Unifying Futures and Spot Market: Overbooking-Enabled Resource Trading in Mobile Edge Networks

Minghui Liwang, *Member, IEEE*, Ruitao Chen, Xianbin Wang, *Fellow, IEEE*, and Xuemin Shen, *Fellow, IEEE*

***Abstract*— Securing necessary resources for edge computing processes via effective resource trading becomes a critical technique in supporting computation-intensive mobile applications. Conventional onsite spot trading could facilitate this paradigm with proper incentives, which, however, incurs excessive decision-making latency/energy consumption, and further leads to underutilization of dynamic resources. Motivated by this, a hybrid market unifying futures and spot is proposed to facilitate resource trading among an edge server (seller) and multiple smart devices (buyers) by encouraging some buyers to sign a forward contract with seller in advance, while leaving the remaining buyers to compete for available resources with spot trading. Specifically, overbooking is adopted to achieve substantial utilization and profit advantages owing to dynamic resource demands. By integrating overbooking into futures market, mutually beneficial and risk-tolerable forward contracts with appropriate overbooking rate can be achieved relying on analyzing historical statistics associated with future resource demand and communication quality, which are determined by an alternative optimization-based negotiation scheme. Besides, spot trading problem is studied via considering uniform/differential pricing rules, for which two bilateral negotiation schemes are proposed by addressing both non-convex optimization and knapsack problems. Experimental results demonstrate that the proposed mechanism achieves mutually beneficial player's utilities, while outperforming baseline methods on critical indicators, e.g., decision-making latency, resource usage, etc.**

***Index Terms*— Futures trading, spot trading, overbooking, forward contract, mobile edge networks.**

Manuscript received May 22, 2021; revised September 28, 2021; accepted January 2, 2022. Date of publication January 14, 2022; date of current version July 12, 2022. This work was supported in part by the Discovery Program of Natural Sciences and Engineering Research Council of Canada (NSERC) under Grant RGPIN2018-06254 and in part by the Canada Research Chair Program. The associate editor coordinating the review of this article and approving it for publication was A. Schmeink. *(Corresponding author: Xianbin Wang.)*

Minghui Liwang was with the Department of Electrical and Computer Engineering, Western University, London, ON N6A 5B9, Canada. She is now with the Department of Information and Communication Engineering, Xiamen University, Xiamen 361005, China (e-mail: minghuilw@xmu.edu.cn).

Ruitao Chen and Xianbin Wang are with the Department of Electrical and Computer Engineering, Western University, London, ON N6A 5B9, Canada (e-mail: rchen328@uwo.ca; xianbin.wang@uwo.ca).

Xuemin Shen is with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: sshen@uwaterloo.ca).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TWC.2022.3141094.

Digital Object Identifier 10.1109/TWC.2022.3141094

I. Introduction

THE evolution of wireless technologies and explosive proliferation of smart devices have enabled a wide range of mobile applications [1], [2], e.g., online gaming, augmented/virtual reality and healthcare monitoring, etc., which have attracted significant number of users. However, many of the aforementioned applications are computation-intensive and require complicated onboard processing, posing great challenges to smart devices with limited computing resources and capability. Besides, limited battery power supply presents another major difficulty for intensive data processing, exchange, and decision-making on a single mobile device, that may hinder the application completion in real-time [3]–[5]. One feasible solution to overcome these challenges is cloud computing, which, however, may potentially incur transmission delays, and burdens on cloud servers as well as backhaul links [1]. To further address these drawbacks, edge computing [3], [4], [6] has become a popular paradigm by exploring distributed computing/storage/communication capability at the edge of mobile networks, and thus offers flexible and cost-effective computing services for resource-constrained smart devices.

A. Motivation

Ensuring the needed resources for edge-assisted computing processes often relies on a certain form of resource trading, where a smart device can offload a certain amount of task data to the edge server, via wireless link to the nearby access point (AP, e.g., base station, roadside unit, etc.), while paying for the acquired resources and computing services. However, conventional trading mechanisms (e.g., onsite spot trading) may face significant challenges caused by the dynamic nature of resource trading market under mobile edge network architecture.

1) Motivation of Futures-Based Resource Trading: To facilitate resource provisioning with proper incentives, onsite spot trading has been widely adopted which allows resource buying and selling among sellers (e.g., servers) and buyers (e.g., mobile devices, where sellers and buyers are collectively known as players) under real-time and on-demand mode. Specifically, players in spot trading can reach an agreement on factors such as the amount of trading resources and the relevant price based on current network/platform/application-related conditions, e.g., resource supply/demand and wireless

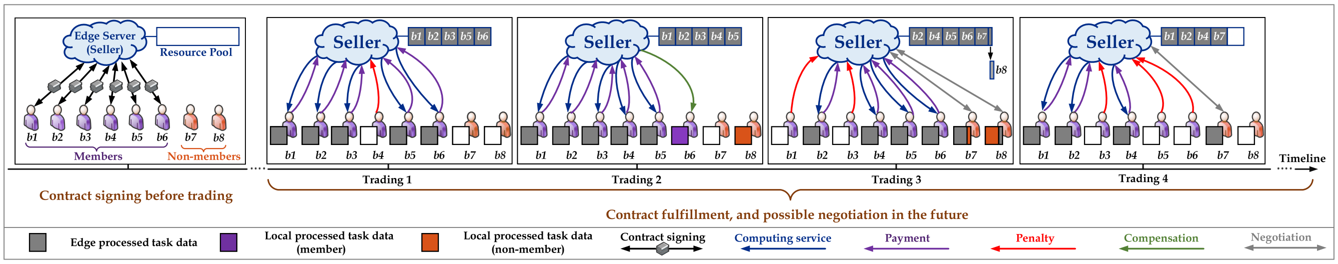

Fig. 1. The proposed overbooking-enabled resource trading framework, timeline, and several trading examples ($S = 5$, $|\boldsymbol{\mathcal{B}}| = 8$, $\kappa = 6$). Specifically, no onsite decision-making latency can be costed in Trading 1 and Trading 2, while $\boldsymbol{b_4}$ has to pay penalty for "no show" (Trading 1); $\boldsymbol{b_6}$ receives compensation due to insufficient resource supply (Trading 2). Trading 3 and Trading 4 have incurred certain decision-making latency, where $\boldsymbol{b_1}$ and $\boldsymbol{b_3}$ pay penalty for "no show" while $\boldsymbol{b_7}$ and $\boldsymbol{b_8}$ compete for available resources under onsite mode (Trading 3); besides, $\boldsymbol{b_3}$, $\boldsymbol{b_5}$ and $\boldsymbol{b_6}$ have to pay penalty while $\boldsymbol{b_7}$ negotiates with the seller for possible price and the amount of trading resources without competing with others (Trading 4).

channel quality at present. However, spot trading may lead to undesirable performance degradation, which are detailed below:

- *Latency on decision-making*: Spot players often have to spend extra latency to reach the final trading consensus during a trading, which may dramatically reduce the available time for dynamic resource sharing, especially for networks with moving devices and limited trading period, as well as time-varying resources. For example, considering 2 seconds as the connection duration between a smart device and the nearby AP, while 1.5 seconds of which has been costed for onsite decision-making. Under this circumstance, there are only 0.5 seconds left for the actual computing service. Besides, resource condition (e.g., supply and demand) can also be fluctuant during a long decision-making procedure. Thus, timely resource provisioning represents a major challenge under dynamic networks when applying spot trading mechanism.
- *Energy consumption on decision-making*: During each trading, spot players may suffer from extra energy consumption to reach the trading consensus, bringing difficulties to power- and battery-constrained mobile devices. For example, a long decision-making procedure can cause certain battery loss, that directly decreases the endurance of a smart device. Consequently, designing an energy-efficient trading mechanism is considered to be urgent and critical.

Motivated by the abovementioned challenges, *futures trading* [5] is considered as an effective *presale* mode, which enables a forward contract between the resource owner and each resource requestor, with contract terms such as the amount of trading resources, the relevant resource price, and default clause, etc., via analyzing historical data. Specifically, the pre-signed contract will be fulfilled accordingly during each practical trading in the future, without any further onsite discussion. Nevertheless, although futures can achieve commendable decision-making latency and energy consumption, it may also be risky due to the insufficient and inaccurate knowledge of historical statistics. Thus, this work investigates a novel hybrid market via unifying both futures trading and spot trading under mobile edge network architecture.

2) Motivation of Overbooking: One conventional resource presale paradigm refers to equal-booking, where the amount of booked resources generally does not exceed the resource owner's maximal resource supply, which, however, may face significant challenges incurred by "no show"[1] of resource requestors. Specifically, "no show" is generally caused by factors such as the mobility of mobile devices, and unreliable wireless communication links, etc., which could result in variation of resource demands, and thus prevent the utilization of confirmed resources. For example, a smart device that has run out of power will not have task execution requirement during a trading and thus no longer need to use the reserved resources stipulated in pre-signed contract as usual. Besides, unpredictable mobility and willingness of buyers can also lead to "no show", e.g., a mobile device outside of any AP's coverage is facing difficulties to get access to resources of the edge server, which is thus absent from the trading. Therefore, *overbooking* [7] has been adopted in this paper, that encourages a certain overbooking rate, where the total available capacity of resource seller can be less than the amount of resources booked by multiple buyers, since some of them may not be able to consume the resources they are entitled to, or may cancel a trading ("no show") [8]. Although overbooking may seem risky, it represents a popular practice in a wide range of commercial domains such as airlines [9], hotels [10], bandwidth reservation [11], [12], etc. For instance, aiming to maximize the occupancy (and thus revenue), airlines routinely overbook tickets by ensuring the maximum number of passengers on a flight; otherwise, flights often depart with up to 15% seats empty (without overbooking), and thus incurs unsatisfying resource utilization and economic losses [8]. Take Fig. 1 as an example, although the theoretical maximal resource supply of the seller can only process 5 tasks in parallel, it overbooks resources to 6 buyers in order to deal with possible "no show" cases, where the corresponding overbooking rate can be calculated by $(6-5)/5 = 20\%$. In generall, overbooking can efficiently achieve substantial utilization and profit advantages in supporting dynamic and unpredictable resource demands in a trading market.

Driven by the abovementioned motivations, this paper investigates a hybrid market via unifying both futures and spot trading under mobile edge networks, and proposes a novel overbooking-enabled resource trading mechanism. Specifically, we consider an edge server with limited resource supply as resource seller, and multiple smart devices with computation-intensive tasks as resource buyers, each of which

[1] In real-life networks, "no show" represents a common phenomenon in many sectors such as airlines and hotels, indicating that a person who has booked a flight ticket or a hotel room but finally does not show up.

may purchase computing service from the seller by offloading certain amount of task data through wireless communication. The proposed trading mechanism effectively alleviates the unexpected latency and cost (e.g., energy and battery consumption) on trading decision-making, and greatly improves the resource utilization and time efficiency.

B. Related Work

1) Resource Trading Mechanism: Existing works devoted to resource trading roughly fall into three categories: *i).* spot trading (also known as onsite trading), where players reach a trading agreement relying on the current conditions, such as online game [13], [14], auction [15]–[17], and bilateral negotiation [18], [19]; *ii).* futures trading, where players sign forward contracts over buying or selling a certain amount of resources at predetermined price in advance, that will be fulfilled during each trading in the future, where existing studies mainly investigate electricity market [20]–[22], spectrum resource trading [23], [24], and edge computing-assisted networks [5], [25]; and *iii).* resource trading in hybrid market where both futures and spot trading are allowed [26], [27]. A multi-user non-cooperative offloading game was investigated by *Wang et al*. [14], intended to maximize the utility of each vehicle via a distributed best response algorithm. The VM auction among edge clouds and mobile users was studied as an n-to-one weighted bipartite graph matching problem by *Gao et al*. in [15], based on a greedy approximation algorithm. In [16], *Liwang et al*. studied a Vickrey–Clarke–Groves-based reverse auction mechanism of vehicle-to-vehicle resource trading and suggested a unilateral matching-based mechanism. In [17], *Gao et al*. developed a truthful auction under computing resource trading market via considering graph tasks, while providing both the optimal and an efficient sub-optimal algorithms. *Shojaiemehr et al*. in [18] proposed a novel negotiation strategy to enhance the satisfaction of both trading parties while supporting negotiation of composite cloud service. In [19], *Wang et al*. presented a smart contract-based negotiation framework while providing a Bayesian Nash equilibrium of service providers which offer flexible QoS. However, the procedure to reach a trade-related decision usually results in excessive latency and energy consumption [5], [23], [28], which further pose challenges to spot trading players. Take online auction as an example, the winners gain the eventual auction contract while there is no such compensation for the losers who have also spent extra time and energy during decision making. Moreover, the latency from bidding to practical computing service delivery can greatly impact the quality of experience and the utilization if resources are reserved while waiting for the auction results [5], [8], [23]. Therefore, futures has been emerged as a practical paradigm and extensively adopted in commodity exchange markets. Benefit from the pre-signed forward contracts, the unexpected latency/energy consumption on decision-making can be efficiently decreased. *Khatib et al*. in [20] proposed a systematic negotiation scheme, through which, a generator and load can reach a mutually beneficial forward bilateral contract in electricity markets. In [21], *Conejo et al*. addressed the power producer's optimal involvement problem in a futures electricity market, aiming to hedge against the risk of pool price volatility. In [22], *Morales et al*. investigated scenario reduction techniques to accurately convey the uncertainties in futures market trading in electricity markets. In spectrum resource trading market, *Sheng et al*. in [23] proposed a futures-based spectrum trading mechanism to alleviate trading failures, and trading unfairness caused by price fluctuation. In [24], *Li et al.* introduced a futures market to manage the financial risk in spectrum trade and discovering future price. Topics associated with futures-based resource trading have rarely been studied in mobile edge networks, where factors such as unpredictable nature of resource supply and demand, as well as the ever-changing channel quality between resource provider and requestor caused by mobile users' mobility, pose difficulties to trading mechanism design. We were among the first to address such challenges [5], [25]. In our previous work [25], we investigated a futures-based resource trading approach in edge computing-enabled internet of vehicles, where a risk-tolerable and mutually beneficial forward contract was designed through estimating the historical statistics of future resource supply/network condition. An energy-aware resource trading mechanism under edge computing-assisted UAV networks was proposed [5], where both forward contract design and power optimization problems were analyzed.

Although futures brings benefits, it may also be risky due to the lacking and inaccurate knowledge of historical data. Motivated by which, several works also consider the integration of both the futures and spot market. In [26], *Gao et al*. focused on the optimal spectrum allocation among unlicensed secondary users in a hybrid market, which maximized the secondary spectrum utilization efficiency. *Vanmechelen et al*. in [27] proposed a hybrid market in which a low-latency spot market coexists with a higher latency futures market, to deal with the significant delay of the allocation decision procedure of grid resources.

2) Overbooking: "Booking" refers to a presale manner (rather than spot trading), where "overbooking" presents the presale of a volatile commodity or service in excess of actual supply, which has been shown to provide substantial utilization and profit advantage in handling "no shows" [8], which, however, has been neglected in most previous mentioned works (namely, these works mainly consider equal-booking where the amount of resources for sale equals to the actual supply). The widespread adoption of overbooking techniques focus on many fields such as airlines and hotels [9], [10], spectrum reservation [11], [12], storage market [29], network slicing [30]–[32], cloud computing [7], [33]–[36], and fog computing [37]. Specifically, *Liu et al*. in [11] proposed an opportunistic link overbooking scheme for an edge gateway to improve its link efficiency, and developed an integrated analytical framework for determining the suitable link overbooking factor. In [12], *Adebayo et al*. proposed a spectrum reservation prediction algorithm for wireless infrastructure providers to reduce the probability of overbooking since it costs certain penalties. *Gao et a*l. in [29] proved that overbooking strategy plays an important role in improving storage renting efficiency. *Zanzi et al*. in [30] deployed an overbooking network slices solution and a 5G network slice broker as an entity in charge of mediating between vertical network slice requests and physical network resources availability. Additionally, in [31], *Zanzi et al*. proposed an orchestration through a dashboard, allowing requesting network slices on-demand, monitored their performance once deployed and displayed the achieved multiplexing gain through overbooking. In [32], *Sexton et al*. employed the practice of overbooking to increase resource utilization when offering auxiliary resources in network slicing. Among existing works considered overbooking, the most similar studies with this work fall into cloud computing environment such as [7], [33]–[36], and fog computing [37]. *Tomas et al*. in [7] focused on implementing an autonomic risk-aware overbooking architecture capable of increasing the resource utilization of cloud data centers by accepting more virtual machines than physical available resources. In [33], *Son et al*. proposed a service level agreement (SLA)-aware dynamic overbooking strategy in software defined networking (SDN)-based cloud data centers, which jointly leveraged virtualization capabilities and SDN for virtual machine (VM) and traffic consolidation. *Alanazi et al*. in [34] introduced an integrated resource allocation framework for data centers that minimizes the number of active physical machines through dynamic VM placement while ensuring that SLAs of admitted VMs are not violated. *Rahimzadeh et al*. proposed a cloud resource management system that overbooks backup VMs by optimizing the overbooking rate tradeoff in [35]. In [36], *Yao et al*. presented an optimal overbooking policy to maximize resource providers' profits in cloud federation and enhance cloud users' experiences. *Zhang et al*. [37] studied a dynamic resource allocation model through overbooking mechanism to maximize the total welfare of fog servers. Although "resource overbooking" has been applied in several fields and achieves good performance, few of them paid attention to computing resource trading problem in edge networks. Besides, characteristic features in wireless communications, e.g., varying channel qualities, etc., also bring difficulties to trading mechanism design.

