Unknown Worker Recruitment with Budget and Covering Constraints for Mobile Crowdsensing

Guoju Gao[†‡], Jie Wu[‡], Zhaoyang Yan[§], Mingjun Xiao[†✉], and Guoliang Chen[†]
[†]School of Computer Science and Technology / Suzhou Institute for Advanced Study,
University of Science and Technology of China, Hefei, P. R. China
[‡]Department of Computer and Information Sciences, Temple University, USA
[§]Department of Computer Science and Technology, Shandong University, Weihai, P. R. China
[✉]Correspondence to: xiaomj@ustc.edu.cn

***Abstract*—Mobile crowdsensing, through which a requester can recruit a group of crowd workers via a platform and coordinate them to perform some sensing tasks, has attracted lots of attention recently. However, most of the existing mobile crowdsensing systems assume that the qualities of workers are known in advance. Based on this assumption, they study the task assignment and worker recruitment problems. Unfortunately, the qualities of workers are generally *unknown* in reality, so the platform must find the tradeoff between exploring and exploiting the qualities by using reinforcement learning. At the same time, all sensing tasks are required to be covered in each round (covering constraint), and the requester usually has a limited budget (budget constraint). In this paper, we study how to recruit unknown workers under the budget and covering constraints so that the total expected achieved qualities can be maximized. To this end, we model the problem as a combination of a maximum weight matching problem and a special multi-armed bandit problem. We first consider that the recruitment costs of workers are homogeneous and propose a recruitment algorithm with a performance guarantee. Then, we study the heterogenous case and devise a heuristic algorithm. Finally, we demonstrate the performances of our algorithms through extensive simulations.**

***Index Terms*—Mobile crowdsensing, multi-armed bandits, maximum weight matching, budget and covering constraints.**

I. Introduction

Recent years have witnessed the increasing proliferation and popularity of smartphones in day-to-day life. These smartphones with powerful sensing, computation, communication and storage abilities can be considered as powerful mobile sensors. In order to fully utilize the resources of these smartphones, a new paradigm called mobile crowdwsensing [12, 17] has attracted lots of attention. In mobile crowdsensing, a requester can coordinate a group of mobile users carrying smartphones to perform some large-scale sensing tasks that one single user cannot deal with. Due to the unparalleled advantages, lots of crowdsensing-based applications such as traffic information collection, noise pollution collection, water pollution monitoring, urban WiFi characterization, and so on, have been fully developed [4, 8, 20, 22].

A typical mobile crowdsensing system mainly consists of three parties: a platform, requesters and crowd workers. After generating some sensing tasks, a requester would submit these tasks and his budget to the platform. Here, these sensing tasks will be performed in multiple rounds while the total recruitment cost cannot exceed the given budget (i.e., budget constraint). In each round, all tasks are required to be completed (i.e., covering constraint). For example, the requester wants to collect the traffic state in some intersections of an urban city every 10 minutes from 7:00 a.m. to 9:00 p.m., so that he can draw the real-time traffic map of this city. On the other hand, there are lots of registered crowd workers distributed in the city. Each worker can perform each task with different qualities and costs. A natural question is how to recruit suitable workers in each round so that the total achieved qualities can be maximized under the budget and covering constraints.

Most existing works [10, 20] assume that the qualities of each worker performing each task are known in advance, and then they mainly study the quality maximization or cost minimization problems under various constraints. To solve this type of optimization problems, they generally model the worker recruitment problem as a special set cover problem with various constraints. However, in real life, the qualities of workers conducting tasks are *unknown* in prior. In this situation, how to recruit suitable workers in each round to perform these tasks is quite challenging. To this end, we adopt the famous reinforcement learning technique, i.e., multi-armed bandits [3, 23], during the worker recruitment process.

In fact, only a few literatures [12, 17, 21] on mobile crowdsensing involve multi-armed bandit algorithms. Thereinto, [12] proposes a context-aware hierarchical online learning algorithm for performance maximization in mobile crowdsensing, where a worker's performance includes its acceptance rate and quality; [17] studies how to maximize the task completion in the face of uncertainty, while task arrivals are dynamic and worker reliability is unknown; [21] investigates how to select most informative contributors with unknown costs for budgeted crowdsensing, and then model the problem as a budgeted multi-armed bandit problem based on stochastic assumptions. However, the existing work [12, 17, 21] either neglects the budget constraint or considers that the crowdsensing only includes one sensing task. Different from the existing researches, we consider that mobile crowdsensing has multiple tasks and all tasks must be covered in each round. We study how to recruit suitable workers with unknown qualities in each round so that the total achieved qualities can be maximized, under the budget and covering constraints.

We summarize three main challenges as follows. First, it is

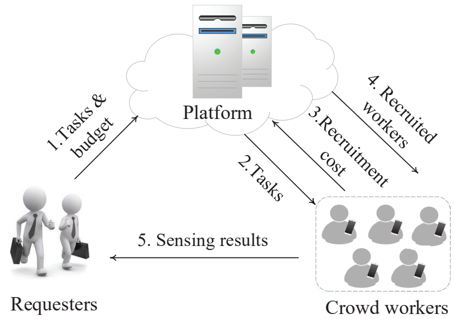


Fig. 1. Illustration of the main procedures in the mobile crowdsensing.

very hard to find the tradeoff between exploring and exploiting the qualities of workers during the recruitment process. Especially, our problem with the budget and covering constraints, which is actually a special combinational multi-armed bandit problem, differs from the general multi-armed bandit problem. Second, due to the covering constraint, our unknown worker recruitment problem can be modeled as a combination of the maximum weight matching problem and the multi-armed bandit problem. This makes it more complicated to apply the bandit algorithm in our problem directly. Third, in the mapped maximum weight matching problem, each edge (i.e., worker-task pair) involves not only the quality values but also the recruitment cost values. This also makes our problem different from the trivial maximum weight matching problem.

In this paper, we introduce and extend the Upper Confidence Bound (UCB) adopted in multi-armed bandit algorithms to estimate the qualities of workers. We first study a simple case where the recruitment costs of workers are homogeneous. For this scenario, the total number of rounds operated is determined, which means the budget constraint is removed. After combining the multi-armed bandits with maximum weight bipartite matching, we apply the maximum weight matching algorithm to maximize the total UCB-based qualities in each round. For the heterogeneous case where the costs of workers are different, we also model our problem as a combination of multi-armed bandits and maximum weight matching. Since the total recruitment costs in each round are unfixed and the workers here involve quality and cost simultaneously, the selection criterion in the heterogeneous case becomes the ratio of UCB-based quality and cost. More specifically, our major contributions are summarized as follows:

- We consider a more realistic mobile crowdsensing scenario in which the platform has no knowledge about the qualities of workers in advance, and then propose an unknown worker recruitment problem with the budget and covering constraints.
- We first study the homogeneous case where the recruitment costs of workers are uniform. By combining multi-armed bandits with maximum weight matching, we propose an efficient worker recruitment algorithm. We also analyze the worst regret guarantee in this case.
- We then investigate the heterogeneous case in which the workers involve the quality and cost simultaneously. Here, we apply and extend the maximum weight matching algorithm by using the new selection criterion in terms of the combination of UCB-based quality and cost.
- We conduct extensive simulations to evaluate the significant performances of the proposed algorithms.

The remainder of the paper is organized as follows. We first describe the crowdsensing system model and introduce the optimization problem in Section II. Then, we design two worker recruitment algorithms for homogeneous and heterogeneous cases in Section III and Section IV, respectively. In Section V, we evaluate the performances of the proposed algorithms. After reviewing the related work in Section VI, we conclude the paper in Section VII.

