for MCS systems, which are susceptible to both uncertain tasks without real-time constraints and inference attacks, poses a considerable challenge. To tackle this issue, we consider a scenario where the arrival of uncertain tasks without real-time constraints is governed by a probability distribution and employ the exponential mechanism, a technique from differential privacy, to safeguard the privacy of worker bids. Our proposed solution, HERALD*¹, which satisfies truthfulness, individual rationality, and differential privacy, while also having low computational complexity and social cost. The main contributions of this paper are:

- *Mechanism:* In contrast to previous approaches [9]–[27], our work introduces a new incentive mechanism named HERALD*, which integrates differential privacy. Specifically, HERALD* is tailored for uncertain tasks without real-time constraints that are expected to arrive based on a probability distribution (in our study, we consider the uniform distribution), enabling the platform to gather sensory data in advance. Additionally, we employ the exponential mechanism to protect the privacy of worker bids.
- *Desirable Properties:* HERALD* can effectively encourage worker participation and achieves a set of desirable properties, such as truthfulness, individual rationality, differential privacy, low computational complexity, and low social cost. Unlike other incentive mechanisms such as those proposed in [9]–[13] for traditional MCS, HERALD* is specifically designed for uncertain tasks without real-time constraints and is not limited to collecting large amounts of sensory data. Furthermore, we demonstrate that HERALD* has a competitive ratio of $\mathcal{O}(\ln \ln l)$ in terms of expected social cost, where l and n represent the number of task subsets and tasks published in advance, respectively. We also prove that HERALD* preserves $\frac{\epsilon l}{2}$ -differential privacy for both linear and logarithmic score functions, where $\epsilon > 0$ is a constant.
- *Evaluations:* In addition to HERALD*'s desirable properties, we performed comprehensive simulations to verify its efficacy. Our findings indicate that HERALD* outperforms current approaches by exhibiting a lower anticipated social cost and total payment while simultaneously providing differential privacy.

The subsequent sections of this paper are structured as follows: Section II provides a discussion of the related literature, while Section III presents an introduction to the preliminaries. Section IV outlines the design and theoretical analysis of HERALD*. In Section V, extensive simulations are conducted

¹The name HERALD* is from incentive meCHanism for uncERtain tasks without real-time constraints under differential privacy in mobile crowd sensing

to validate the properties of the proposed mechanism. Lastly, Section VI concludes the paper.

II. RELATED WORK

Numerous incentive mechanisms [9]–[27] have been proposed for MCS systems since attracting a considerable number of workers is crucial. In addition to truthfulness and individual rationality, these mechanisms often aim to ensure the benefits of both workers and the platform.

The authors of [9]–[11] proposed mechanisms to minimize the social cost, while [12], [13] aimed to maximize the platform's profit. In contrast, [14]–[17] focused on minimizing the platform's payment, and [20], [21] designed mechanisms to maximize social welfare. A novel crowdsourcing assignment strategy proposed in [22] considered fair task allocation for workers. In addition to the above objectives, there have been efforts to achieve other objectives as well. For instance, Bhattacharjee *et al.* in [23] incentivized workers to act honestly by evaluating the quantity and quality of their data. Gong *et al.* in [24] introduced an incentive mechanism to encourage workers to submit high-quality data, while Liu *et al.* in [25] addressed the problem of multi-resource allocation by devising a truthful double auction mechanism.

Moreover, several studies investigated privacy-preserving methods in mobile crowdsourcing. For example, Lin *et al.* [29] proposed a general privacy-preserving framework for incentivizing crowdsensing using two score functions. Hu *et al.* [26] developed a privacy-preserving incentive mechanism for dynamic spectrum sharing crowdsensing, while Han *et al.* [27] focused on privacy-preserving in budget-limited crowdsensing. Yang *et al.* [30] incorporated both additive secret sharing and local differential privacy technologies. Wei *et al.* [31] investigated location privacy-preserving in spatial crowdsourcing. Some studies have also used cryptography techniques [32], [33], or blockchain [18], [19], [34] to protect privacy in mobile crowdsourcing, but they do not consider the strategic behavior of the participants.

Most previously mentioned works have focused on static sensing tasks, where the platform has complete knowledge of the task information beforehand. However, in realistic environments with resource constraints, sensing tasks are often uncertain, meaning that the platform has incomplete knowledge of the task information. In contrast to these previous works, this paper addresses the issue of uncertain tasks while also ensuring the bidding privacy of participants.

III. PRELIMINARIES

In this section, we provide an overview of the system and discuss the objectives that guided its design.

A. System Overview

In this study, we focus on a mobile crowdsourcing system (MCS) consisting of a cloud-based platform and a group of participating workers (represented as $\mathcal{W} = \{1, 2, \dots, m\}$). The platform is assumed to have a priori knowledge of a set of n sensing tasks, $\mathcal{T} = \{\tau_1, \dots, \tau_n\}$, and all future requested tasks will belong to \mathcal{T} . Note that index n is a

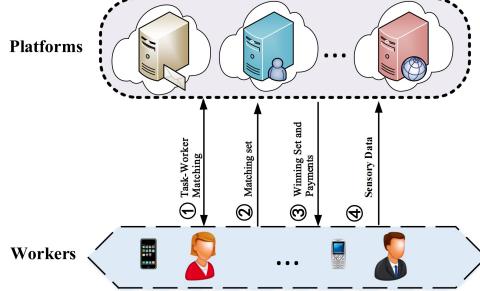


Fig. 2. Framework of HERALD*.

variable. The platform divides these n tasks into l task subsets, $\mathcal{Y} = \{\Gamma_1, \Gamma_2, \dots, \Gamma_l\}$, such that $\bigcup_{j=1}^l \Gamma_j = \mathcal{T}$. The constraint $l < mn$ is set to ensure that the count of potential task subsets is bounded by the collective interactions between the workers and tasks. Additionally, the number of tasks within each subset Γ_j is randomized. To ensure task quality and prevent monopolization, we require that each task $\tau_i \in \mathcal{T}$ must be covered by no less than two task subsets. This condition is practical since the platform typically possesses some prior knowledge about the tasks that need to be performed. For instance, in a traffic monitoring application, the task set may comprise the number of all the intersections on a particular road. This task set remains static over time and has no real-time constraints. A similar application could be the monitoring of road curves. The system framework is depicted in Fig. 2, and its operation is elaborated below.

Incentive Mechanism for Uncertain Tasks without Real-time Constraints: As illustrated in Fig. 2, before the actually requested tasks arrive, the platform first matches workers to task subsets in a task-worker matching phase (step ①). For each task subset $\Gamma_j \in \mathcal{Y}$, each worker i submits a bid b_i . We assume that the probability that task subset Γ_j is matched with worker i is $Pr(b_i)$. Note that in each round of task-worker matching, each worker can only participate once, and in the l rounds of matching, each worker i only has one unique bid b_i . This means that for each task subset Γ_j , the probability that worker i is matched is $Pr(b_i)$. The platform randomly assigns the task subset to the workers according to this probability distribution. To ensure task quality and prevent monopolization, we require that each task $\tau_i \in \mathcal{T}$ is matched by at least two different workers. After the task-worker matching phase, the platform receives the matching set \mathcal{P} , which consists of matching pairs (Γ_j, b_i) for $\Gamma_j \in \mathcal{Y}$ and $i \in \mathcal{W}$, where b_i is worker i 's bid for executing task subset Γ_j (step ②). Upon receiving a matching set \mathcal{P} , the platform proceeds to determine the winning set \mathcal{S} and the corresponding payment p_i to each winning worker i . Among them, the winning set \mathcal{S} consists of winning matching pairs, and the workers contained in the winning matching pairs are selected to perform tasks based on their bids and matching task sets. As indicated by step ③. Additionally, the platform collects sensory data from the winning workers to enable prompt responses to future requests, as per step ④. It is worth noting that the platform collects sensory data for tasks in \mathcal{T} in advance, as it lacks any knowledge about future tasks, which are assumed to arrive according to a probability distribution.

In this mechanism, a worker who is not selected to execute any tasks, as a “loser”, receives zero payment. We use the notation $\vec{p} = (p_1, \dots, p_m)$ to represent the payment profile of the workers, which is initialized to be zero. If we denote the cost of worker i as c_i in HERALD*, we can define the utility of the worker as

$$u_i = \begin{cases} p_i - c_i & , \text{if worker } i \text{ wins,} \\ 0 & , \text{otherwise.} \end{cases} \quad (1)$$

Where the cost refers to the anticipated expenditure workers estimate before starting a task, including time commitment, battery usage, storage needs, computing demands, and communication resources [41]. These factors are crucial considerations for workers in deciding to participate in the MCS ecosystem. Therefore, the platform must provide compensation that adequately offsets these incurred costs, ensuring workers receive non-negative benefits. Without loss of generality, we assume that the bid b_i of each worker i is constrained within the range of $[b_{min}, b_{max}]$, where b_{min} is normalized to 1 and b_{max} is a constant. We use the notation $\vec{b} = (b_1, \dots, b_m)$ to represent the bid profile of the workers for convenience. Additionally, it is assumed that for each worker i who has a specific subset of tasks, denoted as Γ_i , there exist other workers j who have task subsets Γ_j that include the tasks of worker i , i.e., $\Gamma_i \subseteq \bigcup_j \Gamma_j$.

B. Design Objectives

The main objective of this manuscript is to establish that HERALD* exhibits the subsequent advantageous features. Given that workers may exhibit selfish or strategic behavior, it is plausible that any worker i may present a bid b_i that differs from their actual cost c_i of performing all the tasks in Γ_i . Consequently, we aim to develop an incentive mechanism that satisfies truthfulness, which is defined as follows.

Definition 1 (Truthfulness). *An incentive mechanism is truthful if for any worker $i \in \mathcal{W}$, his/her utility is maximized when bidding his/her true cost c_i .*

According to Definition 1, our goal is to ensure that workers make truthful bids to the platform. In addition to truthfulness, we also strive to achieve another desirable property known as individual rationality, which is defined below:

Definition 2 (Individual Rationality). *An incentive mechanism is individually rational if for any worker $i \in \mathcal{W}$, his/her utility u_i satisfies $u_i \geq 0$.*

Nonetheless, if the platform discloses the bidding results that contain the winning bidders and their rewards directly, it may expose the privacy of the participants to inference attacks, thereby impeding the advantages of the platform. Since any changes in workers' bids can considerably influence the ultimate bidding results, particularly the payments, an adversarial worker may deduce the bids of other workers based on the distinct payments received in various auctions.

Adversary Model Analysis: The driving force behind the implementation of robust privacy protection mechanisms within HERALD* is the imperative to counter the looming

threat of malicious worker inference attacks. These attacks possess the potential to inflict substantial disruption upon the operation and integrity of the Mobile Crowd Sensing (MCS) platform. Malicious workers can exploit the bids of fellow participants, thereby gaining insights into their strategies and subsequently employing strategic bid manipulations. If left unaddressed, these manipulations can wreak havoc on the core operations of the platform. Within the MCS framework, the bids submitted by workers play a pivotal role in the decisive winning selection phase. Malicious workers, through the manipulation of bids, can exert undue influence on the outcomes of this phase, thereby imperiling fairness and introducing imbalances into the selection process. The repercussions of malicious worker inference attacks extend even further, reaching into the critical domain of payment determination. Bid manipulation, if unchecked, can culminate in the unjust distribution of compensation, directly impacting the interests of both the platform and its users. This misalignment of incentives, in turn, has the potential to lead to a reduction in participation and the erosion of trust within the MCS ecosystem.

To protect against this inference attack and ensure the privacy of the workers' bids, it is necessary to devise a mechanism that fulfills differential privacy, which is defined as follows.

Definition 3 (Differential Privacy [35]). *A randomized mechanism M preserves (ϵ, δ) -differential privacy if for any two input sets A and B with a single input difference, and for any set of outputs $\mathcal{O} \in \text{Range}(M)$,*

$$\Pr[M(A) \in \mathcal{O}] \leq \exp(\epsilon) \times \Pr[M(B) \in \mathcal{O}] + \delta. \quad (2)$$

If $\delta = 0$, we say that M preserves ϵ -differential privacy.

Let $\Pr(b_i)$ represent the probability that worker i is matched with any task subset $\Gamma_j \in \mathcal{Y}$ when their bid is b_i . To safeguard the privacy of the workers' bids, it is necessary to decrease the impact of the distinct bids of workers on the eventual bidding outcome. To do this, we introduce differential privacy, specifically using the exponential mechanism which is defined as follows.