C. Novelty and Contribution

To the best of our knowledge, this paper is among the first to study overbooking-enabled resource trading among an edge server (resource seller) and multiple smart devices (resource buyers), via considering a hybrid market integrating both futures and spot trading modes. Specifically, in futures market, two major issues are considered: *i)*. overbooking rate design: players determine a feasible overbooking rate, which can be reflected by the number of buyers that can sign the forward contract with the seller (we name these buyers as members); and *ii)*. forward contract design: players determine a reasonable forward contract, which includes the price of resource, the penalty that a member has to pay to the seller if it breaks the forward contract, and the compensation that a member with task can receive if the seller cannot offer computing service due to overbooking. In spot market, the remaining buyers who have not signed forward contract (we name these buyers as non-members) can compete for available resources (if any) based on the current network/market conditions. Major contributions are summarized as follows:

• This paper introduces a novel hybrid resource trading market via integrating both futures and spot modes under mobile edge network architecture, which effectively alleviates extra latency and cost (e.g., energy consumption, etc.) on trading decision-making. Specifically, overbooking is adopted which allows a larger number of members than the seller's maximal capacity, that greatly supports the improvements on both resource utilization and time efficiency.

• To capture the unpredictable random nature of the resource trading market, two key uncertainties are considered: buyer's task arrival (namely, the participation of each buyer), which directly affects the resource demand; and the varying wireless channel quality, which reflects the unstable network condition caused by factors such as the mobility of each buyer. Specifically, buyers are divided into members and non-members, where members can sign a forward contract with the seller in advance, which will be fulfilled during each future practical trading; while non-members with tasks have to compete for available resources under a spot trading manner.

• The proposed mechanism considers solving two key problems associated with different markets. The resource trading problem in futures market mainly relies on designing the feasible forward contract and overbooking rate, which is formulated as a multi-objective optimization (MOO) problem aiming to maximize both the seller's and the members' expected utilities, via analyzing historical statistics of the abovementioned key uncertainties (task arrival condition and wireless channel quality). Moreover, possible risks that players may undergo during each practical trading are evaluated as constraints. To tackle this problem, an efficient bilateral negotiation scheme is proposed that facilitates the players reaching a consensus on futures trading.

• In spot market, resource trading is defined as a MOO problem that aims to maximize the seller's and each non-member's utilities, based on the current resource supply and demand, as well as wireless channel qualities, upon considering both uniform pricing and differential pricing rules. To address the spot trading problem under various pricing rules, we propose a bilateral negotiation-based scheme through solving non-convex optimization problem and knapsack problem, within polynomial time.

• Comprehensive simulation results demonstrate that the proposed overbooking-enabled resource trading mechanism in hybrid market achieves mutually beneficial players' utilities, while outperforming baseline methods on significant indicators such as decision-making latency and cost, task completion time, as well as time and resource utilization.

II. System Model

A. Key Definition and System Overview

Considering a futures and spot integrated resource trading market containing multiple resource buyers represented by set $\boldsymbol{\mathcal{B}} = \left\{\boldsymbol{b_1}, \ldots \boldsymbol{b_m}, \ldots, \boldsymbol{b_{|\mathcal{B}|}}\right\}$, where each buyer may have a computation-intensive task during a trading; and a resource seller with limited computing resources denoted by Sd^{comp} (e.g., CPU cycles), where S represents a positive integer, and d^{comp} indicates the required amount of computing resources per task. Specifically, the overall resource demand may exceed the available resource supply in the studied market (namely, $S < |\boldsymbol{\mathcal{B}}|$). Fig. 1 shows the corresponding framework, timeline, and several trading examples associated with the proposed resource trading; while key definitions are introduced as follows:

Definition 1 (Futures Market and Member): In futures market, some of the buyers can sign a forward contract with *the seller in advance, which will be fulfilled during each practical trading without any further negotiation. Correspondingly, a buyer with a forward contract is named as a member, and the number of members is denoted by κ.*

Definition 2 (Spot Market and Non-Member): In spot market, a buyer without forward contract can purchase computing resources from the seller under a real-time and on-demand mode through spot trading, we regard these buyers as non-members.

Definition 3 (Performer): A performer indicates a member who has task execution requirement during a trading.

Definition 4 (Practical Performer): A practical performer indicates a performer who can practically obtain the required resources and service offered by the seller during a trading.

Definition 5 (Defaulter): A defaulter indicates a member who is absent from a trading ("no show"), although it has signed a forward contract with the seller in advance. Thus, each defaulter has to pay penalty to the seller for breaking the forward contract.

Definition 6 (Volunteer): A volunteer indicates a performer who has to process its task locally since the seller fails to afford performers' task execution requirements due to overbooking; correspondingly, each volunteer will receive compensation from the seller.

Definition 7 (Forward Contract and Contract Term): A forward contract represents a trading agreement between the seller and each member, which will be fulfilled accordingly during each practical trading. In a contract, three key terms are considered: p, q, and r, where p indicates the agreed unit price of resources, q denotes the unit penalty that a defaulter has to pay to the seller, and r refers to the unit compensation for each volunteer from the seller ($r > 0$). Specifically, p is supposed to be larger than q, which also corresponds to real-life trading market since a buyer definitely prefers to purchase a commodity at p since a high penalty will lead to worse utility, which may result in unsatisfying trading experience and a waste of limited resources.

Definition 8 (Overbooking Rate): The overbooking rate κ^o denotes the ratio of overbooked resources to the overall resource supply, which can be calculated by $\kappa^o = (\kappa - S)/S$ (e.g., $\kappa^o = 20\%$ associated with futures market in Fig. 1).

Notably, a feasible overbooking rate can greatly support time efficiency and substantial resource utilization. For example, at most two non-members cost latency on onsite decision-making as shown by Fig. 1, rather than three if considering equal-booking. Besides, the proposed overbooking-enabled trading mechanism achieves 100% resource usage in Trading 1, Fig. 1, although one of the members is absent ("no show") from the trading.

This paper investigates an interesting resource trading mechanism via considering the following key problems under different markets: *i).* futures market focuses on designing a risk-aware and mutually beneficial forward contract between seller and each buyer, as well as the feasible overbooking rate κ^o (namely, an appropriate number of members κ) by analyzing historical statistics (e.g., the task arrival of each buyer and channel quality between each buyer and the seller), to maximize the expected utilities of the two parties; and *ii).* spot market pays attention to design the spot trading mechanism among seller and non-members upon considering different pricing rules, to help players obtain better utilities under real-time and on-demand manner.

B. Modeling of Buyers

Considering a set of buyers $\boldsymbol{\mathcal{B}}$, where each buyer $\boldsymbol{b_m} \in \boldsymbol{\mathcal{B}}$ may have a task that needs to be processed during a trading, denoted by a 7-tuple $\boldsymbol{b_m} = \{d^{size}, d^{comp}, f^b, e^{loc}, e^{tran}, \alpha_m, \gamma_m\}$. Specifically, d^{size} and d^{comp} indicate the data size (e.g., bits), and the required amount of computing resources (e.g., CPU cycles) of the buyer's task, respectively.[2] f^b denotes the local computing capability (CPU cycles/s) of each buyer, e^{loc} describes the local computing power consumption (Watt), which represents the average energy cost per unit time of local computing [50] via considering several factors, e.g., power consumption of local computing/storage, standby power of mobile device, etc; while e^{tran} represents the transmission power (Watt) of $\boldsymbol{b_m}$. Two key uncertainties are considered to describe the random and unpredictable nature of the trading process: α_m and γ_m.[3] Specifically, $\alpha_m \in \{0,1\}$ represents each buyer's task arrival during each trading (namely, if the buyer requires resources and will attend a trading, or not), which is a discrete random variable obeying the Bernoulli distribution denoted by $\alpha_m \sim \mathrm{B}(\{1,0\},\{a,1-a\})$. Thus, probability mass function (PMF) of α_m is given by (1).

$$\Pr(\alpha_m = i) = \begin{cases} a, & i = 1 \\ 1-a, & i = 0 \end{cases} \tag{1}$$

To better capture the uncertainty of wireless communications, as well as the unpredictable mobility of mobile devices (e.g., a general scenario is considered without caring about the exact moving path of each buyer), γ_m is applied to describe the varying channel quality between buyer $\boldsymbol{b_m}$ and the nearby AP. Specifically, γ_m is modeled as a continuous random variable (that can be seen as a function of factors such as fading, path loss and noise [1]), which obeys an uniform distribution [5], [23] in interval $[\varepsilon_1, \varepsilon_2]$, denoted by $\gamma_m \sim \mathrm{U}(\varepsilon_1, \varepsilon_2)$. Notably, we assume that buyers are independent and identically distributed (i.i.d),[4] e.g., buyers may have similar daily activities such as workers in Apple park. More importantly, resource demand during each trading relies on the overall participations of buyers rather than a single buyer, which may follow a different distribution such like an approximate discrete Gaussian distribution upon considering $|\boldsymbol{\mathcal{B}}| = 4$ and $a = 50\%$. For notational simplicity, let $\boldsymbol{\mathcal{A}} = \{\alpha_1, \ldots, \alpha_m, \ldots, \alpha_{|\boldsymbol{\mathcal{B}}|}\}$ and

[2]For analytical simplicity, this paper assumes the same task data size and thus the amount of computing resources [1], [5]. Specifically, the proposed overbooking-enabled resource trading mechanism can also be applied when considering different data sizes, where a large task can be divided into several virtual tasks and each of which is of the same data size, although mathematical derivations of some models should be updated (e.g., the expected utility of each member, risks associated with members). Several existing works with similar ideas can support this assumption [5], [17], [51].

[3]In this paper, trading statistics of uncertainties α_m and γ_m are supposed to be known based on the historical records [5], [23].

[4]The proposed problems and solution designs can also be applied when considering non-i.i.d buyers, although some calculations should be updated. Besides, new problems such as member selection may be incurred. However, these changes will not impact our basic intention of unifying both futures and spot market, as well as the overbooking-enabled trading mechanism. Individual personalities and possible cooperations among buyers will be investigated in our next work.

$\boldsymbol{\mathcal{Y}} = \{\gamma_1, \ldots, \gamma_m, \ldots, \gamma_{|\boldsymbol{\mathcal{B}}|}\}$ denote the vectors of random variables α_m and γ_m.

1) Task Completion Time and Energy Consumption: For each buyer, the local task completion time is calculated as $t^{loc} = \frac{d^{comp}}{f^b}$, and the relevant local energy consumption is thus given by $c^{loc} = e^{loc} t^{loc} = \frac{e^{loc} d^{comp}}{f^b}$ [1], [3], [4], [38], [50]. Additionally, task completion time of buyer $\boldsymbol{b_m}$ when it offloads a certain amount of task data to the seller is defined by the following equation (2), where $e^{tran}\gamma_m$ indicates the received SNR [39] of the AP from buyer $\boldsymbol{b_m}$.

$$t_m^{edge} = \left(\frac{\lambda_m d^{size}}{W\log_2(1+e^{tran}\gamma_m)} + \frac{\lambda_m d^{comp}}{f^s}, \frac{(1-\lambda_m)d^{comp}}{f^b}\right)^+ \quad (2)$$

where λ_m $(0 \leqslant \lambda_m \leqslant 1)$ denotes the offloading rate of $\boldsymbol{b_m}$; symbol $(i,j)^+$ refers to the larger value between i and j; $\lambda_m d^{size}$ and $\lambda_m d^{comp}$ denote the amount of data offloaded to the seller, and the relevant required resources, respectively. W represents the bandwidth of the wireless channel[5] between each buyer and the seller, and $W\log_2(1+e^{tran}\gamma_m)$ indicates the relevant data transmission rate. Moreover, f^s depicts the seller's computing capability (e.g., CPU cycles/s), which is considered as a stable value (e.g., the value of f^s doesn't change with seller's workloads), as supported by existing works [1], [3]. Correspondingly, the relevant energy consumption c_m^{edge} of $\boldsymbol{b_m}$ is defined by the following (3), where the left item denotes the energy costed by uploading a proportion of task data to the seller, while the right item represents the overall local energy consumption on handling the remaining task data by the buyer itself.

$$c_m^{edge} = \frac{e^{tran}\lambda_m d^{size}}{W\log_2(1+e^{tran}\gamma_m)} + \frac{e^{loc} \times (1-\lambda_m) \times d^{comp}}{f^b} \quad (3)$$

2) Utility, Expected Utility, and Risks of Member in Futures Market: We use m to represent the index of members hereafter, to avoid notational redundancy. For analytical simplicity, the first κ buyers $\boldsymbol{b_1}, \boldsymbol{b_2}, \ldots, \boldsymbol{b_\kappa}$ are considered as members,[6] which are encouraged to offload the whole task[7] to the seller (e.g., $\lambda_m = 1$). Correspondingly, the utility U_m^{PP} of a member $\boldsymbol{b_m}$ who is a practical performer is defined as the weighted sum of the time and energy saved from enjoying the computing service, minus the payment of required resources, as given by (4):

$$U_m^{PP} = \omega_1\left(t^{loc} - t_m^{edge}\right) + \omega_2\left(c^{loc} - c_m^{edge}\right) - pd^{comp}, \quad (4)$$

where ω_1 and ω_2 are positive weight coefficients that reflect the sensitiveness on latency and energy consumption associated with each buyer. Correspondingly, the utility U^{DE} of a defaulter can be calculated as $U^{DE} = -qd^{comp}$, indicating that a member has to pay a penalty when it breaks the forward contract. Owing to overbooking, some of the performers with the worst channel qualities[8] may be selected as volunteers when the seller fails to support task execution requirements of members during a trading (e.g., $\sum_{m=1}^{m=\kappa}\alpha_m > S$). Thus, we define the utility of each volunteer as $U^{VO} = rd^{comp}$, indicating the compensation from the seller. Correspondingly, the number of volunteers V can be expressed by (5).

$$V = \sum\nolimits_{m=1}^{m=\kappa}\alpha_m - \left(\sum\nolimits_{m=1}^{m=\kappa}\alpha_m, S\right)^- \quad (5)$$

Let v_m be the volunteer selection indicator, where $v_m = 1$ denotes $\boldsymbol{b_m}$ is chosen as a volunteer; $v_m = 0$, otherwise. Then, the utility of member $\mathcal{U}^{Mem}(p,q,r,\kappa,\boldsymbol{\mathcal{A}},\boldsymbol{\mathcal{Y}})$ is formulated by the following (6).

$$\begin{aligned}\mathcal{U}^{Mem}&(p,q,r,\kappa,\boldsymbol{\mathcal{A}},\boldsymbol{\mathcal{Y}}) \\ &= \sum\nolimits_{m=1}^{m=\kappa}\left(\alpha_m U_m^{PP} + (1-\alpha_m)U^{DE}\right) \\ &\quad - \sum\nolimits_{m=1}^{m=\kappa} v_m U_m^{PP} + U^{VO}V \end{aligned} \quad (6)$$

Since the random nature of the resource trading market poses great challenges to maximize the members' utility directly, we consider the expected value of $\mathcal{U}^{Mem}(p,q,r,\kappa,\boldsymbol{\mathcal{A}},\boldsymbol{\mathcal{Y}})$ as (7). Notably, (7) does not consider which specific member will be chosen as a volunteer.

$$\begin{aligned}&\overline{\mathcal{U}^{Mem}}(p,q,r,\kappa,\boldsymbol{\mathcal{A}},\boldsymbol{\mathcal{Y}}) \\ &= \mathrm{E}\left[\sum\nolimits_{m=1}^{m=\kappa}\left(\alpha_m U_m^{PP} + U^{DE} - \alpha_m U^{DE}\right)\right] \\ &\quad + \mathrm{E}[V]\left(\mathrm{E}\left[U^{VO}\right] - \mathrm{E}\left[U_m^{PP}\right]\right) \\ &= \kappa\mathrm{E}[\alpha_m] \times \mathrm{E}\left[U_m^{PP}\right] - \kappa qd^{comp} + \kappa qd^{comp}\mathrm{E}[\alpha_m] \\ &\quad + rd^{comp}\mathrm{E}[V] - \mathrm{E}\left[U_m^{PP}\right]\mathrm{E}[V], \end{aligned} \quad (7)$$

where $\mathrm{E}[\cdot]$ denotes the mathematical expectation, and we can simply have $\mathrm{E}[\alpha_m] = a$. Specifically, $\mathrm{E}[V]$ is given by (8), shown at the bottom of the next page, and $\mathrm{E}\left[U_m^{PP}\right]$ is calculated by (9),

$$\begin{aligned}\mathrm{E}\left[U_m^{PP}\right] &= \left(\frac{\omega_1 + \omega_2 e^{loc}}{f^b} - \frac{\omega_1}{f^s} - p\right)d^{comp} \\ &\quad - \frac{\ln 2 d^{size}(\omega_1 + \omega_2 e^{tran}) \times \int_{\mathbb{C}_1}^{\mathbb{C}_2}\left(\frac{e^y}{y}\right)dy}{We^{tran}(\varepsilon_2 - \varepsilon_1)}, \end{aligned} \quad (9)$$

where $\mathbb{C}_1 = \ln 2 \times \log_2(1+e^{tran}\varepsilon_1)$ and $\mathbb{C}_2 = \ln 2 \times \log_2(1+e^{tran}\varepsilon_2)$, for notational simplicity. Detailed derivations of (8) and (9) are given in Appendix A. Combine (7), (8), and (9), we rewrite (7) as (10) and (11) considering $\kappa \leqslant S$

[5]For analytical simplicity, we do not consider interference of wireless communication, as supported by existing works [1], [40], and innovative techniques such as OFDMA [41].