II. System Model & Problem

A. System Overview

Consider a typical crowdsensing system which consists of a platform, lots of registered requesters and crowd workers, and time is divided into a series of time slots (called "round") in the system. A requester first generates some crowdsensing tasks, e.g., noise pollution information collection, traffic state collection, etc, and then submits these tasks to the platform. These sensing tasks are continuous (e.g., collecting traffic state of a city from 7:00 a.m. to 9:00 p.m.) and location-related (e.g., Points of Interest). This means that the platform needs to continuously recruit workers to conduct the tasks for a period of time, and the recruited workers must go to specific locations to perform them. In fact, each worker must consume a certain cost (time, effort, etc.) when conducting sensing tasks. Hence, a worker can only perform one sensing task in each round. At the same time, in order to obtain the overall sensing effect for a requester, all sensing tasks must be conducted simultaneously in each round. In reality, covering all sensing tasks in each round is necessary. For instance, in order to construct the real-time traffic map of a city, the local government always wants to collect traffic condition in multiple locations simultaneously in each round. Moreover, a requester usually has a limited budget to complete the sensing task.

After receiving the sensing tasks and the limited budget from a requester, the platform publishes the tasks to all registered workers. Then, each worker estimates all location-related sensing tasks, and replies to the platform with the tasks he can complete, as well as his corresponding recruitment cost. Note that here the qualities of each worker performing each sensing task are *unknown*. Thus, the platform needs to learn the quality values and at the same time maximize the total achieved qualities in all rounds. The main procedures of the crowdsensing system are shown in Fig.1.

In this paper, we concentrate on the unknown worker recruitment problem with budget and covering constraints for mobile crowdsensing, where the term of "unknown" means that the platform has no knowledge about the ability (i.e., quality) of a worker performing a crowdsensing task. Actually, the unknown worker recruitment problem with the budget and covering constraints falls into the dilemma between "exploitation" and "exploration". On the one hand, the platform always wants to select the best set of workers to conduct

TABLE I
DESCRIPTION OF MAJOR NOTATIONS.

Variable	Description
$\mathcal{N}$, $\mathcal{M}$	the sets of workers and sensing tasks, respectively.
N, M	the numbers of workers and tasks, respectively.
i, j, t	the indexes for workers, tasks, and rounds (slots).
$c_{i,j}$	the cost of the worker i conducting the task j.
B	the limited budget given by the requester.
$x_{i,j,t}$	the observed quality of i conducting j in round t.
$q_{i,j}$	the mean of the distribution $\{x_{i,j,t} \vert t \geq 1\}$.
$\pi_{i,j,t}$	$\pi_{i,j,t} \in \{0,1\}$ means the recruitment decision.
(i,j)	the worker-task pair.
Φ	the recruitment strategy in all rounds.
Φ_t, C_t	the recruitment strategy and total cost in round t.
$\mathbb{E}[\cdot]$	the expected function.
$\overline{x}_{i,j}(t)$	the average quality value until the t-th round.
$n_{i,j}(t)$	total times of i performing j until the round t.
$\widehat{x}_{i,j}(t)$	the UCB-based quality in homogeneous case.
$\widetilde{x}_{i,j}(t)$	the UCB-based quality in heterogeneous case.

the tasks according to the estimated qualities so far (i.e., exploitation). Here, the quality values will be revealed after workers complete the related tasks. On the other hand, in order to seek the optimal set of workers in each round, the platform will also recruit workers to conduct different tasks so that it can discover the qualities totally (i.e., exploration). Additionally, the constraints of covering all tasks in each round make the worker recruitment problem more challenging.

In the crowdsensing system, we use t to denote the index of round, and we consider N workers and M sensing tasks in the system, denoted as $\mathcal{N} = \{1, \cdots, i, \cdots, N\}$ and $\mathcal{M} = \{1, \cdots, j, \cdots, M\}$, respectively. Furthermore, we let $c_{i,j}$ denote the cost of the worker $i \in \mathcal{N}$ conducting the task $j \in \mathcal{M}$. At the same time, the requester has a limited budget to complete these sensing tasks (i.e., recruit workers), which is denoted as B. We assume that B is enough so that it can at least cover the initializing cost, i.e., $B > \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{M}} c_{i,j}$.

In addition, we use a normalized nonnegative random variable $x_{i,j,t} \in [0, 1]$ to denote the quality of the worker i completing the task j in the t-th round. Actually, for $\forall i \in \mathcal{N}$ and $\forall j \in \mathcal{M}$, $\{x_{i,j,1}, x_{i,j,2}, \cdots, x_{i,j,t}, \cdots\}$ follows an unknown independent and identically distribution with an unknown expectation $q_{i,j}$. Also, $x_{i,j,t}$ and $x_{i',j',t'}$ are independent for $\forall i, i' \in \mathcal{N}$, $\forall j, j' \in \mathcal{M}$ and $\forall t, t' \geq 1$. Here, if the worker i is not recruited to conduct the task j in round t, we have $x_{i,j,t} = 0$. Else, the quality values $x_{i,j,t}$ will be revealed. Actually, how to estimate the qualities of each worker accurately and meanwhile to maximize the total achieved qualities of all sensing tasks under the limited budget is challenging. Especially, all sensing tasks are required to be covered in each round, making our problem more complicated.

B. Problem

In the crowdsensing system, each sensing task must be assigned to one and only one worker and meanwhile, each worker can only perform no more than one task in each round. Thus, we assume that the number of workers is larger than that of tasks, i.e., $N \geq M$. If $N < M$, we can expand the alternative worker set $\mathcal{N}$ by inviting more mobile users to participate in the crowdsensing system. When a task $j \in \mathcal{M}$ is assigned to a worker $i \in \mathcal{N}$, we say that the task is "covered", and we use the worker-task pair (i, j) to describe the relationship. In this paper, we focus on finding the optimal worker-task pairs in each round, so that all sensing tasks can be "covered" and the total qualities of all tasks can be maximized under a given budget. This problem is quite challenging because the qualities of workers are not known to the platform. Thus, the platform must find the tradeoff between the exploration (i.e., searching the optimal assignment in each round) and exploitation (i.e., utilizing the best results so far). Even though the qualities of workers are known, the recruitment problem is still difficult. This is because the worker recruitment problem involves the maximum bipartite matching problems in which the edges (i.e., worker-task pair) include two factors: cost and quality.

For the simplicity of following descriptions, we use $\Phi = \{\pi_{i,j,t} | i \in \mathcal{N}, j \in \mathcal{M}, t \geq 1\}$ to denote the recruitment strategy, in which $\pi_{i,j,t} = 1$ means that the worker i will be recruited to perform the task j in the t-th round, and otherwise $\pi_{i,j,t} = 0$. When $\pi_{i,j,t} = 1$, the worker i must perform the task j in the round t, and then gets a reward $c_{i,j}$. Only after the task j is conducted, the corresponding quality in this round (i.e., $x_{i,j,t}$) can be observed by the system. Here, we use Φ_t to denote the solution in the t-th round. In fact, Φ_t is a maximum quality bipartite matching with cost. Due to the limited budget of a requester, the recruitment process is finite, and we use T to denote the total rounds. Based on this, we have $\Phi = \cup_{t=1}^{T} \Phi_t$. Also, we let $C_t = \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{M}} c_{i,j} \cdot \pi_{i,j,t}$ denote the total recruitment cost in the t-th round.

The objective of platform is to maximize the total expected obtained qualities under the budget and covering constraints. More specifically, the problem is formalized as follows:

$$Maximize: \quad \mathbb{E}\Big[\sum_{t=1}^{T} \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{M}} \pi_{i,j,t} \cdot x_{i,j,t}\Big] \tag{1}$$

$$Subject\ to: \quad \sum_{t=1}^{T} C_t = \sum_{t=1}^{T} \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{M}} \pi_{i,j,t} \cdot c_{i,j} \leq B \tag{2}$$

$$\sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{M}} \pi_{i,j,t} = M \text{ for } \forall t \geq 1 \tag{3}$$

$$\sum_{i\in\mathcal{N}} \pi_{i,j,t} = 1, \sum_{j\in\mathcal{M}} \pi_{i,j,t} \leq 1 \text{ for } \forall t \geq 1 \tag{4}$$

$$\pi_{i,j,t} \in \{0, 1\} \text{ for } \forall i \in \mathcal{N},\ j \in \mathcal{M},\ t \geq 1 \tag{5}$$

Here, Eq. (2) and Eq. (3) mean the budget constraint and the covering constraint, respectively, while Eq. (4) indicates that in each round each task can only be assigned to exactly one worker, and each worker can conduct at most one task. Furthermore, we suppose that there exists at least one feasible solution for the optimization problem in each round. That is, all sensing tasks can be covered in each round. This is reasonable because we can expand the alternative worker set $\mathcal{N}$ by inviting more mobile users to participate in the crowdsensing system, until the solutions to the optimization problem appear. Additionally, we summarize the commonly used notations throughout the paper in Table I.

