Long-Term or Temporary? Hybrid Worker Recruitment for Mobile Crowd Sensing and Computing

Minghui Liwang, Member, IEEE, Zhibin Gao, Member, IEEE, Seyyedali Hosseinalipour, Member, IEEE, Zhipeng Cheng, Xianbin Wang, Fellow, IEEE, and Zhenzhen Jiao, Senior Member, IEEE

Abstract—This paper explores an interesting worker recruitment challenge where the mobile crowd sensing and computing (MCSC) platform hires workers to complete tasks with varying quality requirements and budget limitations, amidst uncertainties in worker participation and local workloads. We propose an innovative hybrid worker recruitment framework that combines offline and online trading modes. The offline mode enables the platform to overbook long-term workers by pre-signing contracts, thereby managing dynamic service supply. This is modeled as a 0-1 linear integer programming problem with probabilistic constraints on service quality and budget. To address the uncertainties that may prevent long-term workers from consistently meeting service quality standards, we also introduce an online temporary worker recruitment scheme as a contingency plan. This scheme ensures seamless service provisioning and is likewise formulated as a 0-1 ILP problem. To tackle these problems with NP-hardness, we develop three algorithms, namely, i) exhaustive searching, ii) unique index-based stochastic searching with risk-aware filter constraint, iii) geometric programming-based successive convex algorithm. These algorithms are implemented in a stagewise manner to achieve optimal or near-optimal solutions. Extensive experiments demonstrate our effectiveness in terms of service quality, time efficiency, etc.

Index Terms—Hybrid worker recruitment, mobile crowd sensing and computing, offline and online trading, overbooking, risk.

I. INTRODUCTION

T

THE past decade has witnessed an enormous increase in the number of intelligent Internet of Things (IoT) devices embedded with powerful processors and sensors, e.g., camera, GPS, gyroscope. This development has led to the rise of a wide range of innovative data- and computation-intensive applications (e.g., traffic information gathering, urban WiFi characterization, weather forecasting, and social services) [1] that aim to process the distributively collected data at the network edge via modern computation techniques [2], [3], [4], [5]. Traditional techniques such as sensor networks have been separately implemented over real-world networks, which, however, may impose high installation cost, suffer from limited spatial coverage, and face difficulties in integrating the distributed mobile computing/storage resources [6].

To this end, Mobile Crowd Sensing and Computing (MCSC) at network edge has been introduced [6], [7]. MCSC represents a novel platform that enables ordinary users to contribute sensed/generated data from their mobile devices, while aggregating and processing heterogeneous crowd sensed data at the network edge for intelligent service provisioning [7], [8], e.g., intelligent urban traffic management. The MCSC platform empowers IoT devices to function as mobile device clouds (MDCs) [9], [10], equipped with on-board processors and resources. These MDCs can locally preprocess the data they collect, filtering out unnecessary information, which not only reduces data redundancy but also eases the burden on device-to-edge communication and the computational load on edge servers.

MCSC generally constitutes a three-tiered hierarchical network: i) an edge-based MCSC platform as a centralized controller and data processing center (service requestor), ii) multiple Point-of-Interests (PoIs, a PoI represents a certain region where the MCSC platform is interested in its information and data) coordinated by the edge server, and iii) heterogeneous
mobile smart devices (service providers) which are referred to as workers. Each PoI can announce various tasks with different service quality requirements (e.g., data quality, pre-processing delay, etc.), while a set of workers around the corresponding PoI are recruited to offer sensing and computing services [11]. Nevertheless, pre-processing a massive amount of collected data on IoT devices and transmitting the corresponding results to the MCSC platform may require individual efforts [11], bringing crucial issues, e.g., extra computation and communication costs. Also, the selfishness of participants (especially workers) [12] may hinder a smooth service provisioning procedure, which calls for proper monetary incentives [6]. These challenges advocate studying MCSC as a service trading market, where the MCSC platform pays commission to each worker according to its contribution to a specific task. To put it succinctly, in this paper, we study trading-empowered MCSC. During a trading, each recruited worker pre-processes its collected raw data which is required for the successful execution of an MCSC task locally, and transmits the corresponding results to the platform, while receiving a certain payment.

### A. Motivations

Existing studies on MCSC have been focused on designing online and offline trading mechanisms for worker recruitment. Online MCSC considers the current information associated with workers and tasks during each practical trading (e.g., information associated with a network snapshot), to conduct online decision-making (e.g., worker-task assignment) [13]. Although online trading can capture the current task/worker/network conditions and achieves relatively accurate decisions, it faces major difficulties to cope with the random and dynamic nature of mobile networks. For example, mobile devices may choose to participate or to be absent from a trading due to i) uncertain mobility, and ii) uncertain dynamic local workloads, which result in dynamic and unwarranted service supply. Furthermore, time-varying wireless network conditions impose uncertainties on the availability of sensing and computing services, e.g., a large data transmission delay can be incurred by poor wireless channel quality between a worker and the gateway of the MCSC platform. Major challenges associated with online MCSC caused by the uncertainties are detailed below:

- **Extra latency incurred by online decision-making:** Temporal variations of system states (e.g., time-varying channel quality between each worker and the gateway of MCSC platform) require MCSC platform and workers (together referred to participants) to spend excessive amount of time to analyze the current tasks/workers/networks-related information to reach the final agreement during every trading, which reduces the amount of time that can be used for practical sensing and computing. Specifically, since an online decision only works for the current trading, an accumulated latency will be incurred upon considering a large number of online trading.

- **Extra energy consumption incurred by online decision-making:** A long online decision-making procedure can lead to excessive energy consumption, and subsequently a considerable carbon emission and air pollution in long-term [14]. This factor can cause both performance degradation and undesired trading experience, especially in wireless networks with energy-limited mobile devices [2], [15].

- **Unexpected trading failures:** Online trading may lead to undesired trading failures, and low quality of experience for participants. A typical example is online auctions, where a limited number of winners (winning workers) obtain the eventual employment contracts, while no compensation is offered to the losers (workers who fail in a trading) which also have spent time/energy on bidding/negotiating/waiting. Trading failures can significantly impact the trading experience of participants especially for workers, making them no longer willing to offer services.

We are thus motivated to develop cost-effective and time-efficient service provisioning approaches for MCSC in wireless networks, given the above-mentioned drawbacks of online trading. Offline (in-advance) decision-making offers a feasible solution with less impact on the real-time service delivery [16], [17], [18]. Nevertheless, most of studies on offline trading design for MCSC generally consider known prior information (e.g., prior knowledge of the workers’ locations/resources), which can be impractical to obtain in real-world networks [13], [19].

Given the advantages of online and offline trading modes, this paper introduces a novel hybrid worker recruitment mechanism integrating both offline trading that relies on historical trading statistics [18], and online trading that considers the current worker/task/network conditions. In our developed platform, the offline trading mode determines long-term workers for each task by signing risk-aware contracts in advance, taking into account the historical characteristics of workers/tasks/networks. Long-term workers will offer sensing and computing services to the corresponding tasks during each trading while receiving pre-negotiated payments, without any further negotiation/bargain/bidding with the MCSC platform. This strategy significantly reduces the overhead (e.g., time and energy) of online MCSC decision-making. Specifically, our considered offline trading mode enables the MCSC platform to overbook [21], [22] long-term services from workers, which implies that, the overall service quality promised in pre-signed contracts for a task can be higher than that of its actual demand, given the dynamic service supply. Overbooking helps offering satisfactory quality of service when some long-term workers cannot offer services at some particular trading times, or opt out of a trading [16], e.g., when they are located outside of the corresponding PoI due to their mobility. Although it is important to recruit enough workers for each task, the number of workers is usually constrained by each task’s budget. Thus, the overbooking should be carefully investigated to alleviate the risk of going over budget.

Although overbooking is allowed, the overall service quality offered by long-term workers for a task may be unsatisfying owing to the uncertain workers’ participation and fluctuant quality of service. To this end, we develop an online trading mode as an alternative Plan B to achieve better service quality for MCSC tasks. In the online trading mode, the platform selects temporary workers during each trading when the service requirement (e.g., data quality) of some tasks are not met by long-term workers.

1Namely, workers those who have not been selected as long-term ones are called “temporary workers” in this paper.
B. Relevant Investigations

We review related works devoted to the worker recruitment problem in mobile crowd sensing (MCS),\(^2\) where worker recruitment mechanism is designed while assuming that workers’ information (e.g., sensing qualities) are known in advance [12], [23], [24], [25], which is impractical in some real-world scenarios.

Accordingly, a few existing works have been dedicated into scenarios with partially unknown information, e.g., unknown sensing qualities and locations of workers. Such studies can either be related to economic behaviors [11], [13], [19], [26], [27], [28], [43], [44], [46], [47], [48] or not [41], [42], [45]. Regarding service trading markets, Wang et al. in [13] considered location-aware and location diversity-based dynamic crowd sensing systems with moving workers and stochastic arrival tasks. Née Müller et al. [26] introduced a context-aware hierarchical online learning algorithm to maximize the performance of MCS under unknown workers. In [27], Xiao et al. investigated the worker recruitment problem with unknown service qualities, aiming to maximize the total sensing quality under a limited budget, while ensuring workers’ truthfulness and individual rationality. Similarly, Gao et al. in [19] assumed that workers’ sensing qualities and costs are unknown, and focused on continuous sensing tasks. Assuming that sensing quality needs to be protected from disclosure, Zhao et al. in [28] modeled the worker recruitment under unknown service quality as a differentially private multi-armed bandit game. Wu et al. [43] investigated a utility-based sensing task decomposition and subcontract algorithm, by establishing direct collaboration between mobile nodes (namely, workers). In [44], Gao et al. utilized UAVs to optimize the sensing coverage and data quality, by jointly considering the optimization of task allocation and trajectory scheduling. In [46], Zhang et al. focused on two task types, namely, long-duration (e.g., time-sensitive) and short-duration (e.g., data-intensive) tasks, and proposed a budget re-distribution algorithm for worker recruitment in cloud-aided edge networks. Li et al. in [47] studied a POI-tagging App-assisted incentive mechanism (PTASIM), which explores the cooperation with POI-tagging App for mobile edge crowdsensing. Specifically, PTASIM requests App to tag some edges to be POI, and further guides App users to perform tasks at the corresponding location. In [48], Zhang et al. proposed a mobility prediction-based multi-task allocation method, by using workers’ historical trajectories more comprehensively via fuzzy logic system to achieve better predictions, while designing a global heuristic searching algorithm to optimize the overall task completion rate. Among existing works, the most relevant work is [11], where Xu et al. designed a pricing scheme which concerns with: the payment offered to the workers, based on their reputations (obtained from historical performance) and their actual performance (during practical trading). Nevertheless, although the above-mentioned studies have investigated the worker recruitment problem from various perspectives, our paper introduces a different angle, as discussed below.

\(●\) A resource overbooking-empowered hybrid trading mode: To the best of our knowledge, our study is among the first to introduce a hybrid trading mode by integrating both offline and online trading, which is more applicable in real-world networks with dynamics. For example, in our considered market, a task can be completed with the help of both pre-signed long-term workers and temporary workers (as backup services). More importantly, this paper encourages tasks to overbook services in coping with the uncertain resource supply, representing a novel research highlight to the literature. Due to overbooking, we further incorporate soft budget constraints into the mechanism design, where the budget of each task can slightly fluctuate, which is more practical as compared to hard budget assumption that does not allow any changes (e.g., [12]).