[6]In the proposed resource trading market, sign the forward contract with any κ of the buyers has no impact on the solution design since all the buyers are i.i.d. For example, considering $|\mathcal{B}| = 6$ and $\kappa = 3$, the seller contracts with buyers $\boldsymbol{b_1}$, $\boldsymbol{b_2}$, and $\boldsymbol{b_3}$ makes no difference with that with buyers $\boldsymbol{b_4}$, $\boldsymbol{b_5}$, and $\boldsymbol{b_6}$.

[7]Notably, members are often given more rights, e.g., there is generally no quantitative limitation on purchasing commodities, while a non-member may suffer from the relevant limitation. Thus, we consider binary offloading for members, namely, members own the right to enjoy more resources (and thus computing service) under a reasonable price; while non-members have to compete for available limited resources where partial offloading is allowed in the proposed spot market. More importantly, applying different offloading rules will not impact our problem although some mathematical derivations may have to be updated.

[8]In this paper, we consider a fair volunteer selection scheme where the performers with the worst channel qualities will be selected as volunteers during each trading, since all the buyers are i.i.d.

and $\kappa > S$, respectively.

$$\begin{aligned}&\overline{\mathcal{U}^{Mem}}\left(p,q,r,\kappa \leqslant S,\boldsymbol{\mathcal{A}},\boldsymbol{\mathcal{Y}}\right)\\&\quad=\kappa a\mathrm{E}\left[U_m^{PP}\right]-\kappa q d^{comp}+\kappa a q d^{comp}\end{aligned} \tag{10}$$

$$\begin{aligned}&\overline{\mathcal{U}^{Mem}}\left(p,q,r,\kappa > S,\boldsymbol{\mathcal{A}},\boldsymbol{\mathcal{Y}}\right)\\&\quad=\left(\kappa a-\mathrm{E}[V]\right)\mathrm{E}\left[U_m^{PP}\right]-\kappa q d^{comp}\\&\qquad+\kappa a q d^{comp}+r d^{comp}\mathrm{E}\left[V\right]\end{aligned} \tag{11}$$

In this paper, we consider two key risks for members. First, the risk of a member $\boldsymbol{b_m}$ (not a volunteer) suffering from a non-positive utility[9] (abbreviate to "MRisk") is defined as the probability that its utility is too close to or less than U_{min} (U_{min} denotes a value approaching to zero), expressed by the following (12).

$$\begin{aligned}&\mathcal{R}^{MRisk}\left(p,q,\boldsymbol{\mathcal{A}},\boldsymbol{\mathcal{Y}}\right)\\&=\Pr\left(\frac{\alpha_m U_m^{PP}+(1-\alpha_m)U^{DE}}{U_{min}}\leqslant \xi_1\right)\\&=\begin{cases}0, & \mathbb{C}_5<0\\ 1-a, & 0\leqslant \mathbb{C}_5<\mathbb{C}_3-\dfrac{\mathbb{C}_4}{\log_2\left(1+e^{tran}\varepsilon_1\right)}\\ 1-a+a\left(\dfrac{2^{\frac{\mathbb{C}_4}{\mathbb{C}_3-\mathbb{C}_5}}-1-e^{tran}\varepsilon_1}{e^{tran}\left(\varepsilon_2-\varepsilon_1\right)}\right), & \mathbb{C}_3-\dfrac{\mathbb{C}_4}{\log_2\left(1+e^{tran}\varepsilon_1\right)}\leqslant \mathbb{C}_5\leqslant \mathbb{C}_3-\dfrac{\mathbb{C}_4}{\log_2\left(1+e^{tran}\varepsilon_2\right)}\\ 1, & \mathbb{C}_5>\mathbb{C}_3-\dfrac{\mathbb{C}_4}{\log_2\left(1+e^{tran}\varepsilon_2\right)}\end{cases}\end{aligned} \tag{12}$$

Particularly, ξ_1 denotes a positive threshold coefficient; $\mathbb{C}_3=\frac{\omega_1 d^{comp}+\omega_2 e^{loc}d^{comp}}{f^b}-\frac{\omega_1 d^{comp}}{f^s}+qd^{comp}-pd^{comp}$, $\mathbb{C}_4=\frac{\omega_2 e^{tran}d^{size}+\omega_1 d^{size}}{W}$, and $\mathbb{C}_5=\xi_1 U_{min}+qd^{comp}$, which are constants under any given p and q, for notational simplicity. Then, the risk of a performer being selected as a volunteer[10] (abbreviate to "VRisk") is given by (13). Apparently, a larger value of $\mathcal{R}^{VRisk}(\kappa,\boldsymbol{\mathcal{A}})$ leads to a higher risk of being selected as a volunteer, which greatly impacts the members' trading experience. Derivations of (12) and (13) are provided in Appendix B.

$$\begin{aligned}&\mathcal{R}^{VRisk}(\kappa,\boldsymbol{\mathcal{A}})\\&\quad=\begin{cases}0, & 0\leqslant\kappa\leqslant S\\ a-\sum_{i=0}^{i=S-1}C_{\kappa-1}^{i}a^{i+1}(1-a)^{\kappa-1-i}, & \kappa>S\end{cases}\end{aligned} \tag{13}$$

3) Utility of Non-Members in Spot Market: For analytical simplicity, let n be the index of non-members, where $n \in \{\kappa+1,\ldots,|\boldsymbol{\mathcal{B}}|\}$ (notably, there are no non-members when $\kappa=|\boldsymbol{\mathcal{B}}|$). During each trading, if the seller's resources are not fully occupied by tasks of members, each non-member with task execution requirement ($\alpha_n = 1$) can compete for the remaining resources based on the current channel quality, where partial offloading is allowed. Correspondingly, we define the utility $\mathcal{U}_n^{NonM}$ of each non-member $\boldsymbol{b_n}$ as (14), where g_n denotes the unit price of resources that $\boldsymbol{b_n}$ has to pay during each trading.

$$\begin{aligned}&\mathcal{U}_n^{NonM}\left(g_n,\lambda_n,\alpha_n,\gamma_n\right)\\&\quad=\alpha_n\left(\omega_1\left(t^{loc}-t_n^{edge}\right)+\omega_2\left(c^{loc}-c_n^{edge}\right)-g_n\lambda_n d^{comp}\right)\end{aligned} \tag{14}$$

C. Modeling of Seller

1) Utility, Expected Utility and Risk of Seller in Futures Market: Suppose that an edge server owns Sd^{comp} resources (e.g., CPU cycles). Utility of the seller contains two key factors: i) the revenue U^{IN} obtained from practical performers and defaulters, and ii) the total refunds and compensations U^{OUT} the seller has to pay for volunteers when the available resources fail to afford the members' task execution requirements owing to overbooking. Correspondingly, U^{IN} is defined as the following (15).

$$U^{IN}=pd^{comp}\sum_{m=1}^{m=\kappa}\alpha_m+qd^{comp}\sum_{m=1}^{m=\kappa}(1-\alpha_m) \tag{15}$$

Moreover, U^{OUT} is calculated by (16).

$$U^{OUT}=(p+r)d^{comp}V \tag{16}$$

Correspondingly, utility of the seller is considered as the difference [11] between U^{IN} and U^{OUT}, as shown in (17) below.

$$\mathcal{U}^{SelF}\left(p,q,r,\kappa,\boldsymbol{\mathcal{A}}\right)=U^{IN}-U^{OUT} \tag{17}$$

Expected utilities of seller are given by (18) and (19), shown at the bottom of the next page, considering $\kappa\leqslant S$ and $\kappa>S$, respectively.

[9]We consider U_{min} in estimating MRisk since buyers do not accept negative utilities in single spot market. Thus, buyers can be attracted to participate in futures market as long as their risks of obtaining negative utilities can be controlled to some extent.

[10]Although each volunteer will receive a compensation from the seller, this will always lead to bad trading experiences for those members who cannot enjoy computing service even they have signed the forward contracts.

[11]Suppose that price p, penalty q, and compensation r are of the same unit in this paper, e.g., US dollar, which is also common in real-life trading market. Thus, we do not consider weight coefficients between U^{IN} and U^{OUT}.

$$\mathrm{E}\left[V\right]=\begin{cases}0, & \kappa\leqslant S\\ \kappa a-\left(\sum_{i=0}^{i=S-1}iC_{\kappa}^{i}a^{i}(1-a)^{\kappa-i}+S\sum_{i=S}^{i=\kappa}C_{\kappa}^{i}a^{i}(1-a)^{\kappa-i}\right), & \kappa>S\end{cases} \tag{8}$$

In the proposed resource trading market, the seller always prefers to achieve a larger utility than it expects.[12] Thus, we define the risk of the seller as the probability that $\mathcal{U}^{SelF}(p,q,r,\kappa,\boldsymbol{\mathcal{A}})$ is too close to or less than $\overline{\mathcal{U}^{SelF}}(p,q,r,\kappa,\boldsymbol{\mathcal{A}})$, which is given by (20).

$$\mathcal{R}^{SRisk}(p,q,r,\kappa,\boldsymbol{\mathcal{A}}) = \Pr\left(\frac{\mathcal{U}^{SelF}(p,q,r,\kappa,\boldsymbol{\mathcal{A}})}{\overline{\mathcal{U}^{SelF}}(p,q,r,\kappa,\boldsymbol{\mathcal{A}})} \leqslant \xi_2\right), \tag{20}$$

where ξ_2 represents a positive threshold coefficient. According to (18), risk of the seller under $\kappa < S$ is calculated by (21), where $\mathbb{C}_6 = \frac{\xi_2 \overline{\mathcal{U}^{SelF}}(p,q,r,\kappa\leqslant S,\mathcal{A})}{d^{comp}(p-q)} - \frac{q\kappa}{(p-q)}$ for notational simplicity:

$$\mathcal{R}^{SRisk}(p,q,r,\kappa\leqslant S,\boldsymbol{\mathcal{A}}) = \begin{cases} 0, & \mathbb{C}_6 < 0 \\ \sum_{i=0}^{i=\lfloor \mathbb{C}_6 \rfloor} C_\kappa^i a^i (1-a)^{\kappa-i}, & 0 \leqslant \mathbb{C}_6 \leqslant \kappa \\ 1, & \mathbb{C}_6 > \kappa \end{cases} \tag{21}$$

Considering $\kappa > S$, risk of the seller is given by (22), based on (19).

$$\mathcal{R}^{SRisk}(p,q,r,\kappa > S,\boldsymbol{\mathcal{A}}) = \begin{cases} 0, & \mathbb{C}_7 < (0, S(p-q) - (\kappa - S)(q+r))^- \\ \sum_{i=0}^{i=\left\lfloor \frac{\mathbb{C}_7}{p-q} \right\rfloor} C_\kappa^i a^i (1-a)^{\kappa-i} + \sum_{i=\left\lceil \frac{S(p-q)-\mathbb{C}_7}{q+r} + S \right\rceil}^{i=\kappa} C_\kappa^i a^i (1-a)^{\kappa-i}, & (0, S(p-q) - (\kappa-S)(q+r))^- \leqslant \mathbb{C}_7 \leqslant S(p-q) \\ 1, & \mathbb{C}_7 > S(p-q) \end{cases} \tag{22}$$

where $\mathbb{C}_7 = \frac{\xi_2 \overline{\mathcal{U}^{Sel}}(p,q,r,\kappa>S,\boldsymbol{\mathcal{A}})}{d^{comp}} - q\kappa$ for notational simplicity. Notably, let $\sum_{i=0}^{i=\left\lfloor \frac{\mathbb{C}_7}{p-q} \right\rfloor} C_\kappa^i a^i (1-a)^{\kappa-i} = 0$ when $\frac{\mathbb{C}_7}{p-q} < 0$, and $\sum_{i=\left\lceil \frac{S(p-q)-\mathbb{C}_7}{q+r} + S \right\rceil}^{i=\kappa} C_\kappa^i a^i (1-a)^{\kappa-i} = 0$ when $\left\lceil \frac{S(p-q)-\mathbb{C}_7}{q+r} + S \right\rceil > \kappa$. Derivations associated with (21) and (22) are detailed by Appendix C.

[12]We consider the expected value of seller's utility in estimating the corresponding risk, since the seller can definitely receive positive profits as long as there are resource demands in single spot market. Thus, the risk of receiving unexpected utility has to be controlled to some extent, to encourage the seller to join futures market.

2) Utility of Seller in Spot Market: Note that non-members with task execution requirements get chances to compete for available resources due to the possible "no shows" of members. Let binary indicator $x_n = 1$ denote that the seller decides to trade with non-member $\boldsymbol{b_n}$, and $x_n = 0$ otherwise; while $\boldsymbol{\mathcal{X}} = \{x_n | n \in \{\kappa+1, \ldots, |\boldsymbol{\mathcal{B}}|\}\}$ depicts the relevant trading decision vector. Moreover, let $\boldsymbol{\mathcal{G}} = \{g_n | n \in \{\kappa+1, \ldots, |\boldsymbol{\mathcal{B}}|\}\}$ present the price vector, and $\boldsymbol{\Lambda} = \{\lambda_n | n \in \{\kappa+1, \ldots, |\boldsymbol{\mathcal{B}}|\}\}$ indicate the offloading rate vector of non-members. Correspondingly, the seller's utility in spot market is defined by (23).

$$\mathcal{U}^{SelS}(\boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{G}}, \boldsymbol{\Lambda}, \boldsymbol{\mathcal{A}}) = d^{comp} \sum_{n=\kappa+1}^{n=|\boldsymbol{\mathcal{B}}|} x_n g_n \lambda_n \alpha_n \tag{23}$$

III. Problem Formulation and Solution Design in Futures Market

The proposed futures market mainly considers designing both the forward contract (e.g., p, q, and r) and overbooking rate (e.g., κ^o, which is equivalent to the design of κ), where the relevant problem is formulated by $\boldsymbol{\mathcal{F}_1}$ with two objectives (24a) and (24b), under constraints C1-C7. Specifically, (24a) describes that the seller aims to maximize its expected utility while meeting the tolerable risk (constraint C1); (24b) indicates the maximization of the expected utility of members under acceptable tolerant risks (constraints C2 and C3).

$$\boldsymbol{\mathcal{F}_1}: \begin{cases} \underset{p,q,r,\kappa}{\arg\max}\, \overline{\mathcal{U}^{SelF}}(p,q,r,\kappa,\boldsymbol{\mathcal{A}}) & \text{(24a)} \\ \underset{p,q,r,\kappa}{\arg\max}\, \overline{\mathcal{U}^{Mem}}(p,q,r,\kappa,\boldsymbol{\mathcal{A}},\boldsymbol{\mathcal{Y}}) & \text{(24b)} \end{cases}$$

$$\begin{aligned} s.t.\ & \mathrm{C1}: \mathcal{R}^{SRisk}(p,q,r,\kappa,\boldsymbol{\mathcal{A}}) \leqslant \xi^S; \\ & \mathrm{C2}: \mathcal{R}^{MRisk}(p,q,\boldsymbol{\mathcal{A}},\boldsymbol{\mathcal{Y}}) \leqslant \xi^M; \\ & \mathrm{C3}: \mathcal{R}^{VRisk}(\kappa,\boldsymbol{\mathcal{A}}) \leqslant \xi^V; \\ & \mathrm{C4}: 1 \leqslant \kappa \leqslant |\boldsymbol{\mathcal{B}}|; \\ & \mathrm{C5}: U_m^{PP} > 0, \quad \forall\, m \in \{1, \ldots, \kappa\}; \\ & \mathrm{C6}: p \geqslant p_{min}^{Sel}; \\ & \mathrm{C7}: p > q, \quad r > 0. \end{aligned}$$