III. Homogenous Cost

We first study the homogeneous case where the recruitment cost of each worker conducting each task is uniform. In fact, the unknown worker recruitment problem here is still challenging because the budget and covering constraints make the worker selection nontrivial. We first introduce the basic solution and then present the detailed algorithm. Finally, we analyze the performance guarantee of the proposed algorithm.

A. Basic Solution

Since one worker can only perform at most one task and one task must be assigned to one worker (covering constraint) in each round, the worker recruitment can be seen as a special maximum weight bipartite matching problem in which the qualities of workers can be considered as the weights of edges. Note that the qualities of workers here are not known a priori. Moreover, due to the homogeneous recruitment cost, i.e., $c_{i,j} = c$ for $\forall i \in \mathcal{N}$ and $\forall j \in \mathcal{M}$, we get that the total recruitment cost in each round is fixed, i.e., $c \times M$. Based on this, the total rounds of the user recruitment problem is constrained by $\lfloor \frac{B}{c \times M} \rfloor$ (budget constraint).

Then, the focus of the worker recruitment problem is how to explore the qualities of workers accurately and at the same time to maximize the total expected achieved qualities in $\lfloor \frac{B}{c \times M} \rfloor$ rounds. In order to learn the qualities of workers effectively, we adopt the famous online reinforcement learning technique (i.e., multi-armed bandits [16, 19]) in this paper. However, our problem including the budget and covering constraints differs from the general (combinatorial) multi-armed bandits. Thus, the techniques of (combinatorial) multi-armed bandits [3, 5] cannot be applied in our problem directly. To this end, we will propose an effective worker recruitment algorithm in the homogeneous case, where we adopt the Upper Confidence Bound (UCB) to evaluate the qualities of workers.

More specifically, we model the unknown worker recruitment problem as a combination of the maximum weight bipartite matching problem and the special combinatorial multi-armed bandits problem. Here, the stopping round is determined in the homogeneous case, i.e., $\lfloor \frac{B}{c \times M} \rfloor$. According to this, we just study how to solve the unknown worker recruitment problem with covering constraint in each round. Since the number of tasks is less than that of workers, i.e., $M \leq N$, the recruitment algorithm (i.e., a special maximum weight matching algorithm) must output M worker-task pairs in each round. Since our problem is actually a special maximum weight bipartite matching problem, we use $\mathcal{G} = \{\mathcal{N} \cup \mathcal{M}, \mathcal{E}, \widehat{\mathcal{X}}\}$ to denote the bipartite graph where $\mathcal{N}$ and $\mathcal{M}$ mean the sets of left and right vertices, respectively, $\mathcal{E}$ indicates the set of edges (i.e., worker-task pairs) in the bipartite graph, and $\widehat{\mathcal{X}}$ represents the weights of the corresponding edges. Here, the weights of edges are defined as the UCB-based qualities of workers performing tasks, which will be presented in the next paragraph. For simplicity of following descriptions, we first let $\Phi_t = \{(i,j) | \pi_{i,j,t} = 1\}$ denote the set of the selected edges, where $\Phi_t \subseteq \mathcal{E}$. In this paper, the terms $(i,j) \in \Phi_t$ and $\pi_{i,j,t} = 1$ have the same meanings.

Algorithm 1 Recruitment Algorithm with Homogeneous Cost

Require: $\mathcal{N}$, $\mathcal{M}$, B, and $c_{i,j} = c$ for $\forall i \in \mathcal{N}$, $\forall j \in \mathcal{M}$

Ensure: $\Phi_t = \{(i,j) | \pi_{i,j,t} = 1, \forall i \in \mathcal{N}, \forall j \in \mathcal{M}\}$ for $\forall t \geq 1$.

Initialization: $T = \lfloor \frac{B}{cM} \rfloor$, $t = 0$, $Q(t) = 0$, $n_{i,j}(t) = 0$ and $\overline{x}_{i,j}(t) = 0$ for $\forall i \in \mathcal{N}$ and $\forall j \in \mathcal{M}$;

Platform builds the bipartite graph $\mathcal{G} = \{\mathcal{N} \cup \mathcal{M}, \mathcal{E}, \widehat{\mathcal{X}}\}$, where $\widehat{x}_{i,j}(t) = \overline{x}_{i,j} + \sqrt{\frac{(M+1)\ln t}{n_{i,j}(t)}} \in \widehat{\mathcal{X}}$ is initialized to 0;

while $t \leq T$ **do**

  if $n_{i,j}(t) = 0$ for $\forall (i,j) \in \mathcal{E}$ **then**

    // The platform explores the qualities of workers;

    $t \Leftarrow t + 1$;

    Obtain the matching Φ_t including (i,j) based on $\mathcal{G}$ in terms of weight value $\widehat{x}_{i,j}(t) \in \widehat{\mathcal{X}}$;

    Output the qualities $x_{i,j,t}$ for $\forall (i,j) \in \Phi_t$;

    Update the two matrixes $(n_{i,j}(t))_{N \times M} = 0$ and $(\overline{x}_{i,j}(t))_{N \times M} = 0$ according to Eqs. (6) and (7);

    $Q(t) = Q(t-1) + \sum_{(i,j) \in \Phi_t} x_{i,j,t}$;

  end if

  $t \Leftarrow t + 1$;

  Conduct the maximum weight matching algorithm [7, 9] in terms of the weight $\widehat{x}_{i,j}(t)$, and output Φ_t;

  Obtain the qualities $x_{i,j,t}$ for $\forall (i,j) \in \Phi_t$;

  Update the two matrixes $(n_{i,j}(t))_{N \times M} = 0$ and $(\overline{x}_{i,j}(t))_{N \times M} = 0$ according to Eqs. (6) and (7);

  $Q(t) = Q(t-1) + \sum_{(i,j) \in \Phi_t} x_{i,j,t}$;

end while

Output: Φ_t for $t \in [1, T]$, and $Q(T)$;

Here, we first introduce two $N \times M$ matrixes to record the average quality values (denoted as $(\overline{x}_{i,j}(t))_{N \times M}$) and the total times (denoted as $(n_{i,j}(t))_{N \times M}$) of each worker performing each task until the t-rounds, respectively. After the matching (denoted as Φ_t) is determined in round t, the two matrixes will be updated as follows.

$$\overline{x}_{i,j}(t) = \begin{cases} \frac{\overline{x}_{i,j}(t-1) \cdot n_{i,j}(t-1) + x_{i,j,t}}{n_{i,j}(t-1)+1}; & (i,j) \in \Phi_t, \\ \overline{x}_{i,j}(t-1); & (i,j) \notin \Phi_t. \end{cases} \tag{6}$$

$$n_{i,j}(t) = \begin{cases} n_{i,j}(t-1) + 1; & (i,j) \in \Phi_t, \\ n_{i,j}(t-1); & (i,j) \notin \Phi_t. \end{cases} \tag{7}$$

In order to obtain the matching result (i.e., Φ_t) which has the maximum expected total qualities in each round, we first introduce the expression of the UCB-based quality value. That is, we use $\widehat{x}_{i,j}(t)$ to denote the UCB-based quality value according to the average quality value $\overline{x}_{i,j}(t)$, that is,

$$\widehat{x}_{i,j}(t) = \overline{x}_{i,j}(t) + \sqrt{\frac{(M+1)\ln t}{n_{i,j}(t)}}. \tag{8}$$

In the modeled bipartite graph $\mathcal{G}$, we let the UCB-based quality values denote the weights of edges. Based on this, the set of weights $\widehat{\mathcal{X}}$ is determined. Then, we can adopt the maximum weight matching algorithms [7, 9] to output the selected worker-task pairs Φ_t according to the bipartite $\mathcal{G} =$

$\{\mathcal{N}\cup\mathcal{M},\mathcal{E},\widehat{\mathcal{X}}\}$. Similar to the solution in the multi-armed bandit problem, we first explore the qualities of each worker-task pair in the initial stage. After that, we estimate the UCB-based quality of each worker-task pair. Then, we focus on selecting suitable worker-task pairs in each round. Note that whether in the exploration stage or in the exploitation stage, the covering constraint must be met in each round.