Definition 4 (The Exponential Mechanism [35]). *The exponential mechanism $\mathcal{M}_E(x, u, R)$ selects and outputs an element $r \in R$ with probability proportional to $\exp(\frac{\epsilon u(x, r)}{2\Delta u})$, i.e., $\Pr[\mathcal{M}_E(x, u, R) = r] \propto \exp(\frac{\epsilon u(x, r)}{2\Delta u})$, where x is the input set and u is a utility function that maps input/output pairs to utility scores, $\Delta u = \max_{r \in R} \max_{x, y: ||x - y||_1 \leq 1} |u(x, r) - u(y, r)|$ is the sensitivity of the utility function u , and ϵ is a small constant.*

We have integrated the above exponential mechanism into HERALD* to safeguard the input, which consists of the task set and bid, while the output is the worker-task matching result. This ensures that changes in the input do not alter the output, preserving the differential privacy of bids. This method effectively shields the bids from being identified by adversaries, particularly during critical phases such as winner selection and payment determination, which rely on these protected bids. The following theorem is derived from the exponential mechanism mentioned above.

Theorem 1 ([35]). *The exponential mechanism $\mathcal{M}_E(x, u, R)$ preserves $(\epsilon, 0)$ -differential privacy.*

To ensure that changes in workers' bids do not significantly affect the final bidding results, the exponential mechanism is implemented to achieve differential privacy in HERALD*. This mechanism guarantees that any changes in the input data remain hidden, thus safeguarding the privacy of individual bids. Moreover, this protection extends seamlessly to subsequent phases, as these phases are entirely based on the protected bids. This makes it difficult for malicious workers to infer the bidding details of other workers based on the final bidding results. Similar to previous work [26], [27], [29], [31], we incorporate randomization into the incentive mechanism's outcome to achieve differential privacy.

Apart from the above objectives, we also strive for HERALD* to possess a low anticipated social cost, taking into account the probability distribution of the tasks in \mathcal{T} arrive. We assess the competitive ratio of the system's expected social cost as a measure to accomplish this aim.

Definition 5 (Competitive Ratio on Expected Social Cost). *Suppose the tasks in the sensing task set \mathcal{T} arrive in a probability distribution, for any set \mathcal{A} of k tasks that may arrive at the same time from \mathcal{T} , let $\mathcal{S}(\mathcal{A}, W)$ refer to the winning set chosen by the mechanism such that $\mathcal{A} \subseteq \cup_{(\Gamma_j, c_j) \in \mathcal{S}(\mathcal{A}, W)} \Gamma_j$ and $\Gamma_j \cap \mathcal{A} \neq \emptyset$ for $\forall i \in \mathcal{S}(\mathcal{A}, W)$, $C(\mathcal{S}(\mathcal{A}, W)) = \sum_{(\Gamma_j, c_j) \in \mathcal{S}(\mathcal{A}, W)} c_j$ be the corresponding **social cost**, and $C_{OPT}(\mathcal{A}, W)$ be the minimum social cost of a requested task set \mathcal{A} , respectively. The competitive ratio on expected social cost is defined as $\max_k \mathbb{E}_{\mathcal{P} \in \mathcal{F}} [\mathbb{E}_{\mathcal{A} \subseteq \mathcal{T}} [C(\mathcal{S}(\mathcal{A}, W))] / \mathbb{E}_{\mathcal{A} \subseteq \mathcal{T}} [C_{OPT}(\mathcal{A}, W)]]$, where $\mathbb{E}_{\mathcal{P} \in \mathcal{F}} [\cdot]$ is the expectation over all matching set results \mathcal{F} and $\mathbb{E}_{\mathcal{A} \subseteq \mathcal{T}} [\cdot]$ is the expectation over all sets of possibly k arriving tasks in the future.*

It is worth noting that a worker may be included multiple times in the winning set, having the same cost but with a different subset of tasks since each worker can match with several subsets of tasks. Additionally, it is possible for some tasks in the task set \mathcal{A} to be identical. Thus, when referring to $\mathcal{A} \subseteq \mathcal{T}$ in the evaluation of the competitive ratio, we imply that every task in \mathcal{A} also belongs to \mathcal{T} , as the tasks in \mathcal{T} are distinct. Moreover, the expectation $\mathbb{E}_{\mathcal{A} \subseteq \mathcal{T}} [\cdot]$ is determined by the variability of the set \mathcal{A} of k requested tasks. For convenience of notation, the subscript $\mathcal{A} \subseteq \mathcal{T}$ is omitted in the rest of this paper, and this expectation is expressed as $\mathbb{E}[\cdot]$.

Ultimately, we strive for HERALD* to be computationally efficient, and we define this objective as follows.

Definition 6. *An incentive mechanism is computationally efficient if it can be executed within polynomial time.*

In essence, our aims are to guarantee that the proposed mechanism is both truthful and individually rational, as well as being differentially private, while also having a small social cost and computational complexity.

IV. INCENTIVE MECHANISM FOR UNCERTAIN TASKS WITHOUT REAL-TIME CONSTRAINTS

In this section, we introduce an incentive mechanism designed for uncertain tasks without real-time constraints, and we demonstrate that our mechanism satisfies the properties of truthfulness, individual rationality, and differential privacy. Additionally, we examine the competitive ratios on the expected social cost of the mechanism, as stated in Theorem 5. Furthermore, we evaluate the computational complexity of the mechanism, as presented in Proposition 1, in addition to the aforementioned properties.

A. Design Rationale

The design of an incentive mechanism must take into account the possibility of tasks arriving simultaneously, as this can affect the mechanism's construction. In the offline scenario, all tasks arrive simultaneously, and the platform has prior knowledge of all tasks. However, in the case of uncertain tasks, the number of tasks arriving simultaneously is also uncertain, and the probability distribution of the number of arriving tasks is different. To clarify this concept, consider the following straightforward example.

Example 1. *In this example, the platform possesses a sensing task set $\mathcal{T} = \{\tau_1, \tau_2, \tau_3\}$ containing three tasks, each of which has a probability of $\frac{1}{3}$ to arrive in the future. As a result, the task arrivals follow a uniform distribution. If a single task is set to arrive in the future, it could be τ_1 , τ_2 or τ_3 with the same probability $\frac{1}{3}$. While, if two tasks arrive simultaneously in the future, they may be $\{\tau_1, \tau_1\}$, $\{\tau_2, \tau_2\}$ and $\{\tau_3, \tau_3\}$ with the same probability $\frac{1}{9}$, and may be $\{\tau_1, \tau_2\}$, $\{\tau_1, \tau_3\}$ and $\{\tau_2, \tau_3\}$ with the same probability $\frac{2}{9}$. Furthermore, if three tasks simultaneously arrive in the future, they may be $\{\tau_1, \tau_2, \tau_3\}$ with probability $\frac{2}{9}$; $\{\tau_1, \tau_1, \tau_1\}$, $\{\tau_2, \tau_2, \tau_2\}$ and $\{\tau_3, \tau_3, \tau_3\}$ with the same probability $\frac{1}{27}$; and may be $\{\tau_1, \tau_2, \tau_2\}$, $\{\tau_1, \tau_3, \tau_3\}$, $\{\tau_1, \tau_1, \tau_2\}$, $\{\tau_1, \tau_1, \tau_3\}$, $\{\tau_2, \tau_2, \tau_3\}$, and $\{\tau_2, \tau_3, \tau_3\}$ with the same probability $\frac{1}{9}$.*

As demonstrated in the example, our approach differs from existing works. Rather than assuming a fixed number of tasks arriving simultaneously, we propose HERALD*, an adaptive incentive mechanism that adjusts to varying numbers of tasks as they arrive. The first phase of HERALD* involves matching workers to all possible task subsets, before the actual task requests are received. Subsequently, an estimated number of tasks arriving simultaneously is inputted into HERALD*. Based on the different input numbers, HERALD* will produce varying outcomes for both winning selection and payment determination.

B. Design Details

Within this section, we shall furnish an in-depth account of the operational mechanics of HERALD*. This mechanism consists of three phases: the task-worker matching phase (Alg. 1), the winning selection phase (Alg. 2), and the payment determination phase (Alg. 3).

Task-Worker Matching Phase: Before the real requested tasks arrive, the platform first matches workers for all task

Algorithm 1 HERALD*: Task-Worker Matching

Input: The task set \mathcal{T} , the set of task subsets $\mathcal{Y} = \{\Gamma_1, \Gamma_2, \dots, \Gamma_l\}$ such that $\bigcup_{j=1}^l \Gamma_j = \mathcal{T}$, worker set \mathcal{W} , each worker's bid, a score function u and its sensitivity Δu , a small constant ϵ .

Output: Matching set \mathcal{P} .

- 1: $\mathcal{P} \leftarrow \emptyset$
- 2: $S = \sum_{i \in \mathcal{W}} \exp(\frac{\epsilon u(b_i)}{2\Delta u})$
- 3: **for** each worker $i \in \mathcal{W}$ **do**
- 4: Calculate the probability $Pr(b_i)$ that worker i with his/her unique bid b_i is matched with each task subset $\Gamma_j \in \mathcal{Y}$: $Pr(b_i) = \frac{\exp(\frac{\epsilon u(b_i)}{2\Delta u})}{S}$
- 5: **end for**
- 6: **for** each task subset $\Gamma_j \in \mathcal{Y}$ **do**
- 7: Select the worker i randomly according to the computed probability distribution to match the task subset Γ_j
- 8: $\mathcal{P} \cup \{(\Gamma_j, b_i)\}$
- 9: **end for**
- 10: **return** The matching set \mathcal{P} of the task subset Γ_j and its matched worker i 's bid.

subsets. We refer to this as the task-worker matching phase. As can be seen in Alg. 1, for each task subset $\Gamma_j \in \mathcal{Y}$, the probability that worker i with the unique bid b_i is matched is $Pr(b_i)$ which is calculated in Lines 2-4. In particular, as we introduced in definition 4, we employ the exponential mechanism to achieve the differential privacy of the bidding results. Thus we set $Pr(b_i) \propto \exp(\frac{\epsilon u(b_i, r)}{2\Delta u})$, where $r \in R$ is the output and u is a score function that maps input/output pairs to utility scores, $\Delta u = \max_{r \in R} \max_{x, y: \|x - y\|_1 \leq 1} |u(x, r) - u(y, r)|$ is the sensitivity of the utility function u , and ϵ is a small constant. For each round of matching shown in Lines 5-8, the platform eventually assigns the task subset to the worker randomly according to the computed probability distribution. To ensure task quality and prevent monopolization, we require that each task $\tau_i \in \mathcal{T}$ is matched by at least two different workers.

Upon completion of the task-worker matching phase, the platform proceeds to identify the winning set \mathcal{S} and the corresponding payment p_i for each winning worker i based on the submitted matching set \mathcal{P} . We refer to these as the winning selection phase and payment determination phases, which are illustrated in Alg. 2 and Alg. 3, respectively.

Winning Selection Phase: As can be seen in Alg. 2, in order to obtain the sensory data of the uncertain tasks, we define a *selection threshold* (ST) $T \geq 0$ (Line 2) that remains unchanged throughout the implementation process of HERALD*. There are two distinct forms of selection employed in HERALD* during each iteration: *Type I Selection* and *Type II Selection*, which are chosen by cost-effectiveness (CF) (Lines 4-6) defined as follows.

Definition 7 (Cost-effectiveness). *Let \mathcal{T} be the set of tasks whose sensory data is not collected at the beginning of an iteration in HERALD*. For each matching pair (Γ_j, b_i) , the CF of a worker i during each iteration is defined as $\frac{b_i}{|\Gamma_j \cap \mathcal{T}|}$.*

Algorithm 2 HERALD*: Winning Selection

Input: The task set \mathcal{T} , matching set \mathcal{P} , each worker's bid, the number k of tasks arriving simultaneously.

Output: The winning set \mathcal{S} .

- 1: $\mathcal{S} \leftarrow \emptyset$
- 2: Calculate the selection threshold $T = 64\mathbb{E}[C_{OPT}(\mathcal{A}, W)]$, where \mathcal{A} is the set of k possibly simultaneously arriving tasks from the sensing task set \mathcal{T}
- 3: **while** $\mathcal{T} \neq \emptyset$ **do**
- 4: **for** each matching pair $(\Gamma_j, b_i) \in \mathcal{P}$ **do**
- 5: Calculate the cost effectiveness (CF) $\frac{b_i}{|\Gamma_j \cap \mathcal{T}|}$, where b_i is the bid of worker i for executing the task subset Γ_j
- 6: **end for**
- 7: *Type I Selection:*
- 8: **if** $\exists (\Gamma_j, b_i) \in \mathcal{P}$, s.t $\frac{b_i}{|\Gamma_j \cap \mathcal{T}|} \leq \frac{T}{|\mathcal{T}|}$ **then**
- 9: Among the task subsets whose CFs are less than $\frac{T}{|\mathcal{T}|}$, the worker $i \in \mathcal{W}$ that matches the task subset Γ_j with the minimum CF value is selected as the winner
- 10: *Type II Selection:*
- 11: **else**
- 12: Among the workers matched by all task subsets, the worker $i \in \mathcal{W}$ whose bid is the lowest and whose task subset Γ_j she matched contains at least one undiscovered task is selected as the winner
- 13: **end if**
- 14: $\mathcal{S} \leftarrow \mathcal{S} \cup \{(\Gamma_j, b_i)\}$
- 15: $\mathcal{P} \leftarrow \mathcal{P} \setminus \{(\Gamma_j, b_i)\}$
- 16: $\mathcal{T} \leftarrow \mathcal{T} \setminus \Gamma_j$
- 17: **end while**
- 18: **return** \mathcal{S} .