Specifically, ξ^S, ξ^B, and ξ^V are positive threshold coefficients on risk control. Constraints C1-C3 denote the acceptable tolerant risks of seller and members. Constraint C4 limits the practicable number of members (particularly, if the players fail to sign forward contract, let $\kappa = 0$). Constraint C5 represents the individual rationality of members in this market, describing that each practical performer will receive at least non-negative utility from a trading even under a poor channel quality (e.g., $\gamma_m = \varepsilon_1$). Additionally, constraint C6 indicates that p should be larger than the seller's tolerable minimum price p_{min}^{Sel} (e.g., the minimum price reflects the seller's cost for processing a task such as energy consumption, etc.); while C7 describes the relationships among p, q, and r. To facilitate the analysis, we integrate C5 and C6 as C8, where $p_{max}^{Mem} = \frac{\omega_1 + \omega_2 e^{loc}}{f^b} -$

$$\overline{\mathcal{U}^{SelF}}(p,q,r,\kappa \leqslant S,\boldsymbol{\mathcal{A}}) = \kappa a p d^{comp} + \kappa q d^{comp} - \kappa a q d^{comp} = \kappa d^{comp}(q - aq + ap) \tag{18}$$

$$\overline{\mathcal{U}^{SelF}}(p,q,r,\kappa > S,\boldsymbol{\mathcal{A}}) = \kappa d^{comp}(q - aq - ar) + (p+r)\, d^{comp}\left(\sum_{i=0}^{i=S-1} i C_\kappa^i a^i (1-a)^{\kappa-i} + S \sum_{i=S}^{i=\kappa} C_\kappa^i a^i (1-a)^{\kappa-i}\right) \tag{19}$$

$\frac{\omega_1}{f^s} - \left(\frac{\omega_1 d^{size} + \omega_2 e^{tran} d^{size}}{W d^{comp} \log_2(1+e^{tran}\varepsilon_1)}\right)$ denotes the maximum tolerable price of each member.

$$\text{C8: } p_{min}^{Sel} \leqslant p < p_{max}^{Mem} \tag{25}$$

Note that $\boldsymbol{\mathcal{F}_1}$ represents a MOO problem, which, however, is difficult to be solved directly by state-of-the-art methods, e.g., weighted sum method [2], [42], weighted metric method [43], and ϵ-constrained method [44]. A common situation where players do not know the complete information of each other, e.g., the seller is unaware of each buyer's local capability f^b, local power cost e^{loc}, weight coefficients ω_1 and ω_2) poses great challenges to merge objectives (24a) and (24b) together, if applying weighted sum method. Weighted metric method can only be adopted under given optimal players' expected utilities, which also represents a significant challenge. Besides, it is difficult to determine the values of weight coefficients for the above-mentioned weighted-related methods, where inappropriate weights may lead to unfairness. The ϵ-constrained method generally encourages a single optimization objective while turning the remaining into constraint(s), which, however, fails to meet our intention of designing trading mechanism that is mutually beneficial to both seller and buyers. Additionally, each objective in $\boldsymbol{\mathcal{F}_1}$ ((24a) and (24b)) refers to a mixed integer nonlinear programing (MINLP) problem [45], which considers determining both continuous (e.g., p, q and r) and integer variables (e.g., κ), that further complicates the solution design.

Consequently, bilateral negotiation is considered as an efficient approach which can facilitate the negotiation among players with conflicting objectives [18] to reach the final trading consensus on p, q, r, and κ. Since all the buyers are i.i.d, a trusted agent[13] is applied as the representative of members to negotiate with the seller. Besides, Δp, Δq, and Δr are applied as the granularities of price, penalty, and refund, respectively, to promote the quotation procedure. Specifically, we propose an alternative optimization-based bilateral negotiation mechanism to solve problem $\boldsymbol{\mathcal{F}_1}$, as detailed by Algorithm 1, where p^*, q^*, r^* constitute the final contract term, κ^* denotes the final agreement on number of members, while $\mathbb{C}_q$ and $\mathbb{C}_q$ are positive constants to constrain the upper limits of penalty and compensation. Specifically, the corresponding overbooking rate can be calculated as $(\kappa^* - S)/S$. In Algorithm 1, the representative agent of buyers first determines the acceptable range of κ (line 2) under constraint C3, to avoid overmany volunteers. Then, under given price $FuturesP_i$, penalty $FuturesQ_j$ and compensation $FuturesR_l$ in each quotation round, the seller first decides its acceptable range of κ (e.g., $\boldsymbol{K_{i,j,l}^{Sel}}$) while meeting its tolerable risk (line 6); while the agent checkes if the current price and penalty meets MRisk (line 7). If $\boldsymbol{K_{i,j,l}^{Sel}} \cap \boldsymbol{K^{Mem}} \neq \emptyset$, the agent chooses a κ from set $\boldsymbol{K_{i,j,l}^{Sel}} \cap \boldsymbol{K^{Mem}}$ that maximizes the expected utility of members (lines 8-9), where the relevant items will be saved into a candidate set $\boldsymbol{CTerm}$ (line 10). Specifically, if $\boldsymbol{K_{i,j,l}^{Sel}} = \emptyset$, seller can directly raise the value of penalty (line 12) since the current expected utility may be unsatisfying; otherwise, the seller will adjust either the value of compensation, penalty, or price to start another quotation, as shown by lines 15-17. After all the quotations, the seller chooses a group of price, penalty and compensation, as well as the number of available members from candidate set $\boldsymbol{CTerm}$, that maximizes its expected utility, as the final trading agreement on both contract terms and overbooking rate (lines 18-20); otherwise, players fail to sign forward contract.

[13]The seller does not have to negotiate with every buyer since all the buyers are i.i.d. Thus, a trusted agent (e.g., AP, or one of these buyers as representative, etc.) is supposed to be a representative of members, negotiates with the seller on forward contract terms and overbooking rate. Once the trading consensus has been reached, the relevant buyers who get the membership can sign the forward contract with seller.

The computational complexity associated with Algorithm 1 mainly relies on the overall quotation rounds (e.g., $Count$ in Algorithm 1), which can generally be expressed by $O\left(Count\right)$. However, the quotation procedure may be paused or stopped by several possible conditions. For example, the seller has to raise penalty directly since the current one cannot meet its risk on utility (lines 11-12 in Algorithm 1); or raising any factor (p, q, r) will no longer be accepted by any buyer (line 7 in Algorithm 1). Thus, the lower and upper bound of computational complexity associated with Algorithm 1 can be calculated by $O\left(1\right)$, and $O\left(\left(p_{max}^{Mem} - p_{min}^{Sel}\right)\mathbb{C}_q\mathbb{C}_r/\Delta_p\right)$, respectively. Optimality of the proposed Algorithm 1 is discussed from the following two perspectives: *i)*. the maximum of (24b) can be reached under every possible combination of price, penalty and compensation (line 9) since buyers are given the right to choose the optimal value of κ; while *ii)*. the optimum of (24a) can be achieved over the obtained candidate set, since the seller is given right to make the final determination on forward contract and overbooking rate (line 19).

IV. Problem Formulation and Solution Design in Spot Market

Spot trading may occur when the following two cases happen concurrently: *i)*. $\sum_{m=1}^{m=\kappa} \alpha_m < S$, namely, the seller has available resources for onsite sale after meeting the requirements of members; *ii)*. $\sum_{n=\kappa+1}^{n=|\boldsymbol{\mathcal{B}}|} \alpha_n > 0$, namely, there exist resource demands from non-members, e.g., at least one non-member has task execution requirement. Correspondingly, resource trading in spot market is generally formulated by problem $\boldsymbol{\mathcal{F}_2}$, where the seller, and each non-member with task execution requirement is aiming to maximize its own utility, based on the current network/market conditions, as shown by the following (26a), and (26b).

$$\boldsymbol{\mathcal{F}_2}: \begin{cases} \arg\max\limits_{\boldsymbol{\mathcal{X}},\boldsymbol{\mathcal{G}}} \mathcal{U}^{SelS}\left(\boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{G}}, \boldsymbol{\Lambda}, \boldsymbol{\mathcal{A}}\right) & \text{(26a)} \\ \arg\max\limits_{\lambda_n} \mathcal{U}_n^{NonM}\left(g_n, \lambda_n, \alpha_n, \gamma_n\right), & \\ \forall\ \alpha_n = 1, n \in \{\kappa+1, \ldots, |\boldsymbol{\mathcal{B}}|\} & \text{(26b)} \end{cases}$$

$$\begin{aligned} s.t.\ & \text{C9}: \lambda_n \triangleq 0, \quad \forall\ U_n^{NonM}\left(g_n, \lambda_n, \alpha_n, \gamma_n\right) \leqslant 0; \\ & \text{C10}: 0 \leqslant \lambda_n \leqslant 1, \\ & \text{C11}: g_n \geqslant p_{min}^{Sel}, \quad \forall\ \alpha_n = 1,\ n \in \{\kappa+1, \ldots, |\boldsymbol{\mathcal{B}}|\}; \\ & \text{C12}: \boldsymbol{\Lambda}^{\mathrm{T}}\boldsymbol{\mathcal{X}} \leqslant S', \end{aligned}$$

where $S' = S - \sum_{m=1}^{m=\kappa} \alpha_m$ (apparently, $S' d^{comp}$ indicates the remaining resources available for non-members). Constraints C9 ensures the non-negative utility of each non-member, C10 and C11 (similar with C6) constrain the values of offloading

Algorithm 1: Proposed Bilateral Negotiation in Futures Market (Solving Problem $\mathcal{F}_1$)

Input : $a,\ \varepsilon_1,\ \varepsilon_2,\ d^{comp},\ d^{size},\ f^b,\ f^s,\ e^{loc},\ e^{tran},\ W,\ |\mathcal{B}|,\ S,\ \omega_1,\ \omega_2,\ \xi^S,\ \xi^M,\ \xi^V,\ \Delta p,\ \Delta q,\ \Delta r,\ \mathbb{C}_q,\ \mathbb{C}_r$
Output: $p^*,\ q^*,\ r^*,\ \kappa^*$

1 **Initialization:** $FuturesP_1 \leftarrow p_{min}^{Sel}$, $FuturesQ_1 \leftarrow \Delta q$, $FuturesR_1 \leftarrow \Delta r$, $i = j = l \leftarrow 1$, $\boldsymbol{CTerm} \leftarrow \emptyset$, $Count \leftarrow 0$,
2 The agent first determines the acceptable range of κ denoted by $\boldsymbol{K^{Mem}}$ while meeting constraints C3 and C4,
3 **while** $FuturesP_i \leqslant p_{max}^{Mem}$ **do**
4   **while** $FuturesQ_j \leqslant \mathbb{C}_q \times \Delta q$ **do**
5   **while** $FuturesR_l \leqslant \mathbb{C}_r \times \Delta r$ **do**
6   The seller determines its acceptable range of κ denoted by $\boldsymbol{K_{i,j,l}^{Sel}}$, while meeting C1 and C4,
7   The agent checks if the current price and penalty meets constraint C2, if yes, continue the negotiation; otherwise, go to step 18,
8   **if** $\boldsymbol{K_{i,j,l}^{Sel}} \cap \boldsymbol{K^{Mem}} \neq \emptyset$ **then**
9         $\kappa_{i,j,l} \leftarrow \arg\max_{\kappa} \overline{\mathcal{U}^{Mem}}\left(FuturesP_i, FuturesQ_j, FuturesR_l, \boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{Y}}, \kappa\right)$, $\kappa \in \boldsymbol{K_{i,j,l}^{Sel}} \cap \boldsymbol{K^{Mem}}$ % the agent chooses the value of κ that maximizes the members' expected utility,
10         $\boldsymbol{CTerm} \leftarrow \boldsymbol{CTerm} \bigcup \left\{FuturesP_i, FuturesQ_j, FuturesR_l, \kappa_{i,j,l}\right\}$,
11   **else if** $\boldsymbol{K_{i,j,l}^{Sel}} = \emptyset$ **then**
12         $l \leftarrow 1$, $j = j+1$, $FuturesQ_j \leftarrow FuturesQ_{j-1} + \Delta q$, $Count \leftarrow Count + 1$,
13   **else**
14         $Count \leftarrow Count + 1$, break, % jump out of the current while loop
15       $l \leftarrow l+1$, $FuturesR_l \leftarrow FuturesR_{l-1} + \Delta r$, $Count \leftarrow Count + 1$,
16     $l \leftarrow 1$, $j \leftarrow j+1$, $FuturesQ_j \leftarrow FuturesQ_{j-1} + \Delta q$, $Count \leftarrow Count + 1$,
17   $l \leftarrow 1$, $j \leftarrow 1$, $i \leftarrow i+1$, $FuturesP_i \leftarrow FuturesP_{i-1} + \Delta p$, $Count \leftarrow Count + 1$,
18 **if** $\boldsymbol{CTerm} \neq \emptyset$ **then**
19   $\{p^*, q^*, r^*, \kappa^*\} \leftarrow \arg\max_{p,q,r,\kappa} \overline{\mathcal{U}^{Sel}}\left(p, q, r, \kappa, \boldsymbol{\mathcal{A}}\right)$, $\{p, q, r, \kappa\} \in \boldsymbol{CTerm}$, % the seller chooses a set of p, q, r and κ from $\boldsymbol{CTerm}$ that maximizes its expected utility;
20 **else**
21   Players fail to sign the forward contract,
22 **end algorithm**

rate and unit resource price, respectively; C12 restricts available seller's resources in spot market, where $\boldsymbol{\Lambda}^{\mathrm{T}}$ denotes the transpose of vector $\boldsymbol{\Lambda}$. Similar with $\mathcal{F}_1$, we consider bilateral negotiation among the seller and non-members to solve $\mathcal{F}_2$, via considering two pricing rules [46]: uniform pricing and differential pricing.

A. Spot Trading Under Uniform Pricing

Considering spot trading that follows uniform pricing rule where the seller charges all the non-members with the same price. Thus, $\mathcal{F}_2$ can be further considered as the following $\mathcal{F}_3$, by applying a uniform price g, where $\forall n \in \{\kappa+1, \ldots, |\mathcal{B}|\}$, $g_n = g$.

$$\mathcal{F}_3: \begin{cases} \arg\max\limits_{\boldsymbol{\mathcal{X}}, g} \mathcal{U}^{SelS}\left(\boldsymbol{\mathcal{X}}, g, \boldsymbol{\Lambda}, \boldsymbol{\mathcal{A}}\right) & \text{(27a)} \\ \arg\max\limits_{\lambda_n} \mathcal{U}_n^{NonM}\left(g, \lambda_n,\ \alpha_n, \gamma_n\right), \\ \forall\ \alpha_n = 1, n \in \{\kappa+1, \ldots, |\mathcal{B}|\} & \text{(27b)} \end{cases}$$

$$\begin{aligned} s.t.\ & \text{C10; C12;} \\ & \text{C13}: \lambda_n \triangleq 0, \quad \forall\ U_n^{NonM}\left(g, \lambda_n, \alpha_n, \gamma_n\right) \leqslant 0; \\ & C14:\ g \geqslant p_{min}^{Sel}. \end{aligned}$$

Constraints C13 and C14 are similar with C9 and C11. Notably, (27a) depicts a binary knapsack problem with the weight λ_n, and the value $g\lambda_n$ for a non-member $\boldsymbol{b_n}$; while (27b) in $\mathcal{F}_3$ represents a non-convex optimization problem under any given g. Apparently, (27a) is NP-complete which poses difficulty to find efficient algorithms, thus, we apply dynamic programming [46] to solve the binary knapsack problem in pseudo-polynomial time (e.g., by using the kp01 software package in MATLAB). Moreover, algorithm for obtaining the optimal offloading rate (solve (27b)) is detailed by Appendix D. Pseudocode on solving $\mathcal{F}_3$ is given by Algorithm 2, where $\boldsymbol{\mathcal{X}}^*$ and $\boldsymbol{\Lambda}^*$ indicate the final trading decision vector and offloading rate vector, respectively; and g^* denotes the relevant final agreed unit price of resource. Specifically, only non-members with tasks are considered in the proposed spot market, as depicted by lines 6-5. Lines 6-8 indicate that under a given price $SpotP_i$, each non-member $\boldsymbol{b_n}$ with task decides the optimal offloading rate $\boldsymbol{\Lambda_n^i}$ that maximizes its utility; while lines 9-10 shows that the seller will stop raising price if all the non-members decide to process their tasks locally, mainly owing to an excessive price. In line 12, the seller determines a trading decision vector that maximizes its utility under price $SpotP_i$ by solving a 0-1 knapsack problem, and saves the relevant utility u_i, price $SpotP_i$, trading vector $\boldsymbol{\mathcal{X}_i}$, and offloading vector $\boldsymbol{\Lambda_i}$ in to a candidate set $\mathbb{X}$. After all the quotations, seller selects the solution that enables the largest value of its utility from $\mathbb{X}$ (line 16).