When determining the selected worker-task pairs Φ_t in round t, the workers will perform the corresponding sensing tasks. Then, the quality values of workers conducting the corresponding tasks can be observed by the platform (i.e., $x_{i,j,t}$ for $(i,j)\in\Phi_t$). After that, the two $N\times M$ matrixes, i.e., $(\overline{x}_{i,j}(t))_{N\times M}$ and $(n_{i,j}(t))_{N\times M}$, will be updated according to Eqs. (6) and (7), respectively.

B. Detailed Algorithm

Based on the above solution, we present the detailed algorithm as shown in Alg. 1. More specifically, we first initialize the values t, T, $Q(t)$, $n_{i,j}(t)$, $\overline{x}_{i,j}(t)$ in Step 1. The platform builds the bipartite graph $\mathcal{G}$ and the weights are initialized to 0 in Step 2. In Steps 4-10, the platform explores the qualities of workers, that is, it conducts the matchings in multiple rounds so that all worker-task pairs will be included at least one times. The round index t, the total achieved qualities $Q(t)$, the two matrixes $(n_{i,j}(t))_{N\times M}$ and $(\overline{x}_{i,j}(t))_{N\times M}$ will be updated accordingly. After that, the platform focuses on the worker recruitment problem by using the explored qualities in Steps 12-17. Here, the maximum weight matching algorithms are adopted to select the worker-task pairs in each round. After Φ_t is determined, the values of $x_{i,j,t}$ for $\forall(i,j)\in\Phi_t$ can be observed, and then $Q(t)$, $(n_{i,j}(t))_{N\times M}$ and $(\overline{x}_{i,j}(t))_{N\times M}$ will be updated accordingly. After the budget is exhausted, i.e., $t=\lfloor\frac{B}{cM}\rfloor$, the algorithm terminates and outputs the total achieved qualities and the recruitment results in each round.

Moreover, we analyze the computation complexity of Alg. 1 here. First, the algorithm contains $T=\lfloor\frac{B}{cM}\rfloor$ rounds. Second, in each round, the computation complexity is dominated by the maximum weight matching algorithm in Steps 7 and 13, which is denoted as $O(N^2M^2)$. By combining this, we get that the overall computation overhead is $O(\lfloor\frac{B}{cM}\rfloor N^2M^2)$.

C. Performance Analysis

In this section, we analyze the performance guarantee of Alg. 1. For the bipartite graph $\mathcal{G}=\{\mathcal{N}\cup\mathcal{M},\mathcal{E},\widehat{\mathcal{X}}\}$, we let Π denote the set of all matchings. Here, $\forall\Phi\in\Pi$ is an $M\times M$ matching and (i,j), a worker-task pair, is an element of a matching Φ. Among all matchings, we first identify the optimal matching, denoted as Φ^*. Based on the expected means of quality distribution and the covering budget, we have

$$\Phi^*=\text{argmax}_{\Phi\in\Pi}(\textstyle\sum_{(i,j)\in\Phi}q_{i,j}). \tag{9}$$

Note that in this paper we always let * denote the corresponding identification of the optimal matching. For simplicity, we define Δ_k as the total quality difference between the optimal matching Φ^* and one matching $\Phi_k\in\Pi$, i.e.,

$$\Delta_k=\textstyle\sum_{(i,j)\in\Phi^*}q_{i,j}-\sum_{(i,j)\in\Phi_k}q_{i,j}. \tag{10}$$

At the same time, we have $\Delta_{min}=\text{argmin}_{\Phi_k\in\Pi}\Delta_k$ and $\Delta_{max}=\text{argmax}_{\Phi_k\in\Pi}\Delta_k$. Here, we also let k denote the corresponding identification of a non-optimal matching.

Then, we analyze the regret of our proposed algorithm, denoted as $R(B)$ in which B is the limited budget. Here, the regret means the difference of total achieved qualities between the optimal matching and the obtained matching of our algorithm in each round, that is,

$$R(B)=\lfloor\tfrac{B}{cM}\rfloor\textstyle\sum_{(i,j)\in\Phi^*}q_{i,j}-\mathbb{E}[\sum_{t=1}^{\lfloor\frac{B}{cM}\rfloor}\sum_{(i,j)\in\Phi_t}x_{i,j,t}]. \tag{11}$$

We show that the proposed algorithm can achieve a $O(\ln(B))$ worst-case regret on average. More specifically, we have the following theorem:

***Theorem** 1:* The regret of the proposed algorithm is at most

$$R(B)\le\varphi_1\ln(\frac{B}{cM})+\varphi_2, \tag{12}$$

where $\begin{cases}\varphi_1=\dfrac{4\Delta_{max}M^3(M+1)N}{\Delta_{min}^2}\\ \varphi_2=MN\Delta_{max}\Big(1+\dfrac{M\pi^2}{3}-\dfrac{2M}{(\frac{B}{cM+1})^2}\Big)\end{cases}$ are two constants.

Before proving the worst-case regret bound, we first introduce the concept of Chernoff-Hoeffding inequality in the following lemma.

***Lemma** 1: Chernoff-Hoeffding bound [1].* Suppose $X_1,X_2,\cdots,X_n$ are independent random variables whose values lie in $[0,1]$ and whose expected value is μ, i.e., $\mathbb{E}[X_z|X_1,\cdots,X_{z-1}]=\mu$ for $\forall z\in[1,n]$. Let $X=\frac{1}{n}\sum_{z=1}^{n}X_z$. The Chernoff-Hoeffding inequality gives an exponential bound on the probability that the value of X deviates from its mean μ. That is:

$$\mathbb{P}\Big\{X\ge\mu+\theta\Big\}\le e^{-2n\theta^2},\text{ and }\mathbb{P}\Big\{X\le\mu-\theta\Big\}\le e^{-2n\theta^2}. \tag{13}$$

We here let $\theta_{n_{i,j}(t),t}=\sqrt{(M+1)\ln t/n_{i,j}(t)}$ for simplicity. Also, we introduce $T_{i,j}(t)$ as a counter after the initialization period, which is updated as follows. In each round (e.g., t) after the initialization period, one of the following cases must happen: 1) the optimal matching is played; 2) a non-optimal matching is selected. In the former, $T_{i,j}(t)$ will not change; in the latter, we denote the non-optimal matching as Φ_k. Then, there must exist one edge $(i,j)\in\Phi_k$ such that $(i,j)=\text{argmin}_{(i',j')\in\Phi_k}n_{i',j'}(t)$. Based on this, the corresponding value of $T_{i,j}(t)$ is increased by 1. Here, if there are multiple such edges, we arbitrarily choose one. Since exactly one element in $(T_{i,j}(t))_{N\times M}$ is increased by 1 when a non-optimal matching is selected, the total number of non-optimal matching is equal to the summation of the values in $(T_{i,j}(t))_{N\times M}$. According to this, we get

$$\textstyle\sum_{\Phi_k\ne\Phi^*}T_k(t)=\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}}T_{i,j}(t), \tag{14}$$

and further have

$$\textstyle\sum_{\Phi_k\ne\Phi^*}\mathbb{E}[T_k(t)]=\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}}\mathbb{E}[T_{i,j}(t)]. \tag{15}$$