Algorithm 3 HERALD*: Payment Determination

Input: The worker set \mathcal{W} , winning set \mathcal{S} , each worker's bid.

Output: The payment \vec{p} .

- 1: **for** each winning pair $(\Gamma_j, b_i) \in \mathcal{S}$ **do**
- 2: Define a *copy set* $\mathcal{T}_j \leftarrow \Gamma_j$
- 3: Build a *covering set* $\mathcal{W}_i = \{\ell | \forall \ell \in \mathcal{W} \setminus \{i\}, \Gamma_\ell \cap \mathcal{T}_j \neq \emptyset\}$
- 4: Define a *replaced set* $\mathcal{R}_i \leftarrow \emptyset$
- 5: **while** $\mathcal{T}_j \neq \emptyset$ **do**
- 6: Choose a worker $\ell \in \mathcal{W}_i$ whose matched task subset has the minimum CF
- 7: $\mathcal{R}_i \leftarrow \mathcal{R}_i \cup \{\ell\}$
- 8: $\mathcal{T}_j \leftarrow \mathcal{T}_j \setminus \Gamma_\ell$
- 9: **end while**
- 10: $p_i = p_i + \max\{b_i, p_{\mathcal{R}_i}\}$ for $p_{\mathcal{R}_i} = \sum_{\ell \in \mathcal{R}_i} b_\ell$
- 11: **end for**
- 12: **return** \vec{p} .

- *Type I Selection* (Lines 7-8): If there exist any task subsets with cost-effectiveness (CF) values equal to or lower than $\frac{T}{|\mathcal{T}|}$, the platform will choose the worker who matches the task subset with the lowest CF as the winner.
- *Type II Selection* (Lines 9-10): If the CFs of all task subsets exceed $\frac{T}{|\mathcal{T}|}$, the platform will choose a winning

worker from among those matched to all task subsets. This worker must possess the lowest bid and have a task subset that includes at least one unassigned task in \mathcal{T} .

Subsequently, the matching pair that was selected as per the aforementioned criteria are incorporated into the winning set \mathcal{S} (Line 12) and deleted from the matching set \mathcal{P} (Line 13), and the corresponding tasks are also deleted from the task set \mathcal{T} (Line 14). The platform then proceeds to calculate the payment for each winner based on the members of the winning set.

Payment Determination Phase: As shown in Alg. 3, for each winning pair $(\Gamma_j, b_i) \in \mathcal{S}$, the platform defines a *copy set* $\mathcal{T}_j = \Gamma_j$ and builds a *covering set* $\mathcal{W}_i = \{\ell | \forall \ell \in \mathcal{W} \setminus \{i\}, \Gamma_\ell \cap \mathcal{T}_j \neq \emptyset\}$, see Line 2 and Line 3 separately. This means, the intersection of the task set matched by workers in *covering set* \mathcal{W}_i and the task set matched by worker i is not 0. It then derives a *replaced set* denoted as \mathcal{R}_i consisting of workers in \mathcal{W}_i with the least CFs in each iteration such that $\Gamma_j \subseteq \cup_{\ell \in \mathcal{R}_i} \Gamma_\ell$. As shown in Lines 4-9, worker i 's task set Γ_j can be replaced by workers in the *replaced set* \mathcal{R}_i . Since each worker can match multiple task sets at the same time to form multiple matching pairs, among the multiple matching pairs composed of worker i , more than one matching pair may become the winning pair, so the total reward of worker i is equal to the cumulative sum of the rewards of all winning pairs he formed. For each winning pair, worker i 's reward is the maximum value between b_i and the total bids of workers in the *replaced set* \mathcal{R}_i . Therefore, as shown in Line 10, the payment to winner i is $p_i = p_i + \max\{b_i, p_{\mathcal{R}_i}\}$, where $p_{\mathcal{R}_i} = \sum_{\ell \in \mathcal{R}_i} b_\ell$.

Remark 1. Building on real-life applications, scenarios such as environmental monitoring, or smart city implementations are ripe for MCS. In these contexts, tasks may show up uncertainly (e.g., identifying a city block or mapping). Given their unpredictable nature, there's a need to collect data efficiently. Moreover, the sensitivity of the worker's bid involved necessitates strong privacy mechanisms. It is in such scenarios that our algorithm works, providing not just incentive mechanisms but also ensuring the protection of worker bids via differential privacy. This becomes paramount to prevent inference attacks and protect the privacy of participating workers.

Example 2. In this example, the platform has a task set $\mathcal{T} = \{\tau_1, \tau_2, \tau_3, \tau_4, \tau_5\}$ with five tasks and divides them into seven task subsets $\Gamma_1 = \{\tau_1, \tau_2\}$, $\Gamma_2 = \{\tau_2, \tau_3\}$, $\Gamma_3 = \{\tau_3, \tau_1, \tau_4\}$, $\Gamma_4 = \{\tau_4, \tau_5\}$, $\Gamma_5 = \{\tau_4\}$, $\Gamma_6 = \{\tau_2, \tau_5\}$ and $\Gamma_7 = \{\tau_2, \tau_4, \tau_5\}$. There are seven workers whose costs are $c_1 = 1.4$, $c_2 = 1.8$, $c_3 = 2.8$, $c_4 = 2.6$, $c_5 = 3.1$, $c_6 = 3.3$ and $c_7 = 3.6$. As HERALD* is truthful, which will be demonstrated later, it follows that $b_i = c_i$. Before the arrival of the actually requested tasks, the platform first matches workers for all possible task subsets. We refer to this as the **task-worker matching phase** which is shown in Alg. 1. For each task subset $\Gamma_j \in \mathcal{Y}$, the probability that worker i with the unique bid b_i is matched is $Pr(b_i)$. For each round of matching, the platform eventually assigns the task subset to the worker randomly according to the computed probability distribution. To prevent any monopolistic behavior and ensure the quality of the sensing

task, we require that each task $\tau_i \in \mathcal{T}$ be matched by at least two different people. We assume that the matching set $\mathcal{P} = \{(\Gamma_1, b_1), (\Gamma_2, b_2), (\Gamma_3, b_3), (\Gamma_4, b_4), (\Gamma_5, b_5), (\Gamma_6, b_6), (\Gamma_7, b_7)\}$.

We make the assumption that the task arrival follows a uniform distribution. If we set the number of tasks that arrive simultaneously to one, i.e., one task arrives at each time, then the task can be $\tau_1, \tau_2, \tau_3, \tau_4$ or τ_5 with the same probability of $\frac{1}{5}$. Consequently, the selection threshold is $T = 125.44$. The platform then executes the **winning selection phase** as described in Alg. 2. In the first iteration, after computing the cost-effectiveness of all matching pairs, the condition in Line 6 of HERALD* is satisfied. Therefore, the platform performs a type I selection and selects matching pair (Γ_1, b_1) as the winning pair. In the second iteration, the condition in Line 6 still holds, and thus another type I selection is performed, selecting matching pair (Γ_4, b_4) as the winning pair. By repeating this process, we obtain that the HERALD* algorithm selects the final winning set $\mathcal{S} = \{(\Gamma_1, b_1), (\Gamma_2, b_2), (\Gamma_4, b_4)\}$.

Next, the **payment determination phase** is carried out by the platform as shown in Alg. 3. For instance, for worker 1, whose covering set is $\mathcal{W}_1 = \{2, 3, 6, 7\}$, the replace set is $\mathcal{R}_1 = \{2, 3\}$. Thus, the payment for worker 1 is calculated as $p_1 = 1.8 + 2.8 = 4.6$. Similarly, the payments for worker 2 and worker 4 are computed as $p_2 = 1.4 + 2.8 = 4.2$ and $p_4 = 3.6$, respectively. Moreover, when two tasks arrive simultaneously, they can be $\{\tau_1, \tau_1\}, \{\tau_2, \tau_2\}, \{\tau_3, \tau_3\}, \{\tau_4, \tau_4\}, \{\tau_5, \tau_5\}$ with the same probability $\frac{1}{25}$, and $\{\tau_1, \tau_2\}, \{\tau_1, \tau_3\}, \{\tau_1, \tau_4\}, \{\tau_1, \tau_5\}, \{\tau_2, \tau_3\}, \{\tau_2, \tau_4\}, \{\tau_2, \tau_5\}, \{\tau_3, \tau_4\}, \{\tau_3, \tau_5\}, \{\tau_4, \tau_5\}$ with the same probability $\frac{2}{25}$. The selection threshold is $T = 181.248$. Subsequently, the HERALD* platform can execute the stages of **winning selection phase** and **payment determination phase** in a sequential manner in order to derive the set of winning entries and their respective payment amounts.

Remark 2. It is apparent that when the input parameter k (the number of tasks arriving simultaneously) is set to the total number of tasks n in the platform's task set, HERALD* exhibits a probability of $\frac{A_n^n}{n^n}$, which is reduced to an offline incentive mechanism. Several prior studies have explored offline scenarios where the platform has complete knowledge of the task information. Hence, this implies that HERALD* can be employed in a wider range of scenarios than the conventional offline incentive mechanisms.

C. Design of Score Functions

To apply the exponential mechanism to achieve the differential privacy of bidding results, it is necessary to devise score functions. Two score functions, namely a linear score function, and a logarithmic score function are created for this purpose. We will show that they have theoretical bounds on differential privacy and produce different impacts in simulations.

Linear score function: $f_{lin}(x) = -x$. For any worker $i \in \mathcal{W}$, the probability that worker i with bid b_i is matched with any task subset $\Gamma_j \in \mathcal{Y}$ is

$$Pr(b_i) \propto \begin{cases} \exp(-\frac{\epsilon b_i}{2\Delta u b_{max}}), & \text{if } i \in \mathcal{W}, \\ 0, & \text{otherwise.} \end{cases} \quad (3)$$

Since $u = f_{lin}(x) = -x$, we have

$$\begin{aligned} \Delta u &= \max_{r \in R} \max_{x, y: \|x-y\|_1 \leq 1} |u(x, r) - u(y, r)| \\ &= \frac{b_{max} - b_{min}}{b_{max}}. \end{aligned} \quad (4)$$

In order to guarantee that the score function's value is non-negative, we apply the following normalization process.

$$Pr(b_i) = \begin{cases} \frac{\exp(-\frac{\epsilon b_i}{2(b_{max} - b_{min})})}{\sum_{j \in \mathcal{W}} \exp(-\frac{\epsilon b_j}{2(b_{max} - b_{min})})}, & \text{if } i \in \mathcal{W}, \\ 0, & \text{otherwise.} \end{cases} \quad (5)$$

Logarithmic score function: $f_{ln}(x) = -\ln(x)$. For any worker $i \in \mathcal{W}$, the probability that worker i with bid b_i is matched with any task subset $\Gamma_j \in \mathcal{Y}$ is

$$Pr(b_i) \propto \begin{cases} \exp(-\frac{\epsilon \ln \frac{b_i}{b_{max}}}{2\Delta u}), & \text{if } i \in \mathcal{W}, \\ 0, & \text{otherwise.} \end{cases} \quad (6)$$

Since $u = f_{ln}(x) = -\ln(x)$, we have

$$\begin{aligned} \Delta u &= \max_{r \in R} \max_{x, y: \|x-y\|_1 \leq 1} |u(x, r) - u(y, r)| \\ &= \ln \frac{b_{max}}{b_{min}} - \ln \frac{b_{min}}{b_{max}} = -\ln \frac{1}{b_{max}} = \ln b_{max}, \end{aligned} \quad (7)$$

where b_{min} is normalized to 1 and b_{max} is a constant. We also need to normalize the score function to ensure that its value is non-negative

$$Pr(b_i) = \begin{cases} \frac{\exp\left(-\frac{\epsilon \ln \frac{b_i}{b_{max}}}{2 \ln b_{max}}\right)}{\sum_{j \in \mathcal{W}} \exp\left(-\frac{\epsilon \ln \frac{b_j}{b_{max}}}{2 \ln b_{max}}\right)}, & \text{if } i \in \mathcal{W}, \\ 0, & \text{otherwise.} \end{cases} \quad (8)$$

D. Analysis

This subsection will provide evidence that HERALD* conforms to the characteristics outlined in Section III-B.