\(●\) Uncertain factors: We consider various uncertainties in the MCSC network. For example, unpredictable workers’ participation due their mobility. Also, we analyze the service quality under a more fine-grained representation, in which the local workload of each worker is considered (which has rarely been studied in literature). Interestingly, in our considered scenario, each worker can have his own tasks that require multi-dimensional on-board resources, e.g., storage and computing, which can impact the service quality offered to MCSC tasks.

\(●\) Risk evaluation and control: Risks caused by uncertainties have been neglected in many studies. In our market, since the uncertainties may cause economic losses (e.g., over budget) or unsatisfying services, we evaluate and control possible risks that both workers and the platform may face to protect service qualities and workers’ profits.

C. Key Challenges

We aim to solve the worker recruitment problem for MCSC in wireless networks via considering two stages: i) long-term worker determination (offline mode); and ii) temporary worker recruitment during each trading (online mode). We are facing three key challenges during the first stage:

i) How to determine feasible long-term workers for each task? This is challenging since the service quality and participation of workers are unknown before each actual trading. Key challenges can include capturing accurate historical statistics of uncertainties, managing risks, and finding feasible mappings between tasks and appropriate workers in a timely manner.

ii) How to determine long-term contract terms, such as the payment to a hired worker, and the quality of service that this worker should provide? This is crucial because tasks always have varying budgets and diverse service demands. At the same time, insufficient payment may result in negative utility for workers. Also, as this paper introduces a unique perspective on two levels of quality assurance (namely, hard and soft), identifying proper service levels for tasks remains both urgent and critical.

iii) How to decide the overbooking rate for each task under the service quality requirement and the budget constraint? This is a crucial aspect of the design, as a high overbooking rate can cause the total payment to long-term workers to exceed the task budget.

---

\(^2\)This paper considers MCSC, where the raw sensing data can be pre-processed on mobile devices first and then send the useful information to the platform, avoids data redundancy while alleviating heavy burden on both backbone links and edge/cloud servers [9], [10]. However, most existing works have put emphasis on MCS, and neglected the computation and processing aspects of the system to some extent. Thus, MCS is highly relevant to our topic.
while a low rate may lead to subpar service quality. Although temporary workers can be utilized, the online decision-making process to recruit them incurs network costs, such as latency and energy consumption.

For the second stage (i.e., temporary worker recruitment), our key research question is: How to determine temporary workers for a task when the corresponding long-term workers fail to meet its service quality requirements?

In summary, we solve the fundamental problem of Joint task-to-worker association, as well as payment and service level determination in both online and offline trading modes in MCSC networks.

D. Outlines and Summary of Contributions

Our considered scenario involves: i) an MCSC platform as a service requestor (e.g., edge server), ii) PoIs that can generate tasks with service quality requirements and certain budgets, and iii) multiple workers (e.g., mobile devices) with different capabilities. Also, significant uncertainties are modeled and incorporated into our methodology to better capture the random and unpredictable nature of MCSC networks. For each worker, two levels of service quality are considered: hard, and soft quality. The former implies that the worker assigns the highest priority to the scheduled MCSC task, to guarantee a hard service quality assurance by charging a high price. The latter refers to a fluctuating service quality, where the worker will first execute its local workload and then processes the assigned MCSC task, under a lower price. To the best of our knowledge, this work is among the first to study the unknown worker recruitment problem for MCSC networks via considering hybrid trading mode, overbooking, and risk evaluation. Our major contributions are summarized below:

- We introduce a novel hybrid worker recruitment methodology for MCSC service provisioning via unifying both offline and online trading modes, where uncertain workers’ participation and fluctuant workers’ local workloads are considered to capture the random and unpredictable nature of the network.
- The offline worker recruitment mode encourages the MCSC platform to overbook services from long-term workers for each task, in coping with dynamic service supplies, by signing employment contracts with promised payment and service quality level in advance, via analyzing the current network/task/worker conditions.
- Comprehensive simulations relying on both numerical data and real-world dataset demonstrate that our proposed overbooking-enabled and risk-aware hybrid worker recruitment mechanism for MCSC tasks outperforms baseline methods from different perspectives, e.g., long-term service quality, time efficiency (running time), etc.

The rest of this paper is organized as follows. In Section II, we provide an overview of MCSC networks and introduce our modeling. Long-term worker determination and contract design problems, as well as temporary worker recruitment problem are proposed and analyzed in Section III. Experimental results are carried out in Section IV, before drawing the conclusion in Section V.

II. OVERVIEW AND SYSTEM MODEL

A. Overview

We consider an MCSC network consisting of multiple workers gathered via set $W = \{w_i | i \in \{1, 2, \ldots, |W|\}\}$ that can be recruited to collect and compute data for multiple tasks of interest to various PoIs [29], [30] collected via the set $O = \{PoI_j | j \in \{1, 2, \ldots, |O|\}\}$. Each $PoI_j$ periodically generates a certain number of sensing tasks expressed as $s_j = \{s_{j,k} | k \in \{1, 2, \ldots, |s_j|\}\}$, for the execution of recruited workers through a sequence of trading. During each trading, a worker offers service (namely, contribute and pre-process data) to a task $s_{j,k}$ if it stays within the relevant $PoI_j$’s region [11], [23], [32] while charging a certain price. We consider two modes for worker recruitment:

In this paper, we mainly consider periodic sensing tasks, which are also common in real-world networks. For example, a PoI may be interested in the traffic and pedestrian data associated with a certain road intersection during rush hours (e.g., 7:00am-9:00am and 17:00am-19:00am, every workday), and thus periodically generates MCSC tasks every 10 minutes. Correspondingly, there will be a trading in each time slot 7:00am, 7:10am, 7:20am, etc., and contracts can be signed among MCSC platform and workers especially for rush hours. Note that our proposed methodology can also be applied for different task arrival rates. For example, considering two task types, where tasks of type 1 generate every 5 minutes, and tasks of type 2 generate every 10 minutes. Then, our considered hybrid market can allow different contracts for different task types, which can be implemented independently.

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
i) **Offline long-term worker recruitment**: This procedure happens in prior to future trading, where the MCSC platform recruits workers for each task via signing long-term contracts in advance, which will be fulfilled accordingly during each trading without any further negotiations. This mode encourages each task to overbook services from workers to cope with the underlying uncertainties in the system. The terms of contract involves a task (namely, a specific task associated with a specific PoI), a worker, the relevant service quality assurance (hard or soft, which will be introduced in Section II-C), as well as the payment. Workers who have signed contracts with the MCSC platform regarding the tasks of PoI, are referred to long-term workers of PoI.

ii) **Online temporary worker recruitment**: This procedure occurs during each trading. Due to the dynamics and uncertainties associated with MCSC networks, a task may suffer from the risk of receiving unsatisfying service quality. For example, long-term workers who happen to move outside of the target PoI’s region will fail to provide services to their assigned tasks. At this time, MCSC platform can recruit temporary workers without pre-signed contracts for the tasks with unsatisfying service quality, under an online trading mode. Notably, online worker recruitment procedure relies on analyzing the current task/worker/network conditions, which may incur extra latency and energy consumption.

Fig. 1 illustrates the timeline for our proposed hybrid worker recruitment mechanism, which is segmented into two distinct phases: i) before trading, and ii) during trading. During the former phase, the MCSC platform engages workers by establishing long-term contracts. Subsequently, in the trading phase, both long-term workers and the MCSC platform execute their commitments as outlined in the previously agreed contracts, with the potential addition of temporary workers to supplement the workforce as needed. Also, several practical trading examples are illustrated in Fig. 1, e.g., in trading 1, long-term worker w fails to serve tasks s, and s, since it moves outside of both the two associated PoIs’ coverage zone.

**B. Task Modeling**

We denote task s belonging to PoI, as a tuple s = (dQ, dQ), where dQ denotes the tolerable budget of the task, which affects the number of employed workers. Specifically, the budget can be distributed among the two types of workers. For instance, in each practical trading, a task should first pay its long-term workers (who have attended) for the promised prices, while the remaining budget can be utilized for recruiting temporary workers in improving its received service quality. Besides, dQ represents the desired service quality [12], which incorporates the following factors: i) data quality, reliability and accuracy, e.g., high and low-resolution image data can lead to different data qualities; ii) sensing duration, e.g., some workers may need to travel around a certain region for data collection [11], [12], [33]; iii) time for data analyzing (computing), e.g., a worker may have to pre-process the collected raw data locally, to reduce possible data redundancy and heavy computation burden on the MCSC platform; iv) time for uploading data to the MCSC platform through the wireless medium.

**C. Worker Modeling**

We assume that each worker w ∈ W can contribute data and computing service to one task within a PoI region during a trading. To better capture the unpredictable and random nature of
MCSC networks, we consider two key uncertainties\footnote{Note that this paper considers known distributions for these uncertain factors, as also supported by existing studies such as [17], [18], [49]. When confronting unknown distributions, we can follow ideas such as exploratory data analysis (e.g., perform exploratory data analysis on the historical information), fitting probability distributions (e.g., use statistical tools or software to fit probability distributions (e.g., use statistical tools or software to fit probability distributions (e.g., use statistical tools or software to fit probability distributions to the historical data), non-parametric methods (e.g., kernel density estimation), time series analysis (e.g., auto-regressive integrated moving average models), and machine learning-based methods (e.g., long short-term memory), which will be discussed in our future work.) associated with each worker $w_i \in \mathcal{W}$.

- **Uncertain participation:** This uncertainty describes whether a worker $w_i$ is located inside or outside from PoI$_j$ during a trading process, due to its mobility (in the latter of which it cannot offer sensing and computing service to PoI$_j$). For worker $w_i \in \mathcal{W}$, let random variable $\alpha_i$ describe its attendance in different PoIs' regions that follows a discrete distribution\footnote{Note that in our problem formulation (see Section III), to facilitate the analysis, we mainly consider one PoI, where the corresponding distribution of $w_i$ can reasonably be modeled as a Bernoulli distribution independent of time. Moreover, we use different values of $a_i$ to describe various behaviors of workers.} denoted by $\alpha_i \sim \mathcal{D}((0, 1, \ldots, j, \ldots, |\mathcal{D}|), (1 - \sum_{j'=1}^{j} \alpha_{i,j'}, \alpha_{i,1}, \ldots, \alpha_{i,j}, \ldots, \alpha_{i,|\mathcal{D}|}))$, where $0 \leq \sum_{j'=1}^{j} \alpha_{i,j'} \leq 1$. Specifically, $\alpha_i = j$ with probability $\alpha_{i,j}$ captures an event in which worker $w_i$ is within PoI$_j$'s coverage; while $\alpha_i = 0$ with probability $1 - \sum_{j'=1}^{j} \alpha_{i,j'}$ means that worker $w_i$ is absent from the trading (“no show”)\footnote{We consider a Normal distribution for the local workload of each worker, implying a fluctuating service quality to MCSC task completion time. Consequently, we consider two quality levels for the services offered by a worker, namely, service with hard quality assurance and service with soft quality assurance detailed below.} since it is located outside of all PoIs’ regions and can not offer any service. For notation simplicity, let $\mathcal{A} = \{\alpha_1, \alpha_2, \ldots, \alpha_|\mathcal{W}|\}$ where $\alpha_i \in \mathcal{A}$ are independent from each other.