At the same time, we have $T_{i,j}(t)\le n_{i,j}(t)$ for $\forall i\in\mathcal{N}$ and $\forall j\in\mathcal{M}$. Moreover, we let $I_{i,j}(t)$ denote the indicator function in round t where $I_{i,j}(t)=1$ means that $T_{i,j}(t)$ is increased by 1, and let l denote an arbitrary positive integer. Then, we have

$$T_{i,j}(t)=\sum_{t=x+1}^{T}\Big\{I_{i,j}(t)\Big\}\le l+\sum_{t=x+1}^{T}\Big\{I_{i,j}(t),T_{i,j}(t-1)\ge l\Big\},\quad(16)$$

where $T=\lfloor\frac{B}{cM}\rfloor$ and x means the number of initialization period. We further get $x=\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}}n_{i,j}(t)$ in which all elements in $(n_{i,j}(t))_{N\times M}$ just change to a non-zero state. When $I_{i,j}(t)=1$, a non-optimal matching, denoted as $\Phi_t\neq\Phi^*$, has been selected in round t. Next, we get

$$\begin{aligned}T_{i,j}(t)\le l+\textstyle\sum_{t=x+1}^{T}\Big\{\sum_{(i,j)\in\Phi^*}\widehat{x}_{i,j}(t-1)\\ \le\textstyle\sum_{(i,j)\in\Phi_t}\widehat{x}_{i,j}(t-1),T_{i,j}(t-1)\ge l\Big\},\quad(17)\\ \le l+\sum_{t=x}^{T}\Big\{\sum_{(i,j)\in\Phi^*}\widehat{x}_{i,j}(t)\le\sum_{(i,j)\in\Phi_{t+1}}\widehat{x}_{i,j}(t),T_{i,j}(t)\ge l\Big\},\quad(18)\end{aligned}$$

in which $\widehat{x}_{i,j}(t)=\overline{x}_{i,j}(t)+\theta_{n_{i,j}(t),t}$. Then, we have

$$\begin{aligned}T_{i,j}(t)\le l+\sum_{t=x}^{T}\Big\{\min_{0<n^*_{i_1,j_1},\cdots,n^*_{i_M,j_M}\le t}\sum_{(i,j)\in\Phi^*}\widehat{x}_{i,j}(n^*_{i,j})\\ \le\max_{l\le n^{\Phi_{t+1}}_{i_1,j_1},\cdots,n^{\Phi_{t+1}}_{i_M,j_M}\le t}\sum_{(i,j)\in\Phi_{t+1}}\widehat{x}_{i,j}(n^{\Phi_{t+1}}_{i,j})\Big\}\quad(19)\\ \le l+\sum_{t=1}^{T}\sum_{n^*_{i_1,j_1}=1}^{t}\cdots\sum_{n^*_{i_M,j_M}=1}^{t}\sum_{n^{\Phi_{t+1}}_{i_1,j_1}=l}^{t}\cdots\sum_{n^{\Phi_{t+1}}_{i_M,j_M}=l}^{t}\\ \Big\{\sum_{(i,j)\in\Phi^*}\widehat{x}_{i,j}(n^*_{i,j})\le\sum_{(i,j)\in\Phi_{t+1}}\widehat{x}_{i,j}(n^{\Phi_{t+1}}_{i,j})\Big\},\quad(20)\end{aligned}$$

where $n^*_{i_1,j_1}$ and $n^{\Phi_{t+1}}_{i_1,j_1}$ mean the first element in the optimal matching Φ^* and the matching Φ_{t+1}, respectively.

Next, according to the proof in [1, 5, 16], we conclude that at least one of three cases must be true. The detailed expression is similar to [1, 16], so we will not present it here. After applying the Chernoff-Hoeffding bound stated in Lemma 1, we find that the upper bound of the probability for one subcase of the inequalities is less than $t^{-2(M+1)}$. Thus, the ultimate upper bound for the inequalities is at most $Mt^{-2(M+1)}$. Note that for $l\ge\lceil\frac{4(M+1)\ln(t)}{(\Delta_k(t)/M)^2}\rceil$, we get that one of the three cases is false. Based on this, we have

$$\begin{aligned}T_{i,j}(t)\le\lceil\tfrac{4(M+1)\ln t}{(\Delta_k(t)/M)^2}\rceil+\\ \sum_{t=1}^{T}\sum_{n^*_{i_1,j_1}=1}^{t}\cdots\sum_{n^*_{i_M,j_M}=1}^{t}\sum_{n^{\Phi_{t+1}}_{i_1,j_1}=l}^{t}\cdots\sum_{n^{\Phi_{t+1}}_{i_M,j_M}=l}^{t}(2Mt^{-2(M+1)})\quad(21)\\ \le\lceil\tfrac{4M^2(M+1)\ln t}{\Delta^2_{min}}\rceil+M\sum_{t=1}^{T}\Big(2t^{-2}\Big)\quad(22)\\ \le\tfrac{4M^2(M+1)\ln t}{\Delta^2_{min}}+1+M\tfrac{\pi^2}{3}-\tfrac{2M}{(\frac{B}{cM}+1)^2}\quad(23)\end{aligned}$$

Then, we can derive the regret bound as follows.

$$\begin{aligned}R(B)=\textstyle\sum_{\Phi_k\neq\Phi}\Big(\Delta_k\mathbb{E}[T_k(\lfloor\frac{B}{cM}\rfloor)]\Big)\quad(24)\\ \le\Delta_{max}\textstyle\sum_{\Phi_k\neq\Phi}\mathbb{E}[T_k(\lfloor\frac{B}{cM}\rfloor)]\quad(25)\\ \le\Delta_{max}\textstyle\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}}\mathbb{E}[T_{i,j}(\lfloor\frac{B}{cM}\rfloor)]\quad(26)\\ \le\Delta_{max}\Big(\tfrac{4M^3N(M+1)\ln\frac{B}{cM}}{\Delta^2_{min}}+MN(1+M\tfrac{\pi^2}{3}-\tfrac{2M}{(\frac{B}{cM}+1)^2})\Big)\quad(27)\end{aligned}$$

Algorithm 2 Recruitment Algorithm with Heterogeneous Cost

Require: $\mathcal{N}$, $\mathcal{M}$, B, α, and $c_{i,j}$ for $\forall i\in\mathcal{N}$, $\forall j\in\mathcal{M}$
Ensure: $\Phi_t=\{(i,j)|\pi_{i,j,t}=1,\forall i\in\mathcal{N},\forall j\in\mathcal{M}\}$ for $\forall t\ge 1$.

- **Initialization**: $t=0$, $B_t=B$, $Q(t)=0$, $n_{i,j}(t)=0$ and $\overline{x}_{i,j}(t)=0$ for $\forall i\in\mathcal{N}$, $\forall j\in\mathcal{M}$;
- Platform builds the bipartite graph $\mathcal{G}=\{\mathcal{N}\cup\mathcal{M},\mathcal{E},\widetilde{\mathcal{X}},\mathcal{C}\}$, where $\widetilde{x}_{i,j}(t)=\overline{x}_{i,j}(t)+\sqrt{\frac{\alpha\ln t}{n_{i,j}(t)}}\in\widetilde{\mathcal{X}}$ is initialized to 0;
- Platform obtains the matching with minimum cost (i.e., Φ_{min}), so that $C_{min}=\sum_{(i,j)\in\Phi_{min}}c_{i,j}\le C_k$ for $\forall\Phi_k\in\Pi$;
- **while** $B_t\ge C_{min}$ **do**
  - **if** $n_{i,j}(t)=0$ for $\forall(i,j)\in\mathcal{E}$ **then**
    - $t\Leftarrow t+1$;
    - Obtain the matching (i.e., Φ_t) including (i,j) based on $\mathcal{G}$ in terms of the weight $\widetilde{x}_{i,j}(t)\in\widetilde{\mathcal{X}}$;
    - Observe the qualities $x_{i,j,t}$ for $\forall(i,j)\in\Phi_t$;
    - Update the two matrixes $(n_{i,j}(t))_{N\times M}=0$ and $(\overline{x}_{i,j}(t))_{N\times M}=0$ according to Eqs. (6) and (7);
    - $Q(t)=Q(t-1)+\sum_{(i,j)\in\Phi_t}x_{i,j,t}$;
    - $B_t=B_{t-1}-\sum_{(i,j)\in\Phi_t}c_{i,j}$;
  - **end if**
  - $t\Leftarrow t+1$;
  - Conduct the maximum weight matching algorithm [7, 9] in terms of the criterion $\frac{\widetilde{x}_{i,j}(t)}{c_{i,j}}$, and output Φ_t;
  - Obtain the qualities $x_{i,j,t}$ for $\forall(i,j)\in\Phi_t$;
  - Update the two matrixes $(n_{i,j}(t))_{N\times M}=0$ and $(\overline{x}_{i,j}(t))_{N\times M}=0$ according to Eqs. (6) and (7);
  - $Q(t)=Q(t-1)+\sum_{(i,j)\in\Phi_t}x_{i,j,t}$;
  - $B_t=B_{t-1}-\sum_{(i,j)\in\Phi_t}c_{i,j}$;
- **end while**
- **Output**: Φ_t for $t\ge 1$, and $Q(t)$;