Theorem 2. HERALD* is truthful.

The proof is given in Appendix A.

Lemma 1. HERALD* is individually rational.

Proof. Theorem 2 demonstrates that each worker bids their actual cost c_i . The individual rationality of HERALD* is ensured by the fact that the payment made to each winner i is equal to $p_i = p_i + \max\{b_i, b_{\mathcal{R}_i}\} \geq b_i = c_i$. \square

Besides proving truthfulness and individual rationality, we also demonstrate that HERALD* preserves the intended differential privacy of the bidding outcomes. Initially, we examine the impact of the linear score function on the differential privacy outcomes.

Theorem 3. For any constant $\epsilon > 0$, HERALD* with the linear score function preserves $\frac{\epsilon l}{2}$ -differential privacy, where $\epsilon > 0$ is a constant and l is the number of task subsets.

The proof is given in Appendix B. Next, we analyze the effect of the logarithmic score function on the differential privacy results.

Theorem 4. For any constant $\epsilon > 0$, HERALD* with the logarithmic score function preserves $\frac{\epsilon l}{2}$ -differential privacy, where $\epsilon > 0$ is a constant and l is the number of task subsets.

The proof is given in Appendix C. In addition to its truthfulness, individual rationality, and differential privacy, HERALD* exhibits a low level of computational complexity, as can be observed.

Proposition 1. *The computational complexity of the HERALD* is $\mathcal{O}(lm + l^2n^2 + ln^3)$.*

Proof. In order to determine the computational complexity of HERALD*, we must analyze the task-worker matching phase (as outlined in Alg. 1), the winning selection phase (in Alg. 2), and the payment determination phase (as described in Alg. 3), separately.

- 1) *task-worker Matching Phase:* In Alg. 1, the first loop (Lines 3-5) calculates the probability distribution over m iterations with a key operation in Line 4 having $\mathcal{O}(m)$ complexity. The second loop (Lines 6-9) completes task-worker matching in l iterations, where each matching in Line 7 also has $\mathcal{O}(m)$ complexity. Thus, the total computational complexity of this phase is $\mathcal{O}(m + lm)$, which simplifies to $\mathcal{O}(lm)$, accounting for both loops' combined computational requirements.
- 2) *Winning Selection Phase:* In Alg. 2, the main loop (Lines 3–15) ends in up to n rounds. Each round includes type I (Lines 7–8) and type II (Lines 9–10) selections, executed l times, with complexity $\mathcal{O}(l)$ for minimum value identification. Lines 12–14, handling match search and modifications, contribute $\mathcal{O}(n^2)$ complexity. Overall, this phase's complexity is $\mathcal{O}(nl^2 + n^3)$, considering the detailed computations required.
- 3) *Payment Determination Phase:* In Alg. 3, the main loop (Lines 1–9) completes in at most l iterations. Each iteration includes constructing a covering set with complexity $\mathcal{O}(ln^2)$ (Line 3) and forming a replaced set over n iterations (Lines 5–9), contributing $\mathcal{O}(nl + n^3)$ in complexity. Therefore, the overall computational complexity of Alg. 3 is $\mathcal{O}(n^2l^2 + ln^3)$, reflecting the combined computational requirements of constructing covering and replace sets.

By merging the task-worker matching phase, winning selection phase, and payment determination phase, HERALD* has a computational complexity of $\mathcal{O}(lm + l^2n^2 + ln^3)$. We have specified in Section III-A that $l < mn$. Some empirical running times for the three algorithms are documented in Appendix F for further reference. \square

In the following parts, we will demonstrate the competitive ratio of expected social cost attained by HERALD* when the tasks in \mathcal{T} arrive according to a uniform distribution. Specifically, we let $T = 64\mathbb{E}[C_{OPT}(\mathcal{A}, W)]$, where $\mathcal{A} \subseteq \mathcal{T}$ is a subset of k tasks that could potentially arrive concurrently from the task set \mathcal{T} . In order to calculate the competitive ratio of the expected social cost of HERALD*, we analyze the costs of type I selection and type II selection independently.

Lemma 2. *HERALD* achieves a competitive ratio of $\mathcal{O}(\ln n)$ on the expected social cost through type I selection when the tasks in the task set \mathcal{T} are distributed uniformly.*

The proof is given in Appendix D. We still need to establish a bound on the expected social cost of workers in the

winning set \mathcal{S} chosen by HERALD* via type II selection. To accomplish this, we must introduce the following notations.

We define $\mathcal{S}_{II} = \{1, \dots, \ell\}$ as the group of workers in the winning set \mathcal{S} selected by HERALD* via type II selection in the specified sequence. We also define $\tilde{\mathcal{T}}_i$ as the set of tasks whose sensory data is not gathered right before worker i is selected. Additionally, $n_i = |\tilde{\mathcal{T}}_i|$ represents the number of tasks in $\tilde{\mathcal{T}}_i$, and $k_i = n_i \frac{k}{n}$ is the expected count of requested tasks arriving from $\tilde{\mathcal{T}}_i$. We use \mathcal{A}_i to denote the subset of \mathcal{A} that contains requested tasks belonging only to $\tilde{\mathcal{T}}_i$. Furthermore, we define $\mathcal{S}^*(\mathcal{A}, W)$ as the winning set with the least social cost for any \mathcal{A} set. Then, let $\mathcal{S}'(\mathcal{A}_i, W)$ be the subset of $\mathcal{S}^*(\mathcal{A}, W)$ such that for each task $\tau_j \in \mathcal{A}_i$, the worker in $\mathcal{S}'(\mathcal{A}_i, W)$ has the corresponding task subset containing task τ_j and has the least cost among workers in $\mathcal{S}^*(\mathcal{A}, W)$.

Lemma 3. *When the arrivals of tasks in the task set \mathcal{T} follow a uniform distribution, HERALD* achieves a competitive ratio of $\mathcal{O}(\ln \ln)$ on the expected social cost through type II selection.*

The proof is given in Appendix E. By combining Lemma 2 and Lemma 3, the following theorem is established.

Theorem 5. *HERALD* achieves a competitive ratio of $\mathcal{O}(\ln \ln)$ on expected social cost when the arrivals of tasks in the task set \mathcal{T} follow a uniform distribution.*

Based on the results from Theorem 5, we can deduce that HERALD* yields a low expected social cost. Therefore, HERALD* can be applied to many scenarios with uncertain sensing tasks.

V. PERFORMANCE EVALUATION

The subsequent section will showcase the benchmark methods utilized for evaluating the performance of HERALD* and expound on the simulation settings employed in the experiment. Furthermore, the outcomes of the simulation will be presented.

A. Baseline Methods

As very few prior works relate directly to uncertain tasks in the MCS system, we chose to repurpose mechanisms from related optimization problems, such as set covering, to evaluate HERALD*. We compared HERALD* with CONE and COSY in a simulation, all adapted from the set covering problems in [37]. Despite being part of our original incentive mechanism, CONE and COSY's links to optimization problems make them suitable baseline mechanisms.

COST-eFFECTIVENESS greedy auction (CONE): When it comes to uncertain tasks without real-time constraints, the platform is aware that the tasks belonging to set \mathcal{T} are expected to arrive in the future with a probability distribution. To obtain sensory data for these tasks, the platform computes the CF for each matching pair and picks the winning pair, denoted by (Γ_j, b_i) , which has the lowest $CF_{|\Gamma_j \cap \mathcal{T}|}^{b_i}$ value among those of matching pairs in each iteration. The platform then proceeds to acquire the sensory data from worker i .

COST greedY auction (COSY): When it comes to uncertain tasks without real-time constraints, the platform acquires sensory data by examining the bids of matching pairs and determining the winning pair, denoted by (Γ_j, b_i) . This pair has the lowest bid value b_i among those of winning pairs and corresponds to a task subset Γ_j that includes at least one task that has not yet been covered in the current iteration. After selecting the winning pair, the platform proceeds to collect the sensory data from worker i .

The payment determination process in both CONE and COSY is identical to that of HERALD*. It is evident that both CONE and COSY are truthful and individually rational.

B. Simulation Settings

Table I outlines key metrics for various scenarios, including the cost c_i that worker i performs the assigned subset Γ_j of matching tasks, along with the number of tasks $|\Gamma_j|$ in each subset, the number of workers m and the number of sensing tasks n .

We present an assessment of HERALD*, which highlights how the expected social cost and expected total payment are affected by the number of workers and sensing tasks. Specifically, we conduct two evaluations: the first (setting I) fixes the number of sensing tasks at $n = 120$ and varies the number m of workers from 60 to 150 in steps of 5. The second (setting II) maintains the number of workers at $m = 80$ while incrementally varying the number n of sensing tasks from 80 to 160 in steps of 5. In both scenarios, we randomly and independently sample the cost c_i of worker i and the number of tasks in task subset Γ_j from uniform distributions within the intervals $[1, 5]$ and $[15, 20]$, respectively.

We will now examine how worker costs affect the expected social cost and expected total payment yielded by HERALD*. To investigate the impact of worker costs c_i , we consider three different intervals: $[1, 5]$, $[5, 10]$ and $[10, 15]$, in setting III. In this setting, we also randomly sample the number $|\Gamma_j|$ of tasks in the subset from the interval $[15, 20]$. Additionally, we set the number n of sensing tasks to 120, while varying the number m of workers from 60 to 150.

In setting IV, we assess how the number of tasks in subsets influences HERALD*'s social cost and total payment. Specifically, we examine three task intervals in setting IV: $[10, 15]$, $[15, 20]$, and $[20, 25]$. Workers' costs are sampled from $[1, 5]$. We fix the sensing tasks at $n = 120$ and vary worker numbers m from 60 to 150. Across all settings, we maintain a differential privacy coefficient of $\epsilon = 0.1$.

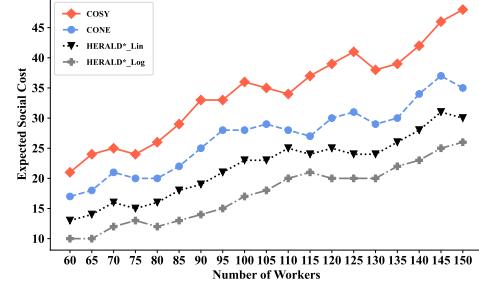
We also investigate how the privacy parameter ϵ affects HERALD*'s social cost and total payment. Due to space limitations, please see Appendix G for details.

C. Simulation Results

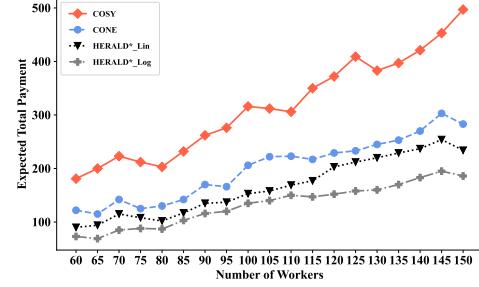
In Fig. 3, we analyze the effect of the number of workers. Specifically, Fig. 3(a) and Fig. 3(b) illustrate the impact on the expected social cost and expected total payment obtained by HERALD*, with results averaged over 100 runs. It is observed that HERALD* performs better than CONE and COSY. Interestingly, the expected social cost and expected

TABLE I
SIMULATION SETTINGS FOR HERALD*.

Settings	individual cost c_i	number $ \Gamma_j $ of each task subset	number m of workers	number n of sensing tasks
I	$[1, 5]$	$[15, 20]$	$[60, 150]$	120
II	$[1, 5]$	$[15, 20]$	80	$[80, 160]$
III	$[1, 5], [5, 10], [10, 15]$	$[15, 20]$	$[60, 150]$	120
IV	$[1, 5]$	$[10, 15], [15, 20], [20, 25]$	$[60, 150]$	120



(a)



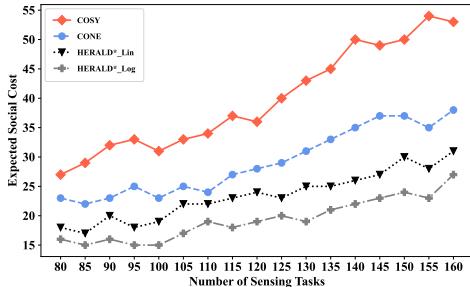
(b)

Fig. 3. (a). Expected social cost versus different numbers of workers for uncertain tasks. (b). Expected total payment versus different numbers of workers for uncertain tasks.