- **Uncertain local workload:** In practice, workers often need to manage their local tasks, which can affect the quality of service when performing MCSC tasks. For instance, a worker’s local tasks may be delayed due to the time required to complete an assigned MCSC task, such as waiting for the release of occupied computing or storage resources.\footnote{Note that in our problem formulation (see Section III), to facilitate the analysis, we mainly consider one PoI, where the corresponding distribution of $w_i$ can reasonably be modeled as a Bernoulli distribution independent of time. Moreover, we use different values of $a_i$ to describe various behaviors of workers.} The local workload of each worker $w_i \in \mathcal{W}$ is modeled by random variable $\beta_i$ where $b_{w_{\beta}}, \leq \beta_i \leq b_{w_{\beta}}$, following a Normal distribution\footnote{We consider a Normal distribution for the local workload of each worker, implying a fluctuating service quality to MCSC task completion time. Consequently, we consider two quality levels for the services offered by a worker, namely, service with hard quality assurance and service with soft quality assurance detailed below.} with mean $\mu_{w_\beta}$ and variance $\sigma_{w_\beta}^2$: $\beta_i \sim \mathcal{N}(\mu_{w_{\beta}}, \sigma_{w_{\beta}}^2, b_{w_{\beta}}, b_{w_{\beta}})$. A large value of $\beta_i$ indicates a heavy local workload that $w_i$ needs to handle. For analytical simplicity, let $\mathcal{B} = \{\beta_1, \beta_2, \ldots, \beta_{|\mathcal{W}|}\}$ where all $\beta_i \in \mathcal{B}$ are independent from each other. Note that workers in the hybrid market are truthful, namely, they will not misreport their local workload (e.g., pretend to have a high workload) in either offline or online modes, due to the following reasons. First, there is no incentive for a worker $w_i$ to report wrong the statistics of its local workload (i.e., the distribution parameters of $\beta_i$) since the asked payment will be large, which may further lead to a trading failure in the offline market. Then, a worker $w_i$ attends the online mode as a temporary worker will not misreport since a large pretended workload can lead to a smaller promised service quality and thus a lower price that a task is willing to pay (due to budget constraint)\footnote{Workers in this paper are truthful since they have no motivations to misreport their information. Also, many existing studies [12], [43], [44] have omitted the truthfulness analysis of workers by directly letting them be truthful.}.

\subsection*{D. Service Modeling}

Participating in different tasks can incur various costs for workers, depending on the complexities and resource requirements of each task [12]. Let $c_{i,j,k}$ denote the fundamental cost that $w_i$ needs to spend as long as performing task $s_{j,k}$, e.g., energy consumption incurred by traveling around the target sensing region. Also, different workers may offer various service qualities of task processing [12], [27], [34], considering factors such as the on-board capabilities of smart devices [12] (e.g., heterogeneous hardware settings), and channel conditions for sensing/data transmission. In addition, the local workload of each worker impacts its service quality. For example, a worker that assigns top priority to its local workload may cause unacceptable MCSC task completion time. Consequently, we consider two quality levels for the services offered by a worker, namely, service with hard quality assurance and service with soft quality assurance detailed below.

- **Service with hard quality assurance** implies that a worker $w_i \in \mathcal{W}$ offers a strict guaranteed service quality $q_{i,j,k,\text{Hard}}$ to task $s_{j,k}$, via promising to assign high priority to the assigned MCSC task under any local workload during each trading ($w_i$ handles its local tasks after the completion of MCSC task). For example, $q_{i,j,k,\text{Hard}}$ can be determined as the theoretically maximum service quality that worker $w_i$ can provide without considering local tasks. Thus, a worker $w_i$ who offers hard service quality assurance can definitely incur a service cost denoted by $r_i q_{i,j,k,\text{Hard}}$, where $r_i$ denotes a positive cost factor, where higher required service quality $q_{i,j,k,\text{Hard}}$ and heavier local workload $\beta_i$ can incur severe cost on worker $w_i$. For example, a higher promised service quality may occupy more resources, while the local tasks should wait for the release of these resources, causing a large cost. From another perspective, the product of the cost factor $r_i$ and the service quality $q_{i,j,k,\text{Hard}}$ can be seen as a unit cost of the occupied resources incurred to the local workload. Let $p_{i,j,k,\text{Hard}}$ indicate the required payment of worker $w_i$ for contributing data to task $s_{j,k}$ under hard quality assurance.

- **Service with soft quality assurance** implies that a worker $w_i$ will assign the highest priority to its local tasks, while thus offering a fluctuating service quality to MCSC task $s_{j,k}$ during each trading (caused by its uncertain local workload). The soft quality is denoted by $q_{i,j,k,\text{Soft}} = q_{i,j,k,\text{Hard}} - r_i q_{i,j,k}$. where, apparently, a large value of workload $\beta_i$ leads to a low value of $q_{i,j,k,\text{Soft}}$. Specifically, $r_i$ measures the performance degradation rate, capturing the degradation of service quality that can be contributed...
to a MCSC task due to the local workload of \( w_i \) (e.g., extra processing time for the assigned task). Let \( p_{i,j,k,\text{Soft}} \) denote the required payment of worker \( w_i \) for providing service to task \( s_{j,k} \) under soft service quality assurance \( (p_{i,j,k,\text{Soft}} < p_{i,j,k,\text{Hard}}) \). Although soft assurance may incur risky service quality, it can lead to opportunistic advantages for both parties. For instance, the MCSC platform may enjoy a high service quality at a low price (e.g., considering \( \beta_i = 0 \), we have \( q_{i,j,k,\text{Soft}} = q_{i,j,k,\text{Hard}} \) and the MCSC platform only pays \( p_{i,j,k,\text{Soft}} \)). For each \( s_{j,k} \in S \), \( q_{i,j,k,\text{Soft}} \) also obeys a Truncated normal distribution according to the distribution of \( \beta_i \in \mathbb{B} \). Table I summarizes key notations used in this paper.

### III. Proposed Hybrid Worker Recruitment for MCSC

#### A. Long-Term Worker Determination and Contract Design (Offline Mode)

We make a rational assumption where the workers and PoIs are independent from each other [12], we focus on analyzing the worker recruitment at one PoI and drop the index \( j \), without loss of generality\(^{11} \) (e.g., now we have \( S = \{ k \mid k \in \{1, 2, \ldots, |S|\} \} \), and \( \alpha_i \sim \text{B}(0, 1, (1 - \alpha_i, \alpha_i)) \), where “B” denotes the Bernoulli distribution).

To achieve better analysis, let \( L = \{ \text{Soft}, \text{Hard} \} \) denote the set of quality assurance levels, while \( \ell \in L \) is the corresponding label of “soft” and “hard”. Specifically, \( x_{i,k,\text{Hard}} = 1 \), and \( x_{i,k,\text{Soft}} = 0 \) or \( x_{i,k,\text{Hard}} = 0 \). Each worker can be assigned to at most one task under hard or soft quality assurance \( (\sum_{s_{k} \in S} \sum_{\ell \in L} x_{i,k,\ell} \leq 1, \forall w_i \in W) \); while a task can recruit multiple workers. We let \( Q = \{ q_{i,j,k,\ell} | w_i \in W, s_k \in S, \ell \in L \} \) indicate the service quality profile, \( P = \{ p_{i,j,k,\ell} | w_i \in W, s_k \in S, \ell \in L \} \) represent the payment profile, and \( X = \{ x_{i,k,\ell} | w_i \in W, s_k \in S, \ell \in L \} \)

\(^{11}\)Possible collaboration among PoIs and cooperation among workers is not considered in this paper, which will be investigated in our future work as an interesting direction.

![Fig. 2. Example of long-term contracts among 4 workers and 2 tasks (the third subscript 1, and 0 denote “Hard”, and “Soft”, respectively for notation simplicity in this figure).](image-url)
\[ \mathbb{P}(X, Q, A) = \mathbb{E} \left[ \sum_{s_k \in S} \sum_{w_i \in W} \alpha_i \sum_{t \in L} x_{i,k,t} | q_{i,k,t} \right] \]
\[ = \sum_{s_k \in S} \sum_{w_i \in W} \alpha_i \sum_{t \in L} x_{i,k,t} | q_{i,k,t}, \]  

(3)

where \( q_{i,k,t} \equiv \mathbb{E}[| q_{i,k,t} ] \) denotes the expected value of service quality. For hard quality assurance, we have \( q_{i,k,t} = q_{i,k,t}^{Hard} \). Also, given the distribution of \( q_{i,k,t} \), \( q_{i,k,t}^{Soft} \) is a bounded random variable \( q_{i,k,t}^{Soft} \leq q_{i,k,t} \leq q_{i,k,t}^{Hard} \), following a normal distribution with mean \( \mu_{q_{i,k,t}} = -r_{i}q_{i,k,t}^{Hard} + q_{i,k,t}^{Hard} \) and variance \( \sigma_{q_{i,k,t}}^2 = (r_{i}\sigma_{w_i})^2 \), where \( q_{i,k,t}^{Soft} = -r_{i}b_{w_i} + q_{i,k,t}^{Hard} \) and \( q_{i,k,t}^{Soft} = -r_{i}b_{w_i} + q_{i,k,t}^{Hard} \) are the lower and upper bound of \( q_{i,k,t}^{Soft} \), respectively. The expectation of \( q_{i,k,t} \) can be calculated as follows

\[ q_{i,k,t} = \mu_{q_{i,k,t}} \]
\[ + \sigma_{q_{i,k,t}} \Phi \left( \frac{q_{i,k,t} - \mu_{q_{i,k,t}}}{\sigma_{q_{i,k,t}}} \right) - \Phi \left( \frac{q_{i,k,t}^{Soft} - \mu_{q_{i,k,t}}}{\sigma_{q_{i,k,t}}} \right), \]

(4)

where \( \Phi() \) and \( \Phi() \) indicate the probability density function (PDF), and cumulative distribution function (CDF) of standard normal distribution.

Since long-term employment contracts are signed among the MCSC platform and feasible workers in advance to the future practical trading, potential risks should be carefully considered when designing the corresponding contracts. Specifically, for task \( s_k \in S \), we consider two major risks as detailed below.

- **Risk of unsatisfying service quality**: Each task is facing the risk of receiving an unsatisfying service quality, mainly due to possible “no show” of long-term workers. We will formulate this risk as the probability of obtaining a service quality less than \( d^Q_k \):

\[ \mathcal{R}^Q_1 (X, Q, A) = \Pr \left\{ \sum_{w_i \in W} \sum_{t \in L} \alpha_i x_{i,k,t} | q_{i,k,t} \leq d^Q_k \right\}, \]

(5)

where \( \lambda^Q_1 \) denotes a positive threshold coefficient. In (5), a larger value of \( \mathcal{R}^Q_1 \) indicates a higher probability of receiving unsatisfying quality for task \( s_k \).