Specifically, computational complexity of the quotation procedure associated with Algorithm 2 is generally denoted by $O(Count)$. Besides, the seller has to solve several 0-1 knapsack problems, where each brings a computational complexity

Algorithm 2: Proposed Spot Trading Under Uniform Pricing (Solving Problem $\mathcal{F}_3$)

Input : $\mathcal{A}$, $\mathcal{Y}$, S', d^{comp}, d^{size}, f^b, f^s, e^{loc}, e^{tran}, W, ω_1, ω_2, Δp
Output: $\mathcal{X}^*$, g^*, Λ^*
Initialization: $SpotP_i \leftarrow p_{min}^{Sel}$, $i \leftarrow 1$, $Count \leftarrow 0$, $\|\Lambda_0\|_1 > 0, \mathbb{X} \leftarrow \emptyset$,
while $SpotP_i \geqslant p_{min}^{Sel}$ **do**
  for $n = \kappa + 1$ *and* $n \leqslant |\mathcal{B}|$ **do**
    if $\alpha_n = 0$ **then**
      $Count_n \leftarrow 0$, $n \leftarrow n + 1$,
      else
        $\lambda_n^i \leftarrow \arg\max_{\lambda_n} \mathcal{U}_n^{NonM}(SpotP_i, \lambda_n, \alpha_n, \gamma_n)$, while meeting C10 and C13, % this problem (27b) represents a non-convex optimization problem which is detailed in Appendix D,
        $\Lambda_i \leftarrow \Lambda_i \bigcup \lambda_n^i$, $n \leftarrow n + 1$, $Count \leftarrow Count + 1$,
    if $\|\Lambda_i\|_1 = 0$ **then**
      jump out of the current while loop, % none of the non-members can accept a higher price, the seller stops quotation,
      else
        $\mathcal{X}_i \leftarrow \arg\max_{\mathcal{X}} {\Lambda_i}^{\mathrm{T}} \mathcal{X}$, while meeting C12, % this problem (27a) refers to a binary knapsack problem which can be solved by dynamic programming;
        $u_i \leftarrow g_i d^{comp} {\Lambda_i}^{\mathrm{T}} \mathcal{X}_i$, $\mathbb{X} \leftarrow \mathbb{X} \bigcup \{u_i, SpotP_i, \mathcal{X}_i, \Lambda_i\}$, $i \leftarrow i + 1$,
        $n \leftarrow n + 1$, $SpotP_i \leftarrow SpotP_{i-1} + \Delta p$,
For all $\alpha_n = 1$, $Count_n \leftarrow Count / \sum_{n=\kappa+1}^{n=|\mathcal{B}|} \alpha_n$,
The seller chooses the largest u_i from set $\mathbb{X}$, where the relevant $SpotP_i, \mathcal{X}_i, \Lambda_i$ stand for the final trading solution $g^*, \mathcal{X}^*$ and Λ^*,
end algorithm

Algorithm 3: Proposed Spot Trading Under Differential Pricing (Solving Problem $\mathcal{F}_2$)

input : $\mathcal{A}$, $\mathcal{Y}$, S', d^{comp}, d^{size}, f^b, f^s, e^{loc}, e^{tran}, W, ω_1, ω_2, Δp
output: $\mathcal{X}^*$, $\mathcal{G}^*$, Λ^*
Initialization: $SpotP_1 \leftarrow p_{min}^{Sel}$, $i \leftarrow 1, \mathbb{X} \leftarrow \emptyset$, $\lambda_n^0 > 0$, $Count_n \leftarrow 0$, $\forall n \in \{\kappa + 1, \ldots, |\mathcal{B}|\}$,
for $n = \kappa + 1$ *and* $n \leqslant |\mathcal{B}|$ **do**
  if $\alpha_n = 0$ **then**
    $n \leftarrow n + 1$,
    else
      while $SpotP_i \geqslant p_{min}^{Sel}$ *and* $\lambda_n^{i-1} > 0$ **do**
        $\lambda_n^i \leftarrow \arg\max_{\lambda_n} \mathcal{U}_n^{NonM}(SpotP_i, \lambda_n, \alpha_n, \gamma_n)$, while meeting C9 and C10, % similar with $\mathcal{F}_3$, this problem (26b) represents a non-convex optimization problem which is detailed in Appendix D,
        if $\lambda_n^i > 0$ **then**
          $\Lambda_n \leftarrow \Lambda_n \bigcup \{SpotP_i, \lambda_n^i\}$, $Count_n \leftarrow Count_n + 1$,
          $i \leftarrow i + 1$, $SpotP_i \leftarrow SpotP_{i-1} + \Delta p$,
          else
            jump out of the current while loop, % non-member $\boldsymbol{b_n}$ cannot accept a higher price, the seller stops the current quotation,
      $n \leftarrow n + 1$, $i \leftarrow 1$,
The seller chooses $\mathcal{X}^*$ that maximizes the value of $\mathcal{U}^{SelS}(\mathcal{X}, \mathcal{G}, \Lambda, \mathcal{A})$ based on set $\bigcup_{\alpha_n=1} \Lambda_n$, where the relevant price set and offloading rate set will be the final solution $\mathcal{G}^*, \Lambda^*$ % this problem (26a) denotes a knapsack problem with grouped items which can be solved by dynamic programming,
end algorithm

denoted by $O\left(\sum_{n=\kappa+1}^{n=|\mathcal{B}|} \alpha_n \times (S - \sum_{n=1}^{n=\kappa} \alpha_n)\right)$, to reach the final trading decision. Similar with Algorithm 1, the optimality of Algorithm 2 is considered from two angles: *i)*. under every given quoted price, each non-member can reach its maximal utility by deciding the optimal offloading rate (as shown in (27b)); while *ii)*. the seller can finally achieve its maximum utility via choosing one uniform price and the relevant non-members as well as offloading rates (as given in (27a)). Fig. 2 shows an example where the seller charges different non-members under the same price during a spot trading.

Non-members	Uniform pricing: price1	Uniform pricing: price2	Uniform pricing: price3	Differential pricing: price1	Differential pricing: price2	Differential pricing: price3
b_{11}	$\lambda_{11} = 93\%$	$\lambda_{11} = 0\%$	$\lambda_{11} = 0\%$	$\lambda_{11} = 93\%$	$\lambda_{11} = 0\%$	$\lambda_{11} = 0\%$
b_{12}	$\lambda_{12} = 85\%$	$\lambda_{12} = 85\%$	$\lambda_{12} = 0\%$	$\lambda_{12} = 85\%$	$\lambda_{12} = 85\%$	$\lambda_{12} = 0\%$
b_{13}	$\lambda_{13} = 100\%$	$\lambda_{13} = 59\%$	$\lambda_{13} = 59\%$	$\lambda_{13} = 100\%$	$\lambda_{13} = 59\%$	$\lambda_{13} = 59\%$
b_{14}	No show	---	---	No show	---	---
b_{15}	$\lambda_{15} = 48\%$	$\lambda_{15} = 48\%$	$\lambda_{15} = 0\%$	$\lambda_{15} = 48\%$	$\lambda_{15} = 48\%$	$\lambda_{15} = 0\%$

Resource price

Fig. 2. Examples associated with spot trading under uniform and differential pricing rules, where the purple-colored blocks indicate the final trading decision. Specifically, buyers $\boldsymbol{b_1} - \boldsymbol{b_{10}}$ are members, $\boldsymbol{b_{11}} - \boldsymbol{b_{15}}$ are non-members, while $\boldsymbol{b_{14}}$ doesn't show up in this trading (namely, $\alpha_{14} = 0$).

B. Spot Trading Under Differential Pricing

Differential pricing rule considers a more general case where the seller can charge different non-members with different prices (although the seller may also consider a uniform price for some non-members as long as the concerned prices can maximize its utility), where problem $\boldsymbol{\mathcal{F}_2}$ in discussed. Specifically, (26a) in $\boldsymbol{\mathcal{F}_2}$ refers to a knapsack problem with grouped items [47], [49], for which the dynamic programming can be applied similar with (27a); while (26b) in $\boldsymbol{\mathcal{F}_2}$ represents a non-convex problem under any given g_n, for which the solution of obtaining the optimal offloading rate is given by Appendix D. Specifically, each non-member with task execution requirement decides an optimal offloading rate based on each quoted price, while the seller determines the final trading vector by changing non-members with different prices, associated the relevant offloading rates, to maximize its utility. Pseudocode for solving $\boldsymbol{\mathcal{F}_2}$ is detailed by Algorithm 3, where $\boldsymbol{\mathcal{X}}^*, \boldsymbol{\mathcal{G}}^*$, and $\boldsymbol{\Lambda}^*$ indicates the final trading decision vector, price vector, and offloading rate vector, respectively. Negotiation procedure between the seller and non-member $\boldsymbol{b_n}$ are mainly shown by lines 5-12, where the seller keeps raising its price until $\boldsymbol{b_n}$ decides to process its task locally, e.g., $\boldsymbol{b_n}$ can no longer afford a higher price. Specifically, under each quoted price, $\boldsymbol{b_n}$ determines the optimal offloading rate that can maximize its utility (line 7). After all the quotation rounds, the seller decides a final trading decision vector by solving a knapsack problem with grouped items, where different prices can be charged to different non-members associated with various offloading rates (line 14).

The computational complexity of the quotation procedure associated with Algorithm 3 can be generally expressed as $O\left(\sum_{n=\kappa+1}^{n=|\boldsymbol{\mathcal{B}}|} Count_n\right)$, where $\sum_{n=\kappa+1}^{n=|\boldsymbol{\mathcal{B}}|} Count_n$ indicates the overall quotation rounds during a spot trading. Besides, computational complexity of the knapsack problem with grouped items for final trading decision (given by line 14) is calculated by $O\left(\text{Number of price-offloading rate pairs} \times \left(S - \sum_{n=1}^{n=\kappa} \alpha_n\right)\right)$ (e.g., there are 8 price-offloading rate pairs associated with differential pricing-based spot trading in Fig. 2). As for the optimality, two perspectives are concerned: *i)*. each non-member can reach its optimal utility under every given quoted price (given in (26b)); while *ii)*. the seller can achieve its maximum utility by charging non-members with different prices (given in (26a)). Fig. 2 shows an example of spot trading under differential pricing, as well as the major differences between the two pricing rules.

V. Experimental Results

This section presents comprehensive simulation results and performance evaluations, illustrating the validity of the proposed overbooking-enabled computing resource trading mechanism. Specifically, simulations are implemented via MATLAB R2019b platform on desktop computer with Intel Core i7-4770 3.40 GHz CPU and 16.0 GB RAM. For notational simplicity, the proposed mechanisms under uniform and differential pricing are abbreviated to "Futures_Spot_OverB_UP", and "Futures_Spot_OverB_DP", respectively.

A. Baseline Method

To achieve better evaluation, key baseline methods in this simulation are considered:

- **Equal-booking-based trading under uniform pricing in futures and spot integrated market (Futures_Spot_EqualB_UP):** In futures market, Algorithm 1 is performed considering $\kappa = S$; in spot market, seller trades with non-members under uniform pricing (Algorithm 2).
- **Equal-booking-based trading under differential pricing in futures and spot integrated market (Futures_Spot_EqualB_DP):** In futures market, Algorithm 1 is performed considering $\kappa = S$; in spot market, seller trades with non-members under differential pricing (Algorithm 3).
- **Spot trading under uniform pricing (Spot_UP):** Without considering futures market, all the trading are performed by following spot trading mode under uniform pricing. Namely, $\kappa = 0$, and Algorithm 2 is performed during each trading.
- **Spot trading under differential pricing (Spot_DP):** Without considering futures market, all the trading are performed by following spot trading mode under differential pricing. Namely, $\kappa = 0$, and Algorithm 3 is performed during each trading.

B. Critical Indicator

Apart from the players' utilities, additional evaluation indicators are considered as follows:

- **Decision-making cost (DMC):** in each trading, DMC denotes the cost (e.g., battery consumption) that players have spent on trading decision-making. Since DMC is difficult to be quantized by a numerical value (e.g., it is challenging to estimate the amount of battery capacity consumed in each quotation); in this simulation, DMC is described by the number of quotations (e.g., value of $Count_n$ in Algorithm 2 and Algorithm 3 for each non-member). Apparently, larger DMC presents heavier cost on onsite decision-making.
- **Decision-making latency (DML):** DML denotes the time that players have spent on trading decision-making, which is estimated by considering the end-to-end delay t^{E2E} of wireless communication channels. Note that members no longer have to spend extra time on trading decision-making, as benefitted from the pre-signed forward contract, in this simulation, DML is mainly considered for non-members. Consequently, DML of a non-member $\boldsymbol{b_n}$ $(n \in \{\kappa + 1, \ldots, |\boldsymbol{\mathcal{B}}|\})$ is calculated by $t_n^{DML} = \alpha_n \times Count_n \times t^{E2E}$.
- **Task completion time (TCT):** Since DML can directly affect the actual TCT of each non-member, for

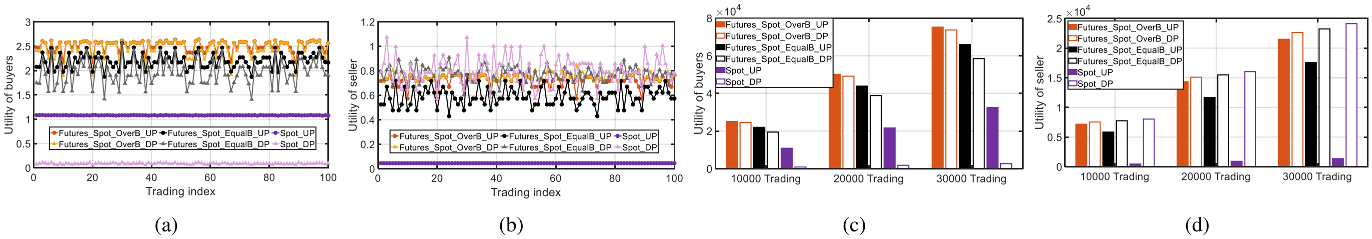

Fig. 3. Short-term and long-term performance on players' utilities.

TABLE I

ADDITIONAL ANALYSIS ON SHORT-TERM PERFORMANCE (ALGO 1: FUTURES_SPOT_OVERB_UP, ALGO 2: FUTURES_SPOT_OVERB_DP, ALGO 3: FUTURES_SPOT_EQUALB_UP, ALGO 4: FUTURES_SPOT_EQUALB_DP, ALGO 5: SPOT_UP, ALGO 6: SPOT_DP)

	Algo 1	**Algo 2**	**Algo 3**	**Algo 4**	**Algo 5**	**Algo 6**
Sum utility of buyers (Figs. 3(a)-Fig. 3(b))	251.61	247.32	220.15	195.70	108.19	8.83
Sum utility of seller (Figs. 3(a)-Fig. 3(b))	72.29	74.94	58.79	77.14	4.57	80.41
Sum task completion time (Figs. 4(a)-4(b))	324.07	328.58	568.26	586.35	904.59	992.20
Sum energy consumption (Figs. 4(a)-4(b))	131.39	135.21	131.47	146.98	131.47	190.77
Sum decision-making cost (Figs. 5(a)-5(b))	13181	12792	53851	52318	109907	106793
Sum decision-making latency (Figs. 5(a)-5(b))	79.09	76.75	323.11	313.91	659.44	640.76
Average time utilization rate (Fig. 6(a))	79.51%	80.97%	42.42%	46.35%	26.39%	35.30%
Average resource utilization rate (Fig. 6(b))	100%	98.41%	100%	93.60%	100%	75.13%

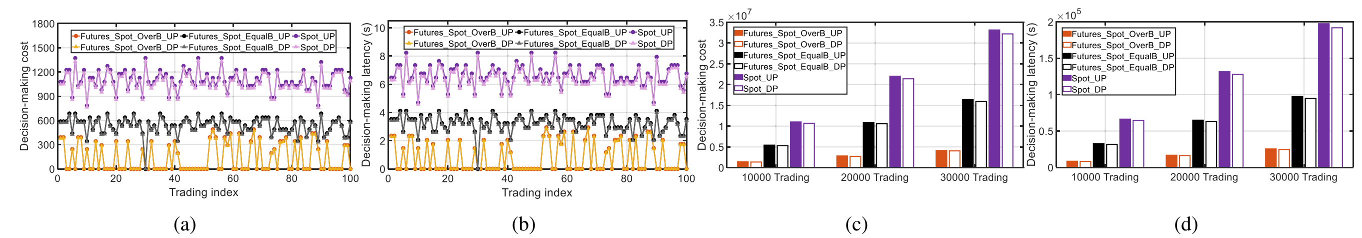

Fig. 4. Short-term and long-term performance on decision-making cost and decision-making latency.