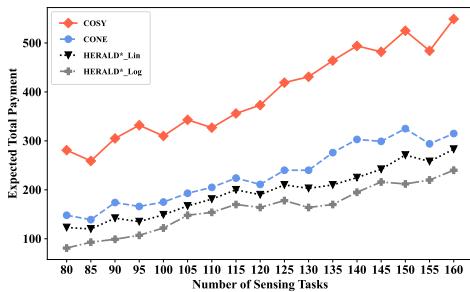
total payment obtained by HERALD* using the logarithmic score function is lower than those obtained using the linear score function. This is due to the logarithmic score function giving higher chances of selection to users with low bids, resulting in a preference for such users.

Fig. 4 examines the impact of the number of tasks on HERALD*'s performance, with results averaged over 100 runs. As depicted in Fig. 4(a) and Fig. 4(b), which show the expected social cost and expected total payment, respectively. Similar to the findings in Fig. 3, HERALD* outperforms CONE and COSY in this setting. Additionally, as the number of tasks increases, the expected social cost and expected total payment of HERALD* also increase due to the need for more workers to collect sensory data. Furthermore, consistent with the earlier results, the expected social cost and expected total payment of HERALD* with the logarithmic score function are lower than those of HERALD* with the linear score function for the same reasons.

Fig. 5 depicts the effect of workers' cost on the performance of HERALD* under the linear score function. Specifically, Fig. 5(a) and Fig. 5(b) illustrate the impact of workers' cost on the expected social cost and expected total payment generated by HERALD*. The results indicate that as the workers' cost increases, both the expected social cost and expected total

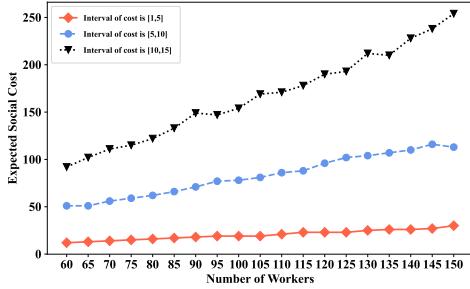


(a)

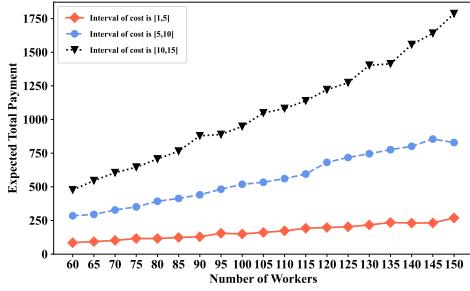


(b)

Fig. 4. (a). Expected social cost versus different numbers of sensing tasks for uncertain tasks. (b). Expected total payment versus different numbers of sensing tasks for uncertain tasks.



(a)

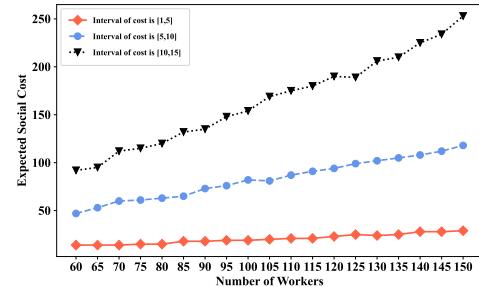


(b)

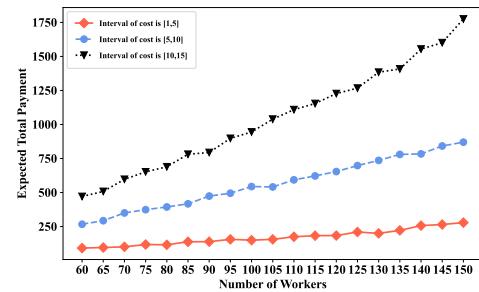
Fig. 5. (a). The impact of worker's cost on the expected social cost obtained by HERALD* for uncertain tasks with the liner score function. (b). The impact of worker's cost on the expected total payment obtained by HERALD* for uncertain tasks with the liner score function.

payment of HERALD* also increase. This is because a higher workers' cost implies that more social cost is required for the same tasks, and the platform has to pay more to the workers compared to the scenario with a lower workers' cost. Similar findings are observed for HERALD* under the logarithmic score function, as shown in Fig. 6.

Fig. 7 displays the influence of the number of matching tasks per worker on HERALD*'s expected social cost and expected total payment under the linear score function. Fig. 7

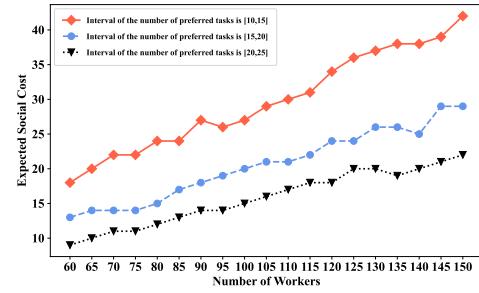


(a)

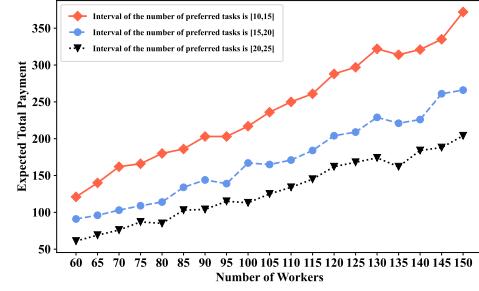


(b)

Fig. 6. (a). The impact of worker's cost on the expected social cost obtained by HERALD* for uncertain tasks with the logarithmic score function. (b). The impact of worker's cost on the expected total payment obtained by HERALD* for uncertain tasks with the logarithmic score function.



(a)



(b)

Fig. 7. (a). The impact of the number of worker's matching tasks on the expected social cost obtained by HERALD* for uncertain tasks with the liner score function. (b). The impact of the number of worker's matching tasks on the expected total payment obtained by HERALD* for uncertain tasks with the liner score function.

(a) and 7(b) highlight this impact, showing a decline in both the expected social cost and expected total payment as the number of tasks per worker increases. This trend is attributed to the reduced need for workers as task allocation per worker rises, leading to lower costs and payments in HERALD*. A similar trend is observed under the logarithmic score function, as depicted in Fig. 8.

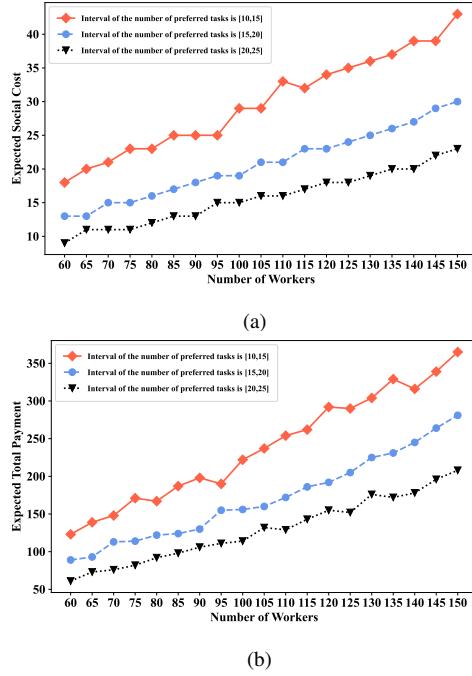


Fig. 8. (a). The impact of the number of worker's matching tasks on the expected social cost obtained by HERALD* for uncertain tasks with the logarithmic score function. (b). The impact of the number of worker's matching tasks on the expected total payment obtained by HERALD* for uncertain tasks with the logarithmic score function.

VI. CONCLUSION AND FUTURE WORK

In this manuscript, we introduce HERALD*, a novel incentive mechanism for a task allocation system in which tasks without real-time constraints arrive randomly according to a probability distribution. Our investigation indicates that HERALD* meets several desirable properties, such as truthfulness, individual rationality, differential privacy, low computational complexity, and low social cost. More specifically, we have shown that HERALD* guarantees $\frac{\epsilon}{2}$ -differential privacy for both linear and logarithmic score functions, and it achieves a competitive ratio of $\ln \ln n$ on expected social cost. We have also validated the effectiveness of HERALD* through both theoretical analysis and extensive simulations.

Moving forward, we plan to tackle the crucial privacy concerns associated with storing sensory data. Our future endeavors include integrating HERALD* with advanced data privacy algorithms (such as [38]–[40]). This integration aims to elevate our platform's data protection standards. Additionally, we plan to tackle “response fatigue” to improve user experience and system usability. We will introduce user-customizable interaction levels, intelligent notifications, and extensive user support to reduce decision overload and enhance the practicality of HERALD* in real-world applications.

REFERENCES

- [1] R. Ding, Z. Yang, Y. Wei, H. Jin and X. Wang, “Multi-Agent Reinforcement Learning for Urban Crowd Sensing with For-Hire Vehicles,” *Proc. IEEE Int. Conf. Comput. Commun. (INFOCOM)*, 2021.
- [2] G. Fan, H. Jin, Q. Liu, W. Qin, X. Gan, H. Long, L. Fu, and X. Wang, “Joint Scheduling and Incentive Mechanism for Spatio-Temporal Vehicular Crowd Sensing,” *IEEE Trans. Mobile Comput.*, vol. 20, no. 4, pp. 1449–1464, 2021.
- [3] M. S. Pan and K. Y. Li, “ezNavi: An Easy-to-Operate Indoor Navigation System Based on Pedestrian Dead Reckoning and Crowdsourced User Trajectories,” *IEEE Trans. Mobile Comput.*, vol. 20, no. 2, pp. 488–501, 2021.
- [4] S. Yu, X. Chen, S. Wang, L. Pu and D. Wu, “An Edge Computing-Based Photo Crowdsourcing Framework for Real-Time 3D Reconstruction,” *IEEE Trans. Mobile Comput.*, vol. 21, no. 2, pp. 421–432, 2022.
- [5] Y. Liu, Z. Yu, B. Guo, Q. Han, J. Su and J. Liao, “CrowdOS: A Ubiquitous Operating System for Crowdsourcing and Mobile Crowd Sensing,” *IEEE Trans. Mobile Comput.*, vol. 21, no. 3, pp. 878–894, 2022.
- [6] W. Zhang, D. Yang, W. Wu, H. Peng, N. Peng, H. Zhang, and X. Shen, “Optimizing federated learning in distributed industrial IoT: A multi-agent approach,” *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 12, pp. 3688–3703, 2021.
- [7] D. Yang, Z. Cheng, W. Zhang, H. Zhang, and X. Shen, “Burst-Aware Time-Triggered Flow Scheduling With Enhanced Multi-CQF in Time-Sensitive Networks,” *IEEE/ACM Transactions on Networking*, 2023.
- [8] D. Yang, W. Zhang, Q. Ye, C. Zhang, N. Zhang, C. Huang, H. Zhang, and X. Shen, “DetFed: Dynamic Resource Scheduling for Deterministic Federated Learning over Time-sensitive Networks,” *IEEE Transactions on Mobile Computing*, 2023.
- [9] Z. Wang, J. Li, J. Hu, J. Ren, Z. Li, and Y. Li, “Towards privacy-preserving incentive for mobile crowdsensing under an untrusted platform,” in *Proc. IEEE Int. Conf. Comput. Commun. (INFOCOM)*, 2019.
- [10] R. Zhou, Z. P. Li, and C. Wu, “A truthful online mechanism for location-aware tasks in mobile crowd sensing,” in *IEEE Trans. Mob. Comput.*, vol. 17, no. 8, pp. 1737–1749, 2018.
- [11] L. Yang, M. Zhang, S. He, M. Li, and J. Zhang, “Crowd-empowered privacy-preserving data aggregation for mobile crowdsensing,” in *Proc. 19th ACM Int. Symp. Mobile Netw. Comput. (MobiHoc)*, 2018.
- [12] M. H. Cheung, F. Hou, and J. Huang, “Delay-sensitive mobile crowdsensing: Algorithm design and economics,” in *IEEE Trans. Mob. Comput.*, vol. 17, no. 12, pp. 2761–2774, 2018.
- [13] Q. Ma, L. Gao, Y. Liu, and J. Huang, “Incentivizing Wi-Fi network crowdsourcing: A contract theoretic approach,” *IEEE/ACM Trans. Netw.*, vol. 26, no. 3, pp. 1035–1048, 2018.
- [14] Y. Qu, S. Tang, C. Dong, P. Li, S. Guo, H. Dai, and F. Wu, “Posted pricing for chance constrained robust crowdsensing,” in *IEEE Trans. Mob. Comput.*, vol. 19, no. 1, pp. 188–199, 2020.
- [15] F. Restuccia, P. Ferraro, S. Silvestri, S. K. Das and G. L. Re, “IncentMe: Effective mechanism design to stimulate crowdsensing participants with uncertain mobility,” in *IEEE Trans. Mob. Comput.*, vol. 18, no. 7, pp. 1571–1584, 2019.
- [16] W. Jin, M. Xiao, M. Li, and L. Guo, “If you do not care about it, sell it: Trading location privacy in mobile crowd sensing,” in *Proc. IEEE Int. Conf. Comput. Commun. (INFOCOM)*, 2019.
- [17] L. Wang, Z. Yu, Q. Han, B. Guo, and H. Xiong, “Multi-objective optimization based allocation of heterogeneous spatial crowdsourcing tasks,” in *IEEE Trans. Mob. Comput.*, vol. 17, no. 7, pp. 1637–1650, 2018.
- [18] Y. Guo, H. Xie, Y. Miao, C. Wang, and X. Jia, “Fedcrowd: A federated and privacy-preserving crowdsourcing platform on blockchain,” *IEEE Transactions on Services Computing*, vol. 15, no. 4, pp. 2060–2073, 2022.
- [19] S. Gao, X. Chen, J. Zhu, X. Dong, and J. Ma, “TrustWorker: A trustworthy and privacy-preserving worker selection scheme for blockchain-based crowdsensing,” *IEEE Transactions on Services Computing*, vol. 15, no. 6, pp. 3577–3590, 2022.
- [20] H. Jin, L. Su, D. Chen, H. Guo, K. Nahrstedt, and J. Xu, “Thanos: Incentive mechanism with quality awareness for mobile crowd sensing,” *IEEE Trans. Mob. Comput.*, vol. 18, no. 8, pp. 1951–1964, 2019.
- [21] L. Jiang, X. Niu, J. Xu, D. Yang and L. Xu, “Incentive mechanism design for truth discovery in crowdsourcing with copiers,” *IEEE Transactions on Services Computing*, vol. 15, no. 5, pp. 2838–2853, 2022.
- [22] F. Basik, B. Gedik, H. Ferhatosmanoglu, and K. Wu, “Fair Task Allocation in Crowdsourced Delivery,” *IEEE Transactions on Services Computing*, vol. 14, no. 4, pp. 1040–1053, 2021.
- [23] S. Bhattacharjee, N. Ghosh, V. K. Shah, and S. K. Das, “QnQ: Quality and quantity based unified approach for secure and trustworthy mobile crowdsensing,” in *IEEE Trans. Mob. Comput.*, vol. 19, no. 1, pp. 200–216, 2020.
- [24] X. Gong, N. B. Shroff, “Truthful mobile crowdsensing for strategic users with private data quality” *IEEE/ACM Trans. Netw.*, vol. 27, no. 5, pp. 1959–1972, 2019.
- [25] X. Liu, and J. Liu, “A Truthful Double Auction Mechanism for Multi-Resource Allocation in Crowd Sensing Systems,” *IEEE Transactions on Services Computing*, vol. 15, no. 5, pp. 2579–2590, 2022.