- **Over-budget risk**: The existing uncertainties (e.g., in the value of \( \alpha_i \) and \( \beta_i \)) call for overbooking workers to complete MCSC tasks [12] under desired service qualities. For example, in a trading process, “no show” of the long-term workers may lead to unsuccessful execution of a task. Nevertheless, tasks are generally constrained by their budgets, limiting the number of employed workers. As a result, each task \( s_k \in S \) is risking overpaying to workers, caused by overbooking. Thus, we formulate the over-budget risk of \( s_k \) as the probability of the payment to the employed workers exceeding the pre-determined budget \( d^B_k \):

\[ \mathcal{R}^B_2 (X, P, A) = \Pr \left\{ \sum_{w_i \in W} \sum_{t \in L} \alpha_i x_{i,k,t} | p_{i,k,t} \leq d^B_k \right\} > \lambda^B_2, \]

(6)

where \( \lambda^B_2 \geq 1 \) represents a positive threshold coefficient.

We then define the utility of worker \( w_i \in W \) during each trading as the net profit for offering sensing and computing service to the assigned MCSC task:

\[ \mathcal{U}^{w_i} (X, Q, P, \alpha_i, \beta_i) = \mathcal{U}^{w_i} (X, Q, P, \alpha_i, \beta_i) \]
\[ = \alpha_i \sum_{s_k \in S} \sum_{w_i \in W} x_{i,k,t} | p_{i,k,t} \leq c_i, \]
\[ + \xi_i \sum_{s_k \in S} \sum_{w_i \in W} x_{i,k,t} | p_{i,k,t} > c_i, \]

(7)

where \( \xi_i \) denotes the weight coefficient, and \( r_{i}q_{i,k,t}^{Hard} \) represents the extra local cost incurred by performing task \( s_k \) with high priority. Maximizing (7) directly is challenging due to the uncertainties; thus, we focus on the expected utility of \( w_i \) given by

\[ \mathcal{U}^{w_i} (X, Q, P, \alpha_i, \beta_i) = \mathcal{U}^{w_i} (X, Q, P, \alpha_i, \beta_i) \]
\[ = \alpha_i \sum_{s_k \in S} \sum_{w_i \in W} x_{i,k,t} | p_{i,k,t} \leq c_i, \]
\[ + \xi_i \sum_{s_k \in S} \sum_{w_i \in W} x_{i,k,t} | p_{i,k,t} > c_i, \]

(8)

On the other hand, each worker \( w_i \in W \) faces the risk of receiving negative utility due to its dynamic local workload. We formulate this risk as the probability of \( w_i \)'s utility being less than the acceptable minimum value \( \mathcal{U}^{w_i}_{min} > 0 \) when participating in the trading as follows:

\[ \mathcal{R}^{w_i} (X, Q, P, \beta_i) = \Pr \left\{ \frac{\mathcal{U}^{w_i} (X, Q, P, \alpha_i, \beta_i)}{\alpha_i \mathcal{U}^{w_i}_{min}} < \lambda^{w_i}_1 \right\}, \]

(9)

2) **Problem Design. Problem Formulation**: The long-term worker recruitment problem aims to design feasible long-term contracts, each of which contains: i) task-worker execution configuration, (i.e., \( x_{i,k,t}, \forall w_i \in W, s_k \in S, t \in L \)), ii) service quality level (i.e., hard or soft), and iii) the corresponding payment \( p_{i,k,t}, \forall w_i \in W, s_k \in S, t \in L \). Such a problem is formulated by the following optimization \( \mathcal{P}_0 \) (given in (10)), aiming to maximize the overall expected utility of the MCSC platform (i.e., the expected value of the sum service quality of tasks), while protecting the expectation of profits of workers (constraint (C6)).

\[ \mathcal{P}_0 : \max \mathcal{U}(X, Q, A) \]

s.t.

\[ (C1) : \mathcal{R}^Q_1 (X, Q, A) \leq \lambda^Q_1, \forall s_k \in S \]
\[ (C2) : \mathcal{R}^B_2 (X, P, A) \leq \lambda^B_2, \forall s_k \in S \]
\[ (C3) : \mathcal{R}^{w_i} (X, Q, P, \beta_i) \leq \lambda^{w_i}_1, \forall w_i \in W \]
\[ (C4) : x_{i,k,t} \in \{0, 1\}, \forall w_i \in W, s_k \in S, t \in L \]

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
(C5) \[ \sum_{i,k,\ell} x_{i,k,\ell} \leq 1, \forall w_i \in \mathbb{W} \]

(C6) \[ \mathcal{U}^s(X_i, Q_i, P_i, \alpha_i, \beta_i) > 0, \forall \mathcal{W}_i \in \mathbb{W} \] \tag{10}

In \( P_0 \), \( \lambda^s_3, \lambda^s_4, \) and \( \lambda^s_5 \) are threshold coefficients within interval (0,1). Constraint (C1) determines the acceptable risk of MCSC platform on receiving unsatisfying services; constraint (C2) controls the tolerable soft budget constraint associated with each sensing task; constraint (C3) considers workers’ individual rationality, and aims to keep the risk of having unsatisfied workers within a limit. Further, constraints (C4) and (C5) ensure a feasible worker recruitment and task assignment: a worker can at most serve one MCSC task during a trading. More importantly, constraint (C6) ensures that each worker can get non-negative utility from a long-term view. Note that \( P_0 \) involves solving a complex problem for obtaining both discrete (e.g., \( x_{i,k,\ell} \)) and continuous variables (e.g., \( p_{i,k,\ell} \)), which is generally NP-hard. Moreover, our considered risk constraints in a probabilistic form (i.e., (C1)-(C3)) further complicates the problem.

**Problem Transformation:** Since the MCSC platform is generally selfish and prefers to hire workers with lower payments, it can offer a payment which is only slightly higher than what a worker requires (as long as each worker get positive utility, e.g., constraint (C6)), to maximize its utility. Subsequently, constraints (C3) and (C6) can be collectively turned into a fixed payment profile \( P^* = \{ p_{i,k,\ell} | w_i \in \mathbb{W}, s_k \in \mathbb{S}, \ell \in \mathbb{L} \} \), where each element (i.e., \( p_{i,k,\text{Hard}}^* \) or \( p_{i,k,\text{Soft}}^* \)) indicates the acceptable payment of each worker under hard/soft quality assurance (describes why (C6) can be automatically satisfied in \( P_1 \) by applying \( P^* \), the derivation of which is detailed in Appendix A).

Correspondingly, we rewrite problem \( P_0 \) as \( P_1 \), where (C2) is transformed into (C7) in the following:

\[ P_1 : \max_{X} \mathcal{U}^s(X_i, Q_i, \mathcal{A}) \]

s.t. (C1), (C4), (C5)

\[ (C7) : \mathcal{R}^s_2(X_i, P^*, \mathcal{A}) \leq \lambda^s_4, \forall s_k \in \mathbb{S}. \] \tag{11}

It is worth noting that transforming \( P_0 \) to \( P_1 \) is reasonable in real-life networks since the MCSC platform generally dominates the worker recruitment procedure. Namely, it is always the employer (MCSC platform) who determines which worker to hire, as long as the tolerable payment requirement of each worker can be met (e.g., constraints (C3) and (C6)).

**Challenges in Solving the Problem:** Solving \( P_1 \) is non-trivial since it still represents a 0-1 integer linear programming (01ILP) problem with NP-hardness. Furthermore, the constraints of \( P_1 \) are challenging to satisfy: each of (C1) and (C7) contains \(|\mathbb{S}|\) probabilistic inequalities. Besides, elements in \( \mathbb{X} \) are inter-inhibitive with each other (e.g., constraint (C5) indicates that if \( x_{i,k,\text{Hard}} = 1 \), then \( x_{i,k,\text{Soft}} \) should be 0), which further imposes significant complications during solution design.

To this end, to tackle \( P_1 \), we first introduce an exhaustive searching algorithm (ESA) which obtains the optimal solution of \( P_1 \) by checking all the possible mappings among tasks and workers as well as their service levels. Note that ESA can incur a high computational complexity due to the large search space, it is only effective for small problem sizes (e.g., few tasks and workers). Then, to achieve a faster trading response and inspired by the state-of-the-art implicit enumeration method, we further propose a sub-optimal algorithm named unique index-based stochastic searching with risk-aware filter constraint (UISRFC), in which our analyzed risks are regarded as filters to eliminate unsatisfying solutions while accelerating the convergence. Nevertheless, although UISRFC can handle large size problems (e.g., large number of tasks and workers), its performance is strictly constrained by the number of performed iterations and the randomness of stochastic searching, which offers no optimality guarantee. For example, few iterations may lead to the failure on obtaining a good solution, while a large number of iterations will definitely raise the time spent on decision-making. Therefore, since \( P_1 \) is highly non-convex, we take one step further and transform it into a more mathematically tractable form to reach better universality in various problem sizes, and develop a geometric programming-based successive convex algorithm (GP-SCA) to solve it. Although it may involve a series of approximations and computations at the platform level, it offers good scalability and works well for diverse problem sizes.

3) **Design of Exhaustive Searching Algorithm (ESA):** Our proposed ESA for offline mode performs exhaustive search over the solution space, according to the statistic analysis on dynamic nature of the market. Specifically, its pseudo-code is given by Algorithm 1, where \( X_n \) denotes \( n \)th solution obtained for \( X \), where \( n \in \{ 0, 1, 2, \ldots \} \), \(|\mathbb{S}| \times |\mathbb{W}|^2 - 1 \) indicates the index of solutions. Line 3 shows that each index will be mapped to the corresponding binary number. For example, considering \(|\mathbb{S}| = 2, |\mathbb{W}| = 3\), the index \( n = 5 \) can be transferred to a binary number 0000000000101, with length \( 2 \times 3 \times 2 = 12 \). Although ESA is simple to implement and can reach the optimality of \( P_1 \), it suffers from high computational complexity of \( \mathcal{O}(2^{(|\mathbb{S}| \times |\mathbb{W}|^2)}) \), which grows exponentially with respect to the value of \(|\mathbb{W}|\) and \(|\mathbb{S}|\). Also, it will converge when all the possible solutions have been obtained.

This makes this algorithm impractical for large problem sizes (e.g., large number of tasks and workers). Motivated by this, we propose UISRFC in the following section.