The theorem holds. □

IV. Heterogenous Cost

In this section, we consider the general case where the recruitment costs of workers are heterogeneous. To address this issue, we first introduce our basic idea and then present the algorithm in detail.

A. Basic Solution

For the homogeneous case, we model the worker recruitment problem as a combination of the combinatorial multi-armed bandit problem and the maximum weight bipartite matching problem. In the heterogeneous case, we also use the special maximum weight bipartite matching problem to model our problem. The difference is that the edges in the bipartite graph contain not only the weight (i.e., the unknown quality) but also the cost. We let $\mathcal{G}=\{\mathcal{N}\cup\mathcal{M},\mathcal{E},\widetilde{\mathcal{X}},\mathcal{C}\}$ denote the special bipartite graph, in which $\mathcal{N}$ (the set of user) and $\mathcal{M}$ (the set of tasks) mean the set of vertices on two sides of the bipartite graph, $\mathcal{E}=\{(i,j)|i\in\mathcal{N},j\in\mathcal{M}\}$ is the set of edges, $\widetilde{\mathcal{X}}=\{\widetilde{x}_{i,j}(t)|(i,j)\in\mathcal{E}\}$ indicates the set of edge weights, i.e., the special UCB-based quality values that will be expressed in the following, and $\mathcal{C}=\{c_{i,j}|(i,j)\in\mathcal{E}\}$ represents the recruitment cost. Actually, even if the edge weights are known in advance, the special maximum weight matching problem is quite complicated, because it involves weight and cost simultaneously.

To solve this problem, we refer to the solution of the homogeneous case. More specifically, we let $(\overline{x}_{i,j}(t))_{N\times M}$ and $(n_{i,j}(t))_{N\times M}$ denote the average qualities of workers and the selected number until the t-th round, respectively. Then, we use $\widetilde{x}_{i,j}(t)=\overline{x}_{i,j}(t)+\sqrt{\frac{\alpha \ln t}{n_{i,j}(t)}}$ to denote the special UCB-based quality values where α is the empirical parameter. We will discover the effects of the parameter α in the simulations. Similar to the solution to the homogeneous case, we also apply the maximum weight matching algorithm [7, 9] in the heterogeneous case. The difference lies in that the selection criteria is changed from $\widehat{x}_{i,j}(t)$ to $\frac{\widetilde{x}_{i,j}(t)}{c_{i,j}}$. That is, the edge weight in the maximum weight matching problem becomes $\frac{\widetilde{x}_{i,j}(t)}{c_{i,j}}$. The heterogeneous case is more challenging than the homogeneous case because the total recruitment cost in each round is different. This means that the total rounds are not determined. Also, we use $\Phi_t=\{(i,j)|\pi_{i,j,t}=1\}$ to denote the solution outputted in round t. After the recruitment result (i.e., Φ_t) in each round is determined, the update of $(\overline{x}_{i,j}(t))_{N\times M}$ and $(n_{i,j}(t))_{N\times M}$ will follow Eq. (6) and Eq. (7), respectively. Based on the presented solution, we propose a new unknown worker recruitment algorithm for the heterogeneous case.

B. Detailed Algorithm

We introduce the detailed worker recruitment algorithm for the heterogeneous case (i.e., Alg. 2). In Step 1, the algorithm initializes the values t, B_t, $Q(t)$ and the two matrixes $(n_{i,j}(t))_{N\times M}$ $(\overline{x}_{i,j}(t))_{N\times M}$. In Step 2, the platform builds the bipartite graph $\mathcal{G}$ in which the edges involve the weights $\widehat{\mathcal{X}}$ and cost $\mathcal{C}$ simultaneously. The weights are denoted by the special UCB-based quality values. That is, for $\widetilde{x}_{i,j}(t)\in\widetilde{\mathcal{X}}$, we have $\widetilde{x}_{i,j}(t) = x_{i,j}(t)+\sqrt{\frac{\alpha \ln t}{n_{i,j}(t)}}$. Here, the values of $\widetilde{\mathcal{X}}$ are updated in each round, and they are initialized to 0. Then, the platform first computes the matching (i.e., Φ_{min}) which has the minimum recruitment cost, in Step 3. When the remaining budget B_t is less than $C_{min}=\sum_{(i,j)\in\Phi_{min}} c_{i,j}$, the recruitment process will terminate, in Step 4. In Steps 5-12, the platform focuses on exploring the qualities of workers. More specifically, when the number of some edges is 0, the matching Φ_t including these edges will be outputted in this round. After observing the values of qualities in this round, i.e., $x_{i,j,t}$ for $(i,j)\in\Phi_t$, the corresponding values in the matrixes $(n_{i,j}(t))_{N\times M}$ and $(\overline{x}_{i,j}(t))_{N\times M}$ will be updated according to Eqs. (6) and (7), in Steps 8-9. Then, the total achieved qualities $Q(t)$ and the remaining budget B_t will be updated in Steps 10-11.

In Steps 13-18, the algorithm mainly exploits the obtained qualities so that the total qualities can be maximized under the remaining budget constraint. More specifically, the platform will conduct the maximum weight matching algorithm [7, 9] according to the criterion $\frac{\widetilde{x}_{i,j}(t)}{c_{i,j}}=\frac{\overline{x}_{i,j}(t)+\sqrt{\alpha \ln t)/n_{i,j}(t)}}{c_{i,j}}$ in Step 14. After getting the recruitment result Φ_t in this round, the recruited workers will go to perform the corresponding tasks. Thus, the platform can observe the relevant qualities $x_{i,j,t}$ for $(i,j)\in\Phi_t$ in Step 15. Same as the procedure of the exploration, the matrixes $(n_{i,j}(t))_{N\times M}$ and $(\overline{x}_{i,j}(t))_{N\times M}$ will be updated in Step 16. After that, the function value of $Q(t)$ and B_t will also be updated in Steps 17-18. At last, when the remaining budget is exhausted, i.e., $B_t<C_{min}$, the algorithm will terminate and output the recruitment results Φ_t for $t\geq 1$ and the achieved total qualities $Q(t)$ in Step 20.

TABLE II
SIMULATION SETTINGS.

Parameter name	default	range
the number of tasks/workers, $N=M$	20	$10-50$
the budget, B	10000	$500-100000$
the recruitment cost, $c_{i,j}$	1	$0-5$
the parameter, α	1	$0.5-10$
the qualities of workers, $x_{i,j,t}$	Gaussian	$0-1$
the mean of Gaussian, $q_{i,j}$	uniform	$0-1$

In addition, we analyze the computation complexity of Alg. 2. Since the budget B is limited, the worker recruitment process is finite. That is, the maximum recruitment round is $\frac{B}{C_{min}}$ where Φ_{min} indicates the matching which has the minimum cost. In each round (i.e., $B_t\geq C_{min}$), the computational overhead is dominated by the maximum weight matching algorithm in Steps 7 and 14, which is $O(N^2M^2)$. Therefore, the unknown recruitment algorithm will result in $O(\frac{B}{C_{min}}N^2M^2)$ computation complexity.