- [26] Y. Hu, and R. Zhang, "Differentially-private incentive mechanism for crowdsourced radio environment map construction," in *Proc. IEEE Int. Conf. Comput. Commun. (INFOCOM)*, 2019.
- [27] K. Han, H. Liu, S. Tang, M. Xiao, and J. Luo, "Differentially private mechanisms for budget limited mobile crowdsourcing," in *IEEE Trans. Mobile Comput.*, vol. 18, no. 4, pp. 934-946, 2019.
- [28] X. Bai, J. Yin, and Y. P. Wang, "Sensor Guardian: prevent privacy inference on Android sensors," *EURASIP Journal on Information Security*, vol. 2017, no. 1, pp. 1-17, 2017.
- [29] J. Lin, D. Yang, M. Li, J. Xu, and G. Xue, "BidGuard: A framework for privacy-preserving crowdsensing incentive mechanisms," in *2016 IEEE conference on communications and network security (CNS)*, 2016.
- [30] M. Yang, I. Tjuiawinata, K. Lam, J. Zhao, and L. Sun, "Secure Hot Path Crowdsourcing With Local Differential Privacy Under Fog Computing Architecture," *IEEE Transactions on Services Computing*, vol. 15, no. 4, pp. 2188-2201, 2022.
- [31] J. Wei, Y. Lin, X. Yao, and J. Zhang, "Differential privacy-based location protection in spatial crowdsourcing," *IEEE Transactions on Services Computing*, vol. 15, no. 1, pp. 45-58, 2022.
- [32] H. Wu, L. Wang, K. Cheng, D. Yang, J. Tang, and G. Xue, "Privacy-Enhanced and Practical Truth Discovery in Two-Server Mobile Crowd-sensing," *IEEE Transactions on Network Science and Engineering*, vol. 9, no. 3, pp. 1740-1755, 2022.
- [33] J. Shu, X. Jia, K. Yang, and H. Wang, "Privacy-preserving task recommendation services for crowdsourcing," *IEEE Transactions on Services Computing*, vol. 14, no. 1, pp. 235-247, 2021.
- [34] H. Wu, B. Düdder, L. Wang, S. Sun, and G. Xue, "Blockchain-based reliable and privacy-aware crowdsourcing with truth and fairness assurance," *IEEE Internet of Things Journal*, vol. 9, no. 5, pp. 3586-3598, 2021.
- [35] C. Dwork, and A. Roth, "The algorithmic foundations of differential privacy," *Foundations and Trends® in Theoretical Computer Science*, vol. 9, no. 3-4, pp. 211-407, 2014.
- [36] Y. Singer, "Budget feasible mechanisms," in *Proc. IEEE Symp. Found. Comput. Sci. (FOCS)*, 2010.
- [37] F. Grandoni, A. Gupta, S. Leonardi, P. Miettinen, P. Sankowski, and M. Singh, "Set covering with our eyes closed," *SIAM Journal on Computing.*, vol. 42, no. 3, pp. 808-830, 2013.
- [38] Z. Sun, Y. Wang, Z. Cai, T. Liu, X. Tong, & N. Jiang, (2021). A two-stage privacy protection mechanism based on blockchain in mobile crowdsourcing. *International Journal of Intelligent Systems*, 36(5), 2058-2080.
- [39] W. Wang, Y. Wang, P. Duan, T. Liu, X. Tong, & Z. Cai, (2022). A triple real-time trajectory privacy protection mechanism based on edge computing and blockchain in mobile crowdsourcing. *IEEE Transactions on Mobile Computing*.
- [40] M. Wang, T. Zhu, X. Zuo, M. Yang, S. Yu, & W. Zhou, (2023). Differentially private crowdsourcing with the public and private blockchain. *IEEE Internet of Things Journal*.
- [41] E. Casella, AR. Khamesi, S. Silvestri, DA. Baker, & SK. Das, (2022). Hvac power conservation through reverse auctions and machine learning. In *2022 IEEE International Conference on Pervasive Computing and Communications (PerCom)* (pp. 89-100). IEEE.



Xikun Jiang received her Ph.D. degree from the Department of Computer Science and Engineering, Shanghai Jiao Tong University, China in 2023. Presently, she holds a postdoctoral position in the Department of Computer Science (DIKU) at the University of Copenhagen (UCPH), Denmark. Her research interests include mobile crowdsensing, online markets, and machine learning.



Chenhao Ying received Ph.D. degree from Department of Computer Science and Engineering, Shanghai Jiao Tong University, China in 2022. Before that, he received the B.E. degree from the Department of Communication Engineering, Xidian University, China, in 2016. He is a research assistant professor of Department of Computer Science and Engineering, Shanghai Jiao Tong University, China. His current research interests include mobile crowd sensing, communication coding algorithms, and wireless communications.



Lei Li is PhD candidate of Department of Computer Science, University of Copenhagen, Copenhagen, Denmark from 2020. Previously, after he received the master degree of University of Science of Thehology of China, he acted as a researcher of SenseTime for deep learning of image analysis. Now he is working on deep learning for 3D point cloud and multi-model analysis. His research interests include machine learning, computer visionand image processing.



research is bridging the formal foundations of computer science and complex industrial applications.



Systems Engineering at Arizona State University, US. Her research interests include data security and privacy protection, mobile crowdsensing/crowdsourcing, and blockchain-based applications.



Haiming Jin received the BS degree from Shanghai Jiao Tong University, Shanghai, China, in 2012, and the PhD degree from the University of Illinois at UrbanaChampaign (UIUC), Urbana, IL, in 2017. He is currently a tenure-track associate professor with the John Hopcroft Center for Computer Science and the Department of Electronic Engineering, Shanghai Jiao Tong University. Before this, he was a postdoctoral research associate with the Coordinated Science Laboratory, UIUC. His research interests include crowd and social sensing systems, reinforcement learning, and mobile pervasive and ubiquitous computing.



Yuan Luo (Member, IEEE) received the B.S. degree in applied mathematics, and the M.S. and Ph.D. degrees in probability statistics from Nankai University, Tianjin, China, in 1993, 1996, and 1999, respectively. From July 1999 to April 2001, he was a Post-Doctoral Researcher with the Institute of Systems Science, Chinese Academy of Sciences, Beijing, China. From May 2001 to April 2003, he was a Post-Doctoral Researcher with the Institute for Experimental Mathematics, University of Duisburg-Essen, Essen, Germany. Since June 2003, he has been with the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China. Since 2006, he has been a Full Professor and the Vice Director of the Department, from 2016 to 2018 and since 2021. His current research interests include coding theory, information theory, and big data analysis.

VII. APPENDIX

A. Proof of Theorem 2

To prove the truthfulness of HERALD*, we will demonstrate its adherence to the criteria outlined in Theorem 6.

Theorem 6 ([36]). *A mechanism satisfies truthfulness only if the following conditions are met:*

- 1) *The selection rule is monotonic: If a matching pair wins by offering a bid of b_i , it will also win if it bids $b'_i \leq b_i$;*
- 2) *Each winning pair is compensated with the critical value: A matching pair will not win if it bids higher than this value.*

Proof. Monotonicity: Given a matching pair (Γ_j, b_i) , we will prove that if it wins with a bid of b_i , it will also win with a bid of $b'_i \leq b_i$. We will demonstrate this in the following two scenarios.

Case 1: During a winning selection phase iteration, if the CF of the winning pair (Γ_j, b_i) satisfies $\frac{b_i}{|\Gamma_j \cap \mathcal{T}|} \leq \frac{T}{|\mathcal{T}|}$, then it implies that it possesses the smallest CF among all the matching pairs. Consequently, it will also win with a bid of $b'_i \leq b_i$.

Case 2: During an iteration, if the CF of the winning pair (Γ_j, b_i) satisfies $\frac{b_i}{|\Gamma_j \cap \mathcal{T}|} > \frac{T}{|\mathcal{T}|}$, then it indicates that it possesses the least cost among the matching pairs and that there exists no matching pair (Γ_J, b_I) such that $\frac{b_J}{|\Gamma_J \cap \mathcal{T}|} \leq \frac{T}{|\mathcal{T}|}$. We must subsequently examine two sub-cases.

Subcase 2.1: If the bid $b'_i \leq b_i$ satisfies $\frac{b'_i}{|\Gamma_j \cap \mathcal{T}|} > \frac{T}{|\mathcal{T}|}$, then it will also win with a bid of b'_i , as b'_i is the lowest and there is no matching pair (Γ_J, b_I) such that $\frac{b_I}{|\Gamma_J \cap \mathcal{T}|} \leq \frac{T}{|\mathcal{T}|}$.

Subcase 2.2: If the bid $b'_i \leq b_i$ satisfies $\frac{b'_i}{|\Gamma_j \cap \mathcal{T}|} \leq \frac{T}{|\mathcal{T}|}$, then it will also win by bidding b'_i , as it is the sole matching pair with a CF that is less than or equal to $\frac{T}{|\mathcal{T}|}$.

Critical Value: If a matching pair (Γ_j, b_i) wins, it follows that worker i 's payment is $p_i = p_i + \max\{b_i, p_{\mathcal{R}_i}\}$, where $p_{\mathcal{R}_i} = \sum_{\ell \in \mathcal{R}_i} b_\ell$. If worker i increases his/her bid to \tilde{b}_i such that $\tilde{b}_i \leq p_{\mathcal{R}_i}$, his/her payments remain constant. However, if $\tilde{b}_i > p_{\mathcal{R}_i}$, we must examine the following two cases during each iteration of the winning selection phase.