$\boldsymbol{b_n}$ ($n \in \{\kappa+1, \ldots, |\boldsymbol{\mathcal{B}}|\}$), TCT of which can be calculated by (28).

$$t_n^{TCT} = x_n \alpha_n \left(\frac{\lambda_n d^{size}}{W \log_2 (1+e^{tran}\gamma_n)} + \frac{\lambda_n d^{comp}}{f^s}, \frac{(1-\lambda_n)\, d^{comp}}{f^b} \right)^+ + (1-x_n) \frac{d^{comp}}{f^b} + t_n^{DML} \quad (28)$$

Besides, TCT of a member $\boldsymbol{b_m}$ ($m \in \{1, 2, \ldots, \kappa\}$) is computed by the following (29).

$$t_m^{TCT} = \begin{cases} \alpha_m \left(\dfrac{d^{size}}{W \log_2 (1+e^{tran}\gamma_m)} + \dfrac{d^{comp}}{f^s} \right), & \boldsymbol{b_m} \text{ is not a volunteer} \\ \dfrac{d^{comp}}{f^b}, & \boldsymbol{b_m} \text{ is a volunteer} \end{cases} \quad (29)$$

• **Time utilization rate (TUR):** TUR represents the time efficiency of each resource trading calculated by (30). Apparently, large TUR refers to better time efficiency of resource trading.

$$\text{TUR} = 1 - \frac{\sum_{n=\kappa+1}^{n=|\boldsymbol{\mathcal{B}}|} t_n^{DML}}{\sum_{m=1}^{m=\kappa} t_m^{TCT} + \sum_{n=\kappa+1}^{n=|\boldsymbol{\mathcal{B}}|} t_n^{TCT}} \quad (30)$$

• **Resource utilization rate (RUR):** RUR indicates the ratio of the amount of resources occupied by the buyers to the seller's total available resources in each trading. Apparently, a large value of RUR presents a better utilization of computing resources.

Major parameters are set as follows: $S = 15$, $|\boldsymbol{\mathcal{B}}| = 30$, $a = 0.76$, $d^{size} = 0.5$Mb, $d^{comp} = 600$cycles/bit$\times d^{size}$, $f^s = 10^{11}$cycles/s, $f^b = 10^9$cycles/s [1]–[4], $e^{loc} = 500$mWatt [50], $e^{tran} = 550$mWatt [5], $\varepsilon_1 = 100$, $\varepsilon_2 = 500$, namely, the value of γ associated with each buyer is randomly chosen from interval $[100, 500]$ during every trading (while the received SNR thus falls within $[17\text{dB}, 24\text{dB}]$ roughly); $W = 6$MHz, $\xi^S = \xi^B = 0.33$, $\xi^V = 0.45$, t^{E2E} is randomly chosen from $[2\text{ms}, 10\text{ms}]$ [48], $\omega_1 = \omega_2 = 0.5$.

C. Performance Evaluation

In this simulation, we analyze the short-term performance upon considering 100 trading, and the long-term performance via simulating large number of trading (e.g., 10000, 20000, and 30000), to evaluate the validity of the proposed overbooking-enabled resource trading mechanism. Fig. 3 and Table I depict the short-term (Figs. 3(a)-3(b)) and long-term performance (Figs. 3(c)-3(d)) on utilities of the seller and buyers. Specifically, Fig. 3(a) and Fig. 3(b) show the sum utility of 30 buyers, and utility of the seller in each trading, respectively; where the proposed Futures_Spot_OverB_UP achieves better players' utilities (both buyers' and the seller's) than Futures_Spot_EqualB_UP in most trading, while outperforming that of Spot_UP in all the trading. Besides, the proposed Futures_Spot_OverB_DP is superior to baseline methods on buyers' utility, although sometimes suffers slightly lower seller's utility than Futures_Spot_EqualB_DP and Spot_DP; the overall seller's utility (100 trading) of which has a small gap in comparison with the two baseline methods under differential pricing, as given by Table I. Figs. 3(c)-3(d)

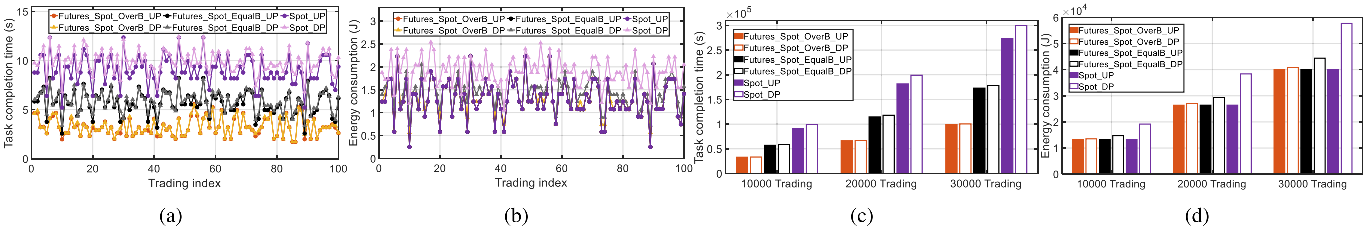

Fig. 5. Short-term and long-term performance on task completion time and energy consumption.

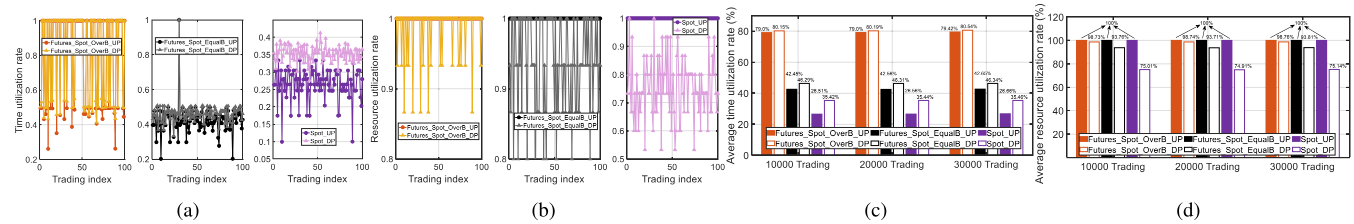

Fig. 6. Short-term and long-term performance on time efficiency and resource utilization.

investigate long-term (cumulative) players' utilities via considering large numbers of trading, by borrowing the idea from Monte Carlo method. As can be seen in Fig. 3(c), the proposed overbooking-enabled mechanism obtains better buyers' utility than baseline methods under both pricing rules owing to that members no longer have to spend extra time on decision-making. In Fig. 3(d), although the proposed Futures_Spot_OverB_DP gets slightly lower long-term seller's utility than Futures_Spot_EqualB_DP, which, however, can achieve better performance on other significant evaluation indicators as described in Figs. 4-6. Specifically, uniform pricing rule only allows the same resource price for different non-members and offloading rates, which can thus represent an extreme case of differential pricing rule. Namely, the seller under differential pricing may achieve larger utility by charging various prices rather than that of uniform pricing; which, in turn, may result in unsatisfying buyers' utility (Figs. 3(c)-3(d)). Besides, spot trading is facing with difficulties to offer mutually beneficial utilities to both parties, since overhead can be caused during onsite negotiation, especially for mobile devices (also see Fig. 4).

Fig. 4 illustrates the short-term (Fig. 4(a) and Fig. 4(b)) and long-term (Fig. 4(c) and Fig. 4(d)) performance on decision-making cost and latency. Benefit from the pre-signed contracts and feasible overbooking rate, the proposed mechanism (where $\kappa^* = 20 > S$) greatly outperforms baseline methods on both DMC and DML (also shown in Table I), which significantly accelerates the service provisioning procedure. Although equal-booking-based trading (where $\kappa^* = S = 15$) may reduce DMC and DML to some extent, which, however, faces challenges to handle "no show" cases (as depicted by Figs. 6(b)-6(d)). Since the players in Spot_UP and Spot_DP have to negotiate a trading agreement before practical service delivery, they may suffer unexpected DMC and DML, which thus pose great challenges to power/battery-constrained mobile devices.

Performance evaluation on task completion time and energy consumption is analyzed by Fig. 5, where Fig. 5(a), Fig. 5(c) and Table I depict that the proposed overbooking-enabled trading mechanism facilitates faster task completion especially when comparing to spot trading, from both short- and long-term perspectives. One major reason refers to the commendable DML brought by the proposed mechanism (Figs. 4(b) and 4(d)), which greatly supports time efficiency since the seller only has to negotiate with non-members during each practical trading. Specifically, considering 10000 trading under uniform pricing in Fig. 5(c), the proposed mechanism (3.307s/trading) achieves 42.23% and 63.55% improvement on task completion time than equal-booking-based method (5.724s/trading) and spot trading (9.072s/trading). Namely, spot buyers may spend roughly 2.7 times longer to complete the same number of tasks, rather than the proposed overbooking-enabled mechanism. In addition, the proposed overbooking-enabled mechanism reaches similar energy consumption comparing with Futures_Spot_EqualB_UP and Spot_UP, while outperforming Futures_Spot_EqualB_DP and Spot_DP on both short- and long-term energy consumption, as demonstrated by Fig. 5(b), Fig. 5(d) and Table I. Investigation on time and resource utilization are detailed in Fig. 6. Thanks to the pre-determined forward contracts among seller and members as well as the overbooking policy, the proposed mechanism enables far better TUR than baseline methods (see Fig. 6(a) and Table I). In Fig. 6(c), the proposed Futures_Spot_OverB_UP achieves averagely 85.98% and 197.78% improvement on time utilization, in comparison with Futures_Spot_EqualB_UP and Spot_UP. Besides, Futures_Spot_OverB_DP obtains averagely 73.37% and 126.56% increase in time utilization, comparing to Futures_Spot_EqualB_DP and Spot_DP. Fig. 6(b), Fig. 6(d), and Table I illustrate that the proposed mechanism offers substantial resource utilization; namely, overbooking provides a commendable reference in handling dynamic resource demand in trading market, e.g., "no shows". Specifically, the proposed mechanism achieves 5.32% and 31.62% improvement on resource utilization under differential pricing, as compared to equal-booking-based and spot trading mechanisms (as depicted by Fig. 6(d)). In summary, the proposed overbooking-enabled trading mechanism under futures-spot integrated market offers mutually beneficial utilities to both parties. Besides, commendable decision-making latency, faster task completion and lower energy consumption can be achieved, while facilitating sufficient time/resource utilization, in comparison with conventional trading methods, e.g., equal-booking-based trading and spot trading.

VI. Conclusion

Motivated by challenges of excessive latency and cost incurred by onsite decision-making, as well as the possible "no show" of smart devices, in this paper, an overbooking-enabled resource trading mechanism considering an edge server (seller) and multiple smart devices (buyers) is investigated under mobile edge network architecture, via integrating both futures and spot market. Specifically, in futures market, mutually beneficial and risk tolerable forward contract as well as the relevant overbooking rate are studied, for which an effective bilateral negotiation scheme is proposed by alternatively optimizing the seller's and members' expected utilities. For the spot trading problem upon considering uniform pricing and differential pricing rules, we propose two bilateral negotiation schemes via addressing non-convex optimization and knapsack problems, by analyzing the current network/market conditions. Experimental results demonstrate that the proposed overbooking-enabled resource trading mechanism achieves mutually beneficial utilities for both the seller and buyers, and outperforms baseline methods on critical indicators such as decision-making latency and cost, as well as time and resource utilization.

Appendix A
Derivation of Expected Utility of Member

Let random variable $X1 = \sum_{m=1}^{m=\kappa}\alpha_m$, and $X2 = (X1, S)^{-}$ for analytical simplicity, we first discuss the PMF of $X1$ as given by (31).

$$\Pr\left(X1 = x\right) = C_\kappa^x a^x (1-a)^{\kappa - x}, \quad x \in \{0, 1, \ldots, \kappa\} \tag{31}$$

Based on (31), we consider $\mathrm{E}\left[X2\right]$ via the following two cases:

- **Case 1** ($\kappa \leqslant S$)**:** $X2 = X1$, we have $\mathrm{E}\left[X2\right] = \mathrm{E}\left[X1\right] = \kappa a$.
- **Case 2** ($\kappa > S$)**:** $X2 = \begin{cases} X1, & X1 < S \\ S, & X1 \geqslant S \end{cases}$ and we have PMF of $X2$ as given by (32).

$$\Pr\left(X2 = x\right) = \begin{cases} C_\kappa^x a^x (1-a)^{\kappa - x}, & 0 \leqslant x \leqslant S-1 \\ \sum_{i=S}^{i=\kappa} C_\kappa^i a^i (1-a)^{\kappa - i}, & x = S \end{cases} \tag{32}$$

Accordingly, $\mathrm{E}\left[X2\right]$ in **Case 2** can be calculated by (33).

$$\mathrm{E}\left[X2\right] = \sum_{i=0}^{i=S-1} i C_\kappa^i a^x (1-a)^{\kappa-i} + S\sum_{i=S}^{i=\kappa} C_\kappa^i a^i (1-a)^{\kappa - i} \tag{33}$$

As a result, $\mathrm{E}\left[V\right]$ is represented by (eq34).

$$\begin{aligned} \mathrm{E}\left[V\right] &= \mathrm{E}\left[X1 - X2\right] \\ &= \begin{cases} 0, & \kappa \leqslant S \\ \kappa a - \left(\sum_{i=0}^{i=S-1} i C_\kappa^i a^i (1-a)^{\kappa - i} \right. \\ \left. + S\sum_{i=S}^{i=\kappa} C_\kappa^i a^i (1-a)^{\kappa-i}\right), & \kappa > S \end{cases} \end{aligned} \tag{34}$$

Let random variable $Y1 = \frac{1}{\log_2(1+e^{tran}\gamma_m)}$ for notational simplicity, the CDF of $Y1$ is given by (35), according to $\gamma_m \sim \mathrm{U}(\varepsilon_1, \varepsilon_2)$.

$$\mathrm{F}_{Y1}\left(y\right) = \begin{cases} 0, & y < \frac{1}{log_2\left(1 + e^{tran}\varepsilon_2\right)} \\ 1 - \frac{2^{\frac{1}{y}} - 1 - e^{tran}\varepsilon_1}{e^{tran}\left(\varepsilon_2 - \varepsilon_1\right)}, & \frac{1}{\log_2\left(1 + e^{tran}\varepsilon_2\right)} \leqslant y \leqslant \frac{1}{\log_2\left(1 + e^{tran}\varepsilon_1\right)} \\ 1, & y > \frac{1}{\log_2\left(1 + e^{tran}\varepsilon_1\right)} \end{cases} \tag{35}$$

The PDF of Y can be obtained as given in (36), based on (35).

$$\begin{aligned} \Pr\left(Y1 = y\right) &= \frac{\partial F_{Y1}\left(y\right)}{\partial y} \\ &= \begin{cases} \frac{\ln 2 \times 2^{\frac{1}{y}}}{y^2 e^{tran}\left(\varepsilon_2 - \varepsilon_1\right)}, & \frac{1}{\log_2\left(1 + e^{tran}\varepsilon_2\right)} \leqslant y \leqslant \frac{1}{\log_2\left(1 + e^{tran}\varepsilon_1\right)} \\ 0, & \text{otherwise} \end{cases} \end{aligned} \tag{36}$$

Accordingly, $\mathrm{E}\left[Y1\right]$ is thus calculated by (37), where $\mathbb{C}_1 = \ln 2 \times \log_2\left(1 + e^{tran}\varepsilon_1\right)$ and $\mathbb{C}_2 = \ln 2 \times \log_2\left(1 + e^{tran}\varepsilon_2\right)$, for notational simplicity.

$$\begin{aligned} \mathrm{E}\left[Y1\right] &= \int_{\frac{1}{\log_2\left(1+e^{tran}\varepsilon_2\right)}}^{\frac{1}{\log_2\left(1+e^{tran}\varepsilon_1\right)}} y \Pr\left(Y = y\right) dy \\ &= \frac{\ln 2 \times \int_{\frac{1}{\log_2\left(1+e^{tran}\varepsilon_2\right)}}^{\frac{1}{\log_2\left(1+e^{tran}\varepsilon_1\right)}} \left(\frac{2^{\frac{1}{y}}}{y}\right) dy}{e^{tran}(\varepsilon_2 - \varepsilon_1)} \\ &= \frac{\ln 2 \times \int_{\mathbb{C}_1}^{\mathbb{C}_2}\left(\frac{e^y}{y}\right) dy}{e^{tran}\left(\varepsilon_2 - \varepsilon_1\right)} \end{aligned} \tag{37}$$

According to (37), $\mathrm{E}\left[U_m^{PP}\right]$ is expressed by (38) and $\overline{\mathcal{U}^{Mem}}(p, q, r, \kappa, \boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{Y}})$ is thus obtained.