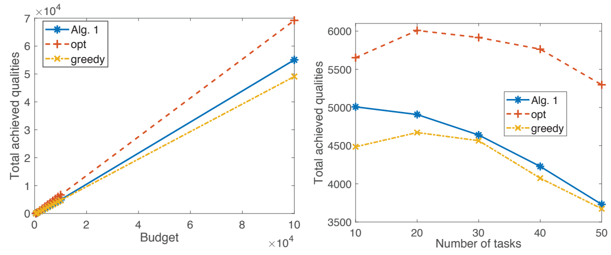
V. PERFORMANCE EVALUATION

In this section, we evaluate the performances of the proposed algorithms with extensive simulations. We conduct the simulations on a computer with Inter(R) Core(TM) i7-8700 CPU @3.20GHz and 32GB RAM under a Windows platform. Moreover, all simulations are implemented in Matlab.

A. Evaluation Methodology

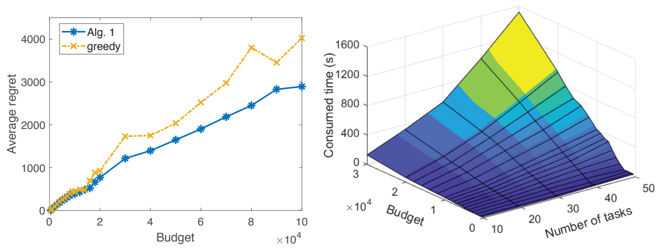
Simulation Settings: Since the number of workers (i.e., N) is required to be not less than that of sensing tasks (i.e., M), we let $N=M$ for simplicity. Moreover, the value of N is selected from $\{10,20,30,40,50\}$. Then, we set the parameters of each worker-task pair (i,j) for $i\in\mathcal{N}$ and $j\in\mathcal{M}$. We let the quality of each worker-task pair (i.e., $x_{i,j,t}$) be randomly sampled from a Gaussian distribution. Here, the Gaussian distribution is truncated to the interval $(0,1]$. Moreover, the expected mean $q_{i,j}$ and the variance of the Gaussian distribution are randomly sampled from the uniform distribution $(0,1)$. Note that the expected mean $q_{i,j}$ needs to be recorded to calculate the optimal recruitment solutions in the homogeneous case. Furthermore, we let the recruitment cost of each worker conducting each task (i.e., $c_{i,j}$) be randomly sampled from the uniform distribution on $(0,5)$. In the homogeneous case, the values of all recruitment cost are set as 1, i.e., $c_{i,j}=1$ for $i\in\mathcal{N}$ and $j\in\mathcal{M}$. Also, we select the requester's budget B from the range $[500,10^5]$. The parameter α is selected from the set $\{0.5,1,2,5,10\}$. The default and range for some parameters are displayed in Table II.

Compared Algorithms: In order to compare our proposed worker recruitment algorithms with others, we design a few



(a) Total quality vs. Budget (b) Total quality vs. Num. of Tasks

Fig. 2. Performances on the total achieved qualities (homogeneous case).

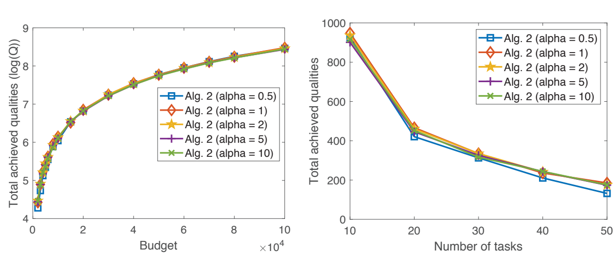


(a) Average regret vs. Budget (b) Consumed time

Fig. 3. Performance on regret and time (homogeneous case).

compared algorithms. Since our proposed algorithms are suitable for the homogeneous and heterogeneous cases, respectively, we divide the compared algorithms into two parts. First, for the homogeneous scenario where $c_{i,j} = 1$, we first design an optimal algorithm, called *opt*, in which the expected mean $q_{i,j}$ is assumed to be known in advance. Thus, *opt* always outputs the worker-task pairs whose total expected mean $q_{i,j}$ is maximum based on the maximum weight matching algorithm in each round. In addition, we also devise a *greedy* algorithm which always selects the worker-task pairs with the maximum UCB-based qualities (i.e., $\widehat{x}_{i,j}(t)$) in turn, until the covering constraint is satisfied. Second, for the heterogeneous scenario, since the worker-task pair involves not only the qualities but also the recruitment costs, the *opt* algorithm designed above cannot be applied here. We evaluate the performances of Alg. 2 by controlling the parameter α.

On the other hand, we track three performance metrics in the evaluations: the accumulative qualities, the average regret and the consumed time. The accumulative qualities mean the total achieved qualities (i.e., $Q(t)$ in the algorithm) when the budget is exhausted. The average regret is the value of the total regret divided by the value $\log(B)$ where B means the budget. Here, since there exists no optimal solution in the heterogeneous case, the metric of average regret can only be adopted in the homogeneous case.



(a) Total quality vs. Budget (b) Total quality vs. Num. of Tasks

Fig. 4. Performance on total achieved qualities (heterogenous case).

B. Evaluation Results

First, we display the evaluation results in the homogeneous settings. We evaluate the performances of accumulative qualities in the homogeneous case (i.e., Alg. 1), and show the simulation results in Fig. 2. We see that our algorithm outperforms the greedy algorithm at any time. When the budget increases, the total achieved qualities in all algorithms will increase accordingly, as shown in Fig. 2(a). Also, we validate the performances by changing the number of sensing tasks, and further get that our algorithm is more effective than greedy. When the number of tasks increases and the budget is fixed (10^4 here), the total qualities overall show a downward trend in Fig. 2(b). This is because that the exploration phase will cost a lot when the number of tasks increases.

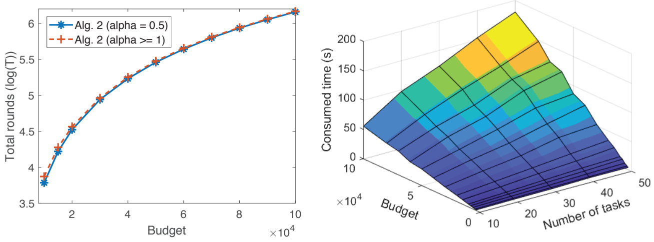
Furthermore, we evaluate the metric of average regret in the homogeneous case and present the results in Fig. 3(a). Since this metric is based on the comparison between the optimal algorithm and other algorithms, the performance of *opt* is not shown here. We then find that our algorithm has less average regret than the *greedy* algorithm, and when the budget rises, the average regret of all algorithms will increase. Additionally, we present the performances of Alg. 1 in terms of computation overhead, as shown in Fig. 3(b). The maximum weight matching algorithm included in our algorithm leads to the relatively high computation overhead. These observations exactly validate our theoretical analysis results.

Second, we demonstrate the simulation results in the heterogeneous settings. When the budget and the number of tasks change, we show the simulation results in Fig. 4. We evaluate our algorithm by controlling the parameter α in the UCB-based quality (i.e., $\widetilde{x}_{i,j}(t) = \overline{x}_{i,j}(t) + \sqrt{\frac{\alpha \ln t}{n_{i,j}(t)}}$). We get that the total achieved qualities have slight discrepancy in terms of α. In addition, we also present the total rounds and the consumed time in the heterogeneous case in Fig. 5. The total rounds of our algorithm under different values of α are almost the same. Here, the consumed time of Alg. 1 is less than that of Alg. 2. This is because the average cost in the heterogenous setting is larger than 1 in the homogeneous setting. These observations still remain consistent with our theoretical analysis results.