Case 1: When CF of matching pair (Γ_j, \tilde{b}_i) satisfies $\frac{\tilde{b}_i}{|\Gamma_j \cap \mathcal{T}|} \leq \frac{T}{|\mathcal{T}|}$, we will prove that there is a matching pair (Γ_q, b_k) where $k \in \mathcal{R}_i$ such that $\frac{b_k}{|\Gamma_q \cap \mathcal{T}|} \leq \frac{\tilde{b}_i}{|\Gamma_j \cap \mathcal{T}|}$. We have $\frac{\tilde{b}_i}{|\Gamma_j \cap \mathcal{T}|} \geq \frac{\sum_{\ell \in \mathcal{R}_i} b_\ell}{\sum_{\ell \in \mathcal{R}_i} |\Gamma_\ell \cap \mathcal{T}|}$. Then let matching pair (Γ_q, b_k) be the one with the minimum CF $\frac{b_k}{|\Gamma_q \cap \mathcal{T}|}$ among all matching pairs where workers belong to \mathcal{R}_i , which means that $\frac{b_k}{|\Gamma_q \cap \mathcal{T}|} \leq \frac{b_\ell}{|\Gamma_\ell \cap \mathcal{T}|}$ for $\forall \ell \in \mathcal{R}_i$, i.e., $b_k |\Gamma_q \cap \mathcal{T}| \leq b_\ell |\Gamma_\ell \cap \mathcal{T}|$. Therefore, we have $b_k \sum_{\ell \in \mathcal{R}_i} |\Gamma_\ell \cap \mathcal{T}| \leq |\Gamma_q \cap \mathcal{T}| \sum_{\ell \in \mathcal{R}_i} b_\ell$, i.e., $\frac{b_k}{|\Gamma_q \cap \mathcal{T}|} \leq \frac{\sum_{\ell \in \mathcal{R}_i} b_\ell}{\sum_{\ell \in \mathcal{R}_i} |\Gamma_\ell \cap \mathcal{T}|}$. Since $\frac{b_k}{|\Gamma_q \cap \mathcal{T}|} \leq \frac{\tilde{b}_i}{|\Gamma_j \cap \mathcal{T}|}$, the platform will select matching pair (Γ_q, b_k) instead of (Γ_j, \tilde{b}_i) in this iteration.

Case 2: When CF of matching pair (Γ_j, \tilde{b}_i) satisfies $\frac{\tilde{b}_i}{|\Gamma_j \cap \mathcal{T}|} > \frac{T}{|\mathcal{T}|}$, we need to consider two subcases.

Subcase 2.1: Once there exist some matching pairs $(\Gamma_q, \tilde{b}_\ell)$ for $\Gamma_q \in \mathcal{Y}$ and $\ell \in \mathcal{R}_i$ such that $\frac{b_\ell}{|\Gamma_q \cap \mathcal{T}|} \leq \frac{T}{|\mathcal{T}|}$, the platform

will select a matching pair $(\Gamma_q, \tilde{b}_\ell)$ among them with the minimum CF instead of the matching pair (Γ_j, \tilde{b}_i) .

Subcase 2.2: Once the CFs of all matching pairs $(\Gamma_q, \tilde{b}_\ell)$ for $\Gamma_q \in \mathcal{Y}$ and $\ell \in \mathcal{R}_i$ satisfies $\frac{b_\ell}{|\Gamma_q \cap \mathcal{T}|} > \frac{T}{|\mathcal{T}|}$, the platform will always find a matching pair $(\Gamma_q, \tilde{b}_\ell)$ with the minimum bid b_ℓ such that $b_\ell \leq p_{\mathcal{R}_i} \leq \tilde{b}_i$, which means that the platform will not select the matching pair (Γ_j, \tilde{b}_i) .

Therefore, the conclusion holds. \square

B. Proof of Theorem 3

Proof. Consider two input bid profiles \vec{b} and \vec{b}' that differ in only one bid. Let $M(\vec{b})$ and $M(\vec{b}')$ denote the task-worker matching results by HERALD* with inputs \vec{b} and \vec{b}' , respectively. We aim to prove that HERALD* achieves differential privacy for an arbitrary sequence of task-worker matching results $\mathcal{I} = \{(\Gamma_1, b_1), (\Gamma_2, b_2), \dots, (\Gamma_l, b_l)\}$ of length l for m workers, where each worker can match k task subsets with $0 \leq k \leq l$. To analyze the relative probability of HERALD* for the given bid inputs \vec{b} and \vec{b}' , we consider:

$$\begin{aligned} \Pr[M(\vec{b}) = \mathcal{I}] &\stackrel{(a)}{=} \prod_{j=1}^l \frac{\exp(-\frac{\epsilon b_j}{2(b_{\max} - b_{\min})})}{\sum_{i \in \mathcal{W}} \exp(-\frac{\epsilon b_i}{2(b_{\max} - b_{\min})})} \\ \Pr[M(\vec{b}') = \mathcal{I}] &= \prod_{j=1}^l \frac{\exp(-\frac{\epsilon b'_j}{2(b_{\max} - b_{\min})})}{\exp(-\frac{\epsilon b'_j}{2(b_{\max} - b_{\min})})} \times \prod_{j=1}^l \frac{\sum_{i \in \mathcal{W}} \exp(-\frac{\epsilon b'_i}{2(b_{\max} - b_{\min})})}{\sum_{i \in \mathcal{W}} \exp(-\frac{\epsilon b_i}{2(b_{\max} - b_{\min})})}, \end{aligned} \quad (9)$$

where equation (a) is derived by formula (5) and b_i, b_j denote the bids of workers matching the task subsets Γ_i, Γ_j , respectively. Then, we prove this theorem in two cases. When $b_k > b'_k$, the value of the first product is at most 1, we have

$$\begin{aligned} \frac{\Pr[M(\vec{b}) = \mathcal{I}]}{\Pr[M(\vec{b}') = \mathcal{I}]} &\leq \prod_{j=1}^l \frac{\sum_{i \in \mathcal{W}} \exp(-\frac{\epsilon b'_i}{2(b_{\max} - b_{\min})})}{\sum_{i \in \mathcal{W}} \exp(-\frac{\epsilon b_i}{2(b_{\max} - b_{\min})})} \\ &= \prod_{j=1}^l \frac{\sum_{i \in \mathcal{W}} \exp(\frac{\epsilon(b_i - b'_i)}{2(b_{\max} - b_{\min})}) \exp(-\frac{\epsilon b_i}{2(b_{\max} - b_{\min})})}{\sum_{i \in \mathcal{W}} \exp(-\frac{\epsilon b_i}{2(b_{\max} - b_{\min})})} \\ &= \prod_{j=1}^l E_{i \in \mathcal{W}} \left[\exp\left(\frac{\epsilon(b_i - b'_i)}{2(b_{\max} - b_{\min})}\right) \right] \\ &\stackrel{(a)}{\leq} \prod_{j=1}^l E_{i \in \mathcal{W}} \left[1 + (e-1)\left(\frac{\epsilon(b_i - b'_i)}{2(b_{\max} - b_{\min})}\right) \right] \\ &\stackrel{(b)}{\leq} \exp\left(\epsilon(e-1)\left(\frac{\sum_{j=1}^l E_{i \in \mathcal{W}}(b_i - b'_i)}{2(b_{\max} - b_{\min})}\right)\right) \\ &\stackrel{(c)}{\leq} \exp\left(\epsilon(e-1)\left(\frac{\sum_{j=1}^l (b_{\max} - b_{\min})}{2(b_{\max} - b_{\min})}\right)\right) = \exp\left(\frac{\epsilon(e-1)l}{2}\right), \end{aligned} \quad (10)$$

where the inequality (a) holds because for all $x \leq 1$, $e^x \leq 1 + (e-1)x$. The inequality (b) holds because for all $x \in R$, $1+x \leq e^x$. In Section III-A, it was mentioned that the bid b_i of every worker i is confined within the interval $[b_{\min}, b_{\max}]$, where b_{\min} is normalized to 1 and b_{\max} is a fixed constant such that inequality (c) is satisfied.

When $b_k \leq b'_k$, the value of the second product is at most 1, we have

$$\begin{aligned} \frac{\Pr[M(\vec{b}) = \mathcal{I}]}{\Pr[M(\vec{b}') = \mathcal{I}]} &\leq \prod_{j=1}^l \frac{\exp(-\frac{\epsilon b_j}{2(b_{max} - b_{min})})}{\exp(-\frac{\epsilon b'_j}{2(b_{max} - b_{min})})} \\ &= \prod_{j=1}^l \exp\left(\frac{\epsilon(b'_j - b_j)}{2(b_{max} - b_{min})}\right) = \exp\left(\frac{\epsilon}{2(b_{max} - b_{min})} \sum_{j=1}^l (b'_j - b_j)\right) \\ &\stackrel{(a)}{\leq} \exp\left(\frac{\epsilon}{2(b_{max} - b_{min})} \times l(b_{max} - b_{min})\right) = \exp\left(\frac{\epsilon l}{2}\right), \end{aligned} \quad (11)$$

in which the inequality (a) holds because of the same reason of inequality (c) in formula (10). Combining the formulas (10) and (11), the proof is completed. \square

C. Proof of Theorem 4

Proof. The proof is analogous to that of the linear score function, and thus we will not repeat the default setting. Instead, we examine the HERALD* relative probability for a particular input bid profile \vec{b} and its perturbed version \vec{b}' :

$$\begin{aligned} \frac{\Pr[M(\vec{b}) = \mathcal{I}]}{\Pr[M(\vec{b}') = \mathcal{I}]} &\stackrel{(a)}{=} \prod_{j=1}^l \frac{\sum_{i \in \mathcal{W}} \exp\left(\frac{-\epsilon \ln \frac{b_j}{b_{max}}}{2 \ln b_{max}}\right)}{\sum_{i \in \mathcal{W}} \exp\left(\frac{-\epsilon \ln \frac{b'_j}{b_{max}}}{2 \ln b_{max}}\right)} \\ &= \prod_{j=1}^l \frac{\exp\left(\frac{-\epsilon \ln \frac{b_j}{b_{max}}}{2 \ln b_{max}}\right)}{\exp\left(\frac{-\epsilon \ln \frac{b'_j}{b_{max}}}{2 \ln b_{max}}\right)} \times \prod_{j=1}^l \frac{\sum_{i \in \mathcal{W}} \exp\left(\frac{-\epsilon \ln \frac{b'_j}{b_{max}}}{2 \ln b_{max}}\right)}{\sum_{i \in \mathcal{W}} \exp\left(\frac{-\epsilon \ln \frac{b_i}{b_{max}}}{2 \ln b_{max}}\right)}, \end{aligned} \quad (12)$$

where equation (a) is derived by formula (8). Then, we prove this theorem in two cases. When $b_k > b'_k$, the value of the first product is at most 1, we have

$$\begin{aligned} \frac{\Pr[M(\vec{b}) = \mathcal{I}]}{\Pr[M(\vec{b}') = \mathcal{I}]} &\leq \prod_{j=1}^l \frac{\sum_{i \in \mathcal{W}} \exp\left(\frac{-\epsilon \ln \frac{b'_j}{b_{max}}}{2 \ln b_{max}}\right)}{\sum_{i \in \mathcal{W}} \exp\left(\frac{-\epsilon \ln \frac{b_i}{b_{max}}}{2 \ln b_{max}}\right)} \\ &= \prod_{j=1}^l \frac{\sum_{i \in \mathcal{W}} \exp\left(\frac{-\epsilon}{2} \left(\frac{\ln \frac{b'_j}{b_{max}}}{\ln b_{max}} - \frac{\ln \frac{b_i}{b_{max}}}{\ln b_{max}}\right)\right) \exp\left(\frac{-\epsilon \ln \frac{b_i}{b_{max}}}{2 \ln b_{max}}\right)}{\sum_{i \in \mathcal{W}} \exp\left(\frac{-\epsilon \ln \frac{b_i}{b_{max}}}{2 \ln b_{max}}\right)} \\ &= \prod_{j=1}^l E_{i \in \mathcal{W}} \left[\exp\left(\frac{-\epsilon \ln \frac{b'_j}{b_i}}{2 \ln b_{max}}\right) \right] \\ &\stackrel{(a)}{\leq} \prod_{j=1}^l E_{i \in \mathcal{W}} \left[1 + (e-1) \left(\frac{-\epsilon \ln \frac{b'_j}{b_i}}{2 \ln b_{max}} \right) \right] \\ &\stackrel{(b)}{\leq} \exp\left(\frac{\epsilon(e-1)}{2} \left(\frac{\sum_{j=1}^l E_{i \in \mathcal{W}} \ln \frac{b_i}{b'_j}}{\ln b_{max}} \right)\right) \\ &\stackrel{(c)}{\leq} \exp\left(\frac{\epsilon(e-1)}{2} \left(\frac{\sum_{j=1}^l \ln \frac{b_{max}}{b_{min}}}{\ln b_{max}} \right)\right) = \exp\left(\frac{\epsilon(e-1)l}{2}\right), \end{aligned} \quad (13)$$

where inequality (a) is valid because, for all $x \leq 1, e^x \leq 1 + (e-1)x$. The inequality (b) is valid because, for all $x \in R, 1 + x \leq e^x$. The inequality (c) holds because of the same reason of inequality (c) in formula (10).