**Algorithm1:** Exhaustive Searching Algorithm (ESA).

| Line | Step |
|------|------|
| 1    | Initialization: \( X_0 = \emptyset \), \( u_0 \leftarrow 0 \), \( n \leftarrow 0 \) |
| 2    | for \( 1 \leq n < 2^{|\mathbb{S}|} \times |\mathbb{W}|^2 - 1 \) do |
| 3    | \( X_n \leftarrow \text{Decimal-to-Binary}([n]) \), map index \( n \) to the corresponding binary number of length \(|\mathbb{S}| \times |\mathbb{W}| \times 2 \), and transfer the binary number to the solution set \( X_n \) |
| 4    | if \( X_n \) meets constraints (C1), (C5), and (C7) then |
| 5    | \( u_n \leftarrow \mathcal{U}^s(X_n, Q_i, \mathcal{A}) \) |
| 6    | if \( u_n < u_{n-1} \) then |
| 7    | \( u_n \leftarrow u_{n-1} \) |
| 8    | \( X_n \leftarrow X_{n-1} \) |
| 9    | \( n \leftarrow n + 1 \) |
| 10   | \( X^* \leftarrow X_n \) |
| 11   | end algorithm |
4) Design of Unique Index-Based Stochastic Searching With Risk-Aware Filter Constraint (UISRFC): Given the discrete nature of the problem, obtaining feasible solutions for \( \mathcal{P}_1 \) (i.e., \( \mathcal{X} \)) is one of the most significant difficulties. To this end, inspired by the state-of-the-art implicit enumeration method [38], which is a special case of branch-and-bound method, we propose UISRFC algorithm (see Algo. 2), which innovatively considers risks as filter constraints, helping with alleviating the unacceptable computational complexity of ESA, and achieving commendable solutions for \( \mathcal{P}_1 \). UISRFC represents an iterative method, where, in each iteration \( m \), it first stochastically chooses an index which corresponds to a unique binary number (line 4), and check if it is a feasible solution (lines 5-6). If not, it deletes the index from index set \( \mathcal{N} \) (to achieve unique index property while thus proving time efficiency), and starts another iteration (lines 8-10); if yes, it considers the filter constraints\(^{12}\) and checks whether they should be updated (lines 11-18). For example, UISRFC updates the lower bound of filter constraint (C\(^{\prime}\)) every time when \( \mathcal{X}_n \) attains a larger value of (3), as shown in line 13. Based on which, any possible solution that fails to meet (C\(^{\prime}\)) will be directly abandoned, without considering other constraints. Furthermore, the upper bound of constraint (C7) should also be adjusted every time a lower over-budget risk has been reached, as given by line 18. Specifically, the update of constraints (C\(^{\prime}\)) and (C7) indicates that a better MCSC platforms’ utility while a lower risk can be achieved by solution \( \mathcal{X}_n \), while solution \( \mathcal{X}_{n+1} \) in the following iteration should be better than \( \mathcal{X}_n \); otherwise, it will be dropped directly (lines 9-10). With the above-mentioned operations, UISRFC will finally converge in the direction of larger utility and lower risk.

Specifically, the computational complexity of UISRFC relies on the number of iterations, as denoted by \( O(M) \), where \( 1 \leq M \leq 2^{2|\mathcal{S}|+|\mathcal{W}|} \). Thus, UISRFC may suffer from performance degradation under a small number of iteration (e.g., it may fail to find a good solution of \( \mathcal{P}_1 \) due to the property of randomness during stochastic searching), upon considering increasing optimization problem sizes. Besides, raising the number of iterations will definitely result in higher running time. For example, considering 5 tasks and 15 workers (namely, \( |\mathcal{S}| = 5, |\mathcal{W}| = 15 \)), there are \( 2^{2\times15\times2} \) potential solutions (although many of them are infeasible) which pose challenges in determining the value of \( M \). Also, the operation of stochastic searching of UISRFC brings difficulties in offering optimality guarantee, while its convergence also depends on the pre-determined number of iterations. To design an algorithm with better generality, we further propose GP-SCA to achieve a trackable version of the optimization problem in the following section.

5) Design of Geometric Programming-Based Successive Convex Algorithm (GP-SCA): To achieve a trackable optimization problem, we next transform \( \mathcal{P}_1 \) and obtain a tractable solution to cope with its NP-hardness, through a series of convex approximations. Since binary variables in \( X \) pose a great challenge on solving \( \mathcal{P}_1 \), we first propose the following inequality to relax binary variables \( x_{i,k, \text{soft}} \) and \( x_{i,k, \text{hard}} \) to continuous ones within interval [0,1]:

\[
\begin{align*}
 x_{i,k, \text{soft}} \left( 1 - x_{i,k, \text{soft}} \right) + x_{i,k, \text{hard}} \left( 1 - x_{i,k, \text{hard}} \right) & \leq 0,
\end{align*}
\]

(12) where \( x_{i,k, \text{soft}} \) and \( x_{i,k, \text{hard}} \) can either be 0, or 1 so that to satisfy (12). We then reformulate \( \mathcal{P}_1 \) (a maximization problem) to \( \mathcal{P}_2 \) (a minimization problem) given in (13) by introducing variable \( y \) and constraint (C9). In \( \mathcal{P}_2 \), constraint (C8) is defined according to (12), while we obtain the tractable versions of (C1) and (C7) based on Markov inequality, given by (C1)\(^{\prime}\) and (C7)\(^{\prime}\), the derivations of which are detailed in Appendix B.

\[
\mathcal{P}_2 : \min_{X, y} y \text{s.t. (C5)},
\]

\[
\begin{align*}
(C1)' : & \left( 1 - \lambda_2^s \right) \Lambda_1 d_k^B - \sum_{w_l \in \mathcal{W}} \sum_{\ell \in \mathcal{L}} a_i x_{i,k,\ell} y_{i,k,\ell} \leq 0, \\
& \forall s_k \in S \\
(C4)' : & \left( 0 \leq x_{i,k,\ell} \leq 1, \forall w_l \in \mathcal{W}, s_k \in S, l \in \mathcal{L} \\
(C7)' : & \sum_{w_l \in \mathcal{W}} \sum_{\ell \in \mathcal{L}} a_i x_{i,k,\ell} y_{i,k,\ell} - \lambda_2^s \Lambda_1 d_k^B \leq 0, \forall s_k \in S
\end{align*}
\]

Algorithm 2: Unique Index-Based Stochastic Searching With Risk-Aware Filtering Constraint (UISRFC).

---

12 A filter constraint generally refers to a constraint that helps filter out bad solutions (e.g., mainly unsatisfied value of the objective function of an optimization problem) before checking other constraints, to accelerate the algorithm.
\((C8) : \sum_{w_i \in W} \sum_{a_k \in S} \sum_{l \in L} (x_{i,k,l} - (x_{i,k,l}))^2 \leq 0\)

\((C9) : \sum_{a_k \in S} \sum_{w_i \in W} \sum_{l \in L} a_k \frac{a_i}{x_{i,k,l}} \leq y\) \hspace{1cm} (13)

In \(P_2\), \((C9)\) constrains the upper bound of \(\frac{1}{f'(X,Q,A)}\) via applying optimization variable \(y\), which achieves the same solution with the original maximization problem. In the following, we exploit the innate characteristics of \(P_2\) and transform it to Geometric Programming (GP) format [39], the solution of which can be obtained via solving a sequence of convex problems. First, we revisit \((C1)'\), \((C7)'\), \((C8)\), and \((C9)\), expressing them as normalized inequalities:

\[(C1)' : \sum_{w_i \in W} \sum_{a_k \in S} a_i x_{i,k,l} \frac{d_k}{d_k} \leq 1, \forall s_k \in S\] \hspace{1cm} (14)

\[(C7)' : \sum_{w_i \in W} \sum_{a_k \in S} \sum_{l \in L} a_i x_{i,k,l} \frac{d_k}{d_k} \leq 1, \forall s_k \in S\] \hspace{1cm} (15)

\[(C8)' : \mu + \sum_{w_i \in W} \sum_{a_k \in S} \sum_{l \in L} (x_{i,k,l})^2 \leq 1\] \hspace{1cm} (16)

\[(C9) : \frac{1}{y} \sum_{a_k \in S} \sum_{w_i \in W} \sum_{l \in L} a_k \frac{a_i}{x_{i,k,l}} \leq 1,\] \hspace{1cm} (17)

where \(0 < \mu \leq 1\) in (16) represents a constant coefficient approaching to 0, which avoids the tightness of constraint \((C8)\), i.e., the right hand side of \((C8)\) is replaced with \(\mu\) instead of 0, which is desired in practical implementation.

We then focus on the non-convex constraints and aim to approximate them via convex functions. Let functions \(f_k^{(C1)}(x)\), \(f_k^{(C8)}(x)\), and \(f_k^{(C9)}(x,y)\) denote the denominator of the fractions in \((C1)'\), \((C8)'\), and \((C9)\), respectively. We upper bound these functions in (18), (19), and (20) shown at the bottom of this page, according to arithmetic-geometric mean inequality [20], where \(\mu_k \triangleq \frac{\mu}{|w_i| \times |s_k| \times |L|}\). Specifically, we solve the problem under an iterative manner, where in (18)–(20) we approximate the above functions for variable \(x = \{x_{i,k,l}\}\) around the fixed-point; \(x_{i,k,l}^{(m)} = \{x_{i,k,l}^{[m]}\}\), where \(m\) denotes the index of the iteration, and \(x_{i,k,l}^{[m]}\) is the solution of the problem at iteration \(m\). Similarly, let \(y_{i,k,l}^{[m]}\) indicate the solution of \(y\) at the \(m\)th iteration. Using these approximations, \((C1)'\), \((C8)'\) and \((C9)\) are further transformed to \((\hat{C}1)'\), \((\hat{C}8)'\), and \((\hat{C}9)\) given by

\[\hat{C}1' : (1 - \lambda_k^s) \lambda_k^{a_i} \frac{d_k}{d_k} \leq 1, \forall s_k \in S\] \hspace{1cm} (21)

\[\hat{C}8' : \sum_{w_i \in W} \sum_{a_k \in S} \sum_{l \in L} x_{i,k,l} \leq 1\] \hspace{1cm} (22)

\[\hat{C}9 : \frac{1}{f_k^{(C8)}(x,y)} \leq 1.\] \hspace{1cm} (23)

Based on the above steps, \(P_2\) can be reformulated a standard GP given by \(P_3\):

\[P_3 : \min_y y\] s.t. \((\hat{C}1)'\), \((\hat{C}4)'\), \((\hat{C}5)'\), \((\hat{C}7)'\), \((\hat{C}8)'\), \((\hat{C}9)\). \hspace{1cm} (24)

Considering the logarithmic change of variables, a standard formed GP can be transformed into a convex optimization problem, which can be solved by using commercial software such as CVX [20] in an efficient manner. Detailed derivations are shown in Appendix C. The corresponding pseudo-code regarding solving \(P_3\) is given in Alg. 3. The computational complexity and convergence of GP-SCA relies on the CVX...
To better verify our commendable performance, we evaluate the benchmark methods associated with online trading mode. First, we implement “Online ESA”, “Online UISRFC”, and “Hybrid GP-SCA” for notation simplicity. To better verify our commendable performance, benchmark methods associated with online trading mode are considered. We implement “Online ESA”, “Online UISRFC”, and “Online GP-SCA”, by drawing inspiration from our previous work on the MCSC networks. To achieve better performance, we conduct both simulations on numerical data (Section IV-B) and a real-world dataset [36] (Section IV-C).