$$\begin{aligned} &\mathrm{E}\left[U_m^{PP}\right] \\ &= \left(\frac{\omega_1 + \omega_2 e^{loc}}{f^b} - \frac{\omega_1}{f^s} - p\right) d^{comp} \\ &\quad - \frac{\ln 2 d^{size}\left(\omega_1 + \omega_2 e^{tran}\right) \times \int_{\mathbb{C}_1}^{\mathbb{C}_2}\left(\frac{e^y}{y}\right) dy}{W e^{tran}\left(\varepsilon_2 - \varepsilon_1\right)} \end{aligned} \tag{38}$$

Appendix B
Derivation of Risks of Member

Based on the pre-determined $Y1 = \frac{1}{\log_2(1+e^{tran}\gamma_m)}$, $\mathcal{R}^{MRisk}\left(p, q, \boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{Y}}\right)$ is rewritten as (39).

$$\begin{aligned} &\mathcal{R}^{MRisk}\left(p, q, \boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{Y}}\right) \\ &= \Pr\left(\frac{\alpha_m U_m^{PP} + (1-\alpha_m) U^{DE}}{U_{min}} \leqslant \xi_1\right) \\ &= \Pr\left(\alpha_m\left(\mathbb{C}_3 - \mathbb{C}_4 Y1\right) \leqslant \mathbb{C}_5\right), \end{aligned} \tag{39}$$

where $\mathbb{C}_3 = \frac{\omega_1 d^{comp}+\omega_2 e^{loc} d^{comp}}{f^b} - \frac{\omega_1 d^{comp}}{f^s} + q d^{comp} - p d^{comp}$, $\mathbb{C}_4 = \frac{\omega_2 e^{tran} d^{size}+\omega_1 d^{size}}{W}$, and $\mathbb{C}_5 = \xi_1 U_{min} + q d^{comp}$, which are constants under any given p and q for notational simplicity. Let random variable $Y2 = \mathbb{C}_3 - \mathbb{C}_4 Y1$, we discuss the CDF of $Y2$ which is given by (40).

$$\mathrm{F}_{Y2}(y) = 1 - \mathrm{F}_{Y1}\left(\frac{\mathbb{C}_3 - y}{\mathbb{C}_4}\right) = \begin{cases} 0, & y < \mathbb{C}_3 - \frac{\mathbb{C}_4}{\log_2(1+e^{tran}\varepsilon_1)} \\ \frac{2^{\frac{\mathbb{C}_4}{\mathbb{C}_3 - y}} - 1 - e^{tran}\varepsilon_1}{e^{tran}(\varepsilon_2 - \varepsilon_1)}, & \mathbb{C}_3 - \frac{\mathbb{C}_4}{\log_2(1+e^{tran}\varepsilon_1)} \leqslant y \leqslant \mathbb{C}_3 - \frac{\mathbb{C}_4}{\log_2(1+e^{tran}\varepsilon_2)} \\ 1, & y > \mathbb{C}_3 - \frac{\mathbb{C}_4}{\log_2(1+e^{tran}\varepsilon_2)} \end{cases} \tag{40}$$

Let random variable $Y3 = \alpha_m Y2$, the CDF of $Y3$ is thus considered by (41).

$$\mathrm{F}_{Y3}(y) = \begin{cases} 0, & y < 0 \\ 1-a, & 0 \leqslant y < \mathbb{C}_3 - \frac{\mathbb{C}_4}{\log_2\left(1+e^{tran}\varepsilon_1\right)} \\ 1-a+a\left(\frac{2^{\frac{\mathbb{C}_4}{\mathbb{C}_3 - y}} - 1 - e^{tran}\varepsilon_1}{e^{tran}(\varepsilon_2 - \varepsilon_1)}\right), & \mathbb{C}_3 - \frac{\mathbb{C}_4}{\log_2(1+e^{tran}\varepsilon_1)} \leqslant y \leqslant \mathbb{C}_3 - \frac{\mathbb{C}_4}{log_2(1+e^{tran}\varepsilon_2)} \\ 1, & y > \mathbb{C}_3 - \frac{\mathbb{C}_4}{\log_2(1+e^{tran}\varepsilon_2)} \end{cases} \tag{41}$$

Accordingly, we recalculate (39) as (42), according to (41).

$$\begin{aligned} &\mathcal{R}^{MRisk}(p,q,\boldsymbol{\mathcal{A}},\boldsymbol{\mathcal{Y}}) \\ &= \Pr(Y3 \leqslant \mathbb{C}_5) \\ &= \begin{cases} 0, & \mathbb{C}_5 < 0 \\ 1-a, & 0 \leqslant \mathbb{C}_5 < \mathbb{C}_3 - \frac{\mathbb{C}_4}{\log_2(1+e^{tran}\varepsilon_1)} \\ 1-a+a\left(\frac{2^{\frac{\mathbb{C}_4}{\mathbb{C}_3 - \mathbb{C}_5}} - 1 - e^{tran}\varepsilon_1}{e^{tran}(\varepsilon_2 - \varepsilon_1)}\right), & \mathbb{C}_3 - \frac{\mathbb{C}_4}{\log_2(1+e^{tran}\varepsilon_1)} \leqslant \mathbb{C}_5 \leqslant \mathbb{C}_3 - \frac{\mathbb{C}_4}{\log_2(1+e^{tran}\varepsilon_2)} \\ 1, & \mathbb{C}_5 > \mathbb{C}_3 - \frac{\mathbb{C}_4}{\log_2(1+e^{tran}\varepsilon_2)} \end{cases} \end{aligned} \tag{42}$$

For VRsik, we first discuss the conditional probability $\Pr(\sum_{m=1}^{m=\kappa-1} \alpha_m > S-1 | \alpha_\kappa = 1)$, describing that a member $\boldsymbol{b_\kappa}$ is under the risk of being selected as a volunteer when it is a performer (e.g., $\alpha_\kappa = 1$, here, we consider member $\boldsymbol{b_\kappa}$ as an example, where the risk is universal for all the members in the proposed market due to that all the buyers are i.i.d.). Let random variable $X3 = \sum_{m=1}^{m=\kappa-1} \alpha_m$, the CDF of $X3$ is expressed by (43).

$$\mathrm{F}_{X3}(x) = \Pr(X3 \leqslant x) = \begin{cases} 0, & x < 0 \\ \sum_{i=0}^{i=\lfloor x \rfloor} C_{\kappa-1}^i a^i (1-a)^{\kappa-1-i}, & 0 \leqslant x \leqslant \kappa - 1 \\ 1, & x > \kappa - 1 \end{cases} \tag{43}$$

Correspondingly, we have $\Pr\left(\sum_{m=1}^{m=\kappa-1} \alpha_m > S-1 \middle| \alpha_\kappa = 1\right)$ calculated by (44).

$$\Pr\left(\sum_{m=1}^{m=\kappa-1} \alpha_m > S-1 \middle| \alpha_\kappa = 1\right) = \begin{cases} 0, & 0 \leqslant \kappa \leqslant S \\ 1 - \sum_{i=0}^{i=S-1} C_{\kappa-1}^i a^i (1-a)^{\kappa-1-i}, & \kappa > S \end{cases} \tag{44}$$

Consequently, the probability of a performer who is undergoing the risk of being selected as a volunteer is given by (45).

$$\mathcal{R}^{VRisk}(\boldsymbol{\mathcal{A}},\kappa) = \begin{cases} 0, & 0 \leqslant \kappa \leqslant S \\ a - \sum_{i=0}^{i=S-1} C_{\kappa-1}^i a^{i+1} (1-a)^{\kappa-1-i}, & \kappa > S \end{cases} \tag{45}$$

Appendix C
Derivation of Risk of Seller

We apply the previous defined $X1$ and $X2$ to describe $\mathcal{R}^{SRisk}(p,q,r,\kappa,\boldsymbol{\mathcal{A}})$ as (46).

$$\mathcal{R}^{SRisk}(p,q,r,\kappa,\boldsymbol{\mathcal{A}}) = \Pr\left((p+r)X2 - (q+r)X1 \leqslant \frac{\xi_2 \overline{\mathcal{U}^{SelF}}(p,q,r,\kappa,\boldsymbol{\mathcal{A}})}{d^{comp}} - q\kappa\right) \tag{46}$$

Consider $\kappa \leqslant S$, we have $X2 = X1$. Thus, $\mathcal{R}^{SelF}(p,q,r,\kappa \leqslant S,\boldsymbol{\mathcal{A}})$ can be calculated by (47), where $\mathbb{C}_6 = \frac{\xi_2 \overline{\mathcal{U}^{SelF}}(p,q,r,\kappa \leqslant S,\boldsymbol{\mathcal{A}})}{d^{comp}(p-q)} - \frac{q\kappa}{(p-q)}$ for notational simplicity.

$$\begin{aligned} &\mathcal{R}^{SRisk}(p,q,r,\kappa \leqslant S,\boldsymbol{\mathcal{A}}) \\ &= \Pr(X1 \leqslant \mathbb{C}_6) \\ &= \begin{cases} 0, & \mathbb{C}_6 < 0 \\ \sum_{i=0}^{i=\lfloor \mathbb{C}_6 \rfloor} C_\kappa^i a^i (1-a)^{\kappa-i}, & 0 \leqslant \mathbb{C}_6 \leqslant \kappa \\ 1, & \mathbb{C}_6 > \kappa \end{cases} \end{aligned} \tag{47}$$

For $\kappa > S$, we consider a random variable Z given by (48),

$$Z = (p+r)X2 - (q+r)\,X1 = \begin{cases} (p-q)\,X1, & X1 < S \\ (p+r)\,S - (q+r)\,X1, & X1 \geqslant S \end{cases} \tag{48}$$

Correspondingly, the PMF of Z can be calculated as (49) based on (31).

$$\Pr(Z=z) = \begin{cases} (1-a)^{\kappa}, & z=0 \\ \cdots \\ C_{\kappa}^{S} a^{S} (1-a)^{\kappa-S}, & z = S(p-q) \\ \cdots \\ a^{\kappa}, & z = S(p-q) - (\kappa-S)(q+r) \\ 0, & \text{otherwise} \end{cases} \tag{49}$$

Apparently, analysis on properties of random variable Z relies on the distribution of $X1$. For example, the CDF associated with Z from 0 to $S(p-q)$ can be calculated via considering the cumulated probability of $X1$ from 0 to S. Correspondingly, we discuss the CDF $\mathrm{F}_Z(z)$ of Z via the following three cases: **Case 1** $(q+r = p-q)$, **Case 2** $(q+r > p-q)$, and **Case 3** $(q+r < p-q)$. Specifically, **Case 1** can be further considered by analyzing three sub-cases (1.1, 1.2, and 1.3), depending on different value ranges of κ:

- **Case 1.1** When $S < \kappa \leqslant 2S$, we have $\mathrm{F}_Z(z)$ shown by (50).

$$\mathrm{F}_Z(z) = \begin{cases} 0, & z<0 \\ \sum_{i=0}^{i=\lfloor \frac{z}{p-q} \rfloor} C_{\kappa}^{i} a^{i}(1-a)^{\kappa-i} + \sum_{i=\lceil 2S - \frac{z}{p-q} \rceil}^{i=\kappa} C_{\kappa}^{i} a^{i}(1-a)^{\kappa-i}, & 0 \leqslant z \leqslant S(p-q) \\ 1, & z > S(p-q) \end{cases} \tag{50}$$

- **Case 1.2** When $\kappa > 2S$, we have the following (51).

$$\mathrm{F}_Z(z) = \begin{cases} 0, & z < (2S-\kappa)(p-q) \\ \sum_{i=\lceil 2S - \frac{z}{p-q} \rceil}^{i=\kappa} C_{\kappa}^{i} a^{i}(1-a)^{\kappa-i}, & (2S-\kappa)(p-q) \leqslant z < 0 \\ \sum_{i=0}^{i=\lfloor \frac{z}{p-q} \rfloor} C_{\kappa}^{i} a^{i}(1-\alpha)^{\kappa-i} + \sum_{i=\lceil 2S - \frac{z}{p-q} \rceil}^{i=\kappa} C_{\kappa}^{i} a^{i}(1-a)^{\kappa-i}, & 0 \leqslant z \leqslant S(p-q) \\ 1, & z > S(p-q) \end{cases} \tag{51}$$

Consequently, CDF of Z in **Case 1** is given as (52) by summarizing (50) and (51):

$$\mathrm{F}_Z(z) = \begin{cases} 0, & z < (0, (2S-\kappa)(p-q))^{-} \\ \sum_{i=0}^{i=\lfloor \frac{z}{p-q} \rfloor} C_{\kappa}^{i} a^{i}(1-a)^{\kappa-i} + \sum_{i=\lceil 2S - \frac{z}{p-q} \rceil}^{i=\kappa} C_{\kappa}^{i} a^{i}(1-a)^{\kappa-i}, & (0, (2S-\kappa)(p-q))^{-} \leqslant z \leqslant S(p-q) \\ 1, & z > S(p-q) \end{cases} \tag{52}$$

Due to space limitation, we omit derivations of **Case 2** $(q+r > p-q)$ and **Case 3** $(q+r < p-q)$, which are similar with **Case 1**. In conclusion, we have $\mathrm{F}_Z(z)$ when $\kappa \geqslant S$ as (53).

$$\mathrm{F}_Z(z) = \begin{cases} 0, & z < (0, S(p-q) - (\kappa-S)(q+r))^{-} \\ \sum_{i=0}^{i=\lfloor \frac{z}{p-q} \rfloor} C_{\kappa}^{i} a^{i}(1-a)^{\kappa-i} + \sum_{i=\lceil \frac{S(p-q)-z}{q+r} + S \rceil}^{i=\kappa} C_{\kappa}^{i} a^{i}(1-a)^{\kappa-i}, & (0, S(p-q) - (\kappa-S)(q+r))^{-} \leqslant z \leqslant S(p-q) \\ 1, & z > S(p-q) \end{cases} \tag{53}$$

Notably, let $\sum_{i=0}^{i=\lfloor \frac{z}{p-q} \rfloor} C_{\kappa}^{i} a^{i}(1-a)^{\kappa-i} = 0$ when $\frac{z}{p-q} < 0$, and $\sum_{i=\lceil \frac{S(p-q)-z}{q+r} + S \rceil}^{i=\kappa} C_{\kappa}^{i} a^{i}(1-a)^{\kappa-i} = 0$ when $\lceil \frac{S(p-q)-z}{q+r} + S \rceil > \kappa$. Correspondingly, risk of the seller upon considering $\kappa > S$ can thus be calculated by (22), according to (53).

Appendix D
Derivation of the Optimal Offloading Rate

Under any given price g_n, we discuss the optimization problem (26b) of maximizing a non-member's ($\alpha_n = 1, n \in \{\kappa+1, \ldots, |\boldsymbol{\mathcal{B}}|\}$) utility in problem $\boldsymbol{\mathcal{F}_2}$ by the following cases.

- **Case 1:** when $\frac{\lambda_n d^{size}}{W\log_2(1+e^{tran}\gamma_n)} + \frac{\lambda_n d^{comp}}{f^s} \leqslant \frac{(1-\lambda_n) d^{comp}}{f^b}$, we have $0 \leqslant \lambda_n \leqslant \mathbb{C}_8$ where $\mathbb{C}_8 = \frac{d^{comp} f^s}{\frac{d^{size} f^s f^b}{W\log_2(1+e^{tran}\gamma_n)} + d^{comp} f^b + d^{comp} f^s}$ for notational simplicity. Thus, (26b) is rewritten as $\boldsymbol{\mathcal{F}_4}$.

In this case, when $\frac{\partial \mathcal{U}_n^{NonM}}{\partial \lambda_n} \leqslant 0$, we have $\lambda_n = 0$; else, we have $\lambda_n = \mathbb{C}_8$.

- **Case 2:** when $\frac{\lambda_n d^{size}}{W\log_2(1+e^{tran}\gamma_n)} + \frac{\lambda_n d^{comp}}{f^s} > \frac{(1-\lambda_n) d^{comp}}{f^b}$, we have $\mathbb{C}_8 \leqslant \lambda_n \leqslant 1$. We reconsider (26b) as $\boldsymbol{\mathcal{F}_5}$ shown

$$\boldsymbol{\mathcal{F}_4}: \mathop{\arg\min}_{\lambda_n \in [0, \mathbb{C}_8]} \left(\frac{\omega_2 e^{tran} d^{size}}{W\log_2(1+e^{tran}\gamma_n)} + g_n d^{comp} - \frac{\omega_1 d^{comp}}{f^b} - \frac{\omega_2 e^{loc} d^{comp}}{f^b} \right) \lambda_n \tag{54}$$

$$\boldsymbol{\mathcal{F}_5}: \mathop{\arg\min}_{\lambda_n \in (\mathbb{C}_8, 1]} \left(\frac{\omega_1 d^{size} + \omega_2 e^{tran} d^{size}}{W\log_2(1+e^{tran}\gamma_n)} + g_n d^{comp} + \frac{\omega_1 d^{comp}}{f^s} - \frac{\omega_2 e^{loc} d^{comp}}{f^b} \right) \lambda_n \tag{55}$$

by (55). In this case, when $\frac{\partial \mathcal{U}_n^{NonM}}{\partial \lambda_n} \leqslant 0$, we have $\lambda_n = \mathbb{C}_8$; else, we have $\lambda_n = 1$. Similarly, problem (27b) can also be solved according to (54) and (55), shown at the bottom of the previous page.