When $b_k \leq b'_k$, the value of the second product is at most 1, we have

$$\begin{aligned} \frac{\Pr[M(\vec{b}) = \mathcal{I}]}{\Pr[M(\vec{b}') = \mathcal{I}]} &\leq \prod_{j=1}^l \frac{\exp\left(\frac{-\epsilon \ln \frac{b_j}{b_{max}}}{2 \ln b_{max}}\right)}{\exp\left(\frac{-\epsilon \ln \frac{b'_j}{b_{max}}}{2 \ln b_{max}}\right)} \\ &= \prod_{j=1}^l \exp\left(\frac{\epsilon}{2 \ln b_{max}} \left(\ln \frac{b'_j}{b_{max}} - \ln \frac{b_j}{b_{max}} \right)\right) \\ &= \prod_{j=1}^l \exp\left(\frac{\epsilon}{2 \ln b_{max}} \ln \frac{b'_j}{b_j}\right) \\ &= \exp\left(\frac{\epsilon}{2 \ln b_{max}} \sum_{j=1}^l \ln \frac{b'_j}{b_j}\right) \\ &\stackrel{(a)}{\leq} \exp\left(\frac{\epsilon}{2 \ln b_{max}} \sum_{j=1}^l \left[\ln \frac{b_{max}}{b_{min}} \right]\right) \\ &= \exp\left(\frac{\epsilon l}{2}\right), \end{aligned} \quad (14)$$

in which the inequality (a) holds because of the same reason of inequality (c) in formula (10). Combining the formulas (13) and (14), the proof is completed, then the proof is completed. \square

D. Proof of Lemma 2

Proof. Assume that HERALD* selects workers for the winning set \mathcal{S} using type I selection in the following order: $\mathcal{S}_I = \{1, \dots, h\}$. Let $\tilde{\mathcal{T}}_i$ denote the set of tasks whose sensory data has not been collected just before worker i is selected. Since HERALD* carries out type I selection, $c_i \leq |\Gamma_i \cap \tilde{\mathcal{T}}_i| \frac{64\mathbb{E}[C_{OPT}(\mathcal{A}, W)]}{|\tilde{\mathcal{T}}_i|}$, where \mathcal{A} is a subset of k tasks possibly arriving simultaneously from \mathcal{T} . Hence, the social cost of workers in \mathcal{S}_I can be bounded by

$$\begin{aligned} \sum_{i \in \mathcal{S}_I} c_i &\leq \sum_{i \in \mathcal{S}_I} \frac{64|\Gamma_i \cap \tilde{\mathcal{T}}_i| \mathbb{E}[C_{OPT}(\mathcal{A}, W)]}{|\tilde{\mathcal{T}}_i|} \\ &\leq 64\mathbb{E}[C_{OPT}(\mathcal{A}, W)] \sum_{t=1}^n \frac{1}{t}, \end{aligned} \quad (15)$$

which is at most $64\mathbb{E}[C_{OPT}(\mathcal{A}, W)] \ln n$, in which Γ_i denotes the worker i 's matching task subset. Therefore, the conclusion holds due to the property of expectation. \square

E. Proof of Lemma 3

Proof. Let us recall that the set of workers with the bids in the winning set \mathcal{S} selected by HERALD* through type II selection is denoted by $\mathcal{S}_{II} = \{1, \dots, \ell\}$. Set $k_{\ell+1} = 0$ and $c_0 = 0$ for notational convenience. Let j be $k_j \geq 8 \ln 2n$ but $k_{j+1} < 8 \ln 2n$. Then, we observe that there are at most $8 \ln 2n$ tasks from $\tilde{\mathcal{T}}_{j+1}$ in expectation. As each of these tasks is executed by a worker whose cost is not greater than the one performing it in $\mathcal{S}^*(\mathcal{A}, W)$, the cost incurred by workers $j+1, \dots, \ell$ is

bounded by $8 \ln 2n \mathbb{E}[C_{OPT}(\mathcal{A}, W)]$. Therefore, the expected cost incurred by using the remaining workers $1, \dots, j$ satisfies

$$\begin{aligned}
& \sum_{i=1}^j c_i \Pr[\mathcal{A} \cap (\Gamma_i \cap \tilde{\Gamma}_i) \neq \emptyset] \leq \sum_{i=1}^j c_i \mathbb{E}[\mathcal{A} \cap (\Gamma_i \cap \tilde{\Gamma}_i)] \\
& \tilde{\Gamma}_{i+1} \subseteq \tilde{\Gamma}_i \setminus \Gamma_i \sum_{i=1}^j c_i \mathbb{E}[\mathcal{A} \cap (\tilde{\Gamma}_i \setminus \tilde{\Gamma}_{i+1})] \\
& \leq \sum_{i=1}^j c_i (k_i - k_{i+1}) \stackrel{c_0=0}{\leq} \sum_{i=1}^j k_i (c_i - c_{i-1}) \\
& \stackrel{(a)}{\leq} \sum_{i=1}^j 16 \mathbb{E}[|\mathcal{S}'(\mathcal{A}_i, W)|] \ln l(c_i - c_{i-1}) \\
& = 16 \ln l \left(c_j \mathbb{E}[|\mathcal{S}'(\mathcal{A}_{j+1}, W)|] \right. \\
& \quad \left. + \sum_{i=1}^j c_i \left(\mathbb{E}[|\mathcal{S}'(\mathcal{A}_i, W)|] - \mathbb{E}[|\mathcal{S}'(\mathcal{A}_{i+1}, W)|] \right) \right) \\
& \stackrel{(b)}{\leq} 16 \ln l \left(\mathbb{E}[C(\mathcal{S}'(\mathcal{A}_{j+1}, W))] \right. \\
& \quad \left. + \sum_{i=1}^j \left(\mathbb{E}[C(\mathcal{S}'(\mathcal{A}_i, W))] - \mathbb{E}[C(\mathcal{S}'(\mathcal{A}_{i+1}, W))] \right) \right) \\
& \leq 16 \ln l \mathbb{E}[C_{OPT}(\mathcal{A}, W)],
\end{aligned} \tag{16}$$

where inequalities (a) and (b) are based on Lemma 3.5 and Lemma 3.4 in reference [37], Γ_i denotes the matching task subset of worker i . It was previously shown that $\sum_{i=j+1}^{\ell} c_i \Pr[\mathcal{A} \cap (\Gamma_i \cap \tilde{\Gamma}_i) \neq \emptyset] \leq 8 \ln 2n \mathbb{E}[C_{OPT}(\mathcal{A}, W)]$. Therefore, the expected cost incurred by workers $1, \dots, \ell$ can be bounded as follows:

$$\sum_{i=1}^{\ell} c_i \Pr[\mathcal{A} \cap (\Gamma_i \cap \tilde{\Gamma}_i) \neq \emptyset] \leq [8 \ln 2n + 16 \ln l] \cdot \mathbb{E}[C_{OPT}(\mathcal{A}, W)]. \tag{17}$$

Then we have

$$\frac{\sum_{i=1}^{\ell} c_i \Pr[\mathcal{A} \cap (\Gamma_i \cap \tilde{\Gamma}_i) \neq \emptyset]}{\mathbb{E}[C_{OPT}(\mathcal{A}, W)]} \leq \mathcal{O}(\ln \ln n). \tag{18}$$

The above is based on one matching set \mathcal{P} among all the matching results \mathcal{F} , we need to take the expectation over all situations, i.e.,

$$\begin{aligned}
& \max_{k \in \{1, \dots, n\}} \mathbb{E}_{\mathcal{P} \in \mathcal{F}} \frac{\mathbb{E}_{\mathcal{A} \subseteq \mathcal{T}}[C(\mathcal{S}(\mathcal{A}, W))]}{\mathbb{E}_{\mathcal{A} \subseteq \mathcal{T}}[C_{OPT}(\mathcal{A}, W)]} \\
& = \sum_{j=1}^{m^l} P_j \times \frac{\sum_{i=1}^{\ell} c_i \Pr[\mathcal{A} \cap (\Gamma_i \cap \tilde{\Gamma}_i) \neq \emptyset]}{\mathbb{E}[C_{OPT}(\mathcal{A}, W)]} \\
& \leq \sum_{j=1}^{m^l} P_j \times \mathcal{O}(\ln \ln n) = \mathcal{O}(\ln \ln n),
\end{aligned} \tag{19}$$

where P_j is the probability of each task-worker matching result. This proof is completed. \square

F. Actual Running Time for HERALD*

To gain a comprehensive understanding of HERALD*'s operational efficiency, we conducted an extensive examination of its actual running times in settings 1 and 2. This process entailed rigorously running the algorithms 100 times across each setting, enabling us to calculate their average running durations. The gathered data, offering a granular view of performance, is vividly represented in Figure 9. As demonstrated in Figure 9 (a), there is a clear proportional relationship

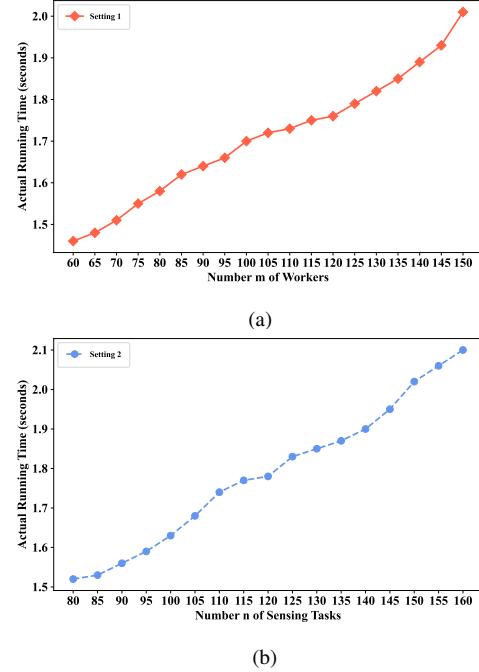


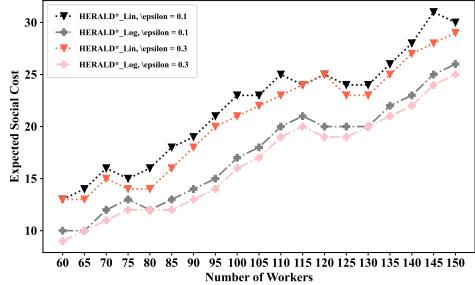
Fig. 9. (a). Actual running time under Setting 1. (b). Actual running time under Setting 2.

between actual time and the number of workers. Similarly, Figure 9 (b) indicates that the actual time corresponds to the number of sensing tasks.

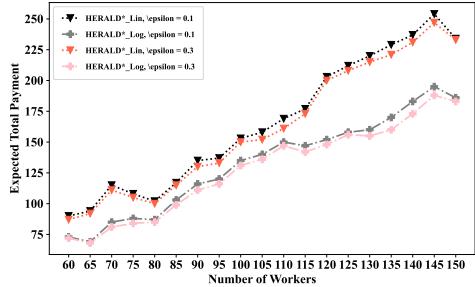
What makes these findings particularly noteworthy is their comparison with the theoretical analysis from Proposition 1. The empirical results present a significantly more efficient performance of HERALD*, with actual running times being substantially lower than what our theoretical models predicted. This disparity not only underscores the practical effectiveness of HERALD* but also points to potential areas for further optimization and refinement in our theoretical frameworks.

G. Impact of Privacy Parameters ϵ on HERALD*'s Performance

Equations (5) and (8) illustrate the probability of worker i with a bid b_i matching a task subset under linear and logarithmic scoring functions. It's evident that at a privacy parameter $\epsilon = 0$, all workers' probabilities follow a uniform distribution. As ϵ increases, the probability and the expected cost for a worker with bid b_i decrease, thereby reducing the expected social cost and expected total payment. However, this also weakens differential privacy protection. Taking Experimental Setting 1 as an example, we set the privacy parameters to $\epsilon = 0.1$ and $\epsilon = 0.3$ respectively, fix the number of sensing tasks n to 120, and change the number of workers m from 60 to 150 in steps 5 to evaluate the impact of privacy parameters on the expected social costs and expected total payments of HERALD* under a logarithmic scoring function and that of a linear scoring function. As shown in the figure 10 below, the expected social cost and expected total payment obtained by HERALD* with a larger ϵ are smaller than those obtained with a smaller ϵ , which is consistent with the above theoretical analysis.



(a)



(b)

Fig. 10. (a). Expected social cost versus different numbers of workers for uncertain tasks. (b). Expected total payment versus different numbers of workers for uncertain tasks.

Interestingly, the expected social cost and expected total payment obtained by HERALD* using the logarithmic score function are lower than those obtained using the linear score function. This is due to the logarithmic score function giving higher chances of selection to users with low bids, resulting in a preference for such users.