VI. Related Work

In this paper, we study the unknown worker recruitment problem with budget and covering constraints for mobile crowdsensing. So far, there have been lots of researches on the worker recruitment problem in crowdsensing, such as [6, 10, 11, 18, 20, 21]. However, most of the existing works assume that the qualities of workers are known in advance, and then focus on the quality maximization or cost minimization problems under various constraints. To solve this type of optimization problems, they generally model the worker recruitment problem as a special set cover problem with various constraints. Unfortunately, the qualities of workers conducting tasks are generally unknown in real life. Thus, how to solve this unknown worker recruitment problem is more practical.

In fact, only a few researches [12, 13, 17, 21] consider that the qualities of workers are unknown in mobile crowdsensing.



(a) Total rounds vs. Budget (b) Consumed time

Fig. 5. Performance on rounds and time (heterogeneous case).

However, they either neglect the budget/covering constraints or consider that the crowdsensing only includes one sensing task. Different from the existing works, we consider that crowdsensing has multiple tasks and all tasks must be covered in each round. We study how to recruit suitable workers with unknown qualities in each round so that the total achieved qualities can be maximized, under the budget and covering constraints. We model our problem as a combination of maximum weight matching and multi-armed bandits. The algorithms for multi-armed bandit [1, 2, 14–16, 19, 23] cannot be applied in our problem directly due to the budget and covering constraints.

Actually, the most related works are [3, 5], in which the concept of combinatorial multi-armed bandit is proposed. The authors in [3] design an algorithm that achieves $O(\log n)$ distribution-dependent regret where n is the number of rounds played, based on an offline (α, β)-approximation oracle. The authors in [5] propose an algorithm, which can achieve regret that grows logarithmically with time and polynomially in the number of unknown variables. This algorithm in [5] only requires linear storage and polynomial computation. Nevertheless, both of them ignore the budget constraints.

VII. Conclusion

In this paper, we study the unknown worker recruitment problem under budget and covering constraints for mobile crowdsensing. We first consider a homogeneous case where the recruitment costs of workers are uniform. By modeling our homogeneous problem as a combination of a multi-armed bandit and a special maximum weight matching, we propose an efficient worker recruitment algorithm which can achieve $O(\ln(\frac{B}{cM}))$ regret where B, c and M mean the budget, uniform cost, and the number of tasks. Second, we focus on the heterogeneous case in which the recruitment costs of worker are different. We also use a special maximum weight matching problem to describe our heterogenous problem. By replacing the UCB-based qualities in the homogeneous case with the ratio of the special UCB-based qualities and the cost, we devise a new worker recruitment algorithm. At last, extensive simulations are conducted to verify the significant performances of the proposed algorithms.

Acknowledgments

This research was supported by the National Natural Science Foundation of China (NSFC) (Grant No. 61872330, 61572457, 61379132, U1709217, 61303206, 61572342, 61502261), NSF grants CNS 1824440, CNS 1828363, CNS 1757533, CNS 1618398, CNS 1651947, CNS 1564128, the NSF of Jiangsu Province in China (Grant No. BK20191194, BK20131174, BK2009150), and Anhui Initiative in Quantum Information Technologies (Grant No. AHY150300). Additionally, the support provided by China Scholarship Council (CSC) during a visit of 'Guoju Gao' to Temple University (Grant CSC No. 201806340014) is acknowledged.

References

[1] P. Auer, N. Cesa-Bianchi, and P. Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3):235–256, 2002.
[2] K. Cai, X. Liu, Y.-Z. J. Chen, and J. C. Lui. An online learning approach to network application optimization with guarantee. In *IEEE INFOCOM*, 2018.
[3] W. Chen, Y. Wang, Y. Yuan, and Q. Wang. Combinatorial multi-armed bandit and its extension to probabilistically triggered arms. *The Journal of Machine Learning Research*, 17(1):1746–1778, 2016.
[4] Y. Duan and J. Wu. Optimizing the crowdsourcing-based bike station rebalancing scheme. In *IEEE ICDCS*, 2019.
[5] Y. Gai, B. Krishnamachari, and R. Jain. Combinatorial network optimization with unknown variables: Multi-armed bandits with linear rewards and individual observations. *IEEE/ACM Transactions on Networking*, 20(5):1466–1478, 2012.
[6] G. Gao, M. Xiao, J. Wu, L. Huang, and C. Hu. Truthful incentive mechanism for nondeterministic crowdsensing with vehicles. *IEEE Transactions on Mobile Computing*, 17(12):2982–2997, 2018.
[7] G. Gao, M. Xiao, and Z. Zhao. Optimal multi-taxi dispatch for mobile taxi-hailing systems. In *IEEE ICPP*, 2016.
[8] K. Han, C. Zhang, and J. Luo. Taming the uncertainty: Budget limited robust crowdsensing through online learning. *IEEE Transactions on Networking*, 24(3):1462–1475, 2016.
[9] J. E. Hopcroft and R. M. Karp. An n^5/2 algorithm for maximum matchings in bipartite graphs. *SIAM Journal on computing*, 2(4):225–231, 1973.
[10] M. Karaliopoulos, O. Telelis, and I. Koutsopoulos. User recruitment for mobile crowdsensing over opportunistic networks. In *IEEE INFOCOM*, 2015.
[11] W. Li, F. Li, K. Sharif, and Y. Wang. When user interest meets data quality: A novel user filter scheme for mobile crowd sensing. In *IEEE ICPADS*, 2017.
[12] S. K. née Müller, C. Tekin, M. van der Schaar, and A. Klein. Context-aware hierarchical online learning for performance maximization in mobile crowdsourcing. *IEEE/ACM Transactions on Networking*, 26(3):1334–1347, 2018.
[13] A. Singla and A. Krause. Truthful incentives in crowdsourcing tasks using regret minimization mechanisms. In *ACM WWW*, 2013.
[14] S. Tang, Y. Zhou, K. Han, Z. Zhang, J. Yuan, and W. Wu. Networked stochastic multi-armed bandits with combinatorial strategies. In *IEEE ICDCS*, 2017.
[15] L. Tran-Thanh, A. Chapman, E. M. de Cote, A. Rogers, and N. R. Jennings. Epsilon–first policies for budget–limited multi-armed bandits. In *AAAI*, 2010.
[16] L. Tran-Thanh, A. Chapman, A. Rogers, and N. R. Jennings. Knapsack based optimal policies for budget-limited multi-armed bandits. In *AAAI*, 2012.
[17] U. ul Hassan and E. Curry. Efficient task assignment for spatial crowdsourcing: A combinatorial fractional optimization approach with semi-bandit learning. *Expert Systems with Applications*, 58:36–56, 2016.
[18] E. Wang, Y. Yang, J. Wu, W. Liu, and X. Wang. An efficient prediction-based user recruitment for mobile crowdsensing. *IEEE Transactions on Mobile Computing*, 17(1):16–28, 2017.
[19] Y. Xia, T. Qin, W. Ma, N. Yu, and T.-Y. Liu. Budgeted multi-armed bandits with multiple plays. In *IJCAI*, 2016.
[20] M. Xiao, J. Wu, S. Zhang, and J. Yu. Secret-sharing-based secure user recruitment protocol for mobile crowdsensing. In *IEEE INFOCOM*, 2017.
[21] S. Yang, F. Wu, S. Tang, T. Luo, X. Gao, L. Kong, and G. Chen. Selecting most informative contributors with unknown costs for budgeted crowdsensing. In *IEEE/ACM IWQoS*, 2016.
[22] X. Zhang, Z. Yang, W. Sun, Y. Liu, S. Tang, K. Xing, and X. Mao. Incentives for mobile crowd sensing: A survey. *IEEE Communications Surveys & Tutorials*, 18(1):54–67, 2015.
[23] D. P. Zhou and C. J. Tomlin. Budget-constrained multi-armed bandits with multiple plays. In *AAAI*, 2018.