A. Benchmark Methods and Basic Parameters

In our simulations, the proposed algorithms are abbreviated to “Hybrid ESA”, “Hybrid UISRFC”, and “Hybrid GP-SCA” for notation simplicity. To better verify our commendable performance, benchmark methods associated with online trading mode are considered. First, we implement “Online ESA”, “Online UISRFC”, and “Online GP-SCA”, by drawing inspiration from our proposed solutions. In these modes, all workers are temporary, and each trading is executed based on the current conditions of workers, tasks, and networks. This is achieved through the application of ESA, UISRFC, and GP-SCA, respectively. Then, another two benchmarks derived from the state-of-the-art methods, namely, Monte Carlo random sampling (MCRS) where task-worker mappings are randomly generated with a certain number of iterations, and service quality-prioritized worker recruitment (SQprefer, inspired by greedy-based algorithms)
where each task prefers hard quality assurances, are also considered\textsuperscript{15} under an online mode. Basic parameters are set as follows: $r_i \in [0.6, 0.8], r_i^l \in [0.1, 0.3], c_{i,k} \in [1.2, 1.8], q_{i,k,\text{Hard}} \in [3.2, 4.8], d_{i}^{1} \in [15, 22], d_{i}^{2} \in [7.2, 8.8], \lambda_{i,k}^1 \in [1, 1.05], \lambda_{i,k}^2 \in (0.95, 1], \lambda_{i,k}^3 \in [0.3, 0.4], \lambda_{i,k}^4 \in [0.1, 0.2], \lambda_{i,k}^5 \in [0.98, 1.01], \lambda_{i,k}^6 \in [0.3, 0.4].$

### B. Performance Analysis on Numerical Parameter Settings

We first conduct simulations on numerical parameters to achieve a better generality of our proposed hybrid service trading mode. To capture diverse behaviors of workers, we consider two sets of parameters to show different distributions of $\alpha_i$ and $\beta_i$, which are Set #1: $\alpha_i \in [64\%, 96\%], \mu_{w_i} \in [2.4, 3.6], \sigma_{w_i} \in [0.4, 0.6], b_{w_i}^+ \in [\mu_{w_i} - 1.44, \mu_{w_i} - 0.96], b_{w_i}^- \in [\mu_{w_i} + 0.96, \mu_{w_i} + 1.44]$; and Set #2: $\alpha_i \in [90\%, 95\%], \mu_{w_i} \in [2.5, 3], \sigma_{w_i} \in [0.1, 0.2], b_{w_i}^+ \in [\mu_{w_i} - 1, \mu_{w_i} - 0.5], b_{w_i}^- \in [\mu_{w_i} + 0.5, \mu_{w_i} + 1]$. Apparently, workers associated with Set #1 are under a more “fluctuant” condition than that of Set #2 (which are rather “stable” and for “peak demand” upon having a larger lower bound of $\alpha_i$). By having the above mentioned two sets of distribution parameters (indicating various statistics of uncertainties), the long-term contracts can be updated accordingly.

1) Expected and Practical Utility of MCSC Platform: To highlight the benefits of our proposed algorithms and the intriguing hybrid worker recruitment mode for MCSC in wireless networks, we initially present the expected service quality of MCSC tasks during the offline stage, using Set #1 as an example (see Fig. 3). Note that Fig. 3(a) takes into account both small-size and large-size networks, where online benchmark methods are not considered, as long-term contracts are exclusively associated with the hybrid mode. Besides, Fig. 3(b)–(h) show the corresponding contracts, where $10^{-4}$ and $10^{-5}$ in the legend denote the convergence criterion of GP-SCA. For example, the algorithm will stop at the $m$th iteration when $|y^{m} - y^{m-1}| \leq 10^{-4}$ by setting $10^{-4}$ as the convergence criterion.

As shown in the left-hand plot of Fig. 3(a), which simulates small-size networks (e.g., 2-3 tasks and 4-6 workers), our proposed Hybrid UISRFC and Hybrid GP-SCA can approximate to the performance of ESA, with a relatively low computational complexity (illustrated by Fig. 6). Besides, a tighter convergence criterion (e.g., $10^{-5}$) can help GP-SCA to reach a larger value of expected service quality. Fig. 3(b)–(e) describe the long-term contracts signing among 2 tasks and 5 workers, where in Hybrid GP-SCA, the expected service quality of MCSC tasks can be improved roughly by 5.5% when applying $10^{-5}$ as the convergence criterion, rather than $10^{-4}$. The right-hand plot of Fig. 3(a) depicts the expected utility of the MCSC platform upon having rather large networks (e.g., 5–10 tasks and 10–16 workers), in which Hybrid ESA has been omitted due to its extremely high computational complexity. Similarly, Hybrid GP-SCA ($10^{-5}$) achieves slightly better performance than GP-SCA with convergence criterion $10^{-4}$, while Hybrid GP-SCA outperforms hybrid URIRFC since the performance of the latter depends heavily on the number of iterations and has randomness due to the large solution space and stochastic searching mode. Fig. 3(f)–(h) depict how long-term contracts are signed among 5 tasks and 15 workers. Interestingly, our solutions allow a certain overbooking rate to support dynamic service supply since some contractual workers may not be able to offer satisfactory services, due to factors such as their mobility and varying wireless channel qualities. For example, services offered by 4 of the 5 contractual workers $w_2, w_9, w_{12}, w_{13}, w_{14}$ associated with task $s_1$ in Fig. 3(f) are sufficient to satisfy $s_1$’s service quality requirement, where the overbooking rate is roughly $(5 - 4)/4 = 25\%$.

We next show the performance on the average practical service quality (per trading) of MCSC tasks from a long-term perspective, in comparison with 5 baseline methods via considering 600 trading in total (i.e., 300 for Set #1, and 300 for Set #2, where the long-term contracts have been updated when having a different set). Similarly, different problem sizes are considered to prove the advantages and generalization of our proposed hybrid worker recruitment mechanism. For small problem sizes shown by Fig. 4(a) and (c), although, in most cases, Online ESA performs better due to its optimality under

\textsuperscript{15}Existing studies, e.g., [38], [41], [43], [47], with similar ideas on benchmark method design can be supportive.

Fig. 3. (a) Performance comparison on the expected service quality of MCSC tasks upon considering small, and large problem sizes under Set #1; (b)–(c) Long-term contract signing associated with 2 tasks and 5 workers in (a); (f)–(h) Long-term contract signing associated with 5 tasks and 15 workers in (a). Note that figures (b)–(h), the yellow, and purple-colored box indicates soft, and hard service assurance, respectively.
the current network/market situation during each practical trading, our proposed methods reach a far better time efficiency (see Fig. 6). Also, our Hybrid ESA achieves a similar average service quality as compared to Online EAS, since our proposed mechanism allows a certain overbooking rate, thus differentiates which from Online ESA where each trading is implemented under strict budget constraints. Besides, although failing to catch the performance of ESA, our proposed Hybrid UISRFC and Hybrid GP-SCA outperform Online UISRFC and Online GP-SCA (for both $10^{-4}$ and $10^{-5}$ as the convergence criterion) mainly thanks to overbooking. All the considered methods can achieve a larger utility than that of MCSC and SQprefer, due to the randomness and greedy attribute of the two benchmarks. More importantly, while our proposed methods may exceed the corresponding budget in a specific trading (due to overbooking and probabilistic constraints), the long-term expenses are generally deemed acceptable (refer to Fig. 5), as attributed to the effective management of risks involved.

Fig. 4(b) and (d) illustrate large size problems upon having different sets of distribution parameters (i.e., Set #1 and Set #2). In Fig. 4(b), our proposed Hybrid UISRFC and Hybrid GP-SCA can reach a similar average service quality performance to that of the corresponding online trading methods, while sometimes obtaining slightly better performance (e.g., 5 tasks/10 workers, as well as 5 tasks/15 workers in Fig. 4(b)), since the proposed overbooking can help with recruiting more long-term workers, under acceptable over-budget risks. For example, revisiting Fig. 3(h), task $s_5$ can obtain commendable service quality during a trading when all the contractual workers $w_2, w_9, w_{11}, w_{15}$ have participated in trading, which cause an over-budget issue. As for Set #2 in Fig. 4(d), since workers are more willing to take part in each trading (i.e., $\alpha_5$ is higher than that in Set #1), the average practical utility of the platform can be higher than that of Fig. 4(b), where the online methods Online UISRFC, Online GP-SCA ($10^{-4}$), and Online GP-SCA ($10^{-5}$) achieve slightly better performance than our proposed hybrid since more feasible solutions can be found when having more available attended workers under budget constraint. However, despite that, our hybrid methods offer a good time efficiency (will be discussed in Fig. 6). Namely, when the service trading market tends to be stable, our hybrid solutions can be closer to the online ones. For example, without uncertainties, e.g., let $a_1 = 1, \forall w_i \in W$ and $\beta, \forall w_k \in W$ be a known constant, the online and hybrid methods can almost be the same. This observation further underscores the adaptability of our hybrid methods to dynamic and uncertain market conditions, which are prevalent in real-world network environments. Significantly, the ESA, UISRFC, and GP-SCA methods, implemented in both hybrid and online modes, consistently outperform benchmark methods such as MCRS and SQprefer.

2) Budget and Payment: We next test two tasks in Fig. 5, and compares the average practical expense of each MCSC task (per trading) for purchasing sensing and computing services from workers, as well as the corresponding budget, from a long-term perspective over simulating 300 trading for Set #1, and Set #2, respectively, upon having different intervals of $d^{k}_B$. Note that the two tasks are separately shown in Fig. 5(a)–(d), to achieve a better visuality (i.e., the horizontal coordinate denotes the index of tasks). As can be seen in Fig. 5(a)–(b) relying on Set #1, although overbooking is encouraged in our proposed hybrid worker recruitment mechanism, the average payment of each MCSC task to recruited workers is still acceptable, as benefited from the well-designed risk control constraint (e.g., constraint (C2) of problem $P_0$). For example, considering $\lambda^s_2 = 0.98$ and $\lambda^s_4 = 0.1$, the risk (namely, probability) of task $s_1$’s payment to sensing and computing services exceeds $98\% \times d^{k}_B$ will be controlled within 10%. Fig. 5(c)–(d) under Set #2 show that the payment from tasks are higher than that of Set #1 since more workers can attend each practical trading due to a large value of $a_1$. Even so, payments in our proposed hybrid methods will not exceed the corresponding budgets thanks to the well-designed risk management. In addition, our proposed mechanism can
achieve a good service quality for tasks while greatly reducing the decision-making time (running time, as illustrated by Fig. 6), below the budgets.