References

[1] C. Yi, J. Cai, and Z. Su, "A multi-user mobile computation offloading and transmission scheduling mechanism for delay-sensitive applications," *IEEE Trans. Mobile Comput.*, vol. 19, no. 1, pp. 29–43, Jan. 2020.

[2] J. Yan, S. Bi, Y. J. Zhang, and M. Tao, "Optimal task offloading and resource allocation in mobile-edge computing with inter-user task dependency," *IEEE Trans. Wireless Commun.*, vol. 19, no. 1, pp. 235–250, Jan. 2020.

[3] T. X. Tran and D. Pompili, "Joint task offloading and resource allocation for multi-server mobile-edge computing networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 1, pp. 856–868, Jan. 2019.

[4] E. El Haber, T. M. Nguyen, and C. Assi, "Joint optimization of computational cost and devices energy for task offloading in multi-tier edge-clouds," *IEEE Trans. Commun.*, vol. 67, no. 5, pp. 3407–3421, May 2019.

[5] 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.

[6] Z. Zhou, X. Chen, E. Li, L. Zeng, K. Luo, and J. Zhang, "Edge intelligence: Paving the last mile of artificial intelligence with edge computing," *Proc. IEEE*, vol. 107, no. 8, pp. 1738–1762, Aug. 2019.

[7] 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. 2014.

[8] K. Chard and K. Bubendorfer, "High performance resource allocation strategies for computational economies," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 1, pp. 72–84, Apr. 2013.

[9] J. Ma, Y. K. Tse, X. Wang, and M. Zhang, "Examining customer perception and behaviour through social media research—An empirical study of the united airlines overbooking crisis," *Transp. Res. E, Logistics Transp. Rev.*, vol. 127, pp. 192–205, Jul. 2019.

[10] N. Haynes and D. Egan, "The perceptions of frontline employees towards hotel overbooking practices: Exploring ethical challenges," *J. Revenue Pricing Manage.*, vol. 137, pp. 1–10, Jan. 2020.

[11] J. Liu, X. Jiang, and S. Horiguchi, "Opportunistic link overbooking for resource efficiency under per-flow service guarantee," *IEEE Trans. Commun.*, vol. 58, no. 6, pp. 1769–1781, Jun. 2010.

[12] A. Adebayo, D. B. Rawat, and M. Song, "Prediction based adaptive RF spectrum reservation in wireless virtualization," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2020, pp. 1–6.

[13] M.-A. Messous, S.-M. Senouci, H. Sedjelmaci, and S. Cherkaoui, "A game theory based efficient computation offloading in an UAV network," *IEEE Trans. Veh. Technol.*, vol. 68, no. 5, pp. 4964–4974, May 2019.

[14] Y. Wang *et al.*, "A game-based computation offloading method in vehicular multiaccess edge computing networks," *IEEE Internet Things J.*, vol. 7, no. 6, pp. 4987–4996, Jun. 2020.

[15] G. Gao, M. Xiao, J. Wu, H. Huang, S. Wang, and G. Chen, "Auction-based VM allocation for deadline-sensitive tasks in distributed edge cloud," *IEEE Trans. Services Comput.*, vol. 14, no. 6, pp. 1702–1716, Nov. 2021.

[16] M. Liwang, S. Dai, Z. Gao, Y. Tang, and H. Dai, "A truthful reverse-auction mechanism for computation offloading in cloud-enabled vehicular network," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4214–4227, Jun. 2019.

[17] Z. Gao, M. Liwang, S. Hosseinalipour, H. Dai, and X. Wang, "A truthful auction for graph job allocation in vehicular cloud-assisted networks," *IEEE Trans. Mobile Comput.*, early access, Feb. 16, 2021, doi: 10.1109/TMC.2021.3059803.

[18] B. Shojaiemehr, A. M. Rahmani, and N. N. Qader, "A three-phase process for SLA negotiation of composite cloud services," *Comput. Standards Interface*, vol. 64, pp. 85–95, May 2019.

[19] P. Wang, J. Meng, J. Chen, T. Liu, Y. Zhan, W.-T. Tsai, and Z. Jin, "Smart contract-based negotiation for adaptive QoS-aware service composition," *IEEE Trans. Parallel Distrib. Syst.*, vol. 30, no. 6, pp. 1403–1420, Jun. 2019.

[20] S. E. Khatib and F. D. Galiana, "Negotiating bilateral contracts in electricity markets," *IEEE Trans. Power Syst.*, vol. 22, no. 2, pp. 553–562, May 2007.

[21] A. J. Conejo, R. Garcia-Bertrand, M. Carrion, Á. Caballero, and A. de Andrés, "Optimal involvement in futures markets of a power producer," *IEEE Trans. Power Syst.*, vol. 23, no. 2, pp. 703–711, May 2008.

[22] J. M. Morales, S. Pineda, A. J. Conejo, and M. Carrion, "Scenario reduction for futures market trading in electricity markets," *IEEE Trans. Power Syst.*, vol. 24, no. 2, pp. 878–888, May 2009.

[23] 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.

[24] H. Li, T. Shu, F. He, and J. Bin Song, "Futures market for spectrum trade in wireless communications: Modeling, pricing and hedging," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2013, pp. 1–6.

[25] M. Liwang, R. Chen, and X. Wang, "Resource trading in edge computing-enabled IoV: An efficient futures-based approach," *IEEE Trans. Services Comput.*, early access, Apr. 2, 2021, doi: 10.1109/TSC.2021.3070746.

[26] L. Gao, B. Shou, Y.-J. Chen, and J. Huang, "Combining spot and futures markets: A hybrid market approach to dynamic spectrum access," *Operations Res.*, vol. 64, no. 4, pp. 794–821, Aug. 2016.

[27] K. Vanmechelen, W. Depoorter, and J. Broeckhove, "Combining futures and spot markets: A hybrid market approach to economic grid resource management," *J. Grid Comput.*, vol. 9, no. 1, pp. 81–94, Mar. 2011.

[28] N. Wu, X. Zhou, and M. Sun, "Incentive mechanisms and impacts of negotiation power and information availability in multi-relay cooperative wireless networks," *IEEE Trans. Wireless Commun.*, vol. 18, no. 7, pp. 3752–3765, Jul. 2019.

[29] X. Gao, K. Wang, and Y. Yu, "To rent or to share?" in *Proc. IEEE Int. Conf. Commun., Control, Comput. Technol. Smart Grids (SmartGridComm)*, Oct. 2018, pp. 1–7.

[30] L. Zanzi, V. Sciancalepore, A. Garcia-Saavedra, and X. Costa-Perez, "OVNES: Demonstrating 5G network slicing overbooking on real deployments," in *Proc. IEEE INFOCOM Conf. Comput. Commun. Workshops (INFOCOM WKSHPS)*, Apr. 2018, pp. 1–2.

[31] L. Zanzi, J. X. Salvat, V. Sciancalepore, A. G. Saavedra, and X. Costa-Perez, "Overbooking network slices end-to-end: Implementation and demonstration," in *Proc. ACM SIGCOMM Conf. Posterc Demos*, Aug. 2018, pp. 144–146.

[32] C. Sexton, N. Marchetti, and L. A. DaSilva, "On provisioning slices and overbooking resources in service tailored networks of the future," *IEEE/ACM Trans. Netw.*, vol. 28, no. 5, pp. 2106–2119, Oct. 2020.

[33] J. Son, A. V. Dastjerdi, R. N. Calheiros, and R. Buyya, "SLA-aware and energy-efficient dynamic overbooking in SDN-based cloud data centers," *IEEE Trans. Sustain. Comput.*, vol. 2, no. 2, pp. 76–89, Apr. 2017.

[34] S. Alanazi and B. Hamdaoui, "Energy-aware resource management framework for overbooked cloud data centers with SLA assurance," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2018, pp. 1–6.

[35] P. Rahimzadeh, Y. Im, G. Jung, C. Joe-Wong, and S. Ha, "ECHO: Efficiently overbooking applications to create a highly available cloud," in *Proc. IEEE 39th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Jul. 2019, pp. 1–11.

[36] M. Yao, D. Chen, and J. Shang, "Optimal overbooking policy for cloud service providers: Profit and service quality," *IEEE Access*, vol. 7, pp. 96132–96147, 2019.

[37] F. Zhang, Z. Tang, M. Chen, X. Zhou, and W. Jia, "A dynamic resource overbooking mechanism in fog computing," in *Proc. IEEE 15th Int. Conf. Mobile Ad Hoc Sensor Syst. (MASS)*, Oct. 2018, pp. 89–97.

[38] Y. He, J. Ren, G. Yu, and Y. Cai, "D2D communications meet mobile edge computing for enhanced computation capacity in cellular networks," *IEEE Trans. Commun.*, vol. 18, no. 3, pp. 1750–1763, Feb. 2019.

[39] F. Liu, E. Bala, E. Erkip, M. C. Beluri, and R. Yang, "Small-cell traffic balancing over licensed and unlicensed bands," *IEEE Trans. Veh. Technol.*, vol. 64, no. 12, pp. 5850–5865, Dec. 2015.

[40] F. Zhou and R. Q. Hu, "Computation efficiency maximization in wireless-powered mobile edge computing networks," *IEEE Trans. Wireless Commun.*, vol. 19, no. 5, pp. 3170–3184, May 2020.

[41] B. Zheng, C. You, and R. Zhang, "Intelligent reflecting surface assisted multi-user OFDMA: Channel estimation and training design," *IEEE Trans. Wireless Commun.*, vol. 19, no. 12, pp. 8315–8329, Dec. 2020.

[42] R. T. Marler and J. S. Arora, "The weighted sum method for multi-objective optimization: New insights," *Structural Multidisciplinary Optim.*, vol. 41, no. 6, pp. 853–862, 2010.

[43] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*. Hoboken, NJ, USA: Wiley, 2001.
[44] C. Zhang, A. K. Qin, W. Shen, L. Gao, K. C. Tan, and X. Li, "ϵ-constrained differential evolution using an adaptive ϵ-level control method," *IEEE Trans. Syst., Man, Cybern. Syst.*, early access, Aug. 2020, doi: 10.1109/TSMC.2020.3010120.
[45] M. LiWang, Z. Gao, and X. Wang, "Energy-aware graph job allocation in software defined air-ground integrated vehicular networks," 2020, *arXiv:2008.01144*.
[46] M. Liu and Y. Liu, "Price-based distributed offloading for mobile-edge computing with computation capacity constraints," *IEEE Wireless Commun. Lett.*, vol. 7, no. 3, pp. 420–423, Jun. 2018.
[47] F. Castillo-Zunino and P. Keskinocak, "Bi-criteria multiple knapsack problem with grouped items," 2020, *arXiv:2006.00322*.
[48] "New services & applications with 5G ultra-reliable low latency communications," 5G Americas, Bellevue, WA, USA, White Paper, Nov. 2018. [Online]. Available: https://www.5gamericas.org/new-services-applications-with-5g-ultra-reliablelow-latency-communications/
[49] L. Chen and G. Zhang, "Packing groups of items into multiple knapsacks," *ACM Trans. Algorithms*, vol. 14, no. 4, pp. 1–24, Oct. 2018.
[50] Z. Zhou, J. Feng, Z. Chang, and X. Shen, "Energy-efficient edge computing service provisioning for vehicular networks: A consensus ADMM approach," *IEEE Trans. Veh. Technol.*, vol. 68, no. 5, pp. 5087–5099, May 2019.
[51] X. Hou *et al.*, "Reliable computation offloading for edge-computing-enabled software-defined IoV," *IEEE Internet Things J.*, vol. 7, no. 8, pp. 7097–7111, Aug. 2020.

Minghui Liwang (Member, IEEE) received the Ph.D. degree from the School of Informatics, Xiamen University, China, in 2019. She is now an Assistant Professor with the School of Informatics, Xiamen University. She was a Post-Doctoral Fellow with the Department of Electrical and Computer Engineering, Western University, ON, Canada, from 2019 to 2021, and a Visiting Scholar with the Department of Electrical and Computer Engineering, North Carolina State University, NC, USA, from 2017 to 2018. Her research interests include wireless communication systems, Internet of Vehicles, cloud/edge/service computing, as well as economic models and applications in wireless communication networks. She has served as a TPC Member for Communication QoS, Reliability and Modeling, held in conjunction with IEEE GLOBECOM 2020; and Selected Areas in Communications: Aerial Communications, held in conjunction with IEEE GLOBECOM 2021.

Ruitao Chen received the B.Eng. and master's degrees in electrical engineering from Tongji University, Shanghai, China, in 2012 and 2015, respectively, and the Ph.D. degree in electrical and computer engineering, Western University, ON, Canada, in 2021. He is currently working towards a post-doctoral fellow position in the area of mobile collaborative computing.

Xianbin Wang (Fellow, IEEE) received the Ph.D. degree in electrical and computer engineering from the National University of Singapore in 2001. He is currently a Professor and the Tier-1 Canada Research Chair at Western University, Canada. Prior to joining Western, he was with the Communications Research Centre Canada (CRC) as a Research Scientist/a Senior Research Scientist from July 2002 to December 2007. From January 2001 to July 2002, he was a System Designer at STMicroelectronics. His current research interests include 5G/6G technologies, the Internet-of-Things, communications security, machine learning, and intelligent communications. He has over 450 highly cited journals and conference papers, in addition to 30 granted and pending patents and several standard contributions. He is a fellow of the Canadian Academy of Engineering and the Engineering Institute of Canada and an IEEE Distinguished Lecturer. He has received many awards and recognitions, including the Canada Research Chair, CRC President's Excellence Award, the Canadian Federal Government Public Service Award, the Ontario Early Researcher Award, and six IEEE Best Paper Awards. He currently serves/has served as the Editor-in-Chief, the Associate Editor-in-Chief, and an Editor/an Associate Editor for over ten journals. He was involved in many IEEE conferences, including GLOBECOM, ICC, VTC, PIMRC, WCNC, CCECE, and CWIT, in different roles, such as the general chair, the symposium chair, the tutorial instructor, the track chair, the session chair, the TPC co-chair, and a keynote speaker. He has been nominated as an IEEE Distinguished Lecturer several times during the last ten years. He is currently serving as the Chair for the IEEE London Section and the Chair of the ComSoc Signal Processing and Computing for Communications (SPCC) Technical Committee.

Xuemin (Sherman) Shen (Fellow, IEEE) received the Ph.D. degree in electrical engineering from Rutgers University, New Brunswick, NJ, USA, in 1990. He is a University Professor with the Department of Electrical and Computer Engineering, University of Waterloo, Canada. His research focuses on network resource management, wireless network security, the Internet of Things, 5G and beyond, and vehicular networks. He is a Registered Professional Engineer of ON, Canada, an Engineering Institute of Canada Fellow, a Canadian Academy of Engineering Fellow, a Royal Society of Canada Fellow, a Chinese Academy of Engineering Foreign Member, and a Distinguished Lecturer of the IEEE Vehicular Technology Society and Communications Society. He received the Canadian Award for Telecommunications Research from the Canadian Society of Information Theory (CSIT) in 2021, the R. A. Fessenden Award in 2019 from IEEE, Canada, the Award of Merit from the Federation of Chinese Canadian Professionals (ON) in 2019, the James Evans Avant Garde Award in 2018 from the IEEE Vehicular Technology Society, the Joseph LoCicero Award in 2015 and the Education Award in 2017 from the IEEE Communications Society (ComSoc), and the Technical Recognition Award from Wireless Communications Technical Committee in 2019 and AHSN Technical Committee in 2013. He has also received the Excellent Graduate Supervision Award in 2006 from the University of Waterloo and the Premier's Research Excellence Award (PREA) in 2003 from the Province of Ontario. He served as the Technical Program Committee Chair/Co-Chair for IEEE GLOBECOM'16, IEEE INFOCOM'14, IEEE VTC'10 Fall, IEEE GLOBECOM'07, and the Chair for the IEEE ComSoc Technical Committee on Wireless Communications. He is the President of the IEEE ComSoc. He was the Vice President for Technical and Educational Activities, the Vice President for Publications, the Member-at-Large on the Board of Governors, the Chair of the Distinguished Lecturer Selection Committee, and a Member of IEEE Fellow Selection Committee of the ComSoc. He served as the Editor-in-Chief for the IEEE INTERNET OF THINGS JOURNAL, *IEEE Network*, and *IET Communications*.