3) Running Time: Time efficiency represents one of the most important factors in service trading market especially with uncertainties. Thus, we adopt Fig. 6 to analyze the performance on time efficiency measured by the average running time (per trading, which can estimate the time consumed by decision-making), upon having 300 trading for Set #1 and Set #2 respectively, while considering small and large problem sizes. Logarithmic function is utilized (in the $y$-axis of Fig. 6) to better illustrate the gap between different methods. As for small problem sizes in Fig. 6(a) and (c), ESA-based methods always incur a large running time to reach the final solution, where our proposed Hybrid ESA, Hybrid UISRFC and Hybrid GP-SCA greatly outperform the corresponding online methods. Moreover, the key reason behind which is that conventional online trading mode should make a decision based on the current network/market information during every trading, which thus incurs a specific delay on decision-making, especially for ESA-based methods. Fortunately, our hybrid methods have determined some long-term worker-task pairs in advance, such an operation can greatly reduce the market scale during each practical trading, as the platform only has to take temporary workers and those tasks with unsatisfying service quality into consideration. For example, in Fig. 6(a), the running time associated with Online ESA reaches over 500 seconds averagely during each trading when considering 2 tasks and 5 workers, which makes it impractical to be implemented in real-world networks particularly with energy-constrained mobile devices. In comparison, our Hybrid ESA offers 0.6473 seconds under the same scenario, which thus achieves far better time efficiency and offers a commendable implementation in real-world mobile networks. Although Online UISRFC and Online GP-SCA obtain far better running time performance than Online ESA, the accumulated delay with the increasing number of trading can still bring challenges to the long-term development of MCSC networks. For instance, Online GP-SCA ($10^{-5}$) (the scenario of 2 tasks and 4 workers in Fig. 6(a)) causes around 0.5636 seconds decision-making delay on average, that may be practicable during one trading, which, however, will cost around 563.6 seconds over 100 trading. For large problem sizes shown by Fig. 6(b) and (d), our proposed Hybrid UISRFC and Hybrid GP-SCA can still offer commendable running time, as benefited by pre-determined long-term workers. On the contrary, the running time associated with Online UISRFC and Online GP-SCA gradually become prohibitive with increasing number of workers and tasks. Specifically, applying Set #2 may suffer from an increasing running time rather than Set #1 e.g., scenario of 10 tasks/16 workers in Fig. 6(b) and (d), since more workers will participant in each practical trading, which thus raise the market scale that the platform needs to handle, especially for online methods. Moreover, although MCRS and SQprefer offer a good performance in time efficiency, they receive unsatisfying practical platform utilities (shown by Fig. 4).

4) Convergence of GP-SCA: Fig. 7 illustrates how our GP-SCA-based solution for $\mathcal{P}_3$ converges to its given convergence criterion ($10^{-5}$). Note that finding the initial feasible point (e.g., $u[0]$ and $v[0]$) is also critical for the convergence speed, for which we show Fig. 7(a) with a rather "bad" initial point, and Fig. 7(b) with a "good" one. As can be seen from these figures, our proposed GP-SCA can exhibit a good performance on convergence, upon considering different problem sizes.

As a summary, the proposed hybrid worker recruitment methodology for mobile crowd sensing and computing that integrates both online and offline trading modes achieves good performance on service quality in comparison with online trading modes, while offering far better time efficiency, which provides a commendable reference for the continuable development of future MCSC networks.

C. Performance Analysis on Real-World Dataset

To better verify the performance of our proposed Hybrid service trading methodology, we then adopt a real-world dataset of Chicago taxi trips [36], which records taxi rides in Chicago from 2013 to 2016 with 77 community areas. Here, we consider the 77th community area as our interested sensing region (e.g., one PoI) and randomly choose 15 taxis as workers, as well as 5 tasks in our simulation (supposed that tasks are randomly distributed in the considered region). Specifically, we estimate the distribution parameters of $\alpha_i$ by counting the number of days that these taxis arrive at the considered region in January 2013. To better quantize service cost $c_{t,k}$, we record three key factors from the dataset: $i$) the traveled distance of each taxi (i.e., the
distance between pick-up locations and drop-off locations); ii) the distance between the current positions of taxis (e.g., pick-up location of taxis) and that of each task; iii) the distance between the position of a taxi after task completion (e.g., the drop-off location of taxis) and location of the corresponding task. As each $c_{i,k}$ stands for a monetary expression, we make it to be proportional to these above-mentioned factors. Besides, note that since this paper offers some distinct elements, e.g., different levels of services and local workload of workers, not all the data (e.g., the values of soft and hard service quality) can be found in a real-world dataset, we should also consider some manual numerical parameter settings similar to Section IV-B (e.g., studies [12], [46] can be supportive with similar ideas on using both real-world and numerical simulation parameters in MCS networks, especially regarding economic behaviors). For example, the hard service quality is chosen to be inversely proportional to the sum of distances in the previous ii) and iii), namely, a long distance leads to a lower quality. Accordingly, we show Fig. 8 to demonstrate our superior performance in comparison with benchmarks, by randomly selecting two groups, namely, 5 tasks and 15 workers (taxis) as well as 10 tasks and 16 workers (taxis). Without loss of generality, we also conduct 300 experiments for each subfigure in Fig. 8. Apparently, our proposed hybrid methods can achieve commendable performance on service quality while maintaining satisfying time efficiency.

V. CONCLUSION

This paper introduces a novel risk-aware hybrid worker recruitment mechanism for MCSC networks, involving diverse uncertainties while integrating both online (long-term worker recruitment) and offline (temporary worker recruitment) trading modes. The optimization targets are formulated as 0-1 integer linear programming problems with NP-hardness. Accordingly, three algorithms, namely, exhaustive searching, unique index-based stochastic searching with risk-aware filter constraint, and geometric programming-based successive convex algorithm, are designed to achieve the corresponding optimal (with high computational complexity) or sub-optimal (with low computation complexity) solutions. Comprehensive simulations demonstrate the effectiveness of our proposed mechanism, in comparison to conventional trading methods. Several promising avenues for future research include exploring worker cooperation, as well as the design of cost-effective and time-efficient algorithms for the challenging NP-hard problems identified. Furthermore, the concept of environment-aware and intelligent renewable contract design is currently viewed as an area closely related to our work.

REFERENCES

[1] J. Liu, Y. Yang, D. Li, X. Deng, S. Huang, and H. Liu, “An incentive mechanism based on behavioural economics in location-based crowdsensing considering an uneven distribution of participants,” IEEE Trans. Mobile Comput., vol. 21, no. 1, pp. 44–62, Jan. 2022.

[2] Y. Qu et al., “Service provisioning for UAV-enabled mobile edge computing,” IEEE J. Sel. Areas Commun., vol. 39, no. 11, pp. 3287–3305, Nov. 2021.

[3] X. Xia et al., “OL-MEDC: An online approach for cost-effective data capturing in mobile edge computing systems,” IEEE Trans. Mobile Comput., vol. 22, no. 3, pp. 1646–1658, Mar. 2023.

[4] X. Xia, F. Chen, J. Grundy, M. Abdelrazek, H. Jin, and Q. He, “Constrained app data caching over edge server graphs in edge computing environment,” IEEE Trans. Serv. Comput., vol. 15, no. 5, pp. 2635–2647, Sep./Oct. 2022.

[5] Z. Zhou et al., “Learning-based URLLC-Aware task offloading for internet of health things,” IEEE J. Sel. Areas Commun., vol. 39, no. 2, pp. 396–410, Feb. 2021.

[6] B. Guo et al., “Mobile crowd sensing and computing: The review of an ever-expanding human-powered sensing paradigm,” ACM Comput. Surv., vol. 48, no. 1, pp. 1–31, 2015.

[7] B. Guo, C. Chen, D. Zhang, Z. Yu, and A. Chin, “Mobile crowd sensing and computing: When participatory sensing meets participatory social media,” IEEE Commun. Mag., vol. 54, no. 2, pp. 131–137, Feb. 2016.

[8] Z. Cheng, M. Liwang, X. Xia, M. Min, X. Wang, and X. Du, “Auction-promoted trading for multiple federated learning services in UAV-Aided networks,” IEEE Trans. Veh. Technol., vol. 71, no. 10, pp. 10968–10974, Oct. 2022.

[9] M. Wang, C. Xu, X. Chen, L. Zhong, Z. Wu, and D. O. Wu, “BC-Mobile device cloud: A blockchain-based decentralized truthful framework for mobile device cloud,” IEEE Trans. Ind. Inform., vol. 17, no. 2, pp. 1208–1219, Feb. 2021.

[10] S. Olaru, “A survey of vehicular cloud network: Trends, applications and challenges,” IEEE Trans. Intell. Transp. Syst., vol. 21, no. 6, pp. 2648–2663, Jun. 2020.

[11] C. Xu, Y. Si, L. Zhu, C. Zhang, K. Sharif, and C. Zhang, “Pay as how you behave: A truthful incentive mechanism for mobile crowdsensing,” IEEE Internet Things J., vol. 6, no. 6, pp. 10053–10063, Dec. 2019.

[12] C. Dai, X. Wang, K. Liu, D. Qi, W. Lin, and P. Zhou, “Stable task assignment for mobile crowdsensing with budget constraint,” IEEE Trans. Mobile Comput., vol. 20, no. 12, pp. 3439–3452, 2021.

[13] X. Wang, R. Jia, X. Tian, X. Gan, L. Fu, and X. Wang, “Location-aware crowdsensing: Dynamic task assignment and truth inference,” IEEE Trans. Mobile Comput., vol. 19, no. 2, pp. 362–375, Dec. 2021.

[14] M. Liwang, X. Wang, and R. Chen, “Computing resource provisioning at the edge: An overbooking-enabled trading paradigm,” IEEE Wireless Commun., vol. 29, no. 5, pp. 68–76, Oct. 2022.

[15] S. Hosseinalipour, A. Nayak, and H. Dai, “Power-aware allocation of graph jobs in geo-distributed cloud networks,” IEEE Trans. Parallel Distrib. Syst., vol. 31, no. 4, pp. 749–765, Apr. 2020.

[16] M. Liwang, R. Chen, X. Wang, and X. Shen, “Unifying futures and spot market: Overbooking-enabled resource trading in mobile edge networks,” IEEE Trans. Wireless Commun., vol. 21, no. 7, pp. 5467–5485, Jul. 2022.

[17] M. Liwang, Z. Gao, and X. Wang, “Let’s trade in the future! a. futures-enabled fast resource trading mechanism in edge computing-assisted UAV networks,” IEEE J. Sel. Areas Commun., vol. 39, no. 11, pp. 3252–3270, Nov. 2021.

[18] S. Sheng, R. Chen, P. Chen, X. Wang, and L. Wu, “Futures-based resource trading and fair pricing in real-time IoT networks,” IEEE Wireless Commun. Lett., vol. 9, no. 1, pp. 125–128, Jan. 2020.

[19] G. Gao, H. Huang, M. Xiao, J. Wu, Y.-E. Sun, and Y. Du, “Budgeted unknown worker recruitment for heterogeneous crowdsensing using CMAB,” IEEE Trans. Mobile Comput., vol. 21, no. 11, pp. 3895–3911, Nov. 2022.
[20] S. Hosseinalipour, A. Rahmati, D. Y. Eun, and H. Dai, “Energy-aware stochastic UAV-Assisted surveillance,” IEEE Trans. Wireless Commun., vol. 20, no. 5, pp. 2820–2837, May 2021.

[21] L. Tomás and J. Tordsson, “An autonomic approach to risk-aware data center overbooking,” IEEE Trans. Cloud Comput., vol. 2, no. 3, pp. 292–305, Jul.–Sep. 2014.

[22] K. Chard and K. Bubendorfer, “High performance resource allocation strategies for computational economics,” IEEE Trans. Parallel Distrib. Syst., vol. 24, no. 1, pp. 72–84, Jan. 2013.

[23] Y. Zhan, C. H. Liu, Y. Zhao, J. Zhang, and J. Tang, “Free market of multi-leader multi-follower mobile crowdsensing: An incentive mechanism design by deep reinforcement learning,” IEEE Trans. Mobile Comput., vol. 19, no. 10, pp. 2316–2329, Oct. 2020.

[24] J. Wang, F. Wang, Y. Wang, D. Zhang, L. Wang, and Z. Qiu, “Social network-assisted worker recruitment in mobile crowd sensing,” IEEE Trans. Mobile Comput., vol. 18, no. 7, pp. 1661–1673, Jul. 2019.

[25] J. Wang, F. Wang, Y. Wang, D. Zhang, B. Lim, and L. Wang, “Allocating heterogeneous tasks in participatory sensing with diverse participant-side factors,” IEEE Trans. Mobile Comput., vol. 18, no. 9, pp. 1979–1991, Sep. 2019.

[26] S. K. Née Müller, C. Tekin, M. van der Schaar, and A. Klein, “Context-aware hierarchical online learning for performance maximization in mobile crowd sourcing,” IEEE/ACM Trans. Netw., vol. 26, no. 3, pp. 1334–1347, Jun. 2018.

[27] M. Xiao, B. An, J. Wang, G. Xiao, S. Zhang, and J. Wu, “CMAB-based reverse auction for unknown worker recruitment in mobile crowdsensing,” IEEE Trans. Mobile Comput., vol. 21, no. 10, pp. 3502–3518, Oct. 2022.

[28] H. Zhao, M. Xiao, J. Wu, Y. Xu, H. Huang, and S. Zhang, “Differentially private unknown worker recruitment for mobile crowdsensing using multi-armed bandits,” IEEE Trans. Mobile Comput., vol. 20, no. 9, pp. 2779–2794, Sep. 2021.

[29] C. H. Liu, Z. Chen, and Y. Zhan, “Energy-efficient distributed mobile crowd sensing: A deep learning approach,” IEEE J. Sel. Areas Commun., vol. 37, no. 6, pp. 1262–1276, Jun. 2019.

[30] C. H. Liu, Z. Dai, Y. Zhao, J. Crowcroft, D. Wu, and K. K. Leung, “Distributed and energy-efficient mobile crowdsensing with charging stations by deep reinforcement learning,” IEEE Trans. Mobile Comput., vol. 20, no. 1, pp. 130–146, Jan. 2021.

[31] X. Zhan, M. Fan, D. Wang, P. Zhou, and D. Tao, “Top-k feature selection framework using robust 0–1 integer programming,” IEEE Trans. Neural Netw. Learn. Syst., vol. 32, no. 7, pp. 3005–3019, Jul. 2021.

[32] Y. Yang, W. Liu, E. Wang, and J. Wu, “A prediction-based user selection framework for heterogeneous mobile CrowdSensing,” IEEE Trans. Mobile Comput., vol. 18, no. 11, pp. 2460–2473, Nov. 2019.

[33] Q. Hu, S. Wang, X. Cheng, J. Zhang, and W. Lv, “Cost-efficient mobile crowdsensing with spatial-temporal awareness,” IEEE Trans. Mobile Comput., vol. 20, no. 3, pp. 928–938, Mar. 2021.

[34] J. Nie, J. Luo, Z. Xiong, D. Niyato, P. Wang, and H. V. Poor, “A multi-leader multi-follower game-based analysis for incentive mechanisms in socially-aware mobile crowdsensing,” IEEE Trans. Wireless Commun., vol. 20, no. 3, pp. 1457–1471, Mar. 2021.

[35] D. R. Barr and E. T. Sherrill, “Mean and variance of truncated normal factors,” IEEE Trans. Ind. Inform., vol. 7, no. 2, pp. 1210–1219, Feb. 2022.

[36] M. Xiao, B. An, J. Wang, G. Xiao, S. Zhang, and J. Wu, “CMAB-based reverse auction for unknown worker recruitment in mobile crowdsensing,” IEEE Trans. Mobile Comput., vol. 21, no. 10, pp. 3502–3518, Oct. 2022.

[37] H. Zhao, M. Xiao, J. Wu, Y. Xu, H. Huang, and S. Zhang, “Differentially private unknown worker recruitment for mobile crowdsensing using multi-armed bandits,” IEEE Trans. Mobile Comput., vol. 20, no. 9, pp. 2779–2794, Sep. 2021.

[38] C. H. Liu, Z. Chen, and Y. Zhan, “Energy-efficient distributed mobile crowd sensing: A deep learning approach,” IEEE J. Sel. Areas Commun., vol. 37, no. 6, pp. 1262–1276, Jun. 2019.

[39] C. H. Liu, Z. Dai, Y. Zhao, J. Crowcroft, D. Wu, and K. K. Leung, “Distributed and energy-efficient mobile crowdsensing with charging stations by deep reinforcement learning,” IEEE Trans. Mobile Comput., vol. 20, no. 1, pp. 130–146, Jan. 2021.

[40] X. Zhan, M. Fan, D. Wang, P. Zhou, and D. Tao, “Top-k feature selection framework using robust 0–1 integer programming,” IEEE Trans. Neural Netw. Learn. Syst., vol. 32, no. 7, pp. 3005–3019, Jul. 2021.

[41] Y. Yang, W. Liu, E. Wang, and J. Wu, “A prediction-based user selection framework for heterogeneous mobile CrowdSensing,” IEEE Trans. Mobile Comput., vol. 18, no. 11, pp. 2460–2473, Nov. 2019.

[42] Q. Hu, S. Wang, X. Cheng, J. Zhang, and W. Lv, “Cost-efficient mobile crowdsensing with spatial-temporal awareness,” IEEE Trans. Mobile Comput., vol. 20, no. 3, pp. 928–938, Mar. 2021.

[43] J. Nie, J. Luo, Z. Xiong, D. Niyato, P. Wang, and H. V. Poor, “A multi-leader multi-follower game-based analysis for incentive mechanisms in socially-aware mobile crowdsensing,” IEEE Trans. Wireless Commun., vol. 20, no. 3, pp. 1457–1471, Mar. 2021.

[44] H. Gao, J. Feng, Y. Xiao, B. Zhang, and W. Wang, “A UAV-assisted multi-task allocation method for mobile crowd sensing,” IEEE Trans. Mobile Comput., vol. 22, no. 7, pp. 3790–3804, Jul. 2023.

[45] X. Li, G. Feng, Y. Sun, S. Qin, and Y. Liu, “A unified framework for joint sensing and communication in resource constrained mobile edge networks,” IEEE Trans. Mobile Comput., vol. 22, no. 10, pp. 5643–5656, Oct. 2023.

[46] Y. Zhang, P. Li, T. Zhang, J. Liu, W. Huang, and L. Nie, “Dynamic user recruitment in edge-aided mobile crowdsensing,” IEEE Trans. Veh. Technol., vol. 72, no. 7, pp. 9351–9365, Jul. 2023.

[47] Y. Li et al., “PTASIM: Incentivizing crowdsensing with POI-Tagging cooperation over edge clouds,” IEEE Trans. Ind. Inform., vol. 16, no. 7, pp. 4823–4831, Jul. 2020.

[48] J. Zhang and X. Zhang, “Multi-task allocation in mobile crowd sensing with mobility prediction,” IEEE Trans. Mobile Comput., vol. 22, no. 2, pp. 1081–1094, Feb. 2023.

[49] W. Zhang, Z. Zhang, S. Zeadally, H.-C. Chao, and V. C. M. Leung, “Energy-efficient workflow allocation and computation resource configuration in distributed cloud/edge computing systems with stochastic workloads,” IEEE J. Sel. Areas Commun., vol. 38, no. 6, pp. 1118–1132, Jun. 2020.

Minghui Liwang (Member, IEEE) is currently an associate professor with the Department of Control Science and Engineering, the National Key Laboratory of Autonomous Intelligent Unmanned Systems, and also with Frontiers Science Center for Intelligent Autonomous Systems, Ministry of Education, Tongji University, Shanghai, China. Her research interests include multi-agent systems, edge computing, distributed learning, as well as economic models and applications.

Zhibin Gao (Member, IEEE) received the BS degree in communication engineering, the MS degree in radio physics, and the PhD degree in communication engineering from Xiamen University, Xiamen, China, in 2003, 2006, and 2011, respectively. He is a professor of Navigation Institute, Jimei University, Xiamen, China. Previously, he worked with Xiamen University, Xiamen, China, as a senior engineer of communication engineering and he was a visiting scholar with the University of Washington, from 2016 to 2017. His research interests include Internet of Vehicles, marine communications, wireless network resource management, and signal processing.

Seyyedali Hosseinalipour (Member, IEEE) is currently an assistant professor with the Department of Electrical Engineering, University at Buffalo, SUNY, Buffalo, NY, USA. His research interests include 6G, machine learning, federated learning, fog and edge computing, and network optimization.

Zhipeieng Cheng is an assistant professor with the School of Future Science and Engineering, Soochow University, Suzhou, China. His research interests include UAV communication networks, cloud/edge/service computing, reinforcement learning and distributed machine learning applications.
Xianbin Wang (Fellow, IEEE) received the PhD degree in electrical and computer engineering from the National University of Singapore, in 2001. He is a professor and a Tier-1 Canada research chair in 5G and Wireless IoT Communications with Western University, Canada. Prior to joining Western University, he was with the Communications Research Centre Canada as a research scientist/senior research scientist from 2002 to 2007. From 2001 to 2002, he was a system designer with STMicroelectronics. His current research interests include 5G/6G technologies, Internet of Things, machine learning, communications security, and intelligent communications. He has more than 600 highly cited journals and conference papers, in addition to more than 30 granted and pending patents and several standard contributions. Dr. Wang is a fellow of the Canadian Academy of Engineering and a fellow of the Engineering Institute of Canada. He has received many prestigious awards and recognitions, including the IEEE Canada R. A. Fessenden Award, Canada research chair, Engineering Research Excellence Award with Western University, Canadian Federal Government Public Service Award, Ontario Early Researcher Award, and nine Best Paper Awards. He is currently a member of the Senate, Senate Committee on Academic Policy and Senate Committee on University Planning at Western. He also serves on NSERC Discovery Grant Review Panel for Computer Science. He has been involved in many flagship conferences, including GLOBECOM, ICC, VTC, PIMRC, WCNC, CCECE, and ICNC, in different roles, such as general chair, TPC chair, Symposium chair, Tutorial Instructor, track chair, session chair, and Keynote speaker. He serves/has served as the editor-in-chief, associate editor-in-chief, and editor/associate editor for more than ten journals. He was the chair of the IEEE ComSoc Signal Processing and Computing for Communications (SPCC) Technical Committee and is currently serving as the Central Area Chair of IEEE Canada.

Zhenzhen Jiao (Senior Member, IEEE) received the PhD degree from the University of Chinese Academy of Sciences in computer science. He was an associate professor with the Institute of Computing Technology, Chinese Academy of Sciences and the Director of a blockchain research laboratory. He is currently the director of the Teleinfo iF-Labs in China Academy of Information and Communications Technology. His research interests include blockchain, self-organized systems and networks, and computing power network.